

Windows Administration *in Realtime*

2 **Letter from the Editor**

Realtime Publishers Is All About You

3 **Answers from the Experts**

How to Start Implementing Virtualization in Tough Economic Times

5 **Product Review**

KeePass

7 **SQL Server 2008 Backups**

Are Your Databases Safe?

By: Richard Siddaway - Explore SQL Server restore and backup issues and learns more about the new backup-related features in SQL Server 2008.

17 **The Deep Dive**

How to Set Up a Basic Windows Cluster

By: Greg Shields - Set up your clustering environment and ready it for hosting the hot, new Hyper-V technology.

22 **Practical PowerShell**

File Compression with WMI

By: Jeffery Hicks - Harness the power of PowerShell to find all files of a given type on a give volume and compress them.

31 **Exclusively Exchange**

Exchange 2007 Services Break Down

By: J. Peter Bruzzese - Step away from administrative tasks to learn more about Exchange 2007's services, dependent services, and the tools you can use to ensure they are up and running.



When people find something good,
word gets out.



VMware makes Windows better.

Over 100,000 companies of all sizes, including 94% of the Fortune 1000, use VMware. Why? Because VMware® Infrastructure delivers greater availability, superior manageability and the broadest support for legacy and current Windows applications.

And when you deploy VMware, you get a proven brand that's the global leader in virtualization—and has been for over a decade. It's experience that matters.

Learn more about the benefits of virtualizing your Windows environment with VMware.

Get your free "getting started kit".
Find out more at www.vmware.com/go/winsmb



Letter from the Editor

Realtime Publishers Is All About You

by Greg Shields

I'm a huge fan of the Mythbusters. I firmly believe that watching every episode of that marvelous show should be one of the prerequisites to true geekdom. I'll admit too that their efforts in weeding out the confirmed myths from those that're busted involve a huge amount of work dedication. Being buried in sand, sucked down a three-story whirlpool, and water skiing behind a cruise ship are all experiences I'm glad to leave off my resume.

You know, this eJournal and Realtime Publishers itself are a lot like the Mythbusters. We see that our job is to help you understand confirmed IT technology and best practices while busting the myths. We bring some of the world's best authors together to give you straight-shooting advice and education about all things IT.

Like every Mythbusters episode, from time to time, we want to reach out to you, our audience. We want you to tell us which technologies you need to learn more about. What products really get you excited? What technologies are on your horizon? For what IT best practices do you still need more education and information?

Let us know! If you've got areas of interest where you're just not getting the right information, drop us a line. If all you can find about a particular technology is marketecture and the white paper spin zone, tell us about it. We'll separate myth from truth and give you the information you need to succeed in your IT job today.

In fact, I'll give you my direct line for any comments or suggestions—send your thoughts and suggestions to gshields@realtimepublishers.net. I'll respond to every message that crosses my Inbox. In the mean time, enjoy this month's eJournal!♦

Answers from the Experts

How to Start Implementing Virtualization in Tough Economic Times

by Don Jones

Q: Our company wants to start using virtualization, but times are tough and money is tight. Where should we start?

A: Both VMware and Microsoft offer free hypervisors. In the case of Microsoft, it's called the Hyper-V Server, and it's technically Windows Server 2008. The offering is limited to the Server Core installation option

and only comes with the Hyper-V role. It doesn't contain any kind of local GUI for managing (you use remote tools), and it doesn't support clustering or quick migration tools. It's also limited to 32GB of host RAM and 4 CPU sockets.

VMware's free offering is its ESXi Server, a 32MB hypervisor that is a subset of the full ESX product. ESXi offers basically the same functionality

and can even be managed with Virtual Center (although its default management tool is the free Virtual Infrastructure Client); the big difference between ESXi and ESX is that the full version offers better manageability, including command-line options such as Windows PowerShell.

ESXi is bundled with server hardware from a number of vendors. They basically ship the ESXi 32MB

CONCENTRATED TECHNOLOGY

MAXIMUM KNOWLEDGE • MINIMUM TIME

Join columnists Don Jones and Greg Shields for informative articles on Windows PowerShell and Windows Server, freebies, techno-geek arguments, off-topic amusements, and even some free tools and resources. Get smarter, faster, and smile while you're doing it.

<http://concentratedtech.com>

hypervisor on an embedded USB drive on the server's motherboard and boot to that by default. That means you get a machine with no "host OS" and minimal overhead for the hypervisor; you essentially plug it in and start building virtual machines (VMs). Given hardware prices these days, it can be incredibly economical—I picked up a Dell PowerEdge 2900 III with 8GB of RAM and two multi-core procs as well as 1.75TB of usable drive space in a RAID 5 array for \$3200 delivered. It handily runs four to six Windows Server VMs, and I've had almost a dozen Linux VMs running at once. I'm not sure it gets much more economical than that.

Keep in mind that the big thing you're not buying is manageability. You're not going to get live VM migration (what VMware calls VMotion), consolidated command

and control over multiple machines, and so forth. But if your needs are limited to one or maybe two host servers, and you're not looking for high availability (clustering and live migration), powerful migration tools, and single-seat administration...well, in that case, the free hypervisors might be all you need. They should be a good start, either way.

Note that the *installable* ESXi (the one you download rather than the one embedded in server firmware) is upgradable to the full ESX. The embedded ESXi is typically *not* upgradable, as stripping it down to 32MB means stripping out all the things that make ESX better.

In the case of Microsoft's free offering, you're getting the full Hyper-V hypervisor and you can always choose to buy the System Center Virtual Machine Manager management tool

(which ain't cheap). You're also not getting any free licenses, and that's a big, yet subtle, difference. For example, if you buy Windows Server 2008 Datacenter Edition and install the Hyper-V role, all the Windows licenses for all your virtual machines are included. With the free Hyper-V (and with ESXi), you have to buy an operating system license for each virtualized copy of Windows. ♦

Don Jones is a co-founder of Concentrated Technology. Join him and cohort Greg Shields for intense Win2008 and Windows PowerShell training—visit ConcentratedTech.com/class for more details. Ask Don a question by visiting ConcentratedTech.com and using the "Contact" page.

Product Review

KeePass

by Eric Schmidt

A daily challenge that every user faces is simply providing a username and password to log in online. Security experts recommend using long and complex passwords that are unique for every site. Yet the reality is that with so many sites requiring usernames and passwords, people often use the same one for each.

A greater risk is that a user's password is not very complex. The reason users rely on simple passwords is simply that they want something that's easy to remember and type. Browsers have attempted to address the issue by incorporating password safes, but they have limited functionality. Further, the actual security of the passwords stored in those safes is questionable. Storing passwords in browsers also limits the flexibility and portability of what is being stored. One alternative is an open source product called KeePass. KeePass is a free, password-management application that offers many benefits over storing passwords in a browser.

KeePass is an exceptionally small application with a simple installation. At first run, it will prompt for a master password and automatically create a database. For extreme security, the product offers an option to require two factors for opening the database. The first factor is a key file stored in a secure location such as a thumb drive, and the second factor is your password. To take security even higher, its key file can be stored on an encrypted thumb drive. By storing the key file on a physical device, the database becomes more secure because you need both pieces to get into the database. KeePass leverages 256-bit AES encryption for the database, and a SHA-256 one-way hash to store the master password, making the database virtually impossible to crack.

KeePass organizes passwords into groups, making it easy to customize organization. In situations in which a large number of passwords is stored, the search feature provides an easy way to find a particular record. Another very useful feature designed for systems administrators is the ability to set expiration dates. This capability is beneficial for accounts that have passwords that have to be changed

at given intervals. By setting an expiration date on accounts, KeePass can display those that have expired, or are about to. This feature enables administrators to monitor accounts that need to be changed and take action.

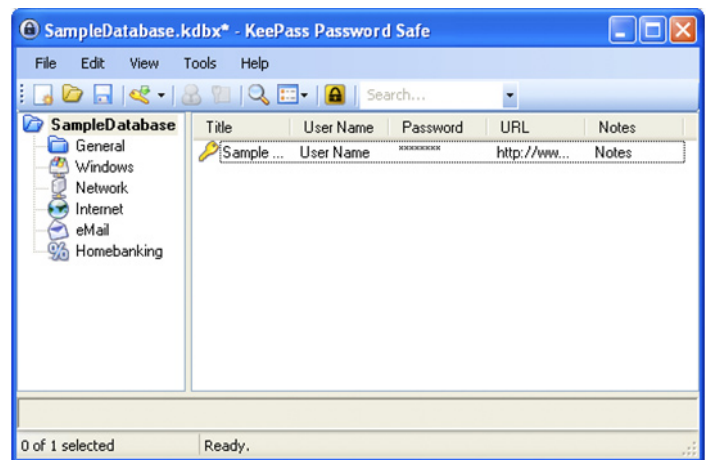


Figure 1: The KeePass console.

KeePass also provides a handy random password generator, which can generate passwords of any length and complexity. One feature often needed is the ability to automatically enter usernames and passwords into the browser. KeePass accomplishes this with an auto-type feature that creates everything from simple usernames, tab orders, and password sequences to very complex scripts with any number of fields. When copy/paste is used instead of auto-type, KeePass has a timer that will clear the clipboard after a period of time.

KeePass is open source with a large number of community-created plugins. Its plugins enable database backup, import/export, and browser enhancements, among others.

One final aspect is its ability to create custom fields to store information that should be encrypted. This data is not necessarily password information, but instead sensitive data such as bank account and credit card numbers.

Overall, KeePass is an excellent password management tool and has the capabilities and security to address both personal and business needs. Its flexibility through custom fields and plugins enables it to be extended to suit the unique requirements of each user. Best of all, it costs you nothing. ♦

Eric Schmidt works as Enterprise Microsoft Security Technologist, with Honors, for Raytheon Company and has worked in Information Technology for 13 years. Eric has a Masters degree in Computer Information Technology and has developed extensive experience in systems administration, engineering, and architecture specializing in Microsoft Active Directory and Systems Management. Eric has been well recognized throughout his career for his contributions to designing and implementing enterprise-wide solutions using Microsoft Windows-based technologies.

SQL Server 2008 Backups

Are Your Databases Safe?

by Richard Siddaway

One of the primary responsibilities of a DBA is to ensure the safety of the data held in the databases they are responsible for administering. This is often expressed in terms of having a sound, and tested, backup policy. In reality, a sound restore policy is required, with the backup policy defining how you generate the source data for the restore process. This article will discuss the restore and backup issues surrounding SQL Server and show some of the new backup-related features in SQL Server 2008.

We will concentrate on backup as it relates to functionality within SQL Server 2008. Previous versions may provide the same, or similar, functionality that is implemented in a different manner and requires a different backup technique. The discussion in this article will concentrate on using the native SQL Server backup and restore tools. Third-party backup tools can be used to produce similar results.

The SQL Server database engine, and the transaction-based databases it supports, tends to get the most attention in terms of backup and restore, but many organizations are also making use of Analysis Services and Reporting Services. We will look at how you can back up and restore these features to ensure your whole environment is protected. SQL Server is often used by other Microsoft products as the backend data store. It is essential that the databases supporting these products, such as SharePoint, are protected to the same level as the databases used to support critical business-process applications.

Restore and Backup Strategy

Before discussing backup methods, we need to resolve the issues around restoring data:

- ▶ What time scale do you have to restore the database?
- ▶ What is the impact of the database being unavailable (what do you need to be able to restore the whole database, a file within the database, or a data page)?
- ▶ Does the database have to be restored in its entirety, in one go, or can it be restored one piece at a time?
- ▶ Is there an environment into which the database can be restored or do you need to obtain new hardware?

These issues will vary between organizations and between databases used in the same organization. Make sure you understand them for your organization and ensure that your organization understands the relative importance of the databases it uses. The wrong time to be arguing about which database to restore first is in the middle of a disaster.

One important point to remember is that high availability does not negate the need for backups. High availability protects the business-process application; it does not protect the data. You can provide varying degrees of high availability through clustering, replication, database mirroring, and even log shipping. However, you also need to back up the data.

A very important part of a backup strategy is testing your capability to restore the database. One of the first things you will be asked in a recovery situation is “How long?” How long until I get my data back? How long until my people can start working again? If you cannot give a reasonable answer to these, and similar, questions, the stress levels during the restore activity will be sky high as the users continually bombard you with variations on the theme of “Is it finished yet?” However, if you have tested the restore procedures, you can truthfully say “Our tests show that it will take 2 hours to restore” (or whatever time is needed), decreasing the tension and pressure.

The other, and probably more important, reason for testing the restore procedures is to make sure you can actually perform the restore and that the backups you are using for restore purposes are readable and usable. Having to admit that you have lost 6-months data because you haven’t checked whether the backups are usable is unlikely to be a career-enhancing opportunity

Recovery Models

When discussing SQL Server backups, we have to consider the recovery model to use for the database. There are three recovery models you can implement:

- ▶ Simple recovery—Automatically reclaims log space, meaning that log backups cannot be taken; the database can be restored only to the point of the last backup
- ▶ Full recovery—Log space is not reclaimed and log backups can be taken; restores can be performed to a point in time, or a particular transaction, using the log backups; the tail of the log can be restored so that the database can be restored to the same state as it had at the point of failure
- ▶ Bulk logged—Minimizes logging activity during bulk operations (for example, a data load); can only recover to the end of backup, as the bulk operations are not logged

migration | optimisation | measurement & management



Centiq Ltd is an IT Services company focussed on premium level IT consulting, architecture, design and implementation services. We are experts in Migration, Optimisation and Measurement & Management.

Centiq Ltd
Unit 1 Charles Park
Charles Way
Cinderhill Road
Nottingham
NG6 8RF

Tel: 0115 951 9666
Fax: 0115 951 9555
email: info@centiq.co.uk
www.centiq.co.uk



Simplify your IT, meet your business needs better

Many businesses grow rapidly in unplanned ways, often the burden of complexity is carried by their IT function.

Centiq helps you maximise your IT investments while ensuring that the IT function meets all your organisation’s overall needs more effectively.

Our tried and tested methodologies help you achieve these objectives, reduce the complexity of your computing environments, and contain the risks of growth and IT expenditure.

Talk to us about

- Infrastructure Optimisation
- Virtualisation
- Active Directory – planning and implementation
- Exchange 2007 upgrades
- SQL server upgrades

Microsoft
GOLD CERTIFIED
Partner

The full recovery model is used for production databases that you need to protect as much as possible. Simple recovery tends to be used for databases that do not change on a frequent basis. If you can perform a full backup after each change, you do not need to backup the logs. Bulk logged tends to be used only when performing bulk operations that you do not need to log.

Backup Types

SQL Server supports a number of backup types. This support gives you the flexibility required to design a backup strategy to meet business requirements. You can begin by considering how you would back up the whole database, as this scenario is easiest to manage. Backup types available are:

Full backup—In a full back, all the data in the database is backed up; the logs are backed up such that the database can be properly restored. In addition, only the data in the full backup needs to be restored to recover the database.

Differential backup—A differential backup will back up only those changes since the last full backup; this type of backup is normally used to speed the backup process. A full backup is taken and then a number of differentials. As each differential takes only the data since the last full backup, a restore consists of the full backup and the last differential (plus any log backups since the differential).

It is possible to perform a partial backup of a database (introduced in SQL Server 2005). Using this approach, a backup of the primary filegroup, all filegroups that are in a read/write state, and any read-only filegroups that are specified to the backup process are taken. A differential partial backup can be taken.

For a database that spans multiple files, a file-based backup regime may give the optimum restore options. A full, or differential, backup can be taken of one or more filegroups or files. This can make the backup process quicker but complicates and extends the restore process if the whole database must be restored. However, if only part of the database has to be restored, the process is potentially easier and quicker. Using a file-based backup strategy requires careful planning to ensure that everything is backed up and that the envisaged restore scenarios can be met.

So far, we have considered only backing up the data in the data files. There is another component in the database that we should consider—namely the transaction logs. If the full or bulk logged recovery model is used, it is possible to take a backup of the logs between full, or differential, backups of the database. The active part of the transaction log is backed up, including all log records not backed up since the last transaction backup. A sequence of log backups starts with a full backup. After that, a series of periodic log backups are taken. The database can be restored using the full backup and all the log backups restored in sequence. The full set of log backups must be applied in the correct order to complete the restore. It is possible to stop the restore of the log backups at a point in time or after a particular transaction. Using a mixture of full and transaction log backups gives the maximum protection to the data but potentially will take the longest time to complete the restore operation.

If a copy of the database is required for a special purpose—for instance, to produce a clone for testing purposes—it is possible to create a Copy-Only backup. This does not affect the database in terms of breaking the sequence of transaction log or differential backups.

Having looked at backup and restore strategies and the backup functionality available with SQL Server, let's now turn our attention to how to perform backups. When we think of SQL Server, we naturally think of backing up the transactional data bases. This is our first stopping point on our backup journey, but it is not the end point. SQL Server 7 went beyond the traditional relation database by introducing the ability to create and use OLAP cubes. This functionality has expanded to become Analysis Services. The databases serving Analysis Services need to be backed up. Subsequent versions of SQL Server introduced Reporting Services, which needs to be backed up. The Full Text catalog also needs to be backed up.

Do you need to keep a copy of SQL Server's configuration? Depending on what functionality you have enabled in your environment, there could be a variety of things you must make sure are backed up correctly. Let's start with the most common backup target—the database engine. The discussion will assume that SQL Server 2008 is being used. Some of the functionality mentioned may not be available in earlier versions of the product.

Database Engine

When you start to think about backing up the databases, you need to think about two sets of databases: the system databases and the user databases that you use with your applications. There are four system databases:

- ▶ Master
- ▶ Msdb
- ▶ Model
- ▶ Tempdb

Tempdb is SQL Server's scratch pad and as such is recreated every time SQL Server is restarted. There is nothing to be gained by backing up this database. The model database is only used to provide a template when a new database is created. If you create objects in the model database, they will be present in every database that you create on the server. In this case, it is essential that the model database is backed up. As it is a very small database, I tend to include it in the backup regime anyway.

The user databases should be backed up in accordance with the strategy you have created for protecting the data. How can you backup your databases?

You can generate a backup by right-clicking a database in SQL Server Management Studio (SSMS) and selecting Tasks, Back Up. This resulting dialog box (see Figure 1) will allow an immediate backup of the database or transaction logs to be taken. This approach is ideal for an *ad hoc* copy backup but is not what you need for a production environment.

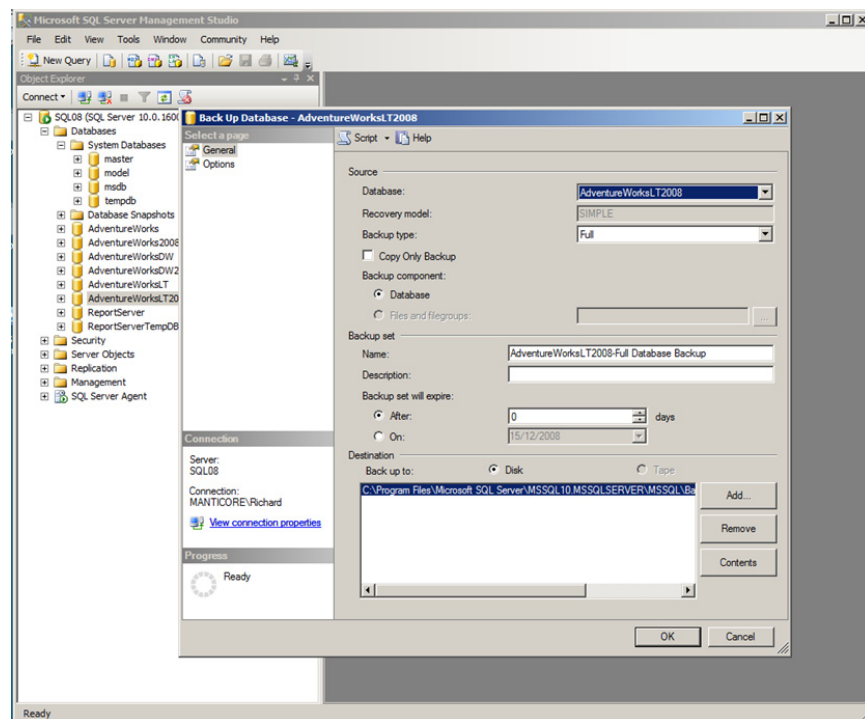


Figure 1: Manual backup.

One way to generate an automated backup for all the databases is to use a maintenance plan. In SQL Server 2008, this plan can be generated by expanding the server node in SSMS, expanding the Management node, and then right-clicking Maintenance Plan. Select New Maintenance Plan, give the plan a name, and the design surface opens (see Figure 2). Maintenance plans exist in previous versions of SQL Server, though they are wizard driven.

Using the design surface is very similar to designing SQL Server Integration Services workflows—not surprising when the maintenance plan is actually an SSIS package. In this case, a Back Up Database Task is dragged from the toolbox to the design surface. Right-clicking on the task and selecting Edit brings up the dialog box shown in Figure 2. At this point, the type of backup; which databases to back up, the folder for the backups (the folder must exist before the plan is run as it cannot be created by the plan), and whether to compress the backup can be chosen. You can even view the T-SQL commands that are used to perform the backup.

The sub plan is where the scheduling takes place with options as to when the plan runs and how frequently. Once the schedule is set, the maintenance plan is passed off to the SQL Server Agent for scheduling.

Maintenance plans can perform other tasks including:

- ▶ Checking database integrity
- ▶ Rebuilding Indexes
- ▶ Reorganizing Indexes
- ▶ Shrinking the database
- ▶ Updating statistics

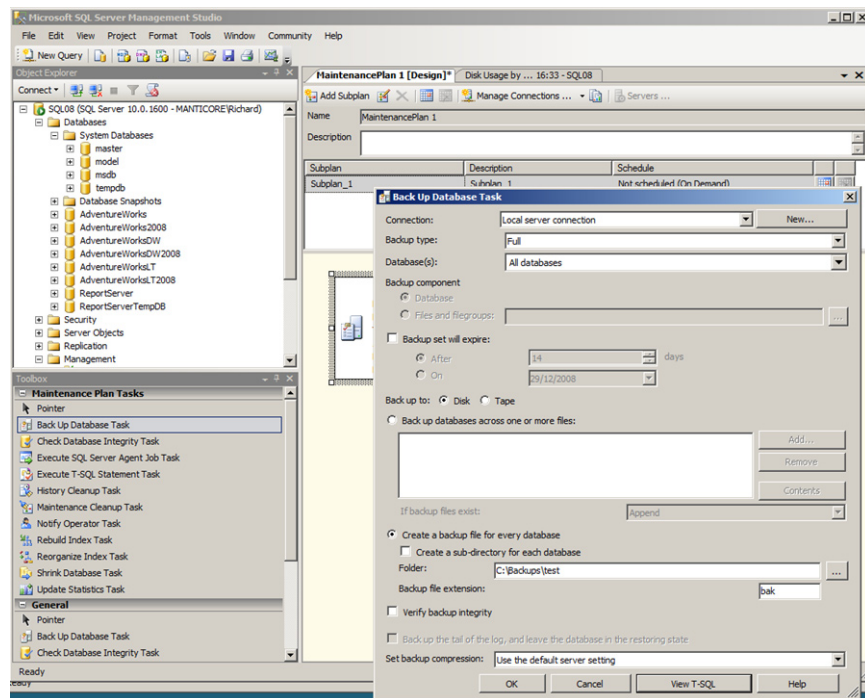


Figure 2: Backup by maintenance plan.

Tasks can be linked in a workflow with a sub plan or sub plans can be created to perform separate tasks. It is also possible to create maintenance plans on a master server and use them on one or more target servers. One very important point to remember is that you should always keep your backup files on a different volume than the database and log files. Doing so will improve the backup performance and is a safeguard. The last thing you want to happen is to lose the database files and your backup files!

One thing to note about backup files is that they can be opened in notepad (assuming the file isn't too big) and that in some areas of the file you can actually read the data! In addition, the backup files can end up occupying a significant amount of disk space, especially if a number of generations of backup are retained. SQL Server 2008 Enterprise Edition introduced the option to produce a compressed backup file. This is achieved by simply adding the **COMPRESSION** keyword into the **WITH** clause of the backup statement. The following statement was copied from the maintenance plan for the database backups and had the keyword added and the name of the backup file changed:

```
BACKUP DATABASE [AdventureWorks2008]
TO DISK = N'C:\Backups\AdventureWorks2008_backup_2008_12_15_compressed.bak'
WITH NOFORMAT, NOINIT, COMPRESSION,
NAME = N'AdventureWorks2008_backup_2008_12_15_compressed',
SKIP, REWIND, NOUNLOAD, STATS = 10
```

Compressing the backup file can have a significant impact on the size of the backup file as is shown in the following table. Backups are not compressed by default, though the default can be changed using `sp_configure`.

Backup type	Backup file size
Uncompressed	186,458 KB
Compressed	45,416 KB

The drawback to using compression is that producing a compressed backup will take significantly more CPU cycles compared with creating an uncompressed backup file. This could have an adverse affect on server performance. One method of preventing this is to use the resource governor that is available in SQL Server 2008. If your database is not in use on a 24x7 basis (so you have a backup window with minimal other activity), the resource governor would not be required.

SQL Server 2008 introduced a security feature known as Transparent Data Encryption (TDE), which performs encryption, and decryption, of the data and log files during read/write operations. TDE operates at the page level within the database. Pages are encrypted before writing to disk and decrypted when read from disk into memory. A full description of TDE is outside of the scope of this article, but it does have an impact on backup.

When using TDE, it is essential that the certificate and the private key are backed up in a reliable manner. The backup files produced from databases that are protected by TDE will also be encrypted using the database encryption key. The certificate and private key must be available to restore these files. If it is not available, the restore will not be allowed to proceed.

It is also possible to script a backup of one or more databases using SQL Server Management Objects (SMO). Introduced in SQL Server 2005, SMO provides a set of .NET classes that can be used to administer SQL Server instances and databases. SMO would have had to have been used through a .NET application when it was first introduced. This has turned a number of DBAs off to using the technology. With the introduction of PowerShell, which is .NET based, you have a scripting and command shell environment in which you can use SMO. SQL Server 2008 is the first version that has PowerShell functionality built-in, making this approach even more accessible. The code to perform a SQL Server database backup would look something like this:

```
## load SMO assemblies
## use $null to prevent display of assembly load information
```



```

$null = [reflection.assembly]::LoadWithPartialName("Microsoft.SqlServer.ConnectionInfo")
$null = [reflection.assembly]::LoadWithPartialName("Microsoft.SqlServer.SmoEnum")
$null = [reflection.assembly]::LoadWithPartialName("Microsoft.SqlServer.Smo")

### set SMO variable
$Smo = "Microsoft.SqlServer.Management.Smo."
$server = new-object ($Smo + "Server") "SQL1"

### database to back up
$bakup = new-object ($smo + "Backup")
$bakup.Database = "AdventureWorks"

### set backup device and type
$date = get-date

$date = $date -replace "/", "-"          # replace / symbols in date part
$date = $date -replace ":", "-"          # replace : symbols in time part
$date = $date -replace " ", "--Time-"    # show time part
$file = "C:\Backups\AdventureWorks" + "--" + $date + ".bak"

$bakup.Devices.AddDevice($file, [Microsoft.SqlServer.Management.Smo.DeviceType]::File)
$bakup.Action = [Microsoft.SqlServer.Management.Smo.BackupActionType]::Database

### invoke backup
$bakup.SqlBackup($server)

```

PowerShell does not load the SMO assemblies by default, so you start by loading the required bits of .NET. You create an object for the SQL Server instance and a backup object. After defining the database you will back up, you create the backup file name and decide the type of backup. The script finishes by starting the backup. The advantage of using PowerShell is that you can create scripts that access a number of remote servers.

Having looked at the backup issues for relational databases, you need to think about the data stored with your analytical databases.

Analysis Services

SQL Server 2008 Analysis Services provides OLAP and Data Mining capabilities. Connecting to Analysis Services in SSMS will show something similar to Figure 3 for an Analysis Services database. What of this do we need to back up and how do we perform the backup?

For backing up Analysis Services databases, you cannot use the T-SQL commands that you used for the relational databases. Instead, you have to use XML for Analysis commands. Yes, the backup command is an XML file! It looks something like the following code:

```

<Backup xmlns="http://schemas.microsoft.com/analysiservices/2003/engine">
  <Object>
    <DatabaseID>Adventure Works DW 2008</DatabaseID>
  </Object>

```

```
<File>C:\backups\Adventure Works DW 2008.abf</File>  
<AllowOverwrite>true</AllowOverwrite>  
<Password>Password1</Password>  
</Backup>
```

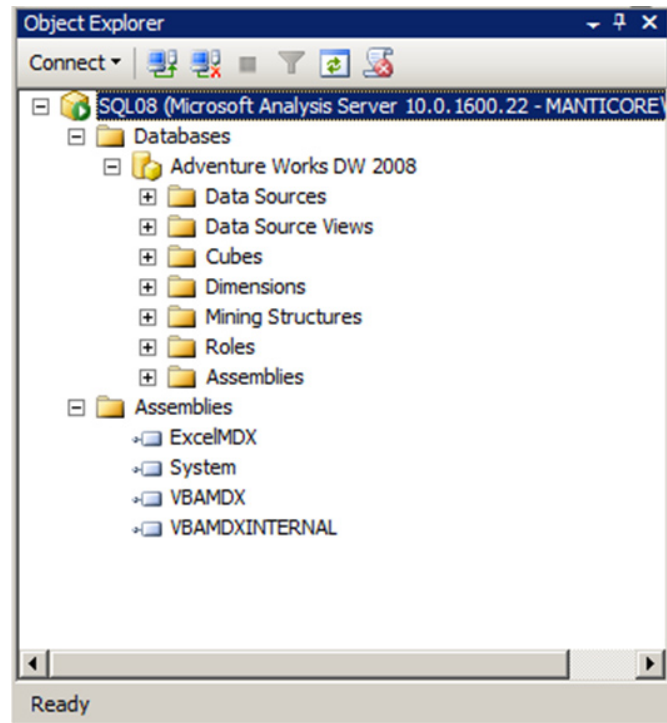


Figure 3: Analysis Server database.

The backup files are compressed by default. If a password is supplied, the backup files are encrypted.

You can perform a manual backup of the Analysis Services database by right-clicking it in SQL Server Management Studio and selecting Backup from the context menu. A dialog box will appear allowing you to select the location of the backup file and to pick from the following options:

- ▶ Allow File overwrite
- ▶ Apply compression
- ▶ Encrypt backup file (in which case a password must be supplied)

If the database uses remote partitions, you can select whether you will include them in the backup. Notice there is a button at the top of the page to create a script of your backup command. Now would be a good time to click it and have the script created in a new query window. Clicking OK causes the backup to start. This is fine for a one-off backup, but you really want to automate your backups.

You can create a SQL Server Job to perform your backup on a scheduled basis. In SSMS, expand the SQL Server Agent node and right-click Jobs, then select New Job. Give the job a name, and then click Steps in the Select a page pane. Add a step by clicking New. Ensure that the type is set to SQL Server Analysis Services Command. The XML that you created when you used the backup dialog box can now be copied and pasted into the command window for the job step. A schedule can be created for the job, and off you go. Note that whether you use a SQL Server Job or perform the task manually, a backup of an Analysis Services database is an all-or-nothing proposition. Your only choice is to back up the whole database. You do not have any options to make the backups more granular.

OLAP Technology

This is not the place for a full discussion of OLAP technology ; however, we can have a brief look at the issues as they affect our backup and restore strategy. There are three types of data that can be found in an Analysis Services database:

- ▶ The underlying source data
- ▶ Aggregations—these are pre-calculated summaries of the data and are used to reduce the time required to answer a query
- ▶ Metadata

The metadata will always be present. Whether or not the other two are present depends on the storage mode used by the database. If this data is not in the Analysis Services database, it will be found in the associated relational database being used as the data source. This database must be backed up as well as the Analysis Services database.

The last consideration for Analysis Services databases is when to perform a backup. This again depends on the storage mode used for the database. If everything is stored in the Analysis Services database (MOLAP), the optimum time to perform a backup is when the database has just been processed and new data imported. If all the data is held in the relational database, a simple backup to protect the metadata when it changes is all that is required. As the backup will produce a small file very quickly, it could be performed after processing. Where a hybrid approach is used, either back up the database or script out the database objects.

Having looked at protecting your relational and analytical databases, the next item on the agenda is Reporting Services.

Reporting Services

Starting as an addition to SQL Server 2000, Reporting Services is now a fully integrated reporting platform. Reporting Services can utilize data from a wide variety of sources as well as SQL Server. Reports can be viewed via a Web connection, integrated with SharePoint, or embedded into an application. As with any piece of business functionality, there are a number of considerations regarding backup.

If you are using SQL Server-based data sources as the basis for your reports, they will have been protected by the techniques we have discussed earlier in the article. What you need to consider is protecting the Reporting Services infrastructure.

Reporting Services creates two databases: reportserver and reportservertempdb. All data related to Reporting Services is stored in these databases. You should incorporate the two databases in your database backup schedules. Reportservertempdb does not store data that needs to be recovered, but the database itself and the table structure need to be restorable in the event of a hardware failure.

Other data that must be backed up includes:

- ▶ Reporting Services encryption keys
- ▶ Configuration files (these can be found in C:\Program Files\Microsoft SQL Server\MSRS10.MSSQLSERVER or the equivalent folder depending on your installation)
- ▶ Data files produced by Report Designer and Model Designer

The backups of these files have to be kept in a synchronized state with the backups of the databases that are dedicated to Reporting Services.

Full-Text Catalog

The full-text search capability in SQL Server is used to work with text-based data and issue queries based on an index of keywords. The full-text search operates on words and phrases using language-based rules. Full-text searches are best used against unstructured data. In SQL Server 2008, the full-text functionality is integrated into the database engine rather than being a standalone process as in previous versions. This makes backup a simpler proposition.

To back up the full-text indexes:

- ▶ Find the full-text indexes (in SQL Server 2008, the full-text catalog is a logical structure comprising the full-text indexes in the database)
- ▶ Identify the filegroups containing the full-text indexes
- ▶ Back up the filegroups containing the full-text indexes

There are T-SQL code examples to achieve these aims in SQL Server Books Online. The article “Backing Up and Restoring a SQL Server 2008 Full-Text Catalog” is a good place to start.

When a filegroup is restored, any full-text indexes are also restored.

Summary

SQL Server has a number of data types that you need to protect. The starting point for any discussion regarding backup is the restore strategy. What time do you have available to perform the restore and do you know the actions you need to perform in order to restore your data?

The majority of SQL Server instances are probably used for relational databases. We have seen that there are a number of backup types available for this type of data and a number of ways to perform the backups. Your challenge is to produce a backup regime that protects your organization’s data and meets its business requirements. ♦

Richard Siddaway is Microsoft Practice Leader for Centiq Ltd, a Microsoft partner specialising in optimisation, measurement, management, and migration involving Microsoft technologies. With more than 20 years experience in various aspects of IT, Richard is currently concentrating on the Microsoft environment at an architectural level – especially around Active Directory (AD), Exchange, SQL Server, and Infrastructure Optimisation. Much of his recent experience has involved migrating enterprises from Windows NT\Exchange 5.5 to Windows 2003\Exchange 2003. Richard has administration experience with several database systems, especially SQL Server. His initial programming background is still useful, as Richard is a very experienced scripter always looking for the opportunity to automate a process. Having discovered PowerShell 18 months ago, he has enthusiastically adopted it in preference to VBScript. Richard founded and currently leads the UK PowerShell User Group. Richard has presented to the Directory Experts Conference 2007, at various events at Microsoft UK, including the last Community Day, and for other UK User Groups.

Richard can be contacted through email at RSiddaway@centiq.co.uk or via his blog at <http://richardsiddaway.spaces.live.com/>

The Deep Dive

How to Set Up a Basic Windows Cluster

by Greg Shields

The new Hyper-V virtualization solution is one of the hottest new Microsoft technologies out on the market. A part of Windows Server 2008, Hyper-V brings some inexpensive yet powerful virtualization capabilities to IT data centers. Best of all, it arrives with no-added cost high-availability features that rely on the Windows Server Failover Clustering feature of Windows Server 2008. Because of that reliance, you can argue that Windows clustering skills might just be the #2 hottest Microsoft technologies—that many admins don't yet realize.

To that end, this month, I'll show you how you can build your own two-node Windows cluster that uses iSCSI for its shared storage. This information is taken from Chapter 10 of the Realtime eBook *The Definitive Guide to Building a Windows Server 2008 Infrastructure*. You can download this chapter or the entire eBook from <http://nexus.realtimepublishers.com/DGBWS2K8I.htm>. With the following information, you'll set up your clustering environment and ready it for hosting Hyper-V.

Next month, we'll talk more about the management of Hyper-V clusters. If you're like a lot of admins, you've probably given Windows clustering a pass. Hyper-V's reliance on clustering, however, will require you to fast-track your learning of this important feature if you want high availability for your virtual machines.

Installing Windows Clustering

For this example, we will build a two-node cluster that will host a cluster service atop the servers \\w2008a.realtime-windowsserver.com and \\w2008b.realtime-windowsserver.com. To simplify this example, we will use two iSCSI data stores and only two network cards. The first iSCSI target will serve as the quorum drive and will be configured with 500MB of space. The second iSCSI target will serve as the shared storage for the hosted cluster service and will be configured with 2GB of space. Although only two network cards are used in this example for simplicity—one for the iSCSI connection and another for the production network—it is strongly recommended that additional network cards are used in production to separate traffic between that needed for the production network, the cluster heartbeat communication, and its connection to iSCSI. Moreover, because the network-based connection to its iSCSI disk can be a single point of failure with only one network card, redundancy in iSCSI network cards is similarly recommended.

Configuring Networking

It is strongly suggested that the connection to the iSCSI target be made over a different network than that which is used for the production network. An example of the IP configuration of each cluster candidate and the iSCSI target server can be set up is shown graphically in Figure 1. Your actual network configuration may differ, but this image shows how the networking is segregated between iSCSI and production networks. Configure the network cards that will connect to the iSCSI target with the proper IP address and subnet mask but leave the gateway and DNS information blank. Also, remove the *Client for Microsoft Networks* service as well as the *File and Printer Sharing for Microsoft Networks* under the properties of the network card. Lastly, remove IPv6 if it is unused on this network.

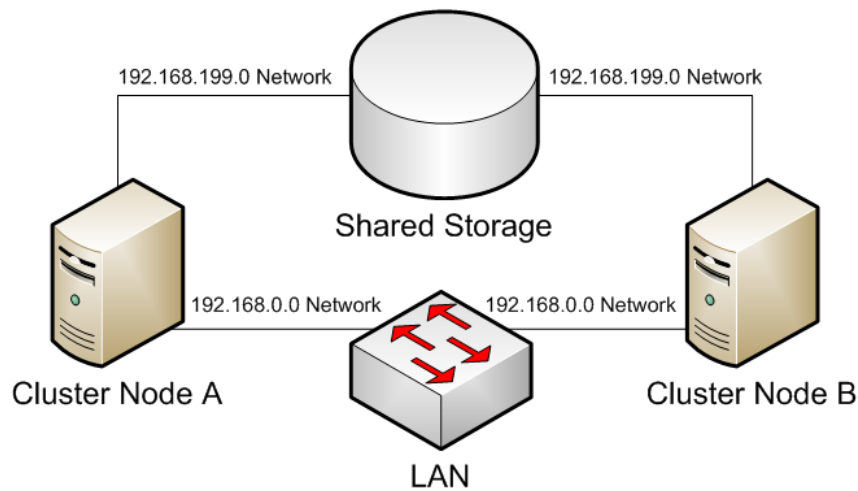


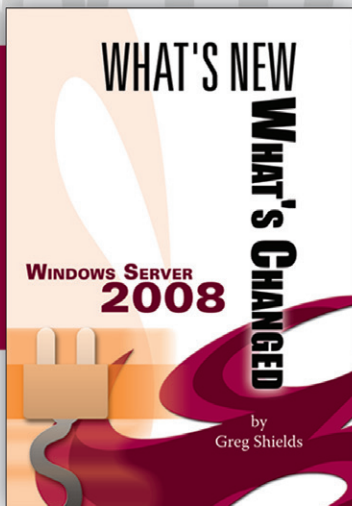
Figure 1: The networking configuration of our two-node cluster. This segregation of networks ensures that traffic routes through the correct network cards.

Configuring the Shared Storage

In this example, the two iSCSI targets have already been configured using the iSCSI data store's management utility. A LUN has been exposed to that iSCSI target and made available to each of the hosts. To connect to that iSCSI LUN on the first host, navigate to *Start | Administrative Tools | iSCSI Initiator*. The first time this tool is run, you will be prompted to start the Microsoft iSCSI Service and set it to Automatic. Click Yes to do so. You will also be prompted to unblock the Microsoft iSCSI Service so that it can operate through the Windows Firewall. Click Yes again to enable this firewall exclusion.



SAPIEN
PRESS



Microsoft has released its next server operating system – Windows Server 2008 – and you need to know more about it. But you don't need the basics. You already know Windows 2003. You just need to know what's new and what's changed in Windows Server 2008. Read-Only Domain Controllers, the Group Policy Central Store, Terminal Server RemoteApps, Fine-Grained Password Policies. This quick and entertaining guide, written by Windows insider Greg Shields does just that. Focusing on the new technologies for installing, managing, and securing Windows Server 2008, you'll quickly ramp up your skills. Save yourself some time and money by skipping the basics and using your existing skills to master Microsoft's new server O/S.

Automate server installations * More effectively manage servers through Server Manager * Gain insight with Reliability and Performance Monitor * Implement powerful new Group Policy * Reduce your attack surface with Server Core * Complete better Active Directory backups * Deploy apps using Terminal Services * Secure your servers with the new Windows Firewall

TABLE OF CONTENTS

Chapter 1: Introduction to Windows Server 2008
Chapter 2: Installing Windows 2008
Chapter 3: Server Management
Chapter 4: Group Policy
Chapter 5: Server Core
Chapter 6: Windows Server Virtualization

Chapter 7: Active Directory
Chapter 8: Terminal Services
Chapter 9: Security & the Windows Firewall with Advanced Security
Chapter 10: IIS 7.0
Chapter 11: Other New & Compelling Features

http://www.sapienpress.com/Windows_Server_08.asp

Greg Shields

If your iSCSI target has special software or device drivers required for its use, this software must be installed prior to moving to the next step.

The Microsoft iSCSI Initiator has six configuration tabs:

- ▶ **General.** This tab displays the name for the iSCSI initiator and provides a location to change that name as well as configure authentication via CHAP. In our example, we will not be configuring authentication for simplicity. However, in a production environment, this authentication protects rogue computers from connecting to exposed iSCSI LUNs over the network and its configuration is considered a best practice. The CHAP secret will need to be entered at both the iSCSI target and initiator to connect.
- ▶ **Discovery.** Click *Add Portal*. In the resulting screen, enter the DNS name or IP address for the iSCSI target that hosts the data storage location. Click *Advanced*. Ensure that the *Local adapter* is set to *Microsoft iSCSI Initiator* and the *Source IP* is set to the IP address for the network card you want to configure for use with iSCSI.
- ▶ **Targets.** If the connection was correctly made on the previous tab, clicking to this tab will automatically display the available drives on the iSCSI target. A picture of how this should look is shown in Figure 2. Click each discovered target and then *Log on*. In the resulting screen, select the *Automatically restore this connection when the computer starts* check box and click *Advanced*. Again set *Local adapter* to *Microsoft iSCSI Initiator*, *Source IP* to the correct source IP for this server's network card, and *Target portal* to the iSCSI target address. If your iSCSI target uses special software that enables multiple connections to the target, you can select the *Enable multi-path* check box. Complete these steps for each discovered drive.
- ▶ **Favorite Targets.** Here you can view the properties of any connected drives. There is no further configuration to be done on this tab.
- ▶ **Volumes and Devices.** Click *Autoconfigure*. If everything has been set up correctly to this point, the *Volume/mount point/device box* should populate with information about the discovered drives.
- ▶ **RADIUS.** For the purposes of this example, there is nothing to do on this tab.

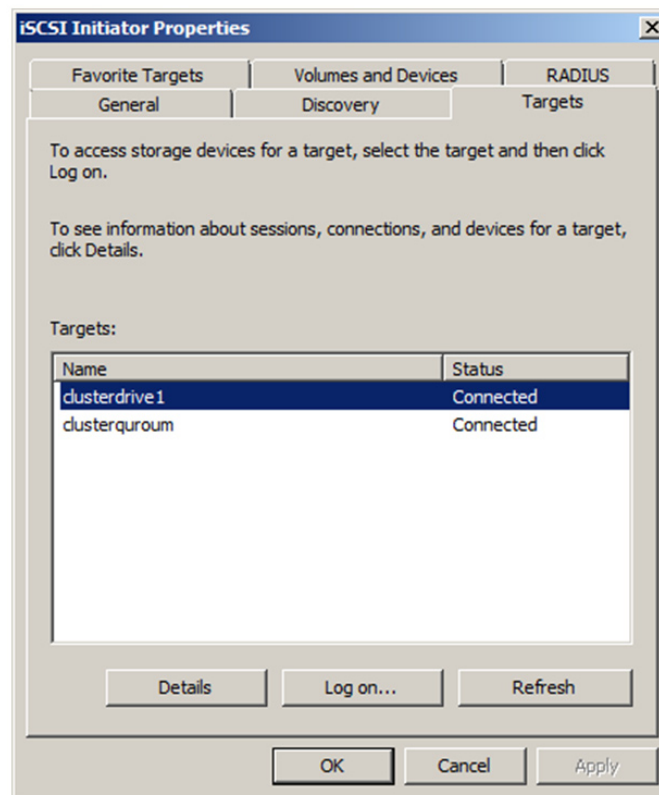


Figure 2: If you've configured everything correctly, your iSCSI drives should appear under the Targets tab.

You will want to complete these steps on both candidate hosts to establish each server's connection to the iSCSI target. Note that at this point, you have made a connection to a raw drive but have not yet initialized or formatted the drive, nor have you added a drive signature to that drive.

To do so, launch Server Manager and navigate to *Storage | Disk Management* on one of the two nodes. The two drives should be present on the node, but both will be displayed in black as *Unallocated disks*. Right-click each disk and select to bring that disk *Online*. Then right-click one of the disks and choose *Initialize Disk*. The Initialize Disk wizard will appear with both disks selected. If this disk will never grow beyond 2TB in size, keep the disk as a *Master Boot Record (MBR)* disk. If you believe the disk will grow beyond that size at some point in the future, convert the disk to a *GUID Partition Table (GPT)* disk. Lastly, right-click each disk and select *New Simple Volume*. Create a new simple volume on each disk, assign a drive letter, and format the disk as NTFS.

Validate and Create the Cluster

By bringing the disk online and formatting it, the disk can be verified by the Cluster Validation Wizard, allowing that process to complete its testing. At this point, navigate to *Administrative Tools | Failover Cluster Management*. There, right-click the top-level node and choose to *Validate a Configuration*. Doing so will launch the Validate a Configuration Wizard, which will prompt for the names of the candidate nodes and the tests to be run. Once run, the wizard will provide an HTML report of the results similar to what is seen in Figure 3. If any errors appear in the running of the wizard, you will need to fix the problem and re-run the wizard until all tests are passed.

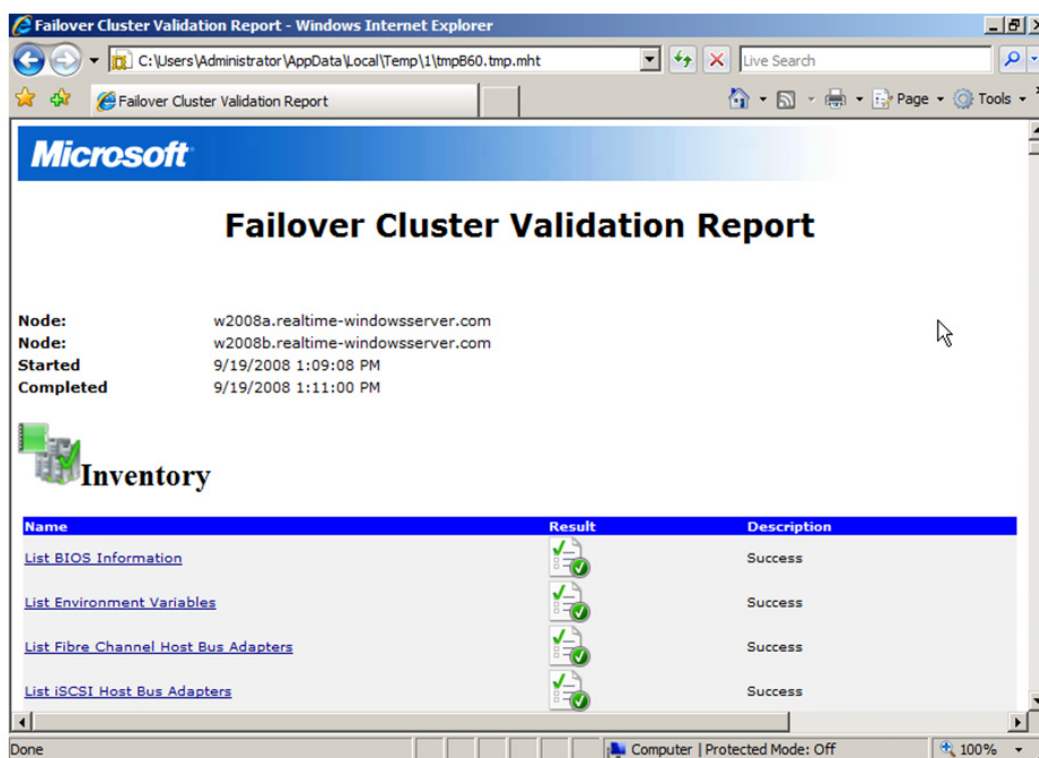


Figure 3: An example of the Cluster Validation Report. All cluster components must pass all tests prior to attempting to create a cluster.

Once you've completed the wizard and fixed any issues discovered in the validation process, it is now time to create your cluster. Do so back in the Failover Cluster Management console by right-clicking the top-level node and choosing *Create a Cluster*. In the resulting wizard, enter the names of the candidate nodes. In the next screen, provide a name as well as an IP address for the cluster itself. This name and address will be used for connecting to the cluster for management. Finally, confirm the creation of the cluster. The wizard will create the cluster and return control when complete.

Ensure that you click *View Report* after the completion of the installation to view the results of the installation process. Some clusters can be installed with warnings that later cause problems.

Post-Installation Quorum Reconfiguration

Once the cluster has completed its installation, it can be managed via the Failover Cluster Management console. Immediately after creation, navigate to this console and verify that all network and storage resources are available and visible in the interface.

The cluster installation wizard by default does not always install the quorum resource to the correct shared disk and sometimes does not always choose the correct quorum model. If either of these conditions is the case, these settings can be changed by right-clicking the cluster name and selecting *More Actions | Configure Cluster Quorum Settings*. In the resulting wizard, it is possible to change the quorum model as well as the shared drive that is to be used for the quorum. Figure 4 shows an example of the screen where the quorum drive can be changed. Click through the wizard to complete the reconfiguration.

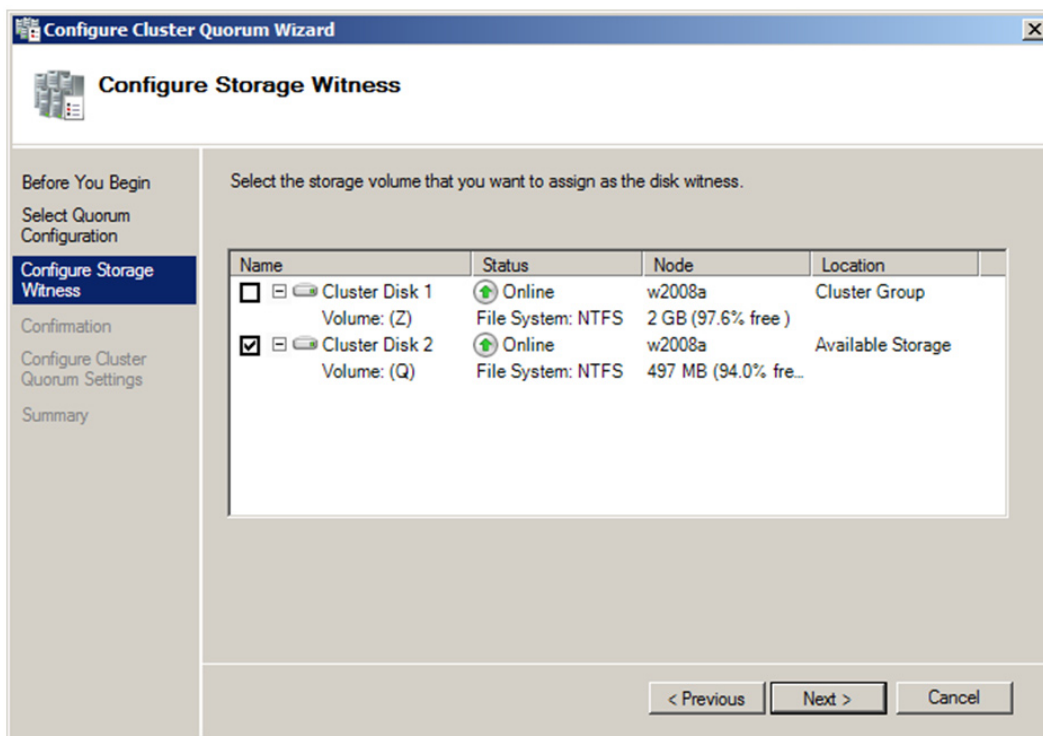


Figure 4: It is possible to adjust the quorum drive and model after the cluster completes its installation.

Learn more about Windows clustering by downloading Chapter 10 of *The Definitive Guide to Building a Windows Server 2008 Infrastructure* at <http://nexus.realtimepublishers.com/DGBWS2K8I.htm>. Next month, I'll show you how you can manage that Hyper-V cluster and ensure the successful failover of resources from node to node. ♦

Greg Shields, MCSE: Security, CCEA, is an independent author, speaker, and consultant, based in Denver, Colorado. With more than 10 years of experience in information technology, Greg has developed extensive experience in systems administration, engineering, and architecture. Greg is a contributing editor for both Redmond magazine and MCPmag.com, authoring two regular columns along with numerous feature articles, webcasts, and white papers. He is also the resident editor for Realtime Publishers' Windows Server Community at www.realtime-windowsserver.com.

Practical PowerShell

File Compression with WMI

by Jeffery Hicks

You can download a zip file with all these scripts from http://www.realtime-windowsserver.com/code/v2n3_Practical_PowerShell.zip.

One “old-school” technique to save on disk space has always been file or folder compression. The recommended best practice is to enable it at the folder level. However, there may be times when you want to compress a group of files based on their type. For example, text and Microsoft Word files compress very well. The challenge is to find all files of a given type on a give volume and compress them, assuming they aren’t already compressed.

We can accomplish this with Windows Management Instrumentation (WMI). For example, you can use a query such as this

```
Select * from CIM_DATAFile where Drivetype='c:' AND Compressed='FALSE' AND  
Extension='txt'
```

The CIM_DATAFile object has a Compress() method that can be invoked for each file returned by the query. You can even create a complex query to find all instances of several different file types:

```
Select * from CIM_DATAFile where Drivetype='C:' AND Compressed='FALSE' AND  
(Extension='txt' OR Extension='log' OR Extension='csv')
```

Naturally, as this is a PowerShell column, there has to be a PowerShell solution. Let’s take a look at my script Compress-FileType.ps1, which you can download from the [Realtime Web site](#).

```
#Compress-FileType.ps1  
  
Param (  
[string]$extensions="txt,bmp,doc,xls,docx,xlsx,ps1,vbs,wsf",  
[string]$drive="c:",  
[string]$computer=$env:computername,  
[switch]$debug  
)  
  
if ($debug) {  
    $debugPreference="Continue"  
}
```



```

Write-Debug "Connecting to drive $drive on $computer"

Write-Progress -Activity "Compress File Types" -status "Getting current freespace for
$drive on $computer"

#get freespace before
$disk=Get-WmiObject win32_logicaldisk -filter "deviceid='$drive'" -computername $computer
$freeBefore=$disk.freespace

Write-Debug "Freespace before is $freeBefore"

#build query
$extFilter="{0}{1}{2}" -f "(Extension='",($extensions.replace(",","'" OR
Extension='"))),"'"

$query="Select CSName,Name,Compressed,Extension,FileSize from CIM_DATAFILE Where
Drive='$Drive' AND Compressed='FALSE' AND $extFilter"

Write-Debug "Query = $query"

Write-Progress -Activity "Compress File Types" `
-status "Running query $query on $computer" `
-currentoperation "Please wait...this may take several minutes..."

Write-Debug "Getting files"
$files=Get-WmiObject -query $query -computername $computer

$filecount=$files.count
Write-Debug "Found $filecount files"

#only process if files were returned
if ($filecount -gt 0) {
    #keep a running count of compressed files
    $iCompressed=0
    $iFailures=0
    $i=0
    Write-Debug "processing files"

    foreach ($file in $files) {
        $i++
        [int]$percent=($i/$filecount)*100
        Write-Debug $file.name
    }
}

```

```

    Write-Progress -Activity "Compress File Types" -Status "Compressing"
    -currentoperation $file.name -percentComplete $percent
    $rc=$file.Compress()
    Switch ($rc.returnvalue) {
        0 { $Result="The request was successful"
            #increment the counter
            $iCompressed++
            #write the WMI file object to the pipeline
            #for later processing
            write $file | select CSName,Name,FileSize,Extension,Compressed
            Break}

        2 { $Result="Access was denied." ; Break}
        8 { $Result="An unspecified failure occurred."; Break}
        9 { $Result="The name specified was invalid."; Break}
        10 { $Result="The object specified already exists."; Break}
        11 { $Result="The file system is not NTFS."; Break}
        12 { $Result="The operating system is not supported."; Break}
        13 { $Result="The drive is not the same."; Break}
        14 { $Result="The directory is not empty."; Break}
        15 { $Result="There has been a sharing violation." ; Break}
        16 { $Result="The start file specified was invalid."; Break}
        17 { $Result="A privilege required for the operation is not held.";
Break}

        21 { $Result="A parameter specified is invalid."; Break}
    } #end Switch

    #display a warning message if there was a problem
    if ($rc.returnvalue -ne 0) {
        $msg="Error compressing: {0}. {1}" -f $file.name,$Result
        Write-Warning $msg
        $iFailures++
    }
    Write-Debug "Result=$result"
}

Write-Progress -Activity "Compress File Types" -status "Getting current freespace for
$drive on $computer"

#get freespace after

```

```

    $disk=Get-WmiObject win32_logicaldisk -filter "deviceid='$drive'" -computername
$computer
    $freeAfter = $disk.freespace

    $freeDiff = $freeAfter - $freeBefore

    Write-Debug "Freespace after is $freeBefore"
    Write-Debug "Freespace difference is $freeDiff"
    Write-Debug "Presenting summary"

    Write-Progress -Activity "Compress File Types" -status "Finished" -completed $True

    Write-Host "Summary" -ForegroundColor CYAN
    Write-Host "*****" -ForegroundColor CYAN
    Write-Host "Computer          : $computer" -foregroundcolor CYAN
    Write-Host "File types         : $extensions" -foregroundcolor CYAN
    Write-Host "Drive              : $drive" -foregroundcolor CYAN
    Write-Host "Total Files        : $filecount" -foregroundcolor CYAN
    Write-Host "Compressed         : $iCompressed" -foregroundcolor CYAN
    Write-Host "Failures           : $iFailures" -foregroundcolor CYAN
    Write-Host "Free bytes before: $freeBefore" -foregroundcolor CYAN
    Write-Host "Free bytes after : $freeAfter" -foregroundcolor CYAN
    Write-Host "Bytes Recovered  : $freeDiff" -foregroundcolor CYAN
}
else {
    Write-Warning "No matching files ($extensions) found on drive $drive on $computer"
}
}#end of script

```

The script takes several parameters, all of which have default values. The extensions parameter is a comma-separated list of file extensions. The drive parameter limits the query to a single volume. Querying the CIM_DATAFile class is notoriously slow, and the more specific the query, the better. You can specify a remote computer name, although performance will probably be better when running locally. As written, the script doesn't support PSCredentials, but you could certainly add it. I've also included a switch parameter `-debug`. If you specify `-debug`, the debug messages I've added to the script will be written to the debug pipeline. This is a handy technique that makes your script "debuggable" from the beginning. Here's how you might run the script:

```
PS C:\> C:\scripts\compress-filetype.ps1 -extensions "txt,log,csv,doc,xls"
```

The script begins using by **Write-Progress** to display a progress message:

```
Write-Progress -Activity "Compress File Types" -status "Getting current freespace for
$drive on $computer"
```

At the end of the script is a brief summary that includes how many bytes were “recovered,” so I grab the current free space on the specified drive:

```
$disk=Get-WmiObject win32_logicaldisk -filter "deviceid='$drive'" -computername $computer
$freeBefore=$disk.freespace
```

To build the query, I do a little fancy string manipulation using the `-f` operator:

```
$extFilter="{0}{1}{2}" -f "(Extension='",($extensions.replace(",","'" OR
Extension='"))),"'"
```

This turns the string “txt,log,csv,doc,xls” to “(Extension=’txt’ OR Extension=’log’ OR Extension=’csv’ OR Extension=’doc’ OR Extension=’xls’). This filter is combined with the main WMI query:

```
$query="Select CSName,Name,Compressed,Extension,FileSize from CIM_DATAFILE Where
Drive='$Drive' AND Compressed='FALSE' AND $extFilter"
```

World's hottest IT topics

- Windows PowerShell 2nd Edition
- Windows PowerShell 3rd Edition
(covers Windows PowerShell v2.0)
- ADSI Scripting
- WSH and VBScript Core
- Windows Server 2008: What's New/What's Changed
- Exchange Management Shell
- Managing Active Directory With Windows PowerShell
- Managing VMware Infrastructure With Windows PowerShell



For more information:
www.sapienpress.com

I found during script development that selecting all properties made for a slow query. You really don't need every property, so my query selects only the bare minimum. If you require any other properties, you will need to edit the query. Once the query is complete, it can be executed using **Get-WmiObject**:

```
Write-Progress -Activity "Compress File Types" `
    -status "Running query $query on $computer" `
    -currentoperation "Please wait...this may take several minutes..."

Write-Debug "Getting files"
$files=Get-WmiObject -query $query -computername $computer
```

The query may take several minutes or more to complete. When finished, the script defines some variables, assuming any files were returned from the query:

```
if ($filecount -gt 0) {
    #keep a running count of compressed files
    $iCompressed=0
    $iFailures=0
    $i=0
}
```

Now the main work can begin. Each file from the collection matching files is processed by the **ForEach-Object** cmdlet. Variables and the progress bar are updated and the file is compressed:

```
foreach ($file in $files) {
    $i++
    [int]$percent=($i/$filecount)*100
    Write-Debug $file.name
    Write-Progress -Activity "Compress File Types" -Status "Compressing" -currentoperation
    $file.name -percentComplete $percent
    $rc=$file.Compress()
```

WMI will return a result object, which the script captures as **\$rc**. By looking at the **ReturnValue** property, the script can assess whether the method was successful. If not, the **ReturnValue** can be associated with an error message. The script accomplishes this with a **Switch** statement:

```
Switch ($rc.returnvalue) {
    0 { $Result="The request was successful"
        #increment the counter
        $iCompressed++
        #write the WMI file object to the pipeline
        #for later processing
        write $file | select CSName,Name,FileSize,Extension,Compressed
        Break}
```



```

2 { $Result="Access was denied." ; Break}
8 { $Result="An unspecified failure occurred."; Break}
9 { $Result="The name specified was invalid."; Break}
10 { $Result="The object specified already exists."; Break}
11 { $Result="The file system is not NTFS."; Break}
12 { $Result="The operating system is not supported."; Break}
13 { $Result="The drive is not the same."; Break}
14 { $Result="The directory is not empty."; Break}
15 { $Result="There has been a sharing violation." ; Break}
16 { $Result="The start file specified was invalid."; Break}
17 { $Result="A privilege required for the operation is not held."; Break}
21 { $Result="A parameter specified is invalid."; Break}
} #end Switch

```

If the result is 0, the compression was successful, the counter is incremented, and the file is written to the pipeline. Otherwise, an appropriate error message is stored in `$Result`. The **Switch** construct will only match `ReturnValue` once, so I've added the **Break** keyword to eke out a little more performance. There's no reason to make any more comparisons once a match has been made.

If there was an error compressing the file, a message is constructed and written to the Warning pipeline using **Write-Warning**:

```

if ($rc.returnValue -ne 0) {
    $msg="Error compressing: {0}. {1}" -f $file.name,$Result
    Write-Warning $msg
    $iFailures++
}

```

I didn't want to take all the fun out of this, so you might enhance this script by logging these messages to a text file. After all the files have been processed, the drive's freespace is grabbed again and the difference calculated.

```

$disk=Get-WmiObject win32_logicaldisk -filter "deviceid='$drive'" -computername $computer
$freeAfter = $disk.freespace
$freeDiff = $freeAfter - $freeBefore

```

All that's left now is to display a summary.

```

Write-Host "Summary" -ForegroundColor CYAN
Write-Host "*****" -ForegroundColor CYAN
Write-Host "Computer      : $computer" -foregroundcolor CYAN
Write-Host "File types       : $extensions" -foregroundcolor CYAN
Write-Host "Drive           : $drive" -foregroundcolor CYAN
Write-Host "Total Files      : $filecount" -foregroundcolor CYAN
Write-Host "Compressed       : $iCompressed" -foregroundcolor CYAN
Write-Host "Failures         : $iFailures" -foregroundcolor CYAN

```

```
Write-Host "Free bytes before: $freeBefore" -foregroundcolor CYAN
Write-Host "Free bytes after : $freeAfter" -foregroundcolor CYAN
Write-Host "Bytes Recovered : $freeDiff" -foregroundcolor CYAN
```

If the WMI query doesn't return any files, a message is written to the warning pipeline:

```
Write-Warning "No matching files ($extensions) found on drive $drive on $computer"
```

I decided to write the summary directly to the console so that it wouldn't be commingled with the other pipelined output. The script's only output to the success pipeline is the file object:

```
#write the WMI file object to the pipeline
#for later processing
write $file | select CSName,Name,FileSize,Extension,Compressed
```

Thus, you could run a PowerShell expression such as this:

```
PS C:\> C:\scripts\compress-filetype.ps1 -extensions "txt,log,csv,doc,xls" | out-file
compressed-results.txt
```

Warnings will be displayed and the summary will be written to the console, but all compressed file information will be written to the text file. I'm including the CSName property, which returns the computer name, so you could process a list of computers and save the results:

```
get-content computers.txt | foreach {
    C:\scripts\compress-filetype.ps1 `
    -extensions "txt,log,csv,doc,xls" -computername $_
}| export-csv results.csv
```

Then you might use PowerShell to analyze the CSV file to see what files were compressed:

```
Import-Csv C:\compresults.csv | Sort CSNAME | group CSNAME | foreach {
    $computername=$_.name
    $_.group | group Extension | foreach {
        $_ | select @{Name="Computername";Expression={$computername}},
        @{Name="Extension";Expression={$_.name.ToUpper()}},
        @{Name="TotalSize";Expression={($_.group | Measure-Object filesize -sum).sum}},
        @{Name="TotalFiles";Expression={($_.group | Measure-Object).count}}
    }
}
```

This expression will give you a summary object for each server and file extension showing the total number of files and total size for that extension.

As with many Practical PowerShell examples, `Compress-FileTypes.ps1` has enhancement opportunities. There are a number of things you might want to add to the script. As I mentioned earlier, you may want to add support for `PSCredentials`. Complete logging of errors and failures might be useful. As written, there's no graceful error handling if a remote computer isn't reachable, so more robust error handling and trapping would be a great addition. Finally, although the script is designed to compress a file, you might want to make the script more flexible by allowing it to also uncompress a file. Here's a hint—there's an **Uncompress()** method.

If you need help enhancing this script or any PowerShell project, feel free to use the forums at ScriptingAnswers.com. ♦

Jeffery Hicks (MCSE, MCSA, MCT) is a Microsoft PowerShell MVP and Scripting Guru for SAPIEN Technologies. Jeff is a 17-year IT veteran specializing in administrative scripting and automation. Jeff is an active blogger, author, trainer, and conference presenter. His latest book is Managing Active Directory with Windows PowerShell: TFM (SAPIEN Press). Follow Jeff at [Twitter.com/JeffHicks](https://twitter.com/JeffHicks) and blog.sapien.com. You can contact Jeff at jhicks@sapien.com.

Exclusively Exchange

Exchange 2007 Services Break Down

by J. Peter Bruzzese

When Exchange 2007 installs, it relies on certain pre-existing services to function properly. In addition, it installs a bevy of new services. This behavior is by no means something new—previous Exchange versions did so as well; however, there are new services specific to Exchange 2007 and some that relate only to a particular server role. In this break down, we'll look into the dependent services, the Exchange services, and tools for working with these services to make sure they are up and running smoothly.

Initial Dependent Services and Features

To install Exchange within your environment, you need to have the infrastructure in place first. A functioning Active Directory (AD) and DNS, for example, precede the installation of Exchange. And then, on the Exchange Server system itself, you must ensure hardware and software specifications are met and important core services are running on the server.

You can quickly see which services are dependent on others (or which components depend on a service) by opening your Services console through Administrative Tools, opening the Properties of any service, and selecting the Dependencies tab (see Figure 1).



VISTA / OFFICE 2007 ROLLOUT

"The key to a smooth **Vista / Office 2007 ROLLOUT** is **ClipTraining**."

- Chris Nichols - Director of IT, Tax Education Support of Iowa

When you give your team the latest software; give them the latest training. ClipTraining supports your team and creates a confidence unattainable with traditional classroom and video training.

LEARN WHAT YOU NEED...
**WHEN YOU
NEED IT.**



www.ClipTraining.com

Email: **info@ClipTraining.com**

Phone: **1-888-611-CLIP (2547)**

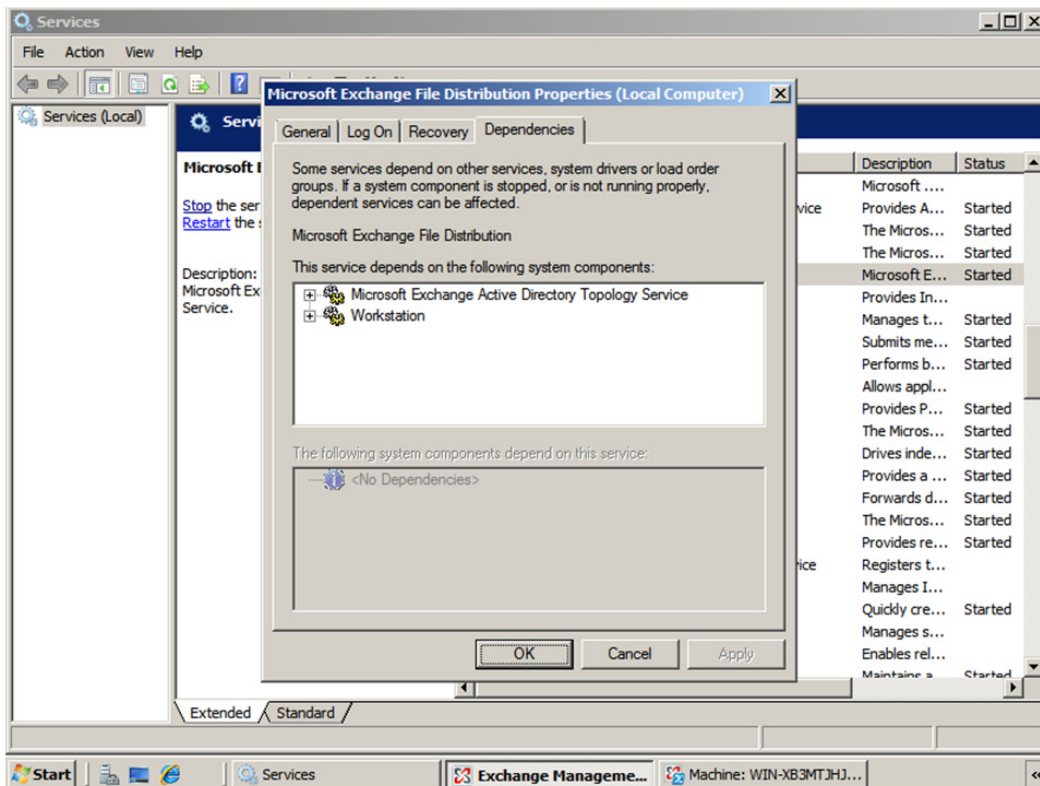


Figure 1: Dependent services.

At times, Exchange services are interdependent on each other; other times, they rely on important system components. For example, on an Edge Transport server, the Microsoft Exchange File Distribution Properties service says it depends on the Workstation services. Some of the services necessary prior to installation are the WWW Publishing Service, Remote Procedure Call (RPC), and Windows Management Instrumentation (WMI). As you research, you will find others.

Exchange Services

The following sections highlight services installed by Exchange Setup; the lists are broken down by server role. For a description of any of these services, select them in Services and view the description.

Mailbox Server Role

- Microsoft Exchange Active Directory Topology (Required)
- Microsoft Exchange Information Store (Required)
- Microsoft Exchange Mail Submission Service (Required)
- Microsoft Exchange Mailbox Assistants (Required)
- Microsoft Exchange Monitoring (Required)
- Microsoft Exchange Replication Service (Required)
- Microsoft Exchange Search Indexer (Optional)
- Microsoft Exchange Service Host (Required)
- Microsoft Exchange System Attendant (Required)
- Microsoft Exchange Transport Log Search (Optional)
- Microsoft Search (Exchange Server) (Optional)

Client Access Server Role

- Microsoft Exchange Active Directory Topology (Required)
- Microsoft Exchange File Distribution Service (Required)
- Microsoft Exchange IMAP4 (Optional)
- Microsoft Exchange Monitoring (Optional)
- Microsoft Exchange POP3 (Optional)
- Microsoft Exchange Service Host (Required)

Hub Transport Server Role

- Microsoft Exchange Active Directory Topology (Required)
- Microsoft Exchange EdgeSync (Optional)
- Microsoft Exchange Anti-spam Update (Optional)
- Microsoft Exchange Monitoring (Optional)
- Microsoft Exchange Transport (Required)
- Microsoft Exchange Transport Log Search (Optional)

Unified Messaging Server Role

- Microsoft Exchange Active Directory Topology (Required)
- Microsoft Exchange File Distribution Service (Required)
- Microsoft Exchange Monitoring (Optional)
- Microsoft Exchange Speech Engine (Required)
- Microsoft Exchange Unified Messaging (Required)

Edge Transport Server Role

- Microsoft Exchange ADAM (Required)
- Microsoft Exchange Credential Service (Required)
- Microsoft Exchange Anti-spam Update (Optional)
- Microsoft Exchange Monitoring (Optional)
- Microsoft Exchange Transport (Required)
- Microsoft Exchange Transport Log Search (Optional)

Working with Services

There are several ways to determine whether your services are up and running. Certainly through the Services console, you can see and adjust the status. Also, from within the Exchange Management Shell, you can run a test cmdlet. If you type

```
Test-ServiceHealth
```

the system provides information about the role running on the server, whether the required services are running, and which services are running or not running (see Figure 2). This method is a great way to determine the cause of a problem during troubleshooting.

Copyright Statement

© 2009 Realtime Publishers, all rights reserved. This eJournal contains materials that have been created, developed, or commissioned by, and published with the permission of, Realtime Publishers (the “Materials”) and this work and any such Materials are protected by international copyright and trademark laws.

THE MATERIALS ARE PROVIDED “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. The Materials are subject to change without notice and do not represent a commitment on the part of Realtime Publishers or its sponsors. In no event shall Realtime Publishers or its sponsors be held liable for technical or editorial errors or omissions contained in the Materials, including without limitation, for any direct, indirect, incidental, special, exemplary or consequential damages whatsoever resulting from the use of any information contained in the Materials.

The Materials (including but not limited to the text, images, audio, and/or video) may not be copied, reproduced, republished, uploaded, posted, transmitted, or distributed in any way, in whole or in part, except that one copy may be downloaded for your personal, non-commercial use on a single computer. In connection with such use, you may not modify or obscure any copyright or other proprietary notice.

The Materials may contain trademarks, services marks and logos that are the property of third parties. You are not permitted to use these trademarks, services marks or logos without prior written consent of such third parties.

Realtime Publishers and the Realtime Publishers logo are registered in the US Patent & Trademark Office. All other product or service names are the property of their respective owners.

If you have any questions about these terms, or if you would like information about licensing materials from Realtime Publishers, please contact us via e-mail at info@realtimepublishers.com. ♦