



**Ensuring  
High Availability with  
Microsoft Exchange Server**

**By Jerry Cochran  
Kieran McCorry  
Evan Morris  
Daragh Morrissey  
Tony Redmond  
Paul Robichaux**



## Contents

<b>Chapter 1: Exchange Server 2003 Clusters</b> .....	<b>1</b>
<i>By Tony Redmond</i>	
Understanding Clusters and How Exchange Operates in Them .....	1
Using Exchange 2000 in Clusters .....	3
What's Changed in Exchange 2003 .....	5
<i>Sidebar: Geoclustering</i> .....	7
<i>By Paul Robichaux</i>	
<i>Sidebar: Higher Availability with GeoCluster</i> .....	8
The Microsoft Experience .....	9
Don't Be a Fool .....	10
<b>Chapter 2: 8 Ways to Improve Your Exchange Cluster</b> .....	<b>11</b>
<i>By Daragh Morrissey</i>	
1. Training .....	11
2. Planning .....	12
3. Redundancy, Redundancy, Redundancy .....	14
4. Stabilize Your Windows Infrastructure .....	14
5. Configuration .....	15
6. Security .....	17
7. Failovers .....	17
8. Tips for Exchange Service Packs .....	20
Better Clusters .....	21

**Chapter 3: Get a Grip on Exchange Data Management . . . . . 22**

*By Kieran McCorry*

Juggling Constraints . . . . .	22
Dealing with Server-Based Data . . . . .	23
Managing User-Maintained Data . . . . .	24
Better Backup and Restore . . . . .	25
<i>Sidebar: Putting Exchange Data Management in Context</i> . . . . .	25
All About Archiving . . . . .	26
<i>Sidebar: Data Management Challenge: How Did We Get Here?</i> . . . . .	27
Get Your Act Together . . . . .	28

**Chapter 4: Exchange Server 2003 and VSS:  
A Way to Improve Recoverability and Availability . . . . . 29**

*By Jerry Cochran*

Volume Snapshot and Volume Cloning Overview . . . . .	29
The VSS Foundation . . . . .	31
<i>Sidebar: Better Backup through Replication</i> . . . . .	34
Exchange 2003 Support for VSS . . . . .	34
Exchange 2003 Backups Using VSS . . . . .	34
Exchange 2003 Recovery Using VSS . . . . .	34
Implications for Exchange Administrators . . . . .	36

**Chapter 5: Exchange Availability Tips & Tricks . . . . . 37**

Exchange Server Availability: The Big Picture . . . . .	37
<i>By Jerry Cochran</i>	
Replication-Based Recovery Servers: Worth the Effort? . . . . .	38
<i>By Jerry Cochran</i>	
Data Replication Technology for Your Exchange Deployments . . . . .	39
<i>By Jerry Cochran</i>	
Tips for Maintaining Messaging Availability . . . . .	40
<i>By Paul Robichaux</i>	
The 7 Habits of Highly Available Exchange Servers . . . . .	41
<i>By Evan Morris</i>	

## Chapter 1

# Exchange Server 2003 Clusters

*By Tony Redmond*

The history of Microsoft Exchange Server clusters is one of peaks and valleys, with cycles of enthusiasm generated by new releases followed by depression as the releases don't work out so well in practice. Exchange Server 5.5 first supported Windows clusters, but the clusters were expensive to deploy and were limited to two nodes configured in an active-passive cluster. Exchange 2000 Server promised clusters spanning as many as four active nodes. However, Microsoft has been forced to rescind that promise and now requires you to maintain a passive node because of memory-fragmentation problems. In addition, you can deploy four-node Exchange 2000 clusters on only Windows 2000 Datacenter Server, so the solution is expensive. The net result is that clusters represent a small percentage of the hundreds of thousands of Exchange servers deployed today. No one is willing to say exactly how many clusters are in production, but anecdotal evidence suggests that fewer than 2 percent of all the deployed Exchange servers are in clusters.

On the surface, clusters seem to be an extremely effective way to achieve high degrees of robustness and reliability. Indeed, UNIX and OpenVMS administrators have been deploying clusters for these reasons for years. So, why haven't Windows administrators been deploying Exchange 2000 and Exchange 5.5 clusters? The reasons why include added complexity, Exchange components that can't run on clusters, the lack of third-party product availability, memory fragmentation, and high costs.

For clusters to work well, the email application and the OS must join forces. Microsoft promised better clusters with Windows Server 2003 and Exchange Server 2003, but has the company keep that promise? To answer this question, you need to know how clusters work, how Exchange in general operates in a cluster, how Exchange 2000 specifically operates in a cluster, and what improvements Exchange 2003 offers.

## Understanding Clusters and How Exchange Operates in Them

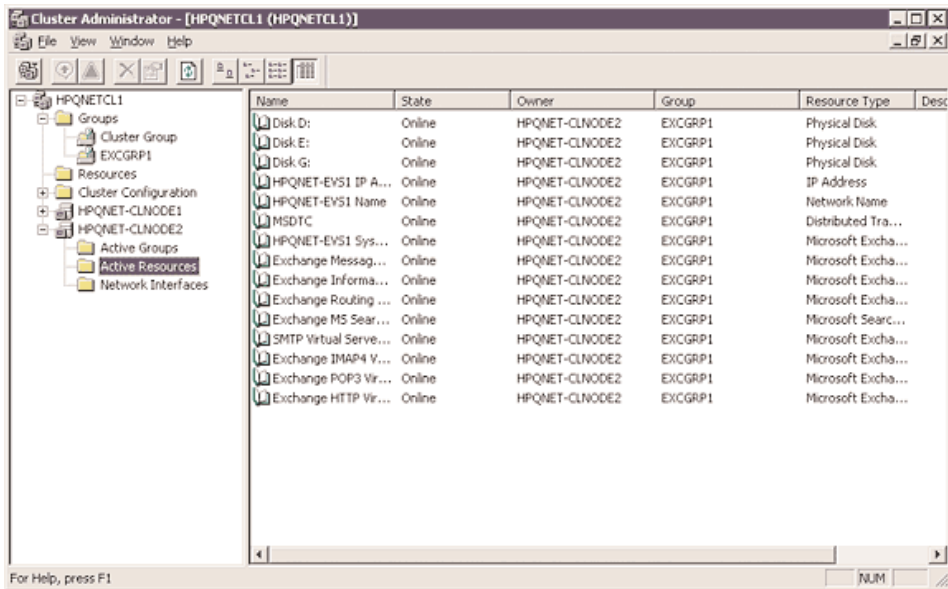
The process of setting up and configuring hardware and software for clusters is more complex than that for standard servers. Clusters typically boast multiple NICs. You need at least one NIC for the public network and one NIC for the cluster "heartbeat," which is the network signal between nodes that lets the nodes know that the cluster is alive and well. Clusters use shared storage instead of direct-connected drives because services depend on being able to move data between nodes when problems occur—and the services can't move the data if the data is restricted to a specific server.

Managing clusters differs from managing standard servers. Instead of controlling the set of services for Exchange or other applications through the Computer Management console, you manage them through the Cluster Administrator console, which Figure 1 shows. In this example, Cluster Administrator shows a set of Exchange services running on a cluster. The console shows the

## 2 Ensuring High Availability with Microsoft Exchange Server

additional resources that combine to form an Exchange virtual server, such as the disks, IP address, and network name.

**Figure 1**  
*Cluster Administrator console*



The concept of virtual servers is crucial to clusters. Exchange runs on a cluster as one or more virtual servers. Each virtual server represents the set of resources (e.g., disks, a network name, the Store) that Exchange needs to provide services to users. Exchange virtual servers run on physical nodes within the cluster. The virtual servers manage the data in mailbox and public stores, which are gathered into storage groups (SGs). An SG is the basic unit of storage for Exchange clusters. If a physical node fails and the cluster has to move work within the cluster, the cluster distributes the SGs from the failed server to other nodes rather than moves individual stores. After the failover, the SGs come under the control of the Exchange virtual server running on that physical node.

After you understand the basic concepts of clusters and how Exchange operates in clusters, you have to face the fact that not all Exchange components can run on a cluster. The reason why is simple. In some cases, the Exchange component is old and wasn't designed to run on anything other than a standard server. Because the component is old and perhaps not needed by the majority of Exchange installations, Microsoft never upgraded the component to support clusters. In other cases, the component is used only in specific circumstances (e.g., for interoperability between Exchange 2000 and Exchange 5.5 servers), so that component doesn't need to support clusters in the long term. Table 1 lists the optional Exchange components and the degree of cluster support for those components.

**TABLE 1: Support of Optional Exchange Components on Clusters Component Status**

Message Transfer Agent (MTA)	The MTA can be active in only one virtual server within the cluster.
Network News Transfer Protocol (NNTP)	NNTP isn't supported largely because of the problems associated with reconfiguring news feeds after a cluster transition.
Microsoft Mail (MS Mail) connector	One instance can be active in a cluster. The dependency is on the post office name, which you must manually reconfigure after a failover.
IM	Not supported (available in Exchange 2000 only).
Site Replication Services (SRS)	Not supported.
Lotus cc:Mail connector	One instance can be active in a cluster. The dependency is on the post office name, which you must manually reconfigure after a failover.
Lotus Notes connector	Not supported.
Novell GroupWise connector	Not supported.
IBM Professional Office System (PROFS) connector	Not supported.
IBM SNADS connector	Not supported.
Active Directory Connector (ADC)	Not supported.
Key Management Service (KMS)	Not supported (available in Exchange 2000 only).
Chat Service	Not supported (available in Exchange 2000 only).
Exchange Conferencing Server	Not supported (available in Exchange 2000 only).

In the past, some Independent Software Vendors (ISVs) didn't support clusters because these vendors preferred to concentrate on the largest market: support software for standard Exchange servers. The lack of add-on software for clustered Exchange servers caused a problem if organizations wanted to deploy the same antivirus, antispam, and backup software and messaging connectors across both standard and clustered servers. Fortunately, this situation has improved tremendously since Exchange 2000 first appeared, and you now have a reasonable choice of add-ons for clustered servers.

## Using Exchange 2000 in Clusters

Exchange 2000 was the first release to support active-active clusters, meaning that every node in the cluster supports an Exchange virtual server at the same time. Unfortunately, active-active clusters ran into virtual-memory fragmentation problems within the Store, and this problem has prevented Exchange 2000 from scaling up as much as it should on a cluster.

As Exchange 2000 runs, Windows allocates and deallocates virtual memory to the Store to map mailboxes and other structures. Virtual memory is sometimes allocated in contiguous chunks, such as the approximately 10MB of memory that's necessary to mount a database. However, as time goes by, providing the Store with enough contiguous virtual memory becomes difficult because the memory becomes fragmented. In concept, this fragmentation is similar to the fragmentation that occurs on disks and usually doesn't cause too many problems, except for cluster state transitions.

During a cluster state transition, the cluster must move the SGs that were active on a failed node to one or more other nodes in the cluster. SGs consist of sets of databases, so the Store has to initialize the SGs, then mount the databases so that users can access their mailboxes. You can track this activity through event ID 1133 in the Application event log. On a heavily loaded cluster, the Store might not be able to mount the databases because not enough contiguous virtual memory is

## 4 Ensuring High Availability with Microsoft Exchange Server

available, in which case you'll see an event such as event ID 2065. Thus, you encounter a situation in which the cluster state transition occurs but the Store is essentially brain-dead because the databases are unavailable. This kind of situation occurs only on heavily loaded systems, but consolidating servers and building big, highly resilient systems are prime driving factors for considering clusters in the first place.

After receiving problem reports, Microsoft analyzed the situation and realized a problem existed when running in active-active mode. Microsoft began advising customers to limit cluster designs and limit the number of concurrently supported clients to 1000 in Exchange 2000, 1500 in Exchange 2000 Service Pack 1 (SP1), and 1900 in Exchange 2000 SP3 and SP2.

The client numbers that Microsoft recommends are based on Messaging API (MAPI) loads. Because MAPI is the most functional and feature-rich protocol, MAPI clients usually generate the heaviest workload for Exchange. Microsoft Outlook Web Access (OWA) clients generate much the same type of demand as MAPI clients. However, other client protocols (e.g., IMAP4, POP3) typically generate lower system demand and can result in a lesser workload for the server. So, organizations might be able to support more client connections than the number of clients Microsoft recommends before the virtual-memory problem appears.

Exchange 2000 SP3 includes a new virtual-memory allocation scheme for the Store. This new scheme changes the way in which Windows allocates and deallocates memory. Experience to date demonstrates that servers running SP3 encounter fewer memory problems on high-end clusters. Thus, Microsoft highly recommends that organizations with large clusters upgrade to Exchange 2000 SP3 or, even better, upgrade the OS to Windows 2003 and deploy Exchange 2003, which better manages memory.

The problems with virtual-memory management have forced Microsoft to express views about how to set up active clusters. Essentially, Microsoft's advice is to keep a passive node available whenever possible, meaning that a two-node cluster should run in active-passive mode and a four-node cluster should have three active nodes and one passive node.

Virtual memory begins to decline as the load on a cluster grows. Exchange logs event ID 9582 when less than 32MB of available memory is present, then flags the same event when no contiguous blocks of virtual memory larger than 16MB exist inside the Store. After Exchange reaches this threshold, the cluster can become unstable and stop responding to client requests, and you must reboot. You might also see event ID 9582 in two other situations:

- Event ID 9582 might appear immediately after a failover to a passive node if the passive node previously hosted the same virtual server. Each node maintains a stub store.exe process, and the structures within the process might have already been fragmented, leading to the error. If this error occurs, you can transition the virtual server to another node in the cluster, then restart the server that has the fragmented memory. If a passive node isn't available, you have to restart the active node. Exchange 2000 SP3 generates far fewer problems of this nature, so you're unlikely to see event ID 9582 triggered under anything but extreme load.
- Incorrect use of the /3GB switch in the boot.ini file can generate event ID 9582. If you're hosting Exchange 2003 on a Windows 2003 server that has more than 1GB of physical memory, you should set the /3GB switch and its associated /UserVa= switch in the boot.ini file so that Windows 2003 has a better balance in its allocation of resources between kernel- and user-mode

## GeoCluster for Exchange Server Clusters

Based on NSI Software's proven Double-Take replication technology, GeoCluster extends the capabilities of Microsoft Cluster Service (MSCS) to create a stretch cluster. While MSCS and clustering in general provides a high level of application availability, MSCS clusters maintain only a single copy of the data and only provide failover within a single site. Built on top of MSCS, GeoCluster offers added data protection, assuring continual access to data and applications across sites. To learn more, please visit: <http://www.nsisoftware.com/pro/geocluster/>

memory. For more information about these switches, see the Microsoft article “[XADM: Event Viewer Log Entries Cite Virtual-Memory Fragmentation on an Exchange 2000 Server](#)” and “[XADM: Using the /Userva Switch on Windows 2003 Server-Based Exchange Servers](#).”

## What's Changed in Exchange 2003

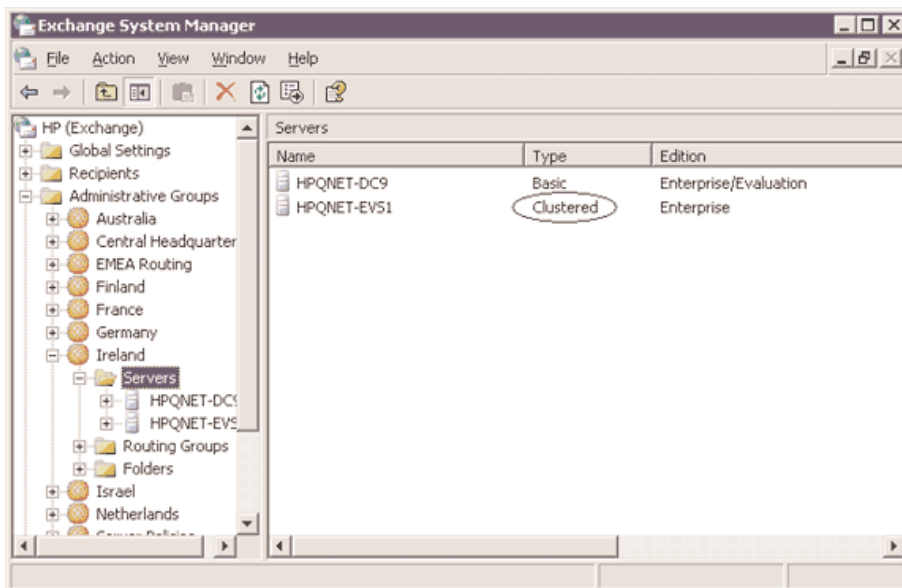
Although you can deploy Exchange 2003 on Windows 2000 servers, Microsoft is fond of saying that Windows 2003 and Exchange 2003 are better together and deliver the optimum functionality because they're designed to work as a team. This statement is true in many respects but is especially true for clusters. I wouldn't recommend deploying an Exchange cluster on anything but Windows 2003 servers. Here are the major improvements in Windows 2003 and Exchange 2003 clusters:

- With Windows 2003 and Exchange 2003, you can configure eight-node clusters. Compared with four nodes, eight nodes provide a lot more flexibility in how you can lay out the servers within a cluster and the roles that the servers take. However, at least one node has to be passive if the cluster supports numerous clients, many connectors, or a heavy processing load. Clusters that support a small number of clients and perhaps run only one SG with a few databases on each active node can typically operate in a fully active mode because virtual-memory fragmentation is less likely to occur.
- The dependency on Datacenter is gone, so you can now deploy clusters without the additional expense that Datacenter introduces.
- Windows 2003 and Exchange 2003 better control virtual-memory fragmentation, which increases the number of MAPI clients that a cluster can support. Windows 2003 and Exchange 2003 also make better use of large amounts of memory (i.e., more than 1GB) when that memory is available to a server. No formal testing has yet established how many concurrent MAPI clients Exchange 2003 supports before it runs into the virtual-memory fragmentation problem, but the fact that Microsoft has deployed clusters that support 4000 mailboxes per node reveals that the limit is high. If a passive node is always available in an active-passive configuration, clusters can support numerous users per active node—perhaps as many as 5000 mailboxes per node. The exact figure depends on the system configuration, the load that the users generate, the types of clients used, and careful monitoring of virtual memory on the active nodes as they come under load.
- You can use drive mount points (otherwise known as NTFS mounted drives) to eliminate the Win2K—Exchange 2000 restriction on the number of available drive letters, which limits the number of available disk groups in a cluster. This improvement is important when you deploy more than 10 SGs across multiple cluster nodes.

## 6 Ensuring High Availability with Microsoft Exchange Server

- Because of Exchange 2003's new resource-dependency model and some tweaks in the way that Exchange 2003 manages failover, Exchange 2003 appears to be faster than Exchange 2000 at transitioning SGs from failed servers to active nodes when problems occur.
- Microsoft made tweaks to Exchange 2003's management interfaces to make life easier for administrators. For example, as Figure 2 shows, the Exchange System Manager (ESM) console now displays details of server types, so you know immediately whether a server is running on a cluster.
- Assuming that you use appropriate hardware and backup software, you can use Windows 2003's Volume Shadow Copy Service (VSS) API to take hot snapshot backups. This improvement is crucial because clusters can't attain their full potential if administrators limit the size of the databases. Limiting the size of databases limits the number of mailboxes that a cluster can host. However, vendors have been slow to ship VSS-compliant products, so don't depend too much on this feature until you see solid products appear.
- The Recovery Storage Group feature lets administrators recover from individual database failures quickly and without having to deploy dedicated recovery servers. This feature is also available when you deploy Exchange 2003 on Win2K servers.

**Figure 2**  
*Exchange System Manager console*



VSS and the Recovery Storage Group aren't cluster-specific features. However, both contribute to higher levels of service availability and reassure administrators who worry that consolidating many standard servers into a large cluster might be putting all their eggs into one basket.

As you can see, these improvements address many of the reasons why Windows administrators haven't implemented clusters. Some of the other reasons why administrators haven't considered

## Geocustering

*By Paul Robichaux*

I've recently seen a surge of interest in an exotic technology. I'm not talking about High-Definition TV (HDTV)—I'm talking about geocustering: building clusters of servers that are in physically separate locations. For example, a company in a metropolitan area such as Detroit could run Exchange Server on a cluster, placing one cluster server at headquarters in Dearborn, Michigan, and another at a factory in Melvindale, Michigan. The company could connect both servers to a storage unit at a facility somewhere else in the area.

Although geocustering can be prohibitively expensive, Exchange does support the technology, which seems to be drawing more and more attention. The appeal of geocustering from a disaster-recovery or business-continuance standpoint is obvious. By physically separating cluster nodes, you gain a high degree of protection from events such as fires, building collapses, and earthquakes. How does geocustering work? Well, the bottom line is optic fiber. Simply put, geocustering extends the length of the optic fiber run used to build a Fibre Channel Storage Area Network (SAN). Of course, a lot of rather complicated technology is involved. For example, long fiber runs require amplifiers; most geocustering implementations use multiple channels (e.g., 32 channels of 32MBps each) which in turn require switching and routing equipment.

You can implement geocustering in one of two ways. You can split the cluster so that each node is in a separate location and all nodes share a common storage system in yet another location. In this case, the limiting factor is the latency that occurs while packets traverse the fiber between each node and the storage unit. And of course, this design is vulnerable to storage failures. The more sophisticated (and expensive) design uses local Fibre Channel SAN storage at each cluster node; the SAN hardware synchronously replicates this storage to the counterpart nodes. The advantage of this method is that major SAN vendors understand synchronous SAN replication and have a lot of experience designing and deploying replicated systems. If you have problems with replication, your vendor will be the first line of support.

As interesting as it is, geocustering isn't for everyone and has significant limitations. First, Exchange doesn't recognize the physical design of the underlying cluster. If the cluster works (e.g., if it has acceptable latency and I/O policies), Exchange will work. If the cluster doesn't work or works only intermittently, the same will be true of Exchange.

Second, geocustering is incredibly expensive. I've seen geocuster implementations that cost several million dollars. All that fiber (and its support infrastructure) costs money, and the vendor-specific replication solutions you need to use are expensive as well. The additional latency caused by the length of the fiber run means that Exchange can't handle as many concurrent I/O operations as it can on a traditional cluster or individual servers, so the number of users you can support on a geocuster shrinks as the fiber run length increases. In effect, you have to spend much more to support the same population of users.

Third, geocustering requires extraordinary caution in design. The I/O profile for Exchange is very demanding, and an improperly designed cluster will provide an endless source of frustrating, difficult-to-fix problems caused by I/O latency. The Windows Server Catalog lists several Windows Hardware Quality Labs (WHQL)-certified geocuster implementations (under Cluster Solutions, Geographically Dispersed Cluster Solution); choose one of these if you want to deploy a ready-made geocuster rather than build your own. You should also work with your SAN vendor to make sure that it will stand behind its recommended design as suitable for (and supported with) Exchange.

The bottom line is that Exchange supports geocustering but that few applications justify its present-day cost or limitations. Still the technology is very cool—almost as cool as HDTV—and might be looking forward to a bright future.

## Higher Availability with GeoCluster

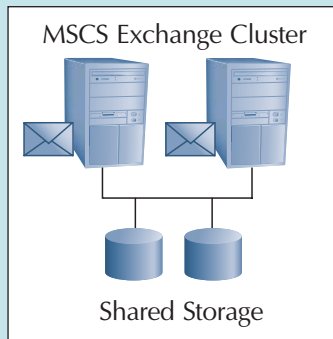
For those who choose to implement Microsoft Cluster Server (MSCS) to take advantage of its strengths in providing local application availability, NSI can address the limitations of MSCS by offering solutions based on GeoCluster. GeoCluster is exclusively designed for MSCS environments to remove its inherent architectural issues while maintaining its superior server, application, and services monitoring and failover capabilities.

To eliminate the risk of a shared storage failure shutting down your entire email system, GeoCluster combined with MSCS provides a solution for redundancy of the storage. GeoCluster allows each of the clustered email and BES servers to have its own copy of the email data on its own storage device. Any failure on the active clustered node would still be handled by the cluster services, but now the storage is equally fault-tolerant and will no longer be a single point of failure.

To protect against a building-wide crisis which could negate an entire traditional MSCS cluster, the GeoCluster nodes have the ability to be a significant distance apart from each other. As GeoCluster requires just a standard IP LAN or VLAN between cluster members, distances are only limited to the quality of the connections to meet the heartbeat requirements of MSCS. Now, should an entire building or even region suffer an outage, a cluster member at a remote site hundreds of miles away can immediately become the active node, taking over for the failed server. Since each node has its own local copy of the data, the cluster would continue to service clients throughout the environment from the now-active remote cluster member. Gone are the worries about loss of email due to just a single copy of the data.

Because GeoCluster runs on top of and leaves all monitoring and failovers to MSCS, it can be seamlessly integrated with most any cluster-aware application, providing some of the highest possible levels of application availability.

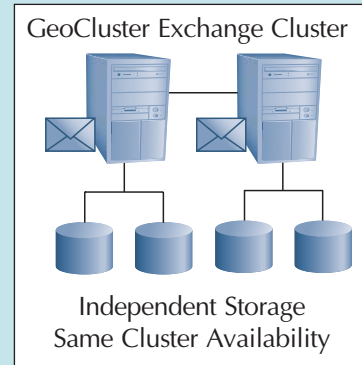
**GeoCluster can provide a geographically dispersed cluster, protecting against local and regional failures while maintaining the application and server availability offered by MSCS.**



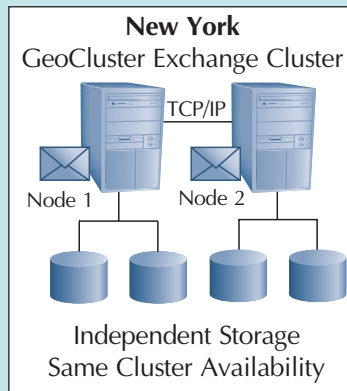
Before GeoCluster  
Single Point of Failure

**GeoCluster employs a “shared nothing” architecture, providing protection against a disk or other hardware failure.**

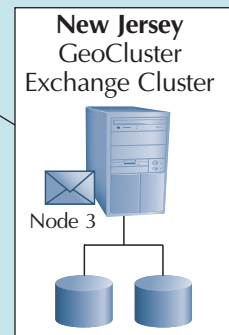
After GeoCluster  
No Single Point of Failure



Independent Storage  
Same Cluster Availability



Independent Storage  
Same Cluster Availability



clusters are going away on their own. For example, as organizations complete their Exchange 2000 deployments or finish migrating from a legacy email system, they no longer need some of the older Exchange components that can't run on a cluster. Even some of the newer components introduced in Exchange 2000 that couldn't run in a cluster have been replaced by new products that can run in a cluster. All the core Exchange 2003 components can run in a cluster. Thus, problems with noncluster compliance are now likely to be found in only third-party products. And even that problem is going away because, as I mentioned previously, an increasing number of ISVs are offering add-ons for clustered servers.

## The Microsoft Experience

Surprisingly, Microsoft never used clusters in its Exchange deployment in the past, but that situation changed dramatically with the arrival of Exchange 2003 and Microsoft's server consolidation program. Microsoft has replaced its old set of standard Exchange servers with a new set of large clusters. The most interesting configuration is the data-center design, which supports 16,000 mailboxes spread across four Exchange virtual servers. The cluster consists of seven physical nodes, four of which handle the load that the four mailbox servers generate. Another server is passive, waiting to spring into action should one of the mailbox servers fail. The mailbox servers (and the passive node) boast substantial power: They're HP ProLiant DL580G2 models with quad 1.9GHz Intel Xeon III processors, 4GB of memory, and a 400MHz front-side bus. Microsoft has enabled hyperthreading on these servers and reports that this feature provides an increase of about 20 percent of CPU headroom. Microsoft follows its own advice and tunes the servers by setting the /3GB switch and setting the /Userva= switch to 3030 (this value is in megabytes) in the boot.ini file.

The two remaining servers—each of which is an HP ProLiant DL380G2 model with two CPUs and 2GB of RAM—handle backup and other administrative functions. Because they're auxiliary servers and don't host Exchange, these servers have lower-specification configurations.

Microsoft's standard mailbox quota went from 100MB to 200MB, although considerable variation exists in actual quotas based on business demand. Not surprisingly, Microsoft has a lot of mailbox data to back up daily. The best backup solutions can stream data to tape as fast as 100GB per hour, but this rate isn't satisfactory for a 16,000-mailbox cluster. For a cluster this size, you want the mailbox servers delivering the best possible response to users and not handling the load that tape backups generate. Microsoft's operations team solved the backup problem by first backing up the disks to volumes that are temporarily available to the mailbox servers. After the disk backups are finished, the volumes are failed over to the auxiliary nodes and moved to the control of the two auxiliary servers, which copy the backup to tape. Moving disks between servers in this fashion demonstrates how to use cluster features to solve administrative problems. In the future, Microsoft plans to deploy VSS-enabled hot snapshot backups in addition to hot snapshot backups. (Hot snapshot backups are designed to complement, not replace, tape backups.)

An HP StorageWorks Enterprise Virtual Array 5000 (EVA5000) Storage Area Network (SAN) manages all the storage. This SAN is an important contributor to Microsoft's large cluster because of its redundancy and management features. Because SGs, transaction logs, and SMTP work directories all require drives, Microsoft's large cluster has numerous drives. Microsoft heavily uses mount points to get around the drive-letter limitation that would otherwise render these drives almost impossible to configure in a satisfactory manner.

## 10 Ensuring High Availability with Microsoft Exchange Server

I haven't heard of similar clusters running in production, so Microsoft might take the blue ribbon for large Exchange clusters with this design. Microsoft reckons that it achieves a better service level with the cluster than it achieved with standard servers. The company says it reached the "four nines" territory (i.e., 99.99 percent availability). However, this feat hasn't been independently audited because Microsoft deployed the cluster with beta versions of Exchange 2003, then continually upgraded the software until the final release. Getting anywhere close to such an uptime record would be remarkable.

### **Don't Be a Fool**

Only fools rush in and deploy clusters. Administrators who plunge into cluster deployment without investing the necessary time to research, plan for, and design clusters generally encounter problems. Exchange 2003 clusters are more complex than standard Exchange servers, and experience demonstrates that you must carefully set up and manage Exchange 2003 clusters to generate the desired levels of uptime and resilience. Only those administrators who take the time up front to properly design clusters and successfully manage those clusters will likely achieve their desired results.

The improvements in Exchange 2003 and Windows 2003 are helping make Exchange clusters a viable option. The early reports of successful deployments of Exchange 2003 clusters, including Microsoft's deployment, are encouraging. The challenge for Microsoft now is to continue driving complexity out of clusters so that installing and managing clusters is as easy as installing and managing standard servers. That day is not yet here.

Still, I remain positive about clusters. Clusters do a fine job, provided that administrators carefully plan cluster configurations and then appropriately manage the clusters. However, the road to clusters has been bumpy, and Microsoft didn't keep its promise about clusters in the past. Work is continuing to improve clusters, but in the interim, if you're interested in clustering Exchange servers, you need to consider all options before making a final decision.

## Chapter 2

# 8 Ways to Improve Your Exchange Cluster

*By Daragh Morrissey*

Clustering your Exchange Server 2003 or Exchange 2000 Server systems can provide the high availability that's so important for a business-critical email application. If you're considering clustering Exchange, you can take several steps to improve your deployment, such as getting cluster-specific training, planning ahead, building extra redundancy into the cluster, and deploying a solid Windows infrastructure before building the cluster.

## 1. Training

Clusters are more complex than single-server Exchange deployments, so you need training that focuses on clustering concepts and operations such as the quorum, failover/failback operations, and using Cluster Administrator. You also need to understand the requirements of clustering-hardware configurations. For example, shared storage must be accessible to all nodes, so you must correctly configure any hardware that manages storage connections (e.g., array controllers, Storage Area Network—SAN—switches) to avoid contention or corruption of databases. Attention to detail is necessary to ensure that you correctly install Windows before installing Exchange and that you install and configure Exchange in the correct sequence to work on a cluster—a process that differs significantly from installing Exchange on one server. For example, to install a two-node, active/passive cluster, you need to perform the following tasks in sequence:

- Run Exchange Setup on node 1.
- Run Exchange Setup on node 2.
- Create a cluster group for the Exchange Virtual Server (EVS).
- Move disk resources that the EVS will use to the Exchange cluster group.
- Create the resources that the EVS requires (e.g., Microsoft Distributed Transaction Coordinator—MSDTC—an IP Address resource, a Network Name resource).
- Create a System Attendant resource for the EVS. As part of this step, you must supply the name of the EVS, the administrative group and routing group in which the EVS will reside, and a shared-storage folder in which Exchange will create and store its databases, transaction logs, and SMTP folders at installation.
- Cluster Administrator automatically creates Exchange cluster resources for the EVS (e.g., the Information Store—IS; HTTP and IMAP servers for the virtual server; the required dependencies for the IP Address and Network Name resources).
- Use Exchange System Manager (ESM) to relocate the Exchange components (i.e., databases, transaction logs, and SMTP folders) to shared-storage drives or folders, according to established best practices. Exchange needs to be able to access these resources from each node as the EVS fails over.

## 12 Ensuring High Availability with Microsoft Exchange Server

You need to have a firm grasp of Microsoft Cluster service clustering concepts (see the Microsoft white paper “[Windows Clustering Technologies—An Overview](#)” for more information about these concepts). You also need to understand the limitations and constraints of running Exchange 2003 or Exchange 2000 on a cluster. For example, the Lotus Notes connector is unsupported on Exchange 2003 or Exchange 2000 clusters, as the Microsoft article “[Status of Exchange 2000 Server and Exchange Server 2003 Components on a Server Cluster](#)” explains. You must deploy additional standard servers to support any components that aren’t supported on clusters. Be aware that many third-party products fall into this category, and I definitely recommend against installing unsupported products on a cluster, given the complexity of clustering.

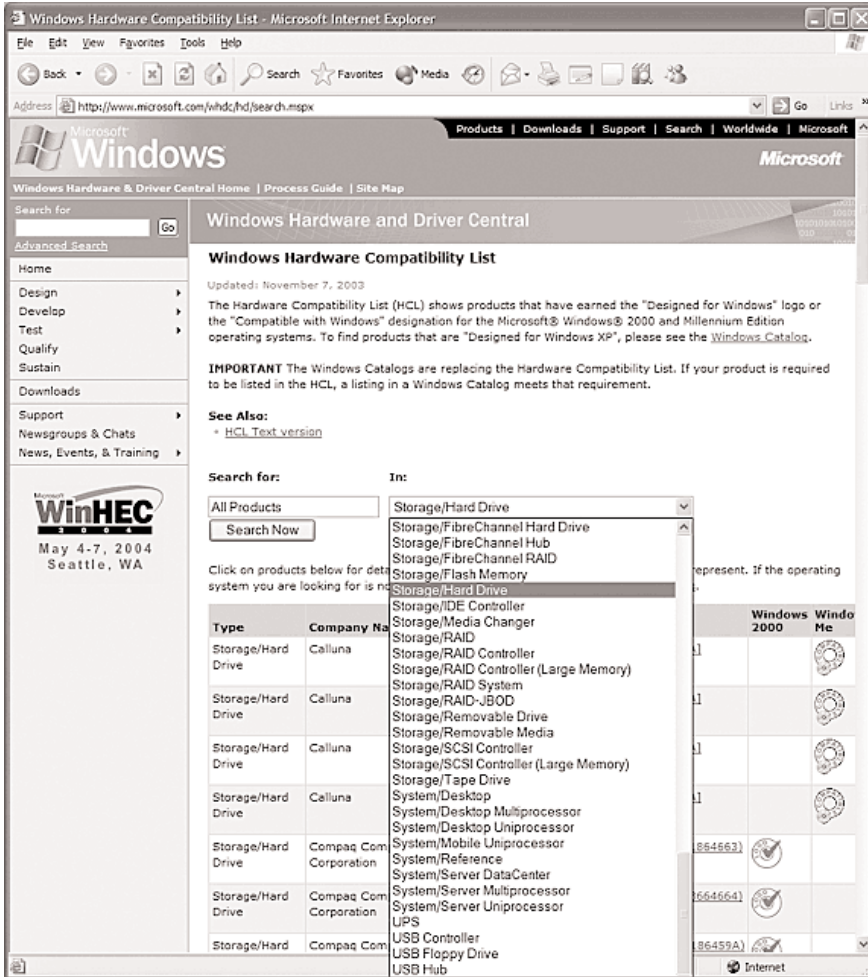
As important as training is, deploying production-quality test clusters that match the specifications of your production clusters can be prohibitively expensive because of the additional hardware necessary (compared with single-server deployments). Therefore, getting the necessary experience on a cluster before deployment is often difficult. To deploy low-cost clusters as training aids, consider using virtual server technology such as VMware or Microsoft Virtual Server. Windows Server 2003 introduces the concept of a local quorum, which lets you deploy single-node clusters. However, you can’t test failover and failback operations or rolling upgrades on this type of cluster.

### 2. Planning

Plan your cluster deployment carefully. A poorly implemented cluster can perform erratically and can increase downtime rather than maximize uptime. When planning, consider hardware specifications, node configuration, and the limitations of Exchange clustering memory management.

Hardware specifications. All the hardware components you use in a Windows 2000 cluster (e.g., disk drives, array controllers) must appear on the [Microsoft Hardware Compatibility List](#) (HCL). For Windows 2003 clusters, consult the [Windows Catalog](#), which replaces the HCL. If you implement hardware that isn’t on the HCL or in the Windows Catalog, Microsoft won’t support your configuration. The HCL lets you view devices by category, as Figure 1 shows; the Windows Catalog does the same.

**Figure 1**  
*The HCL lets you view devices by category*



**Node configuration.** Within a cluster, identically configure each node that can host an EVS, and configure all nodes with identical specifications for memory, disks, CPUs, and so forth. Although Microsoft's Operations and Technology Group (OTG) has implemented clusters with member nodes that have varying hardware configurations, I recommend identical node configurations within a cluster. Clusters are complex, and introducing cluster nodes with varying hardware specifications introduces additional complexity for Cluster Administrator. Also, implementing nodes of varying specifications can lead to inconsistent performance levels as EVSs move between nodes. The Microsoft article "The Microsoft Support Policy for Server Clusters and the Hardware Compatibility List" describes Microsoft's support policy for cluster hardware, and some hardware vendors offer Microsoft-certified and -supported packaged cluster solutions with standard hardware across nodes.

**Limitations of clustering memory management.** Active/active clusters on early Exchange 2000 deployments experienced virtual memory problems. The release version of Exchange 2000 supports a maximum of 1000 connections per node, Exchange 2000 Service Pack 1 (SP1) supports a maximum of 1500 connections, and SP2 supports a maximum of 1900 connections, as does Exchange 2003. (This limitation applies to clusters running on Windows 2003 or Win2K.) However, Microsoft's recommended cluster model for Exchange 2003 and Exchange 2000 is active/passive. Active/passive clusters don't have the same constraints on connections as active/active clusters have. Virtual memory fragmentation is less of a concern with active/passive clusters because the EVS can always start on a passive node.

In Exchange 2003, built-in functionality in ESM enforces active/passive clustering guidelines on clusters with more than two nodes: The number of EVSs you can create is  $(N-1)$ , where  $N$  represents the number of nodes in the cluster. ESM blocks you from creating EVSs that equal or exceed the number of nodes in the cluster. However, you still need to monitor memory fragmentation on active/passive clusters (and standalone servers) with many users. The Microsoft article "[XADM: Monitoring for Exchange 2000 Memory Fragmentation](#)" describes how to configure Performance Monitor to monitor virtual memory usage. For more information about planning Exchange 2003 clusters, see the Microsoft article "[Planning an Exchange 2003 Messaging System](#)."

### 3. Redundancy, Redundancy, Redundancy

The key design principle of any cluster deployment is to provide high availability. In the event of a hardware failure, a failover operation will move resources from the failed node to another node in the cluster. During Exchange failovers, users won't be able to access email folders for a brief time as resources go offline on the failed node and come online on the other node. For each node in a cluster, implement redundant hardware components to reduce the effect of hardware failures and thus avoid a failover. Examples of components in which you can implement redundancy are NICs, power supplies, and host bus adapters (HBAs) or array controllers.

NIC teaming lets multiple NICs act as one virtual NIC, allowing for the failure of one NIC, cable, or switch port (when you split the NICs over multiple network switches) without any interruption of service. I suggest you use NIC teaming on the public (client) network in a cluster, but Microsoft doesn't support teaming on the private (heartbeat) network in a cluster (as the Microsoft article "[Network Adapter Teaming and Server Clustering](#)" explains).

You can connect redundant power supplies to separate power distribution units; if you connect multiple power supplies to the same power distribution unit and that unit fails, power will be lost. Also connect power supplies to a UPS or use a UPS service to protect the datacenter hosting the cluster in the event of a power failure.

If you implement a SAN with your cluster, try to implement redundancy into the connections between your nodes and the SAN to handle failures of the HBA, fibre connections, or SAN switches, without the need to induce node failover. Take care to configure your storage; many of the problems with early Exchange 2000 cluster deployments were storage related.

### 4. Stabilize Your Windows Infrastructure

A stable and resilient Windows infrastructure is a crucial element of any Exchange cluster deployment. For an Exchange 2003 cluster, all Active Directory (AD) domain controllers (DCs) and Global Catalog (GC) servers must run Windows 2003 or Win2K SP3 or later; the DCs and GC servers that

support an Exchange 2000 cluster must run Win2K or later. Exchange 2003 and Exchange 2000 store configuration information in DCs and GC servers. Each DC holds a complete copy of all the objects in the DC's domain, plus a copy of objects replicated in the forestwide Configuration naming context (NC). A GC server holds a complete copy of all objects in the GC server's domain, plus partial copies of objects from all other domains in the forest. DSAccess is the Exchange component that locates and retrieves AD information from DCs and GC servers. From DCs, DSAccess retrieves information about Exchange entities such as administrative groups, connectors, Exchange system policies, and other servers in the Exchange organization. From GC servers, DSAccess retrieves user information such as email addresses and distribution group memberships.

To ensure the stability of the infrastructure that your cluster relies on, you can build redundancy into your Windows 2003 or Win2K organization by implementing multiple GC servers, DNS servers, and WINS servers.

**Multiple GC servers.** Implement two GC servers in the same Windows 2003 or Win2K site and LAN in which your Exchange clusters reside. If DSAccess can't contact a GC server, the System Attendant will fail, causing a failover because the IS resource has a dependency on the System Attendant resource. Implementing two GC servers mitigates the effect of a GC server going offline. If another GC server is available in the site, Exchange will use DSAccess to locate that GC server. If no GC server is available in the site, Exchange will try to use GC servers in other sites, resulting in downtime as DSAccess attempts to locate a GC server. When DSAccess locates a GC server, the System Attendant and IS resources will come back online and service will be restored. Outlook clients also use GC servers to query and retrieve the Global Address List (GAL). If a GC server goes offline, Outlook sessions also are adversely affected. Deploying Outlook 2003 with cached mode enabled can reduce the impact and visibility to users when a GC server goes offline. When working offline in cached mode with no connection to an Exchange server or a GC server, Outlook uses the Offline Address Book (OAB) on the client to access directory information. Some deployments use separate sets of GC servers for additional resilience: Back-end GC servers support Exchange servers, and front-end GC servers provide directory information to Outlook clients.

**Multiple DNS servers.** Windows 2003 and Win2K use DNS to resolve server names to TCP/IP addresses and to locate resources. If a DNS server goes offline and no secondary DNS server is available, Exchange can't resolve server names to TCP/IP addresses and might experience DSAccess errors and nondelivery of mail.

**Multiple WINS servers.** Windows 2003 and Win2K use WINS servers to resolve NetBIOS names to IP addresses; Windows NT networks use WINS for name resolution. The Exchange Server 2003 Deployment Guide states that WINS is necessary for deploying Exchange 2003 or Exchange 2000; Exchange Setup and the ESM use WINS.

## 5. Configuration

As I explain in Step 4, the stability of the Windows infrastructure underlying your cluster is key to the cluster's success. Properly configuring that infrastructure can also improve your cluster's performance. Important configuration steps include setting staggered boot delays for the cluster nodes, obtaining the applicable OS resource kit, and tuning memory.

**Set staggered boot delays.** When power returns after a power failure, each node in your cluster will attempt to access shared storage at the same time. To avoid this conflict, set your preferred passive node's boot delay to be longer than the active node's delay.

To access the delay setting on Windows 2000 servers, right-click My Computer and select Properties from the context menu. Click Advanced, then click Startup and Recovery. On Windows Server 2003 nodes, open the My Computer Properties dialog box, click Advanced, then click Settings under Startup and Recovery. In the Startup and Recovery dialog box, select the Display a list of Operating Systems for \_\_\_ seconds check box and enter the desired delay in the scroll box. Set the active node to 5 seconds and the passive node to 20 seconds. Alternatively, you can manually edit the boot.ini file on each node to implement a specific delay.

**Obtain the OS resource kit.** The Microsoft Windows 2000 Server Resource Kit contains valuable tools for cluster administrators. The resource kit provides approximately 300 utilities that aid management of Active Directory (AD) and Win2K servers, and several of these utilities are specific to clusters. Among the most important are dumpcfg.exe, which manages and records disk signature information; the Cluster Tool (clustool.exe), which backs up and restores cluster configurations; and clusrest.exe, which restores the quorum database. The Microsoft Windows Server 2003 Resource Kit tools include new and improved cluster utilities such as the Cluster Server Recovery Utility (clusterrecovery.exe), which you can use when restoring resource checkpoint files, replacing a failed disk, recovering from disk signature changes, or migrating cluster data to a different disk in the cluster; and the Cluster Diagnostics and Verification Tool (clusdiag.exe), which provides diagnostic tests to verify a cluster's functionality and which assists in reading the cluster log files.

Copy the tools to a standard folder on each cluster node as part of your cluster installation. Having the tools readily available can reduce the amount of time you need to diagnose a clustering problem if one arises. See <http://www.microsoft.com/windows2000/techinfo/reskit/default.asp> for more information about obtaining resource kits or resource kit tools.

**Tune memory.** Exchange 2000 servers that run on Win2K Advanced Server or Win2K Datacenter Server and that have more than 1GB of RAM require you to add the /3GB switch to the startup line, as the Microsoft article “[XGEN: Exchange 2000 Requires /3GB Switch with More Than 1 Gigabyte of Physical RAM](#)” explains. However, using the /3GB switch reduces the number of available Free System page table entries (PTEs), a situation that can cause performance problems—most noticeably the server's loss of network connectivity or blue screens. Microsoft recommends that you monitor the Free System PTE counter under the Performance Monitor's Memory object. If the value drops below 10,000, modify the HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management registry subkey's SystemPages entry, as the Microsoft article “[XADM: An Exchange 2000 Server with the '/3GB' Switch in the Boot.ini File May Lose Network Connectivity Under a Heavy Messaging Load](#)” describes.

Windows 2003, Standard Edition and Windows 2003, Enterprise Edition both support the use of the /3GB switch. However, both editions also support a new switch, /userva, which allows a custom environment size for the application virtual address space and lets you allocate PTEs from boot.ini (rather than from the registry). For Exchange 2003 servers that have more than 1GB of RAM, use /userva=3030 in conjunction with the /3GB switch. For more Exchange 2003 memory-tuning procedures, see the Microsoft article “[How to Optimize Memory Usage in Exchange Server 2003.](#)” Both Windows 2003 and Win2K require a reboot after you make these memory changes.

Another way to reduce virtual memory usage is to minimize the number of storage groups (SGs). Additional virtual memory is used when an SG is mounted, but additional databases within an existing SG have little effect on the amount of virtual memory used.

## Protecting Exchange with Replication

Protecting data has always been important. However, given the heightened awareness around national security and protecting important human and physical assets, having solutions that are cost-effective, hardware independent and scalable is something every IT manager should seriously consider. For many of us, that is Microsoft Exchange. So then, the question becomes "How do I ensure that my Exchange environment is always protected?" Please take a look at how NSI Software's Double-Take solutions for Microsoft Exchange Server is a complete data protection and disaster recovery solution. [http://www.nsisoftware.com/how/nsi\\_exchange\\_protect.asp?eve=657](http://www.nsisoftware.com/how/nsi_exchange_protect.asp?eve=657)

## 6. Security

You need to lock down your Exchange cluster to prevent the spread of W32.Blaster.Worm, the Nimda virus, and other network-based attacks. Viruses can infect systems through file shares, Web browsers, OS vulnerabilities, or email, and your antivirus strategy should address each of these areas. But be careful: When deploying file-based antivirus scanning, be sure to exclude the Exchange database files and transient files (i.e., the Message Transfer Agent—MTA—and mailroot folders). A file-based virus scanner that attempts to disinfect or quarantine an Exchange database or transaction log can prevent Exchange from accessing the database or log, thus causing data corruption. Ideally, the file-based antivirus scanning product you use will let you define these exclusions during installation to ensure that the transient files aren't accidentally included. (For Microsoft's recommendations about which antivirus measures to take for Exchange, see the Microsoft article "XADM: Exchange and Antivirus Software.")

Recent security rollup patches from Microsoft include protection from W32.Blaster.Worm and fix other OS vulnerabilities exploited by virus writers. Use the Microsoft Baseline Security Analyzer (MBSA) to audit cluster nodes for vulnerabilities and to get a list of recommended security updates. (See <http://www.microsoft.com/technet/security/tools/mbsahome.asp> for more information about MBSA.) Alternatively, use the Windows Update service to download the most recent security patches.

If your cluster runs Exchange 2000, set access on the built-in Message Tracking Log share and Address share to Read only. By default, the Everyone security group has full-write access to these shares. The Microsoft article "XADM: The Nimda Virus May Infect the Files in Log Folders on New Exchange 2000 Virtual Servers in a Cluster" describes how to change the permissions. As an added precaution, don't create any shared folders on your Exchange cluster. (By default, Exchange 2003 sets built-in Exchange shares to read-only.)

Implement an antivirus solution to protect your cluster from email viruses. Third-party antivirus products for Exchange (such as those listed at <http://www.microsoft.com/exchange/partners/antivirus.asp>) can scan mailboxes in real time for viruses such as Sobig.F and ILOVEYOU. Be sure to schedule regular virus-pattern updates from the product's vendor. Microsoft introduced the Virus Scanning API (VS API) so that antivirus vendors could develop software that can scan Exchange components such as databases, SMTP queues, and the MTA. Choose an antivirus solution that's VS API—compliant and that runs on clusters.

## 7. Failovers

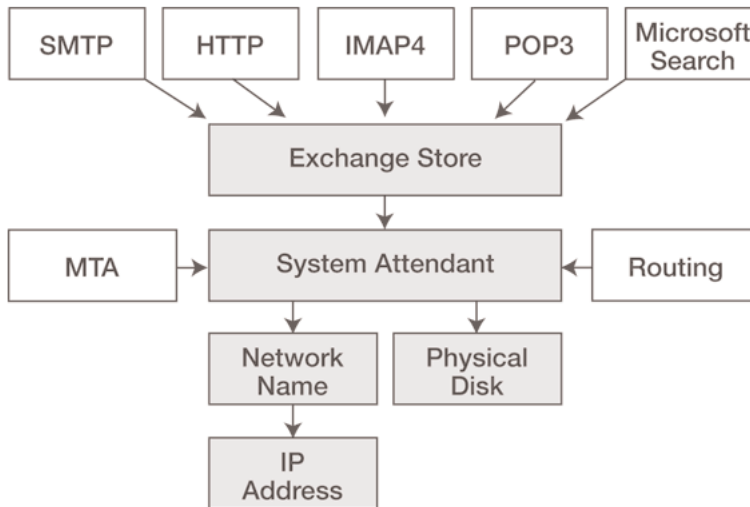
Failover occurs when an Exchange cluster group is moved from one node to another. Microsoft Outlook clients can't access Exchange during a failover, so minimizing failover times is necessary to

provide high availability and meet service level agreements (SLAs). To reduce the impact of failovers, you can deploy Microsoft Office Outlook 2003 clients running in cached mode, which lets users work from a local cache when no network connection is available. Outlook 2003 cached mode handles the loss of network connectivity much more efficiently than earlier versions of Outlook, which must be restarted to handle changes in network connectivity. Outlook 2003 cached mode can detect whether the Exchange server is reachable and seamlessly reconnect and synchronize without any action from the user.

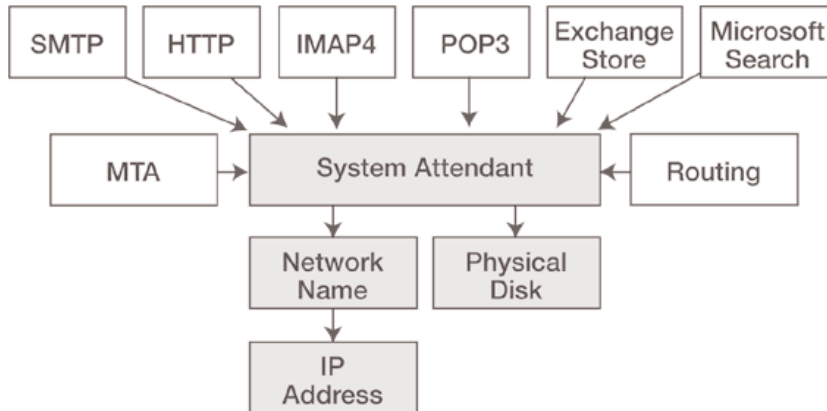
Exchange 2003 can achieve better failover times than Exchange 2000 because Microsoft has enhanced the resource model by flattening the Exchange dependency tree. Figure 2 shows the cluster resource model for Exchange 2000; protocol resources such as HTTP can come online only after the Exchange Store resource has started. Figure 3 shows the resource model for Exchange 2003; protocol resources depend only on the System Attendant resource. This change leads to faster failover times because cluster resources can start in parallel. Microsoft implemented a 3-minute timeout in Exchange 2003, after which, if the failover hasn't happened, the Store process is terminated to expedite the failover. This timeout should result in faster failovers compared with Exchange 2000.

**Figure 2**

*The cluster resource model for Exchange 2000*



**Figure 3**  
*The cluster resource model for Exchange 2003*



Two types of failover exist: planned and unplanned. Let's take a look at each type and how you can best handle each, as well as how monitoring unplanned failovers can help improve your deployment.

**Planned failovers.** Planned failovers usually take place as part of scheduled system maintenance tasks, such as an Exchange service pack rolling upgrade (which I explain in more detail later). To use that task as an example, on a two-node cluster, the process involves installing the service pack upgrade on the passive node (node 2) and rebooting if required. After you complete maintenance on node 2, you use Cluster Administrator to move the Exchange Virtual Servers (EVS) from the active node (node 1) to the passive node and perform the upgrade on node 1. An additional failover operation to fail the EVS back over to node 1 (called failback) will be necessary if node 1 is the preferred node in the cluster. During the planned failover process, the Exchange Resource DLL (exres.dll) takes the Exchange components in the Exchange cluster group offline; dismounts the SGs; stops protocols such as IMAP, POP, and HTTP; and takes the EVS Network Name and IP Address resources offline. Exres.dll then brings the EVS Network Name and IP Address resources, followed by the Exchange resources, online on the other node. The Cluster Service also updates the quorum database.

You can reduce planned failover time by performing the failover outside working hours. During hours of operation, a heavily loaded Exchange server can host as many as 3000 Messaging API (MAPI) connections, each of which must be terminated during the failover. The number of connections decreases outside working hours. Outlook 2003 cached mode, which makes better use of the network, might also help reduce failover times.

After a Windows service pack upgrade, the Store process rebuilds indexes, which can add several minutes to your failover time. The delay depends on the size of your Exchange databases. Event ID 611 in the Application event log indicates that an index rebuild is taking place.

**Unplanned failovers.** Unplanned failovers occur when the node hosting the EVS crashes or loses power. When the active node (node 1) goes down, the Cluster Service detects that the heartbeat connection—and therefore the active node—is no longer available and brings the EVS IP Address and

Network Name resources online on the passive node (node 2). The disks belonging to the EVS are brought online on node 2. `Exres.dll` brings the Exchange cluster resources online. The Store mounts the SGs and performs recovery tasks. The quorum database is updated. If you've designated node 1 as the preferred owner for the cluster, failback will occur to return the EVS to node 1 when it comes back online.

You can't do much to reduce unplanned failover time. However, if you've designated a preferred node in your cluster and your cluster fails over, you can at least schedule failback to occur outside working hours. To set failback times, right-click the Exchange Group in Cluster Administrator, select Properties, and choose the Failback tab. Select the Allow failback option and specify the permitted failback window (according to the 24-hour clock) in the drop-down boxes in the Failback between \_\_\_ and \_\_\_ hours option. Choose a time that doesn't conflict with other events such as backups and online database defragmentations. The failback process, which dismounts and remounts the Exchange databases, would disrupt these tasks.

If your cluster serves as a back end in your messaging organization, you can reduce the time needed to perform unplanned failovers. The Microsoft article "[How to Configure IPsec on an Exchange Server 2003 Back-End Server That Is Running on a Windows Server 2003 Server Cluster](#)" describes procedures for improving unplanned failover times for back-end virtual servers running on Windows 2003 clusters with Exchange 2003 and using IP Security (IPsec) to secure traffic between front-end and back-end servers.

**Monitoring.** One of the best ways to reduce the frequency of unplanned failovers and failover time is to monitor and analyze performance data on an ongoing basis. Windows and Exchange both include built-in monitoring tools. (For large production deployments, a management framework such as Microsoft Operations Manager—MOM—can simplify monitoring.) Before outages, both Windows and Exchange can log events that indicate a hardware or application problem. By actively monitoring event logs, disk performance, and memory usage, you can often prevent many outages—or at least delay them until later in the day when they'll affect fewer users.

## 8. Tips for Exchange Service Packs

As with single-server Exchange implementations, knowing the proper procedures for dealing with the installation and configuration of service packs is an important part of keeping your cluster running smoothly. You can get the best performance by keeping your Exchange service packs up-to-date, performing full backups of the Exchange database before and after service pack installation, verifying permissions, and testing upgrades in a clustered test environment before rolling them out.

Upgrade to the most recent service pack. For clusters running [Exchange 2000, Service Pack 3](#) (SP3) is available for download and incorporates more than 400 bug fixes. Virtual memory fragmentation is a major problem for large Exchange installations, especially for those that host many MAPI clients. SP3 includes an updated version of the Store (`store.exe`) to help address this problem. SP3 also includes several fixes that specifically address clustering concerns, as the Microsoft articles "[XCON: Cluster Failover Process Is Delayed Because of Message Transfer Agent Remote Procedure Call Timeouts](#)," "[XCON: Messages Back Up in Queue When the Virtual Server Is Set to Forward All Messages with Unresolved Recipients](#)," "[XADM: Cluster Service Terminate Function Does Not Kill the Information Store Unless It Times Out](#)," and "[XADM: The Information Store Stops on a Cluster Because of the IsAlive Check](#)" describe.

**Perform full backups.** Take full backups of your Exchange server before and immediately after installing an Exchange service pack. Set aside tapes that are outside your typical backup cycle. Exchange service packs usually include an updated version of store.exe. When you remount the Exchange databases after service pack installation, the service pack upgrades the databases to work with the new Store binary. The differences in Store versions mean that you can't roll back to a backup performed on an earlier service pack. For example, you can't restore to an SP3 server a backup that you took when the server ran SP2. (The Microsoft article "[XADM: Exchange 2000 SP2 Does Not Allow You to Restore Exchange 2000 or Exchange 2000 SP1](#)" contains some background information about Store version mismatch scenarios.) Be sure to update your disaster-recovery servers and procedures to reflect changes in Exchange service packs.

**Verify permissions.** As I explained earlier, you must have Exchange Full Administrator permissions to upgrade an Exchange 2000 cluster. Applying service packs, however, can cause permissions to be reset to their default values. Before and after you apply a service pack, verify permissions on the administrative group in which your EVS resides. (By default, Exchange System Manager—ESM—doesn't show security settings. To enable the Security tab, add the REG\_DWORD entry ShowSecurityPage to the HKEY\_CURRENT\_USER\Software\Microsoft\Exchange\ExAdmin registry subkey and set the entry's value to 1.)

**Test upgrades.** Create a parallel test environment so that you can try out service packs, hotfixes, and third-party products before deploying them in production. (Many third-party products aren't supported on clusters and might require some customization to work properly.) Exactly replicating your production configuration might not be cost-effective, but consider implementing a test cluster that uses virtual server technology, such as VMware, to test third-party products. Cluster-aware third-party applications create cluster resources in Cluster Administrator; you can move these resources between nodes as failover operations are performed. For third-party products that don't create cluster resources, however, be sure that you can automatically shut down and start the products on cluster nodes during failover and failback operations.

One benefit of using clusters with Exchange is the ability to perform rolling service pack upgrades to minimize downtime for end users. A rolling upgrade entails moving the EVS to one node and performing the installation on the passive node, then failing back the cluster and upgrading the other node. For example, suppose you have a two-node cluster in which node 1 is the active node with one EVS (EVS1) and node 2 is the passive node with no active cluster resources. First, back up node 2, apply the Exchange service pack to node 2, then move EVS1 to node 2. Check the Application log for errors and verify that Exchange starts correctly on node 2. Assuming that everything is working as it should, back up node 1 and apply the service pack to that node. Move EVS1 back to node 1, check the Application log for errors, and verify that Exchange starts correctly on node 1. Take a full backup of the cluster, including the system state on each node and the Exchange databases.

## Better Clusters

The guidelines in this chapter can help you achieve success when deploying Exchange 2003 or Exchange 2000 clusters. One last bit of advice for improving performance on Exchange 2000 clusters: Take a look at the improvements in Windows 2003 and Exchange 2003 clusters.

## Chapter 3

# Get a Grip on Exchange Data Management

By Kieran McCorry

Effectively managing Microsoft Exchange Server data is no easy job. Striking a balance between user demands and Exchange performance and stability has never been fun, but these days, it's a must. Email has become a critical business application, current regulatory demands are putting administrators in the hot seat, and you've gone from spinning plates to juggling knives. To get a grip on your Exchange data, you need a multidisciplinary approach that combines clearly defined policies and appropriate technologies (e.g., storage hardware, monitoring and reporting tools, data-management applications). Where do you begin?

First, let me clarify what I mean by *effective data management*. I define it as the practice of securely handling stored Exchange data in a way that optimizes the data's storage while providing adequate access to the data. That said, the best way to get started is to examine the financial, technical, and regulatory constraints that apply to your organization. These factors will influence your options for managing the Exchange data that resides in storage groups (SGs), databases, and user mailboxes (including offline folder stores—OSTs—and personal folder stores—PSTs) while providing efficient backup, recovery, and archiving capabilities.

## Juggling Constraints

Anyone who manages Exchange data is used to balancing sometimes contradictory demands. When you're looking for data-management solutions, three considerations will come into play. You'll need to look at the costs of your various options, the technical limitations of the options, and the types of regulatory requirements that apply to your company.

**Financial constraints.** As messaging traffic continues to grow (in terms of both volume and size) and as more businesses decide to retain messages in online Exchange databases or verifiable offline stores, data storage requirements—and the associated costs—increase. Financial considerations involve more than the cost of extra disks to support larger databases. You must also pay for the required storage infrastructure (e.g., additional storage arrays, backup devices, Storage Area Networks—SANs) and personnel to manage increased data volumes. These costs vary based on the size and nature of your organization. Small organizations of several hundred users might get by with simply increasing storage to satisfy demand, but larger organizations of several thousand users might incur significant costs.

If demand for storage outstrips your capacity to pay for additional resources, you might consider implementing stricter backup policies that retain less unnecessary data, or archiving solutions that help optimize your retention policies. The costs associated with these approaches can often be significantly less than the costs incurred to add more storage in a frantic attempt to satisfy demand.

**Technical constraints.** Even when your company is able and willing to throw money at data storage, unchecked data growth impairs your ability to maintain effective backup and recovery solutions. Even though tape technology (for example) continues to improve, increased data volumes take longer to back up and restore. Thus, the attempt to meet one need (e.g., easy access to data) can reduce your ability to meet other needs (e.g., quick recovery).

Look for a technical solution that balances these requirements. Such balanced solutions combine the management of online Exchange data with nearline or offline archiving solutions. This approach is appealing because it lets you cap the growth of your primary Exchange storage subsystems and archive critical data according to agreed policy limits, but still keeps that data within easy reach.

**Regulatory constraints.** Many organizations have implemented policies that mandate the archiving of email communications. In companies that implement archiving solutions to meet internal standards (rather than external requirements), adherence to these policies is largely a matter of internal corporate governance. Organizations that are regulated by external agencies have much stricter requirements for archiving—or more accurately, compliance. A solid regulatory-compliance system intercepts all email that enters, leaves, or circulates within your organization. When you implement such a system, you can guarantee the archiving of all the data in your environment, wherever it may ultimately reside (e.g., in PSTs, on mobile devices).

Once you know which constraints apply to your organization, you can begin to determine the types of guidelines that you need to place on the data that resides in your Exchange server database files, Microsoft Outlook cache files (i.e., OSTs), and PSTs. You'll also be able to decide which backup, recovery, and archiving solutions will work best in your environment.

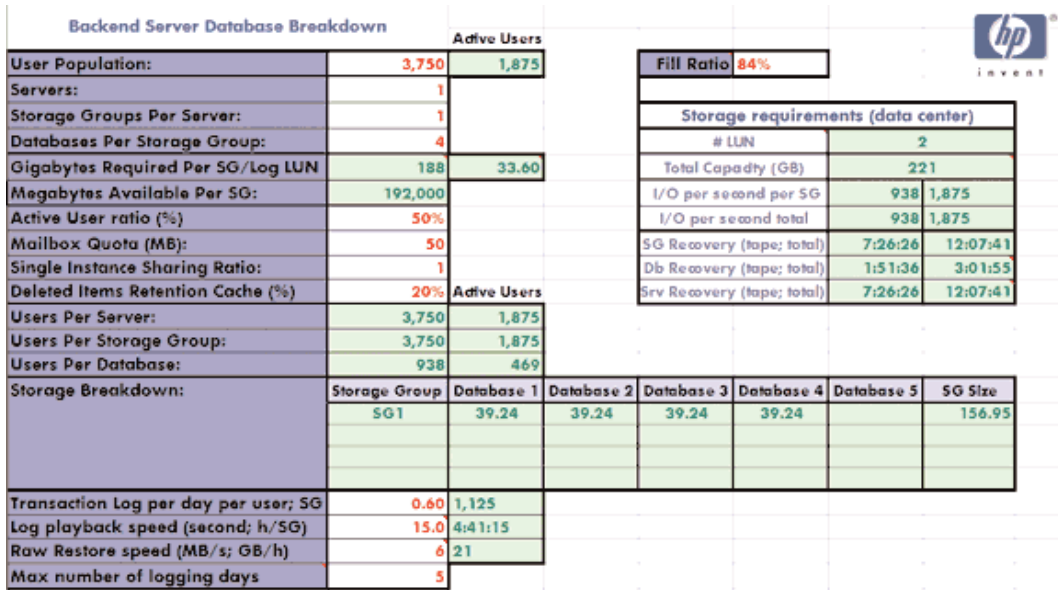
## Dealing with Server–Based Data

Exchange stores email data in databases on the Exchange server (or servers). These databases are arguably the best repository for email content, not least because of the single-instance storage mechanism that exists within each database (though not between databases). In general, data located on the server is more accessible than data in PSTs, at least from a management perspective. Shared information is best stored in Exchange public-folder databases. An Exchange Server 2003 or Exchange 2000 Server machine can contain as many as five databases within an SG and can hold as many as four SGs; thus, one server can contain as many as 20 databases. Established best practice advises against letting your databases exceed 40GB each so that backups—and more importantly, restores—can occur within acceptable time limits.

Storage capabilities determine the maximum number of active users that can be served by a single Exchange system. Exchange storage subsystems must be capable of dealing with the I/O load that the user population will place on the system. The Microsoft guide “[Optimizing Storage for Exchange 2003](#)” suggests that you implement a subsystem that can provide an average of about 0.75 I/Os per second per active user. For most subsystems—even those on high-end SAN platforms—this guideline dictates a practical maximum of just less than 4000 active users per server.

You have to keep the guidelines for database size and user limits, along with other performance factors (e.g., transaction log volumes), in balance when sizing your servers, allocating storage, and setting mailbox limits. Figure 1 shows a typical spreadsheet tool that I use to calculate storage requirements. For example, a disk quota of 200MB is achievable for a server that hosts approximately 4000 mailboxes.

**Figure 1**  
Backend server database breakdown



Aside from using mailbox quotas (which you can apply on a per-database as well as on a per-user basis) to manage Exchange server-based data, you can use Group Policy to configure the Exchange Mailbox Manager to detect and delete old or large messages from users' mailboxes. This approach can help keep mailbox sizes in check before users run up against the dreaded *mailbox quota exceeded* message. If you're worried about accidental deletions, Exchange's Deleted Items Recovery feature, when enabled, lets users recover messages even after the Deleted Items folder has been emptied. You can use this great administrative feature to combat user errors that otherwise would result in costly restore operations, but be careful—it can increase the size of your databases. I've seen empirical evidence suggesting that a deleted-items retention period of just 7 days can cause a database bloat of 10 to 30 percent.

## Managing User-Maintained Data

The data that users store in OST or PST files on desktop or laptop PCs is the most troublesome type of Exchange data to manage because it's distributed and often inaccessible (from an administrative perspective). OST files are a lesser problem because they're simply slave copies of Exchange server mailbox content. In Microsoft Office Outlook 2003 Cached Exchange Mode, the OST is a complete replica of the online Exchange mailbox, whereas in non-Cached Exchange Mode (or in earlier Outlook versions), the local OST contains a subset of the server mailbox data.

**PSTs are a different story.** Mailbox quota restrictions often force users to store important email data in PST files, but these files are usually large (several hundred megabytes or greater) and—when stored locally—typically are excluded from local hard-disk backup procedures, if any even exist. Users often place PST files on network shares, which is certainly better than keeping them on hard

disks. But though backups are simpler when dealing with network share-based PST files, unchecked PST growth can still be problematic. As a PST's size increases, so does its chance of corruption, which can be irreparable. In reality, little benefit is to be had from moving data to a network share-based PST versus keeping the content in a user's mailbox. Furthermore, PSTs are inherently unsecure. You can encrypt PST files, but decrypting utilities are well known and widely available. If users are storing sensitive corporate data in PSTs on, say, laptops, that data is at risk if the laptop is lost or stolen. Even data held in PSTs on network shares must be adequately protected from unauthorized access. Furthermore, if you have legal requirements for archiving or retention, unmanaged PST files can get you in a lot of trouble.

## Better Backup and Restore

Your choice of backup—and more importantly, restore—solutions will depend on the amount of data that you need to process and the speed with which this processing must occur. For server-based data, many enterprise deployments implement procedures that allow for the databases to be restored within 1 hour (your specific Service Level Agreements—SLAs—might provide for variance from this figure). For example, to meet the goal of a 1-hour restoration of 40GB of server data, a tape device must provide restore rates (not just backup rates) of no less than 10MBps. Many backup solutions now involve intermediate backup to disk before eventually streaming off to tape, so initial backup rates (i.e., the rate of the backup-to-disk portion) and restore rates can often be significantly higher than backup-and-restore traditionally associated with tape only.

SAN-based solutions often offer high tape-restore transfer rates; figures in the region of 100GB to 140GB per hour aren't uncommon. Such capability might influence the size limits that you assign to

### Putting Exchange Data Management in Context

You can't look at Exchange Server data management in isolation. When you're planning a management solution, you must consider it in the context of the following:

- Storage infrastructures are often shared among applications, so Exchange storage allocations and platforms are often affected by other enterprise storage initiatives
- Service Level Agreements (SLAs), which are tightly coupled with Exchange storage, are influenced by myriad business and service-related factors
- Exchange storage is influenced by enterprise-wide archiving and compliance initiatives
- Exchange data management decisions are ultimately derived from business drivers and factors

As such, many organizations consider Exchange data management as just one component of an overall enterprise data management framework, which in turn is a component of an overall service management framework.

Formal guidelines to help organizations craft in-house procedures that help deliver a quality IT service have emerged in the IT Infrastructure Library (ITIL) best practice recommendations. The ITIL guidelines are part of an overall IT Service Management Forum (itSMF) initiative; organizations often follow these guidelines in conjunction with the relevant standard (BS15000). A full description of the itSMF initiative and its relevance to Exchange data management is beyond the scope of this piece, but suffice any organization that is gearing itself towards itSMF certification will need to implement policies and procedures that mandate effective data management. (You can learn more about the ITIL at <http://www.ogc.gov.uk/index.asp?id=2261>.)

## Protecting RIM BlackBerry

More and more companies are reaping the benefits of wireless email, allowing employees to remain connected even when not at their computers. This new freedom brings with it the need for protecting this now-critical application to ensure it's availability in the event of any type of failure or outage. This whitepaper discusses the need and options for maintaining your RIM BlackBerry environment during a failure, regardless of the severity. To view this white paper, please click here: [http://www.nsisoftware.com/how/nsi\\_blackberry.asp?eve=657](http://www.nsisoftware.com/how/nsi_blackberry.asp?eve=657)

databases. The ability to backup and restore larger volumes of data faster means that you can implement larger databases, which in turn can mean either increased mailbox quotas for users or more users per server.

Windows Server 2003 provides support for Volume Shadow Copy Services (VSS) which in conjunction with Exchange 2003 offers the capability to take a consistent snapshot of an Exchange database in a matter of seconds. Note that the snapshot is merely a point-in-time view of the disk map for the original database file, so if the physical volumes on which the database resides becomes unavailable, the snapshot is effectively useless (although many vendors attempt to insulate systems from this problem). Therefore, even though databases can be “snapped” in seconds, the snap volume must still be streamed off to some storage medium, typically tape. Accordingly, however, the snapped volume can be restored in a matter of seconds as well. VSS-aware storage subsystems and backup and restore solutions can dramatically influence your data-management framework, but be sure you carefully research and test them before putting them into production.

Exchange 2003 (especially Service Pack 1—SP1) introduces new functionality in the form of the Recovery Storage Group (RSG). The concept is straightforward: If a database from a particular SG becomes unavailable to users and must be restored from backup media, an empty recovery database is made available to users homed in the affected database while that database is being restored from backup. Although none of the users' existing messages will be available during this restore period, the ability to send and receive email is maintained. When the restore is complete, the recovery database (which is now populated with new content) can be merged with the restored database. When properly worked into disaster recovery and restore plans, the RSG concept can positively influence SLAs and maximum database sizes. And SP1's Recover Mailbox Data Wizard simplifies the merging of the restored data with newly created data.

Backing up user-maintained data, such as PSTs, presents greater challenges, as I mentioned earlier. Backups of PSTs on local hard disks are almost impossible to enforce or control because they rely almost solely on the user. PSTs on network shares can be backed up centrally but still seem to offer little advantage over large mailboxes in the Exchange database.

## All About Archiving

Strictly speaking, archiving solutions differ from regulatory-compliance solutions in the following ways:

- Archiving is often user-initiated, in that a user arbitrarily decides to archive an object from his or her Exchange mailbox to an archive store.
- Arbitrary archiving is often complemented by policy-based archiving of expired content to archive stores.

## Data Management Challenge: How Did We Get Here?

Four primary factors contribute to the increasing need for Exchange Server data management solutions. These factors are the increasing use of email, the growth in size of document formats, the increasing availability of inexpensive raw storage, and corporate and regulatory-compliance requirements.

**Increasing use of email.** IDC has predicted that the number of email users worldwide will have increased from 452 million in 2000 to 1.2 billion by 2005. The company similarly predicts that email traffic will increase from an average of 9.7 billion person-to-person messages sent worldwide each day (in 2000) to 35 billion person-to-person messages worldwide per day (in 2005). Considering these numbers, the increased requirements for storage and the resulting data management challenge are unsurprising.

**Growth in size of document formats.** Document formats, especially those produced by Microsoft Office applications, have enlarged file sizes considerably over the past several years. The proliferation of these document types as email attachments has accounted for a considerable increase in storage requirements. A typical Microsoft PowerPoint file can be 1000 times larger than an average text-based email. Microsoft Word files are similarly large (although not quite as culpable as their PowerPoint cousins). Retaining  $x$  number of email messages today is likely to demand a significant increase in overall storage than it required several years ago. Mailbox sizes have typically grown over the past several years, and complex file formats stored in user mailboxes now consume significantly more space than do simple text messages. The Radicati Group reports that the size of a basic text email more than doubled between 2003 and 2004, with an average email being 38.1KB and 469.2KB in size (without and with attachments, respectively); attachment sizes having increased annually by 30 percent.

**Increasing availability of inexpensive raw storage.** Another factor that has contributed significantly to the data management challenge is the sheer availability of raw storage. Many Exchange I/O subsystems are configured for maximum spindle count (smaller Exchange implementations might not be afforded this luxury). As disks have grown in capacity, a pleasant side-effect has been copious quantities of available disk space at a relatively low cost. The Harrow Group's Harrow Technology Report has stated that 20 years ago, a 20MB disk drive cost around \$1200 (approximately \$20 per MB), whereas in 2004 a 200GB disk drive cost around \$100 (approximately \$0.0005 per MB). With this increase in storage capacity and in an effort to address the rising demand for storage, many organizations have set generous mailbox quotas, thus setting the stage for data-management headaches in the future. Mailbox sizes have grown from about 25MB back in 1996 to about 100MB to 200MB today.

Similarly, increases in the performance capabilities of many backup and restore devices have resulted in faster backup and restore times. The earliest DLT tape drives had native transfer rates of about 1.25MBps; modern DLT drives can provide 16MBps transfer rates, and Linear Tape-Open (LTO) drives offer rates of up to 30MBps. These advances in backup-device performance support larger Exchange database sizes, effectively masking a data-management problem.

**Regulatory-compliance requirements.** With the advent of Sarbanes-Oxley, more and more companies need to implement policies that meet the needs of external regulatory compliance agencies as opposed to simply meeting internal guidelines.

- Archiving solutions typically don't guarantee that all messages that are created or sent within a system or that pass through an ingress or egress point will be written to an archive store.

You might be aware that Outlook provides a rudimentary form of archiving whereby the user can configure Outlook to move messages older than a defined age to a PST file. However this approach just moves the data around rather than delivering it to dedicated, protected archive stores, so Outlook archiving isn't a serious contender.

More sophisticated solutions can provide user-initiated and policy-based archiving to a second-tier (or higher) data location. Solutions such as these are effective because they can retain a message stub in the user's Exchange mailbox while moving sizeable attachments or message content to the archive store. If a user wants to review archived content, it's often accessible merely by clicking the message stub, at which point the archived content is retrieved. Thus, Exchange storage consumption is optimized while large content is offloaded to a system more suitable for bulk storage.

This type of archiving solution is often integrated with Exchange's Journaling feature to intercept and trap all messages circulating within an Exchange environment. But when large volumes of traffic are expected or when regulatory-compliance issues dominate, even archiving systems that integrate with Exchange Journaling (which might not provide the non-rewritable, non-erasable storage environment that most regulations stipulate) must integrate with or be replaced by more advanced technologies.

Examples of this form of technology include EMC Centera as well as HP's Reference Information Storage System (RISS). These types of solutions let you store static content, in a non-modifiable format, on disk and usually implement RAID-like technologies to guarantee data integrity and content authentication by means of digital signatures and time stamping. Typically, these solutions implement sophisticated Hierarchical Storage Management (HSM) systems, in addition to providing content indexing and retrieval. When you're dealing with regulatory compliance, HSM functionality is important because of the huge volume of email that can quickly mount up, especially in larger organizations. The average user sends 20 emails per day at an average size of 25KB per message. In an organization of 10,000 users, this estimate correlates to a total of 200,000 messages per day—4.7GB of content per day or 1.7TB per year. If you also need to archive inbound messages, storage requirements can grow significantly. Of course, these are average figures, but I'm aware of one organization with 9400 users that receives between 120GB and 150GB of email per month.

Many organizations choose to implement an archiving solution as a first step when migrating from one Exchange version or organization to another. This technique reduces the amount of data that must be migrated and can speed up the migration process.

## Get Your Act Together

You can no longer ignore the importance of managing Exchange data, especially as email traffic and message size continue to grow and as regulatory-compliance requirements become more commonplace. Users will continue to demand that you retain more data—yet leave it at their disposal—and that you maintain fast recovery times and as little downtime as possible. As an administrator, you must try to meet these demands while operating under your organization's financial, technical, and regulatory constraints. Fortunately, you have many options at your disposal: mailbox quotas, storage technologies, and archiving solutions.

## Chapter 4

# Exchange Server 2003 and VSS: A Way to Improve Recoverability and Availability

*By Jerry Cochran*

The increasing importance of messaging and collaboration as mission-critical services has left Microsoft Exchange Server administrators and implementers looking for new ways to improve server recoverability and availability. In addition, hardware and software vendors are supporting technologies that enable more rapid recovery of server data and applications. Two specific enabling technologies are volume snapshots and volume cloning. These technologies are available in a variety of hardware and software implementations ranging from full-blown snapshot-manager products for Exchange to integration kits for a custom Exchange recovery solution. In the past, regardless of how these technologies were delivered, Microsoft products didn't support snapshots or cloning. Table 1 lists pertinent Microsoft articles that discuss snapshot and cloning support for Exchange 2000 Server, Exchange Server 5.5, and Exchange 4.0. However, with the advent of Windows Server 2003 and Exchange Server 2003, Microsoft has introduced Volume Shadow Copy Service (VSS), which makes snapshots and cloning available natively to Exchange administrators. Let's look at VSS and what it means to Exchange disaster recovery and availability.

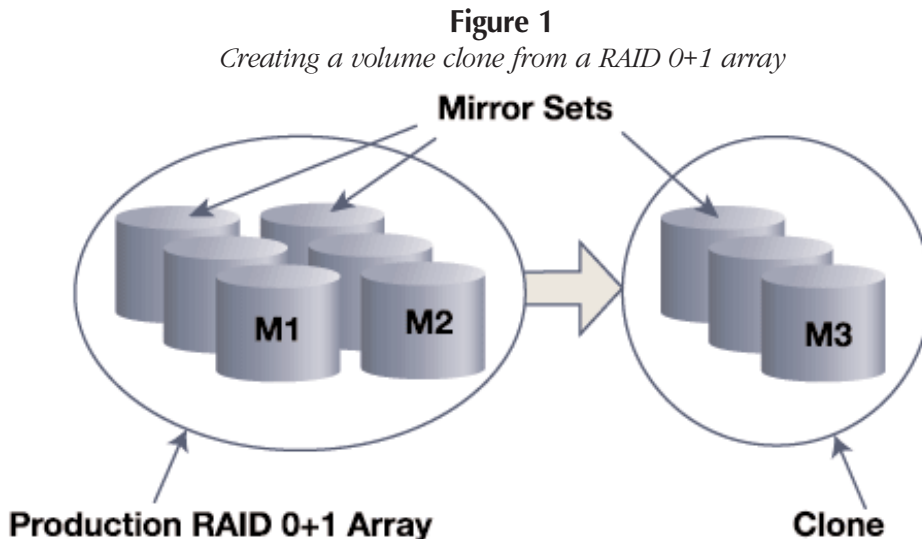
## Volume Snapshot and Volume Cloning Overview

Let's briefly review volume snapshots and volume cloning. Although these technologies aren't new, they're relatively new to the Windows platform. Cloning and snapshot technologies provide Business Continuity Volumes (BCVs)—a mechanism for data duplication and point-in-time copies. On the surface, these technologies might appear to be the same; however, they're quite different in technical implementation.

**Volume snapshots.** A volume snapshot (also known as a copy-on-write snapshot) is a representational metadata mapping of specific volume blocks at the time you create the snapshot. For example, if you create a snapshot of your Exchange database volume, the snapshot represents the blocks on disk that compose your Exchange database and any other files on the volume at the time you create the snapshot. Therefore, after you create a snapshot, you must maintain the original volume blocks for the snapshot to remain intact. As a result, you must copy changes to the volume blocks to another location in the storage pool. From an Exchange viewpoint, this requirement means that if you create a snapshot for the volume on which Exchange data resides, then make a change to a page in the Exchange database, the VSS-supported storage system will typically copy the affected blocks to a special "diff" area in the storage pool allocated from free volume pool space. In this manner, the system preserves the original subset of volume blocks that represent the snapshot. After you create the snapshot, the production data consists of a combination of original unchanged blocks from the snapshot and new blocks of data. The snapshot remains intact and represents the data state at the time the snapshot was created. Creating snapshots is relatively quick and simple—the

VSS-supported storage system creates the volume block mapping, and the snapshot exists. Because a copy-on-write snapshot isn't a complete redundant copy of the data and is subject to disk failures, this technology is less desirable than clones. As a result, snapshot recovery can be problematic if the base volume is lost or corrupted because an administrator must complete several steps to recover the original volume.

**Volume clones.** Similar to volume snapshots, volume cloning isn't a recent development. Cloning comes from RAID 0+1 technology. A clone is an additional member of a RAID 0+1 mirror set. For example, if you have a RAID 0+1 set with three disks mirrored to three disks, you have a two-member RAID 0+1 set. By normalizing another three disks to the existing RAID 0+1 set of six disks, you create a three-member mirror set (i.e., a triple mirror of nine disks). You can add other members to the mirror set as well. By creating multimember mirror sets and then separating members from the set, you enable clones. Because the existing production data has multiple mirrored copies, you can use some of these copies to create point-in-time copies of the data. Unlike a snapshot, a clone is a complete standalone copy of the data. To create a clone, you simply separate one or more members from the production set. The result is a production mirror set that supports the application (i.e., the two-member RAID 0+1 array) and clones (single-member sets) that you've separated from the production data, as Figure 1 shows. In the event of data corruption or loss, you can use a clone to recover system data by making the clone available as the production LUN. Because clones are a complete redundant copy of the data, they're most useful as rapid-recovery mechanisms.



The advantage of volume clones lies in how quickly you can create them. The downside is having to resynchronize an old (or new) member with the primary mirror set, which can take time depending on the size of the disks and the capabilities of the controller and enclosure.

## The VSS Foundation

Because volume snapshots and volume cloning have had limited availability in the Windows space, the OS and applications haven't been able to take full advantage of these technologies. Hardware and software developers have implemented snapshot and clone solutions with little or no exposure or integration with the OS and applications. As a result, third-party vendors have been primarily responsible for supporting these solutions.

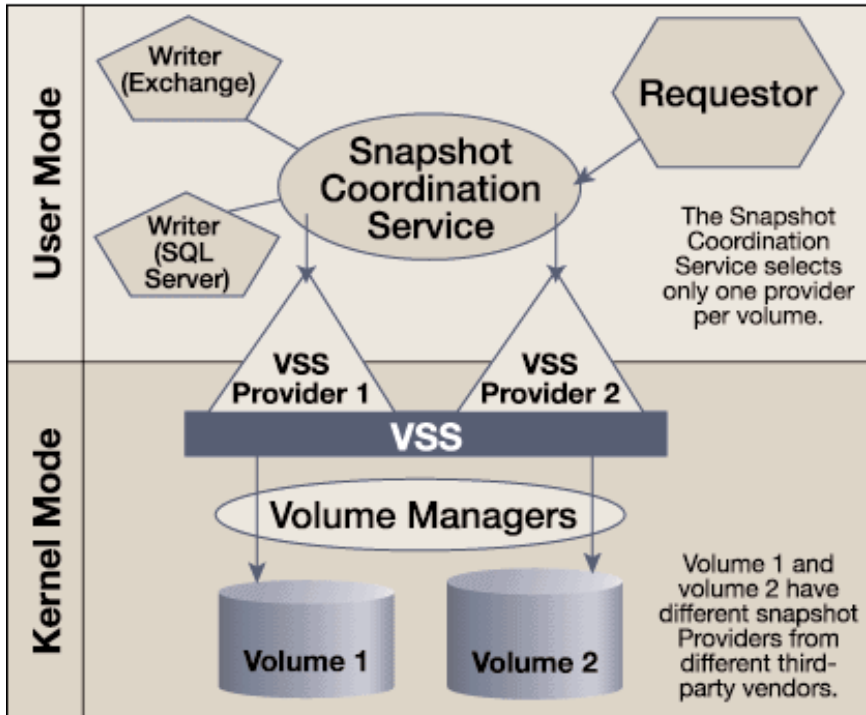
Microsoft has made substantial storage technology investments, including VSS, in Windows 2003. VSS attempts to solve a key problem—the constant expansion of the backup and restore window. Because of today's inexpensive disk space and applications' large appetites for storage, administrators are constantly challenged with data sets that continually grow and disaster-recovery facilities that don't. Administrators have many methods—such as growth management (e.g., archiving, Hierarchical Storage Management—HSM, quotas)—of dealing with this problem, but what they really want is a way to increase backup and (more importantly) restore speeds. If you could increase backup rates from 10GB per hour to 20GB per hour, your backup window would shrink by 50 percent.

VSS addresses one primary concern—that a lot of today's data is online. For example, consider a 24 x 7 file server with thousands of user files. Whenever a typical backup runs, a few files will be open. To complete the backup, you have three choices. First, you can stop the service or session and close the open files. Second, you can skip open files and hope they don't get lost or corrupted before the next backup. But the best solution is to take a snapshot of the data and use the snapshot as the basis for recovery.

VSS provides a framework for using snapshot and cloning technologies with the Windows platform. More specifically, VSS provides services that deliver an infrastructure upon which the OS, applications, and vendors can leverage these technologies.

VSS has three primary goals: to provide application synchronization, including synchronizing application data spread over multiple volumes; to provide discovery and enumeration of snapshots or clones (called Shadow Copies); and to provide a framework in which hardware and software vendors can plug in interoperable Shadow Copy creation components (called Providers). With these goals in mind, VSS on Windows 2003 lets a hardware or software vendor supply a Provider, an application developer expose Shadow Copy packages that contain XML-based metadata (called Writers), and a backup vendor build applications (called Requestors) that can initiate backup and restore operations that leverage these components on a common infrastructure. Figure 2 shows Windows 2003's VSS architecture.

**Figure 2**  
*Windows 2003's VSS architecture*



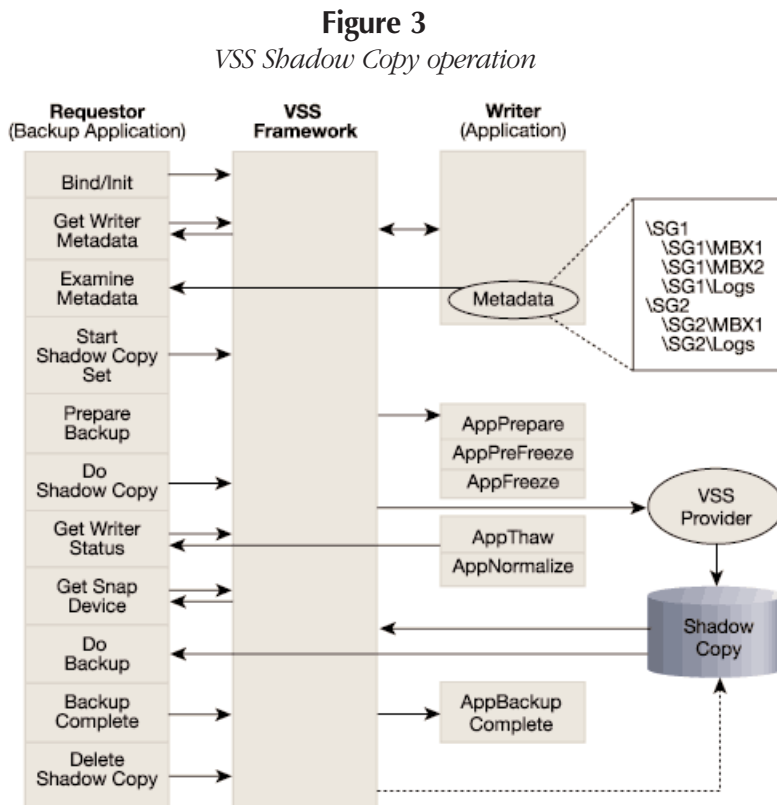
**VSS Providers.** VSS exposes APIs that let vendors VSS-enable their solutions. For a vendor's snapshot or clone technology to function within the VSS framework, that vendor must develop a Provider—components that manage volumes and create clones and snapshots according to a specific vendor's technology and implementation. Typically, a Provider is a process containing some kernel-mode and some user-mode code that persists data about a physical Shadow Copy and exposes that Shadow Copy to the OS or applications. Vendors must build Providers regardless of whether the vendors create hardware- or software-based solutions. In the case of a software-based Provider, the implementation typically consists of a user-mode process coupled with a kernel-mode device driver. Details about the hardware- and software-based solutions and the Provider implementation are at the discretion of the vendors, as long as they follow the VSS framework-implementation rules. Windows 2003 includes a software-based Provider, which the OS implements as a copy-on-write software snapshot.

**VSS Requestors.** Backup and disaster-recovery vendors can develop applications that make use of the VSS architecture, APIs, and implementation rules. To do so, these vendors must develop a Requestor—an automated or GUI-based process or application that requests one or more Shadow Copy sets from one or more volumes. The Requestor is the main process that communicates with the Shadow Copy interface, which coordinates activities between Requestors, Providers, and Writers. The Requestor also communicates with Writers to gather backup components, files, and metadata that the

Writers manage. This communication lets a Requestor select which volumes should be Shadow Copied to complete the requirements of the backup operation. Windows 2003 doesn't include a Requestor.

**VSS Writers.** The most important players in the VSS framework are arguably the applications. An application must carefully expose recovery packages that are specific to the application's technology, implementation, and disaster-recovery requirements and constraints. For example, because Exchange uses a transacted database engine, its requirements are unique, even when compared with similar applications (e.g., Microsoft SQL Server, Oracle). Writers are code and related data embedded in applications and components of those applications to enable VSS compatibility. Writers respond to the Shadow Copy interface to let the application prepare, freeze, and thaw application I/O to ensure that no writes occur on the volume when the Provider creates the Shadow Copy. Through the VSS interface, Writers also respond to Requestors by supplying Writer metadata that includes details about what the Requestor requires to perform Shadow Copy operations for the specific application.

A backup operation that uses VSS is a well-orchestrated process that involves the interaction of each component in the VSS framework. Figure 3 shows a generalized flow and interaction diagram of a backup operation using VSS technology.



## Better Backups through Replication

All business continuity efforts start with protecting your data. Double-Take from NSI Software is the leader in High Availability and Disaster Recovery solutions because of continuous, real-time protection of Windows file systems that compliment your existing tape back-up solution. This Technical Whitepaper will describe how Double-Take provides a "better copy of the data to back up", providing a truly comprehensive data protection solution.

[http://www.nsisoftware.com/how/nsi\\_better\\_backup.asp?par=ENTMag.com&eve=657](http://www.nsisoftware.com/how/nsi_better_backup.asp?par=ENTMag.com&eve=657)

## Exchange 2003 Support for VSS

To support the VSS framework, an application such as Exchange must provide the Writer component. Because Microsoft has no plans to provide a Writer for Exchange 2000, Exchange 5.5, or Exchange 4.0, the company won't support VSS for these versions. However, Exchange 2003, paired with Windows 2003, does provide VSS support for Store backup and recovery. In Exchange 2003, Microsoft has built the Writer functionality into the Store process. This Writer provides the necessary support for Requestors to initiate backup operations for Exchange 2003.

## Exchange 2003 Backups Using VSS

Traditional Exchange API-based backups focused on four backup types for Exchange databases: Full, Incremental, Differential, and Copy. However, the Exchange 2003 Writer supports only a Full backup at the storage group (SG) level. VSS performs Exchange Full backups at the SG level, even though the Exchange Writer treats individual databases as separate components. VSS uses the AddComponent call to add each database component to the Shadow Copy set, which in the case of a Full backup, is the entire SG (i.e., databases or log files). In a Full backup of a SG, VSS creates a complete Shadow Copy of all volumes—the Shadow Copy contains database and transaction log files associated with that SG. In addition, as is the case with non-VSS Full backups, VSS truncates the transaction log files after successfully creating and backing up the Shadow Copy. To truncate the transaction log files, the Shadow Copy set must include all databases. For this reason, Microsoft will use the metadata definition for the Exchange Writer to force the Requestor applications to process only Full backups that have all SG components (i.e., databases or log files) in the Shadow Copy set.

## Exchange 2003 Recovery Using VSS

Although VSS backup for Exchange 2003 is at the SG level, you can recover individual databases from the SG Shadow Copy set. VSS-based restoration of an Exchange 2003 SG is useful when data in one or more databases in the SG is lost or corrupted, but the current log files remain intact on disk; when the current log files on disk are lost or corrupted, but the databases remain intact; or when databases and current log files within an SG are lost or corrupted.

In the context of Exchange 2003 and VSS, only the backup application is responsible for restoring data to disk. The Exchange 2003 database engine, not the Requestor, is responsible for recovering the data to a consistent, up-to-date state through playback of the log file. To do so, the database engine activates existing soft or hard recovery procedures. After the VSS-aware backup application restores the transaction log files and databases, Exchange 2003 remounts and restarts the SG, then the database engine initiates recovery. The database engine determines that the state of the databases isn't consistent with the end of the log file on disk and begins the recovery procedure.

Three Exchange 2003 data restoration scenarios exist, but only two procedures for those scenarios exist. The Roll-Forward recovery and Point-in-Time recovery procedures for restoring data are the same whether you've lost only the SG's log files or you've lost an SG's log files and databases. You use the same procedure because the loss of the log files is a catastrophic failure in Exchange and requires restoring the entire SG. In either case, these recovery options follow a specific step-by-step process:

- The backup application Requestor through the Exchange Writer and APIs takes the SG offline.
- The backup application performs a VSS-based recovery of the volumes required from the SG Shadow Copy set.
- If one LUN per SG is configured, Exchange recovers all databases except those that are intact.
- If multiple LUNs per SG are configured, Exchange recovers only the LUNs with the databases needing recovery from the Shadow Copy set.
- Exchange performs an Extensible Storage Engine (ESE) hard recovery and replays applicable log files for databases being recovered, depending on whether a Roll-Forward recovery or Point-in-Time recovery is occurring.
- The backup application Requestor through the Exchange Writer and APIs brings the SG online.

**Roll-Forward recovery.** In a Roll-Forward recovery, one or more databases in the SG are lost, but the log files are intact on the server at the time of the recovery. In this case, you can selectively restore each of the affected databases from a Full backup of the SG. Within the context of the VSS framework, you select from the SG backup only those database components that correspond to the databases you want to restore. The VSS-aware backup application restores the databases and Exchange recovers the databases and brings them up-to-date from their state at the time of the snapshot by rolling forward through the transaction logs (i.e., Exchange hard recovery). The Roll-Forward recovery option lets you recover backed up data as well as data that has accumulated (e.g., in transaction logs) since the last backup.

**Point-in-Time recovery.** When the SG's log-file volume has been damaged or lost or the log files have been lost or damaged together with some or all of the SG's databases, you must restore the log files from a previous backup, together with all the databases backed up at the time of the last full backup of the SG. Because you can't recover to the point of the failure because the log files and databases since the last backup have been lost or damaged, you can recover only to the point of the last full backup. This process is known as a Point-in-Time recovery. Because this option doesn't provide roll-forward capability, some data will be lost. To provide Point-in-Time recovery, you must restore the databases that you backed up at the time of the Full backup as well as the log files from the Full backup. In addition, you must recover all databases associated with the SG. You can't assume that any of the databases were left in a transaction-consistent state at the time the log files were lost and went offline because the loss of the transaction log is a fatal error that causes the Store to shut down immediately with no guarantee of consistency. Therefore, to ensure that the databases are in a consistent state when you restart the SG, you must return the entire SG to its state at the time of the last Full backup.

## Implications for Exchange Administrators

As organizations move to Windows 2003 and Exchange 2003, the use of VSS-based backup and recovery will become a standard mechanism for Exchange disaster recovery. However, VSS solutions aren't yet proven or readily available. In addition, VSS adds complexity to your disaster-recovery scenario, and we're only beginning to learn the best practices and pitfalls. The non-VSS solutions that exist today let you use snapshot and clone technologies with Exchange. However, these technologies have no native OS or application support. Organizations must rely on the vendors of these solutions for support—both for current non-VSS solutions and for future VSS solutions. Now that Microsoft has shipped Windows 2003 and Exchange 2003 is scheduled for release this year, vendors likely will follow closely with robust VSS Provider and Requestor support.

## Chapter 5

# Exchange Availability Tips & Tricks

## Exchange Server Availability: The Big Picture

*By Jerry Cochran*

How do you measure reliability in your Exchange Server environment? You can use a standard military specification that factors failure rates and the mean time between failures (MTBF) into the total operational period. However, reliability measurements are best suited to individual physical system components. Do these values really mean anything to you as an Exchange administrator?

I prefer to think about availability rather than reliability. However, most Exchange administrators look at availability from a binary point of view (i.e., is your Exchange server up or down?) or as a measurement of the percentage of time a system is available for a given operational period (e.g., 99.999 percent). These views might not be the most effective way to measure availability in your Exchange environment.

When you want to accurately measure Exchange availability, where should you start? First, you need to understand that downtime isn't simply about the server. The poor Exchange server often takes complete blame for an entire outage period, the majority of which isn't necessarily the server's fault. Suppose your Exchange server is down (i.e., unavailable) for 8 hours. Rather than simply blaming Exchange, look deeper. You might discover that you weren't notified of the problem for 2 hours, you took 1 hour to decide what to do, 2 hours to find a good backup tape, and another 3 hours to restore the server to operational status. In this case, you can contribute only a small part of the downtime to software or hardware—most relates to personnel, procedural, and process issues (e.g., monitoring, alerting, disaster recovery). When you understand that downtime and outages actually consist of multiple components, you start to rethink how you measure downtime. For an Exchange deployment, you need to identify the components of downtime, then figure out how to address each component to reduce downtime and thereby increase availability.

You also must come to terms with how you measure availability. Most people think that Exchange availability is synonymous with Exchange server uptime. However, this definition might not provide an accurate picture of true availability (i.e., the availability of Exchange services to users). An Exchange server might be up and running, but that doesn't mean that users can get the services they require. For example, if a user's mailbox is accessible but the user can't get to an important public folder on another server, the Exchange service (or a subset of it) is unavailable—even though the user's mailbox server is running just fine. Likewise, if Exchange points of access (e.g., mailbox and public folder servers) are fully operational but all the bridgehead (i.e., routing) servers are down, mail won't flow between sites and routing groups or the Internet—thus, the Exchange service isn't completely available. The measurement of availability in your environment should be well thought out and should provide a picture of Exchange availability from your business's perspective.

Should you measure Exchange availability from the server's point of view (which can be a bit myopic), or should you primarily consider the client or user's perspective? The best approach seems to incorporate both viewpoints by focusing on the availability of Exchange "service elements," an approach that's in line with defining appropriate service levels for your Exchange environment. Service elements might include message routing, mailbox access, public folder access, protocol (e.g., POP3, IMAP4, HTTP, Messaging API—MAPI) access, recovery services, security protection (e.g., protecting against viruses, blocking spam), and other Exchange functionality that you can treat as a service for measurement purposes.

I'm not talking about something new or revolutionary—I'm encouraging you to change the way in which you think about and measure Exchange availability. Whether you use Exchange 2000 Server or Exchange Server 5.5, reevaluating your definition of downtime and availability and focusing on the availability of service elements can help you get a more accurate picture of how your Exchange environment operates and measure availability in a way that's meaningful to your organization and how it does business.

## Replication-Based Recovery Servers: Worth the Effort?

*By Jerry Cochran*

Many people have asked me about replication-based recovery servers for Exchange Server. The topic is worthy of a white paper, but I attempt to address the basics here.

The goal is to provide a reliable and supportable recovery server that can mirror or copy Exchange Information Store (IS) data in realtime or near-realtime. The primary server should also be able to transparently fail over to the recovery server, and the recovery server should be able to fail back to the primary server. Exchange doesn't inherently provide this type of functionality, but several third-party solutions exist. (Be aware, though, that Microsoft provides limited support for these types of setups.) Of the available solutions, two flavors exist: server mirroring and data replication.

At the core of all the available solutions is some sort of shared storage or I/O mechanism—typically a Storage Area Network (SAN). When you're building a data replication or mirroring solution for a mission-critical application, shared storage or I/O interconnection is imperative. Also, many SAN implementations offer built-in volume or controller mirroring and cloning. This capability simplifies data replication because the SAN hardware and software already support such replication.

Server mirroring solutions are typically proprietary solutions that involve both hardware and software. For example, Marathon Technologies' Endurance products combine special I/O interconnection hardware with specialized software to provide a server mirror of your Exchange server. In addition to mirroring the data at an I/O level, this solution provides a completely redundant hot-backup server that's paired to a production server. Mirroring solutions are virtually lockstep fault tolerant with the production server and data and are great for small server deployments or branch-office Exchange servers that need high availability. However, such products add too much complexity for most environments with large Exchange servers that have many users and stores.

Data replication solutions that build on SAN or Network Attached Storage (NAS) implementations are more mainstream and a bit more flexible. Solutions such as EMC's Symmetrix Remote Data Facility (SRDF) or Compaq's Data Replication Manager (DRM) replicate data volumes (e.g., the

transaction log volume, the database files volume) across controllers that can be across the room or across town. From the perspective of Exchange and the Extensible Storage Engine (ESE)/Joint Engine Technology (JET) database engine, data replication from the Exchange server to the redundant data set is transparent. The key to successful implementation of this solution lies in tuning the I/O replication operations according to the physical distance between redundant data sets and the response time that Exchange requires. Data replication solutions also typically support either synchronous or asynchronous I/Os; choose either, depending on factors such as replication distance, I/O load, and application requirements. As you might guess, Exchange is rather sensitive to I/O problems, so you must ensure that your implementation is well tested and well tuned. Otherwise, your high-availability solution will increase downtime instead of decreasing it.

You can also implement other measures (with or without a data replication solution) to provide more rapid disaster recovery for your Exchange servers. For example, if you use a SAN, you can deploy a Redundant Array of Independent Servers (RAIS)—spare servers that you attach to the shared storage and boot from the SAN in the event that a production server or IS fails.

In building a recovery solution, you need to target the area of Exchange that causes the most downtime in your environment. If you want to protect your Exchange servers and data and can justify the cost of server mirroring or data replication solutions, you can maximize your Exchange deployment's availability.

## FREE Download

Take a test drive of the de facto standard in data replication and protecting Exchange today! Introducing Double-Take from NSI Software. [http://www.nsisoftware.com/how/nsi\\_eval.asp?eve=657](http://www.nsisoftware.com/how/nsi_eval.asp?eve=657)

## Data Replication Technology for Your Exchange Deployments

*By Jerry Cochran*

Most of us are interested in finding new ways to make our Exchange servers more mission-critical. One technique that has started to surface more and more in Exchange Server deployments is data replication. Let's look at how this technology provides additional availability to Exchange.

It's important that I make one statement up front: As with data replication's storage technology relatives—Business Continuance Volumes (BCVs) and Network Attached Storage (NAS)—Microsoft does NOT directly support data replication for Exchange. That being said, you can leverage this technology, provided your data replication vendor supports it for Exchange servers.

Two types of data replication products are currently on the market: software-based and hardware-based. Software-based data replication products use a filter driver layered on top of the OS to mirror I/O operations at the OS I/O subsystem layer. Software-based solutions available include NSI's DoubleTake.

Hardware-based solutions are essentially the same except that I/Os are mirrored at the controller level (in hardware) and are independent of the OS. Products include EMC's Synchronous Remote Data Facility (SRDF), Compaq's Data Replication Manager (DRM), and Marathon's Endurance. In truth,

all of the hardware solutions are partially software-based, as they typically have controller firmware support and require device drivers and services running on the system to function.

You can use data replication for your Exchange servers by mirroring to a remote copy set I/O that occurs to Exchange databases and transaction logs. The remote copy set is an exact copy of the production data. If an Exchange database is lost or corrupted, you can remap these copy sets to the production data set and restart the server (Exchange 5.5) or mount the database (Exchange 2000). In addition, you can combine data replication technology with technologies such as clustering to further enhance availability. In the case of clustering, because you can store remote copy sets nearby or many kilometers away (via storage area network—SAN, LAN, or WAN), you can stretch the cluster across the distance. Even without clustering, you can locate remote copy sets at hot standby sites that provide catastrophe protection for Exchange data.

The challenge with data replication for Exchange comes in the area of transactional integrity. The main reason that Microsoft can't support this technology is that the mirroring operations that occur at the I/O level have no relation to actual database transactions. Also, because remote I/Os may or may not be committed synchronously (depending on the vendor and configuration) or in the same order as they occur at the transaction level, this technology is difficult to embrace from a supportability point of view. Recovery to remote copy sets is also not trivial or automatic. You must map the remote copy set as the production disk unit, and you must check database integrity to properly mount databases. Although you can script and automate this process, it must be done with care and is open to operator error.

I don't want to sell you on data replication for your Exchange deployments, but I do think it's worth a look. Recently, I've talked to more and more organizations that are considering this as a potential solution for more availability for Exchange and other applications. If you choose to look at this technology for your own deployment, factor in the points above, test it, and ensure your vendor of choice can carry the support burden when Microsoft won't.

## Tips for Maintaining Messaging Availability

*By Paul Robichaux*

If someone asked you to identify the single biggest weakness in your Exchange configuration, what would your response be? Some administrators would cite their hardware; others would point the finger at their network or Internet connection, and some would put the onus on their software. However, in a surprising number of cases, the true weak link is lack of knowledge about and planning for your environment. The fastest, least-expensive route to better uptime for your Exchange systems is to become smarter about how you do your job.

Now, let me make clear that I know administrators have a tough job and that products are often to blame for operational failures. We all have horror stories about unreliable hardware or poorly designed software that either did something wrong or made it too hard to figure out how to do things right. However, an unfortunate number of sites are forgoing some simple steps that can help raise their uptimes. In the spirit of helping banish unscheduled downtime (and productively filling what for some people might be a slow holiday period), let me present a few simple suggestions for improving your messaging availability.

**Know your SPOFs.** That Exchange server, laboring away in a corner, that has only one disk and one power supply is a SPOF, or single point of failure. Any component or service whose failure can knock out your Exchange server is a potential SPOF. Some SPOFs, such as electrical service, might be outside your control. Others, such as how much redundancy you specify when you buy hardware, are within your control, although you might not be able to change them now. But what you can certainly do is review your network and Exchange design to make sure you know what SPOFs exist and what to do in case one or more of them fails. In other words, you should know which Global Catalogs (GCs) your Exchange servers routinely talk to (use the Dsdiag tool), what kind of redundancy your hardware has, where you'd get spare parts in case of a failure, and so on.

**Revisit your procedures.** Do you know what to do in case of a virus outbreak on your servers? How about a failure of your RAID controller? What would you do if your office flooded over the weekend? You should have a clear plan of action for every failure or situation that can keep your users from getting their email. And you should write down those plans. Then, if you eat too much holiday turkey, another team member can follow the appropriate plan to restore service while you're sleeping off your overindulgence.

**Poke around.** Are you confident that your performance and availability monitoring is doing a good job? Dust off the Performance Monitor and use it to see what's really happening on your servers. While you're at it, make sure that your server and message monitors are functional—they won't do you any good if they silently fail.

**Play "If I Had a Million Dollars."** If the budget fairy paid you an unexpected visit, what would you spend the money on? New servers? More bandwidth? A team vacation to Cancun? Although you might not be able to actually follow through on these plans, it's always wise to know ahead of time what you need in the future.

**Get smart.** If your schedule and budget allow it, why not take a training class to bolster your knowledge in some Exchange-related area? If that's not possible, spend a few hours each week from now until the end of the year reading a good Exchange book or browsing the dozens of technical white papers that Microsoft makes available on its site.

These steps aren't magic, but if you take the time to carefully assess your SPOFs so that you have a better picture of the real weaknesses behind your network, you'll be much better prepared to fix them. Likewise, my other suggestions call for you to evaluate what you know, what you have, and what you do so that you can make the necessary changes. Happy tweaking!

## The 7 Habits of Highly Available Exchange Servers

*By Evan Morris*

Consulting about Microsoft Exchange Server availability is like watching the Loony Tunes' Wile E. Coyote: Watch for a while, and you can begin to predict the mistakes that lead to the falls. You also learn that the falls aren't as deadly as the pounding that follows close behind. After years of working with Exchange Server organizations, I've identified the factors that can lead to falls from high availability and the disaster recovery mistakes that can make these falls catastrophic. Inspired by Stephen R. Covey's bestseller *The 7 Habits of Highly Effective People* (Simon & Schuster, 1999), I've identified seven factors that help organizations prevent Exchange Server system failures and maintain high availability.

## ***Seek first to Understand Downtime***

Administrators must commit to solving the problems that decrease Exchange Server availability. Such problems fall into one of two categories: planned downtime or unplanned downtime. Planned downtime (e.g., applying service packs, upgrading hardware) is by far the easier category to manage. The best approach, when feasible, is to schedule planned downtime for nonbusiness hours.

Highly available Exchange Server organizations conduct risk assessments of unplanned downtime events. An important part of these assessments is the list you generate of possible downtime events. You can sort this list by the events' relative risks, then concentrate on preventing high-probability, high-impact events (e.g., Software Component A causing Software Component B to behave unexpectedly) and give less attention to the low-probability events (e.g., a meteorite striking your data center).

In my experience, software quality problems—bugs—are most often the cause of unplanned downtime. However, your response to outages—the decisions you make and the procedures you follow—determines the duration of the downtime. Unplanned downtime cycles have several stages, from problem identification through recovery. Understanding these stages and preparing yourself for action helps minimize downtime.

The first stage is notification that a problem exists. Automated notification systems—either built-in or added on—can detect hardware problems before they cause outages. OS- and application-level monitors, such as NetIQ's AppManager Suite and BMC Software's PATROL for Microsoft Exchange 2000 Servers, also aid in early problem detection. Undetected problems can lead to cascading failures that obscure the source problem. For example, suppose a mail connector queue fills Server A's hard disk. If this problem goes unnoticed, it might result in a connector on Server B failing to deliver messages to Server A. Thus, Server B appears to be the source of the problem, which diverts attention from the actual source: Server A.

The second stage is thorough problem analysis. Analysis helps you develop a troubleshooting course of action. The troubleshooting team must react quickly, but mistakes can be costly. The team members need to first isolate the problem to prevent further harm. Then, they must gather information about the problem, whether from tracking logs, Windows event logs, or the server operator's records of system changes.

Implementing and testing your recovery solution is the third stage. But don't consider the downtime cycle complete until the fourth stage: analysis of the lessons you've learned. Most unplanned downtime events contain lessons that can help you prevent a recurrence of the problem.

## ***Put Hardware First***

Hardware is the foundation of availability. Application stability doesn't matter if you don't run your applications on solid hardware. Fault-tolerant hardware often lets you repair hardware faults without taking systems down. Redundant components can keep systems running when the inevitable hardware faults occur. Hot-swappable components let you replace them without downtime.

RAID-protected hard disk subsystems are key to protecting your Exchange servers from the effects of hard disk failure. Best practice is to place Exchange Server log files on a RAID 1 volume and the database on a RAID 5 or, better yet, RAID 0+1 volume.

Storage planning is another important consideration. One organization's Exchange Server administrators told me that migrations to larger storage cabinets and more or larger hard disks were their servers' most significant sources of downtime (corporate policy prevented these administrators from enforcing mailbox limits). The organization was looking into a Storage Area Network (SAN) as a

solution. A SAN provides a high-performance pool of hard disks from which you can allocate storage to servers. SANs also simplify storage expansion, reconfiguration, and backup and recovery. However, transitioning to SAN-based storage can be difficult and can increase downtime.

### ***Clustering for a Win-Win Environment***

Clustering improves application reliability and helps prevent system failures. But the real beauty of clustering is that it can make even unreliable applications highly available to end users. For example, one day Node A in my 2-node Exchange Server 5.5 cluster began failing over to Node B. When I looked in the event log, I noticed that the failovers were occurring at 2-hour intervals. The person who installed the cluster had mistakenly installed an evaluation edition of Windows NT Server. When the 120-day evaluation period had expired, the OS began performing hard shutdowns every 2 hours. Clustering kept our Exchange Server system available to end users until we resolved the problem.

Clustering also helps you manage planned downtime. In a clustered environment, you can fail over Node A's services to Node B, then apply a service pack, hotfix, or upgrade to Node A.

Exchange Server 5.5 permits only 2-node active-passive clustering. Only the active node can perform Exchange Server processing. The passive node can't perform any processing until failover occurs. This limitation has lowered clustering's adoption rate, because 2-node active-passive clustering requires you to spend twice as much money on hardware without increasing processing capacity.

Exchange 2000 active-passive clusters are slightly different from Exchange Server 5.5 clusters: One node runs an Exchange Virtual Server (EVS) and the other has Exchange 2000 and doesn't run EVS until a failover occurs. Exchange 2000 with Service Pack 1 (SP1) permits 2-node active-active clustering on Windows 2000 Advanced Server. However, to ensure failover, you need to carefully distribute active user connections and keep processor utilization within the range that lets failovers occur. You can progress to 4-node clustering (i.e., 3+1 clustering) on Win2K Datacenter. Although you get better returns for your hardware investment when you cluster on Exchange 2000 and Win2K, you must still purchase special storage that lets two or more cluster nodes share a hard disk. Fibre channel SANs are a must for 3+1 clusters.

### ***Back Up with Restores in Mind***

A nasty crash can result in a corrupted Information Store (IS) that won't mount. This situation can necessitate a lengthy recovery process. Checking database integrity can take several hours. Eseutil, Exchange Server's primary integrity check and repair utility, could take an hour to check and repair a 15GB database, even with the fastest disk technology.

To a large extent, the techniques you employ for backing up your IS determine the length of the recovery process. If you plan for disaster recovery, you'll get back on your feet more quickly after a failure. Exchange Server 5.5 availability takes its biggest hit from the unpartitioned IS because when you need to restore this monolithic IS, you need to restore the entire IS. If you run Exchange 2000 Enterprise Server, you can partition the IS, which improves recovery time.

The most common approach to IS backups is doing full nightly backups to tape, then rotating the tapes off site. Database (.edb) restoration from tape drives runs at 15GB to 30GB per hour on the best DLT technology and more slowly on other tape technology or over the network.

Win2K's Ntbackup utility lets you perform online Exchange 2000 and Exchange Server 5.5 backups to disk. You can then back up the resulting .bkf file to tape and rotate the file off site. The advantage of this approach is that in the event of an IS problem, you can go directly to the

disk-based backup set instead of locating and loading a tape. Restores from disk are also typically faster than restores from tape.

If you're willing to spend the extra money, advanced backup techniques—cloning, snapshots, and data replication—lead to much faster recoveries and are approaches to consider as your situation requires (e.g., if you need to satisfy a service level agreement—SLA). Cloning is a function of RAID 0+1 mirroring. The clone is the third member of a triple mirrored set. Extracting the clone requires that you stop the Exchange Server services so that the database is consistent. This action immediately affects uptime, but SLAs typically permit such brief outages if they take place during off-hours.

To run utilities such as integrity checks, you can present the clone to another host on the SAN. You can then take the clone offline and back it up to tape. To restore a database that's been totally lost, you can make the clone stripe set the primary member of a new mirror set, then bring your Exchange Server system back online. Even if your database is large, you're back online in minutes instead of hours. The RAID controller will rebuild the mirror set in the background, with a negligible impact on performance.

A snapshot is a point-in-time copy of a disk. Snapshot software, running on the OS or at the RAID controller level, creates a disk map. As your source disk changes, your snapshot records those changes.

Some snapshot software lets you present the snapshot to other systems. This feature can be valuable if you need to test an application without risking the production database. Snapshots are also handy for individual item or mailbox restores. Restoring from a snapshot is similar to restoring from a tape backup. The traditional method for restoring individual items and mailboxes is to restore the entire IS to a recovery server. With a snapshot, you don't need to wait for a lengthy tape to restore; instead you can mount the snapshot to the recovery server and immediately extract the specific information that you want to recover.

To guarantee database integrity, creating a snapshot requires that you take offline all stores that share a disk volume. (In Exchange 2000, you dismount each Mailbox Store and Public Folder Store individually; in Exchange Server 5.5, you dismount the IS as a whole.) Some vendors provide snapshot technology with online backup capabilities, but database consistency is difficult to guarantee.

Data replication helps protect you against the most serious disasters, such as loss of the data center. Data replication can copy the IS in realtime to a distant location. The underlying technology (e.g., fibre channel, Asynchronous Transfer Mode—ATM) determines how distant this location can be. Data replication solutions typically involve specialized, high-end hardware (e.g., Marathon Technologies' Marathon Exchange Servers, Compaq SANworks Data Replication Manager) or specialized software (e.g., VERITAS Software's Storage Replicator), all of which can be expensive.

### **Monitor Proactively**

Proactively monitoring and maintaining your system can prevent downtime. Exchange Server's basic server and link monitoring tools provide limited functionality compared with third-party tools such as AppManager Suite and PATROL. You can monitor your servers at several levels: network, system hardware, OS, and application. The number of platforms you monitor and how you want the product to integrate with your systems will help you decide which product to use. But more important than what product you use is using it proactively: Respond to all early warnings to prevent detected problems from recurring or becoming more severe.

## ***Sharpen Your Network Defense***

Administrators of highly available Exchange Server organizations defend their systems vigorously against viruses and network attacks. Without a solid defense, you risk taking a hit to availability. I've seen an email virus outbreak shut down Exchange Server systems that previously had great availability track records. Cleaning up the aftereffects of such an outbreak can take hours.

A common network defense myth is that virus detection software is your most important method of protection. Virus scanning protects your systems against older known viruses but can't protect you against new viruses.

You also need to educate your users about how to recognize and dispose of suspicious attachments. You and your users need to configure systems in ways that limit the damage of virus attacks. Microsoft Outlook offers security patches, and Outlook 2002 will offer security options that help control virus attacks.

Although essential, purchasing antivirus software isn't enough. To sharpen your network defense, you need to stay on top of security bulletins and hotfixes. If you run Exchange 2000, you can take advantage of Win2K Server's security benefits.

## ***Synergize Expertise***

Organizations that have the most highly available Exchange Server systems have an amazing amount of inhouse expertise—although they might not have started out with such experts. Even if they did, ever-changing technology levels the field of high technology every few years. What organizations with highly available Exchange Server systems have in common is that they continually develop their inhouse expertise. And what they can't do, they outsource.

To be a high-availability system, a system's downtime must be less than 52 minutes per year. These 52 minutes don't leave much room for outages and planned downtime, so don't be discouraged if your system isn't one of the elite and highly available. Instead of counting downtime minutes, concentrate on developing these seven habits, and one day you'll be the Exchange Server expert whom others seek out.