

Windows Server 2008 Multi-Site Clustering

Technical Decision-Maker White Paper

Published: November 2007

For the latest information, please see   
[www.microsoft.com/windowsserver2008/failover-clusters.mspx](http://www.microsoft.com/windowsserver2008/failover-clusters.mspx)

Contents

[Introduction 1](#_Toc182969535)

[The Business Imperative for Geographically Dispersed Clusters 1](#_Toc182969536)

[Multi-Site Clustering 2](#_Toc182969537)

[How Multi-Site Clusters Work in Windows Server 2008 3](#_Toc182969538)

[Quorum Considerations 3](#_Toc182969539)

[Network Considerations 5](#_Toc182969540)

[Storage Considerations 5](#_Toc182969541)

[Multi-Site Clustering Use Cases 9](#_Toc182969542)

[Exchange Server 2007 10](#_Toc182969543)

[SQL Server 2005 and Other Cluster-Aware Server Workloads 11](#_Toc182969544)

[Conclusion and Final Considerations 13](#_Toc182969545)

[Related Links 14](#_Toc182969546)

# Introduction

High availability comes in many flavors. For many important business applications, highly reliable and redundant hardware provides sufficient uptime. For other business needs, the ability for a critical application to fail over to another server in the same data center is sufficient. However, neither of these server availability strategies will help in the event of truly catastrophic server loss.

For some business applications, even an event as unlikely as a fire, flood, or earthquake can pose an intolerable amount of risk to business operations. For example, the downtime between such an unlikely, large disaster striking and the time that it takes to restore service to a Microsoft® Exchange Server workload or a line-of-business (LOB) application can cost a larger business millions of dollars in productivity. For truly essential workloads, distance can provide the only hedge against catastrophe. By failing server workloads over to servers separated by hundreds of miles, truly disastrous data loss and application downtime can be prevented.

## The Business Imperative for Geographically Dispersed Clusters

Not all server workloads are created equal. Some applications and solutions hold disproportionate value to your organization. These might be the line-of-business application that represents your competitive advantage or the e-mail server that ties your far-flung organization together. The dark side of these essential IT functions is that they provide a hostage to fate—eventually something will take those services offline, severely hampering your company’s ability to operate, or even bringing business operations to a halt.

Redundant hardware on servers, redundant servers at data centers, and effective IT management all play a role in keeping these applications online and available to your employees and your customers. However, none of these precautions can prepare for large-scale server disruptions. Fires, floods, and earthquakes that can destroy or impair an entire data center are relatively rare, yet they do occur and, without adequate preparation, they can cost an organization millions of dollars in lost revenue and production. For truly large disasters, distance between server sites is the only thing that can keep a disruption from turning into a catastrophe.

Geographically dispersed clusters can form an important component in disaster preparation and recovery. In contrast to cold standby servers, the servers in a multi-site cluster provide automatic failover. This reduces the total service downtime in the case of a loss of a business-critical server. Another server in the cluster takes over service from the lost server as soon as the cluster determines that it has lost communication with the server previously running the critical workload, as opposed to users and customers waiting for human administrators to notice the service failure and bring a standby server online. And, because the failover is automatic, it lowers the overall complexity of the disaster-recovery plan.

The lower complexity afforded by automatic failover in a multi-site cluster also reduces administrative overhead. Changes made to applications and the application data housed on the cluster are automatically synchronized between all of the servers in the cluster. Backup and recovery solutions do this with a periodic snapshot of the standalone server being protected; meaning that the standby server may have a longer effective time to recovery. For example, if the backup software takes a snapshot every 30 minutes, even if the standby server is brought online at the disaster recovery site within 10 minutes, if the last 25 minutes of transactions have been lost with the primary server, the recovery might as well have taken 25 minutes (at least from the user’s point of view). Moreover, because the Windows Server® 2008 cluster service is specifically designed to keep application data changes consistent between dispersed clustered servers, it can be easier to keep the servers of a multi-site cluster consistent than a protected server and its designated standby.

Fundamentally, multi-site clustering can be a compelling part of disaster preparation and recovery because it removes human error from the equation. Your disaster recovery plans and procedures may have been prepared months, or even years, ago by people who are no longer around for business realities that may have changed—relying on such is inherently error prone. The primary reason why disaster recovery solutions fail is their dependence on people. Automating those plans in the form of a multi-site cluster can remove that factor and add to the disaster-resilience of your critical server workloads.

## Multi-Site Clustering

Clusters are defined as a set of servers (referred to as *nodes*) that together provide a highly available and highly scalable platform for hosting applications. *Failover clusters* in Windows Server 2008 provide a highly available environment for applications and services, such as Microsoft® SQL Server™, Exchange Server, file, print, and virtualization, by providing failover support: if one node fails while hosting an application, the application is failed over to another node in the cluster.

The failover mechanism in Windows Server 2008 is automatic. Failover clusters provide health monitoring capabilities; if the Windows Server 2008 failover cluster service detects that an application is not responsive, the application has failed, or the node hosting the application has failed, the cluster service ensures that the application fails over to another node in the cluster. The core principal in making server failover work is to have the Windows Server 2008 failover cluster service make certain that, regardless of failures of nodes or communication links between nodes, only a single instance of a given application is running at any point in time. This is crucial to avoiding data corruption or inconsistent results to the clients. To ensure this, all of the nodes that can host the application must be in the same Windows Server 2008 failover cluster. To provide disaster tolerance for these kinds of applications, a single Windows Server 2008 failover cluster must be stretched across multiple geographical sites.

A geographically dispersed (multi-site) cluster is a Windows Server 2008 failover cluster that has the following attributes:

* It has multiple storage arrays, with at least one storage array deployed at each site. This ensures that in the event of failure of any one site, the other site or sites will have local copies of the data that they can use to continue to provide the services and applications.
* Its nodes are connected to storage in such a way that, in the event of a failure of a site or the communication links between sites, the nodes on a given site can access the storage on that site. In other words, in a two-site configuration, the nodes in Site A are connected to the storage in Site A directly, and the nodes in Site B are connected to the storage in Site B directly. The nodes in Site A can continue without accessing the storage on Site B and vice versa.
* Its storage fabric or host-based software provides a way to mirror or replicate data between the sites so that each site has a copy of the data. There is no shared mass storage that all of the nodes access and data must thus be replicated between the separate storage arrays to which each node is attached.

Because of their extreme disaster tolerance, multi-site clusters should be thought of as both a high-availability solution and a disaster recovery solution. The automatic failover of Microsoft multi-site clustering means that your “backup” data on the other nodes of the cluster is available in moments upon failure of your primary site. What is more, automatic failover means that you can quickly failback to your primary site once your servers there have been restored.

# How Multi-Site Clusters Work in Windows Server 2008

Many of the benefits of multi-site clusters largely derive from the fact that they work slightly differently from conventional, local clusters. Setting up a cluster whose nodes are separated by hundreds, or even thousands, of miles will inform the choices you make on everything from the quorum model you choose to how you configure your network and data storage for the cluster. This section outlines some of the considerations unique to multi-site clustering and examines what they mean for this geographically dispersed high-availability, disaster recovery strategy.

## Quorum Considerations

The concept of quorum in Windows Server 2008 moves away from the requirement for a shared storage resource. The concept of quorum now refers to a number of “votes” that must equate to a majority. All nodes and a witness resource (a resource such as a file share that can break ties in the event that they occur) can get a vote to determine majority for cluster membership. This helps eliminate failure points in the old model, where it was assumed that the disk would always be available; if the disk failed, the cluster would fail. This makes the quorum model in Windows Server 2008 particularly well suited to geographically dispersed clusters.

Geographically dispersed clusters configured in Windows Server 2008 can be deployed in a way that automates the failover of applications in situations where the following occurs:

* Communication between sites has failed and the one site is still functioning.
* The other site is down and no long available to run applications.

Of the four quorum models available in Windows Server 2008, two are best suited to multi-site clustering: *Node and File Share Majority* and *Node Majority*. The node-and-file-share-majority quorum configuration enables the creation of up to 16-node clusters with no shared disk (previous versions of Windows Server allowed only two-node node-and-file-share-majority clusters). In this quorum configuration, a file share acts as a *witness*, a tie-breaking vote to the two votes provided by the two cluster nodes. Were it not for such a witness disk, the cluster as a whole would have only two votes total, meaning that the cluster would fail entirely with the loss of either one of the nodes. With the addition of the file share witness, the cluster has a total of three votes, meaning that connectivity can be lost by either one of the nodes (or the witness itself) and the cluster can continue functioning.

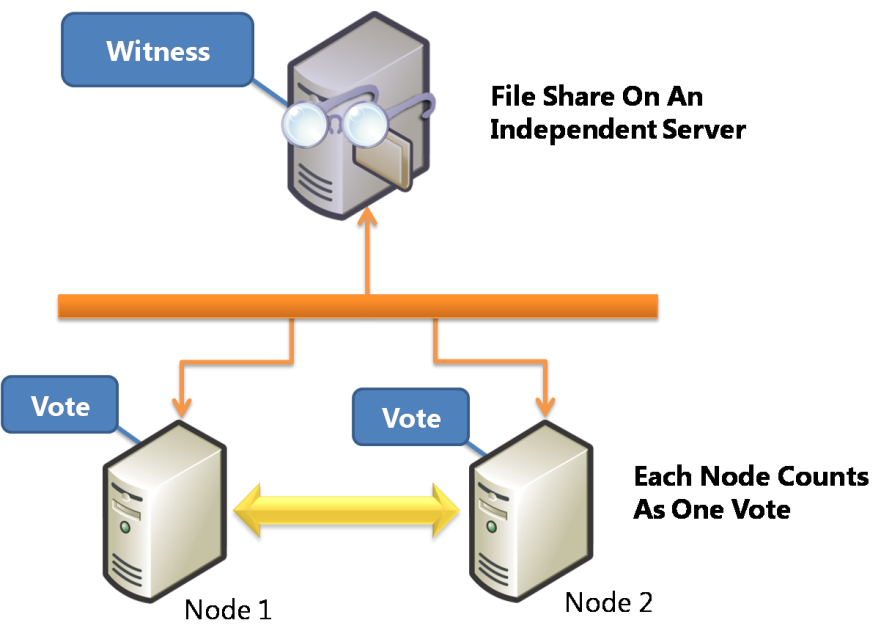


Figure 1 – Two-node node-and-file-share-majority cluster

A cluster quorum configured to use a node-and-file-share majority is an excellent solution for multi-site clusters. The file share witness can reside at a third site independent of either site hosting a cluster node for high disaster resilience. Moreover, a single file server can serve as a witness to multiple clusters (with each cluster using a separate file share witness on the file server).

If a file share witness at a site independent of your cluster sites is not an option for you, you can still use a multi-site cluster with a node-majority cluster configuration. A node-majority cluster consists of three of more cluster nodes without shared storage. In the language of the Windows Server 2008 quorum model, each of the nodes has a “vote” and there is no shared-storage or shared-disk “vote.” The result is that the majority of votes are necessary to operate the cluster: a geographically dispersed cluster with three nodes at three separate sites would continue to function if one of the sites were unavailable, but would cease to function if two sites became unavailable.

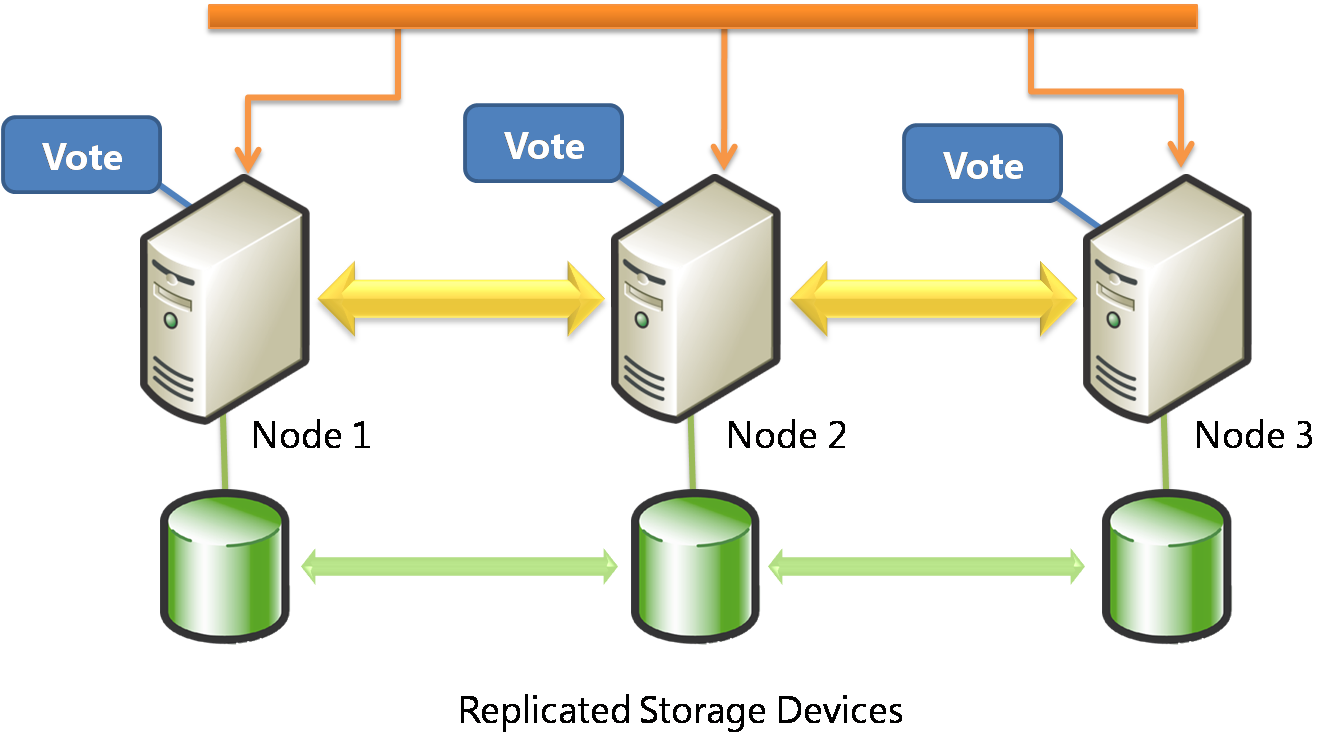


Figure 2 – Node-majority cluster

**Note:** It is not enough to have an even half of the cluster nodes functioning in this model. If four nodes were set up in a node-majority configuration, the cluster would continue to operate with the loss of one node but not with the loss of two nodes. For this reason the node-majority quorum configuration works best with an odd number of cluster nodes.

The node-majority quorum configuration can work when there is more than one cluster node at each site. Take the case of a multi-site cluster consisting of five nodes, three of which reside at Site A and the remaining two at Site B. With a break in communication between the two sites, Site A can still communicate with three nodes (which is greater than 50 percent of the total), so all of the nodes at Site A stay up. The nodes in Site B are able to communicate with each other, but no one else. Since the two nodes at Site B cannot communicate with the majority, they drop out of cluster membership. (Were Site A is to go down in this case, in order to bring up the cluster at Site B, it would require manual intervention to override the non-majority.)

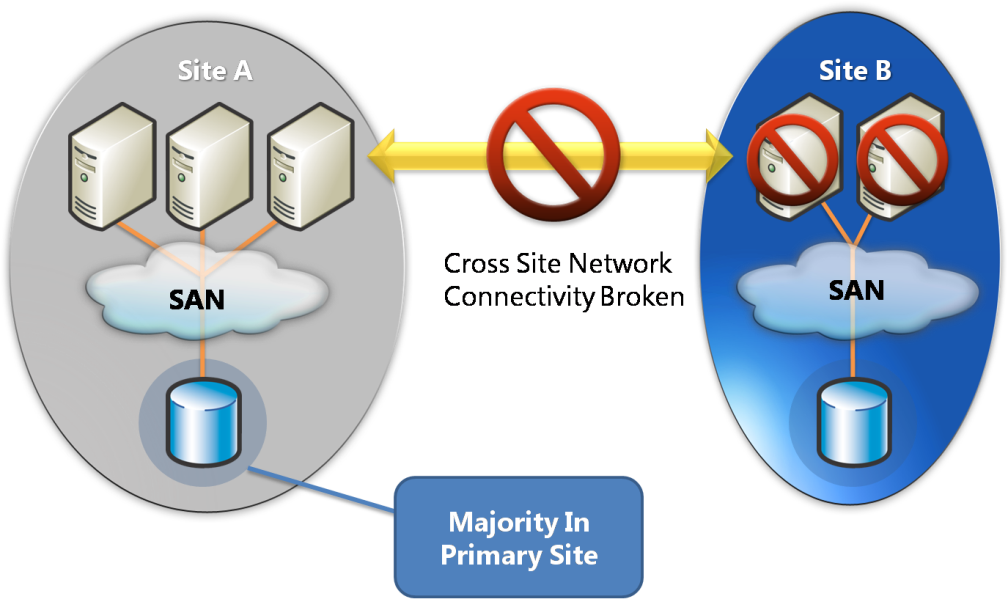


Figure 3 – Node-majority cluster with multiple nodes at both sites

There are a number of aspects of multi-site clustering to consider when planning your geographically dispersed cluster.

## Network Considerations

A major improvement to clustering in Windows Server 2008 is that cluster nodes can now reside on different subnets. As opposed to previous versions of clustering (as in Windows Server® 2003 and Windows® 2000 Server), cluster nodes in Windows Server 2008 can communicate across network routers. This means that you no longer have to stretch virtual local area networks (VLANs) to connect geographically separated cluster nodes, far reducing the complexity and cost of setting up and maintaining multi-site clusters.

One consideration for subnet-spanning clusters can be client response time. Client computers cannot see a failed-over workload any faster than the DNS servers can update one another to point clients to the new server hosting that workload. For this reason, VLANs can make sense when keeping workload downtime to an absolute minimum is your highest priority.

## Storage Considerations

Since there is no shared mass storage for all of the nodes in a multi-site cluster, data must be replicated between the storage devices used by the nodes at each site in the cluster.

Consider the case of a cluster dispersed between two sites, Site A and Site B. Traditionally when a cluster node at Site A makes a change to its local storage, the changes are then replicated off to Site B. The logical unit numbers (LUNs) of the storage at Site A would generally be in a Read-Write mode, while the replica at Site B would be in a Read-Only mode to ensure data consistency. When a failover occurs, Site B must switch its replica LUNs from Read-Only to Read-Write and then in turn replicate its changes to the other site (Site A).

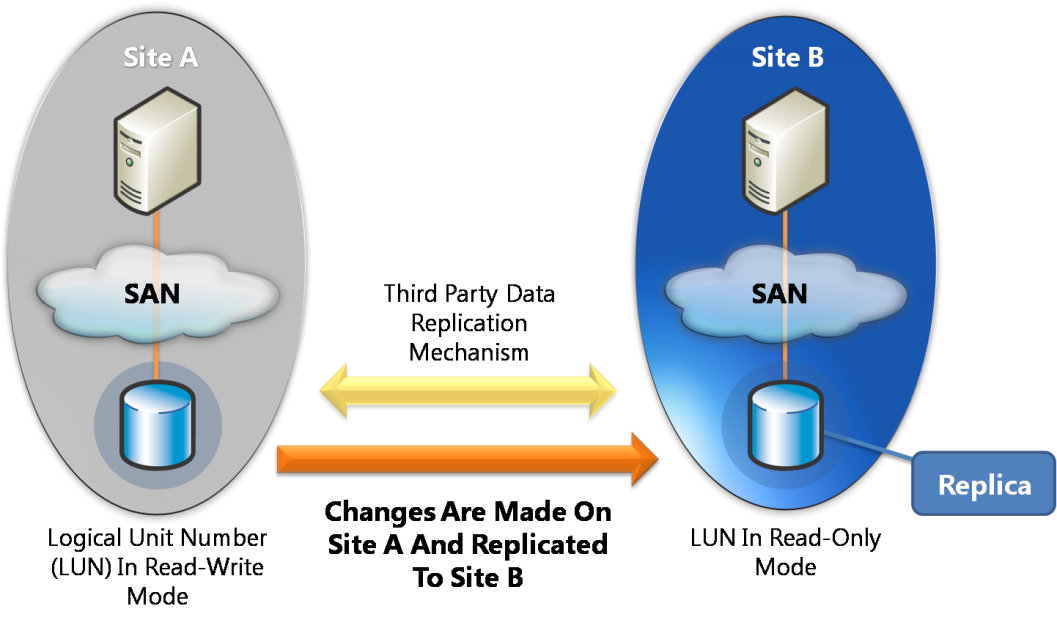


Figure 4 – Multi-site cluster data replication

Data replication is thus the backbone of geographically dispersed clustering. Data can be replicated between sites within a multi-site cluster by different techniques, and at different levels in the clustering infrastructure. At the hardware level (the block level), replication is performed by the storage controllers or by mirroring the software. At the file-system level (that is, replicating file system changes), the host software performs the replication. Finally, at the application level, the applications themselves can replicate data. An example of application-level replication is Microsoft® Exchange Server 2007 Continuous Cluster Replication (CCR). The method of data replication that you use depends on the requirements of the application and the business needs of your organization for your multi-site cluster.

When planning geographically dispersed clusters, you need to understand your data consistency needs in different failure and recovery scenarios and work with the solution vendors to meet your requirements. Different geographically dispersed cluster solutions provide different replication and redundancy strategies. Determine the support requirements of your applications with regard to replication—in geographically dispersed server clusters, the type of data replication is just as important as the level at which it occurs. There are two types of data replication: synchronous and asynchronous.

*Synchronous replication* is when an application performs an operation on one node at one site, and then that operation is not completed until the change has been made on the other sites. So, synchronous data replication holds the promise of no data loss in the event of failover for multi-site clusters that can take advantage of it.

Using synchronous, block-level replication as an example, if an application at Site A writes a block of data to a disk mirrored to Site B, the input/output (I/O) operation will not be completed until the change has been made to both the disk on Site A and the disk on Site B.

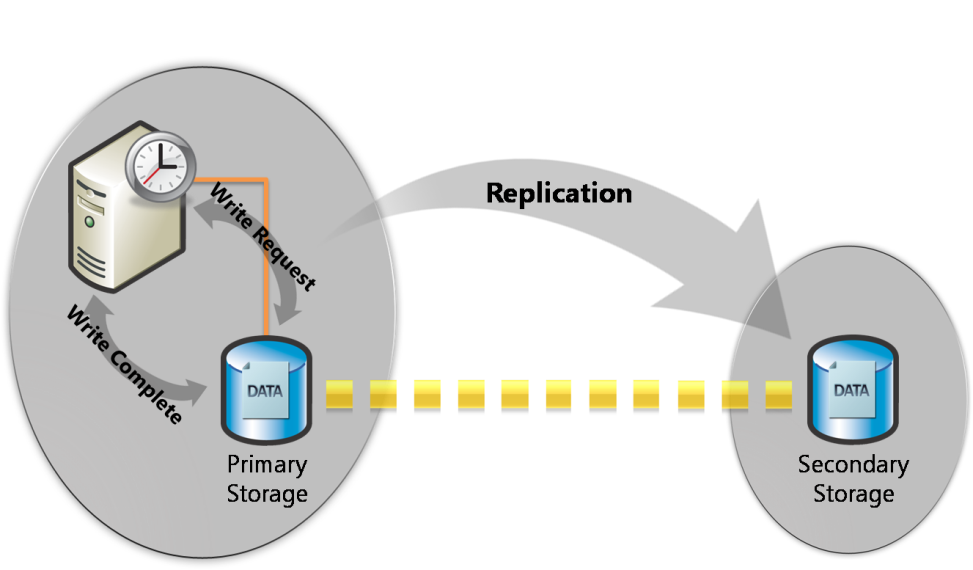
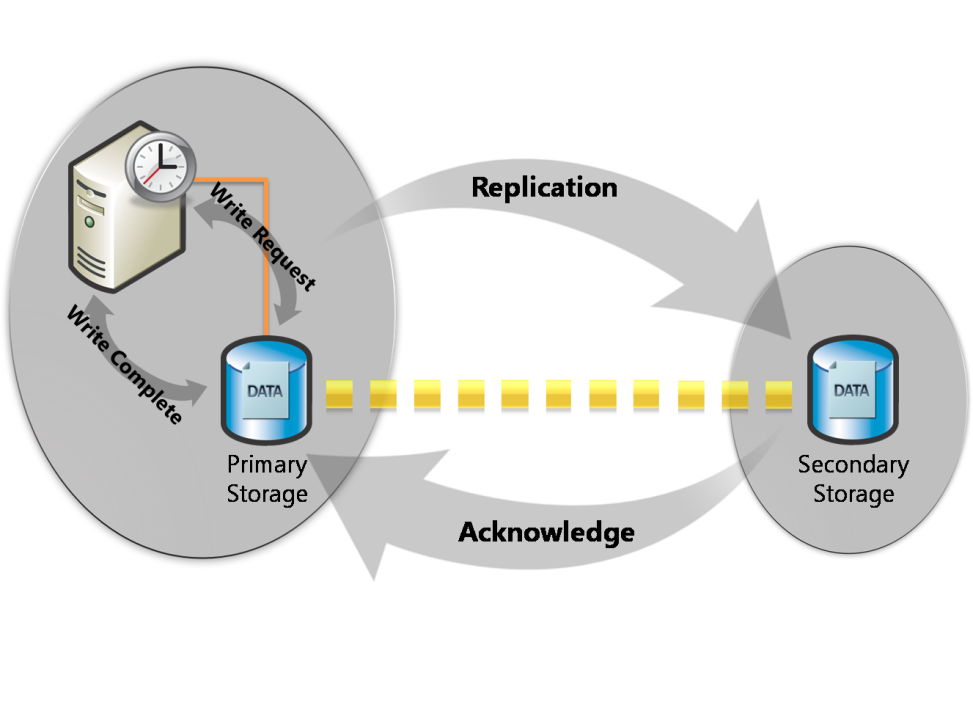


Figure 5 – Synchronous replication

In general, synchronous data replication is best for multi-site clusters that can rely on high-bandwidth, low-latency connections. Typically, this will limit the application of synchronous data replication to geographically dispersed clusters whose nodes are separated by shorter distances. While synchronous data replication protects against data loss in the event of failover for multi-site clusters, it comes at the cost of the latencies of application write and acknowledgement times impacting application performance. Because of this potential latency, synchronous replication can slow or otherwise detract from application performance for your users.

*Asynchronous replication* is when a change is made to the data on Site A and that change eventually makes it to Site B. Multi-site clusters using asynchronous data replication can generally stretch over greater geographical distances with no significant application performance impact. In asynchronous replication, if an application at Site A writes a block of data to a disk mirrored to Site B, then the I/O operation is complete as soon as the change is made to the disk at Site A. The replication software transfers the change to Site B (in the background) and eventually makes that change to Site B.



**Figure 6 – Asynchronous replication**

When exploring the increased performance asynchronous replication affords over clusters using synchronous data replication, the primary consideration to keep in mind is data consistency.

In the event of the hard failure of a node in a multi-site cluster using asynchronous data replication, you must be prepared to lose some data. With asynchronous replication, the data at Site B can be out of date with respect to Site A at any point in time. This is because a node may fail after it has written an application transaction to storage locally but before it has successfully replicated that transaction to the other site or sites in the cluster; if that site goes down, the application failing over to another node will be unaware that the lost transaction ever took place.

Preserving the order of application operations written to storage is also an issue with asynchronous data replication. Different vendors implement asynchronous replication in different ways. Some preserve the order of operations and others do not. This distinction is crucial, because if a solution preserves ordering, then the disk at Site B might be out of date, but it will always represent a state that existed at Site A at some point in the past. This means Site B is *crash consistent*: the data at Site B represents the data that would exist at Site A had Site A crashed at that point in time. Conversely, if a solution does not preserve ordering, the I/O operations might be applied at Site B in an arbitrary order. In this case, the data set at Site B might never have existed at Site A.

**Many applications can recover from crash-consistent states; very few can recover from out-of-order I/O operation sequences.** Hence it is vital that geographically dispersed server clusters never use asynchronous replication unless the order of I/O operations is preserved; if this order is not preserved, the data that is replicated to the second site can appear corrupt to the application and be totally unusable.

Using Distributed File System Replication (DFS-R) in Windows Server 2008 to replicate data across the nodes of a geographically dispersed cluster is tantalizing, but DFS-R is unsuitable for multi-site cluster data replication (nor is it supported). DFS-R only performs its data replication upon closure of a file. This works well for files such as documents, presentations, or spreadsheets, but it will not work for files that are held open, such as databases. Except in the case of Exchange Server 2007 Mailbox servers (discussed later in this paper), you will have to invest in a third-party data replication solution to synchronize the data at each node of your multi-site cluster.

# Multi-Site Clustering Use Cases

This section discusses the role of Windows Server 2008 multi-site clustering in the context of two of its most important use cases: increasing the availability of Microsoft Exchange Server 2007 and Microsoft SQL Server 2005. While these two applications serve very different functions for businesses, the high-availability issues surrounding both have some important commonalities.

Bear in mind that it is generally very easy to have inappropriate expectations regarding availability targets for Exchange Server and SQL Server. It is also easy for organizations to demand higher levels of availability than they are actually willing to pay for before the cost implications are understood. This is often because the availability of a service is a complex issue that spans many disciplines. Many different approaches can be taken to deliver the required levels of availability; multi-site clustering is just one of these, and as you have already seen in this white paper, there are a number of cost implications and considerations for this combined high-availability and disaster recovery approach.

However, availability requirements can frequently be expressed in relatively simplistic terms by the customer and without a full understanding of the implications. This situation can lead to misunderstandings between the customer and the information technology organization, inappropriate levels of investment, and, ultimately, to customer dissatisfaction through inappropriate expectations being set.

One expressed requirement for 99.5 percent availability can be different from another requirement for 99.5 percent availability. One requirement may discuss the availability of the hardware platform alone, and the other requirement may discuss the availability of the complete end-to-end service. Even the definition of complete end-to-end service availability can vary greatly. It is important to understand exactly how any availability requirements are to be measured. For example:

* If all hardware and software on the primary server are functioning correctly and user connections are ready to be accepted by the application, is the solution considered 100 percent available?
* If there are 100 users but 25 percent of them cannot connect because of a local network failure, is the solution still considered 100 percent available?
* If only one user out of the 100 users can connect and process work, is the solution considered only 1 percent available?
* If all 100 users can connect but the service is degraded with only two out of three customer transactions being available, or performance is poor, how does this affect availability measurements?

Multi-site clustering is a high-availability strategy that primarily speaks to hardware platform availability, but how a specific multi-site cluster is configured and deployed will have availability ramifications ranging from the ability of users to connect to the application to the quality of performance of the application. Multi-site clustering can be a powerful solution in dealing with planned and unplanned downtime, but its benefits must be examined against all of the dimensions of application availability.

Beyond the common need to fully examine your availability needs, Exchange Server 2007 and SQL Server 2005 implement Windows Server 2008 multi-site clustering in different ways. Let’s examine some of the specifics of those implementations now.

## Exchange Server 2007

The core of Exchange Server 2007 running on Windows Server 2008 multi-site clusters is Cluster Continuous Replication (CCR). CCR is a high-availability feature of Exchange Server 2007 that combines the asynchronous log shipping and replay technology built into Exchange Server 2007 with the failover and management features provided by the Windows Server 2008 failover cluster service.

For purposes of our discussion, CCR combines the following elements:

* Failover provided by a Windows Server 2008 multi-site cluster.
* Transaction log replication and replay features provided by Exchange Server 2007.
* Message queue feature of the Hub Transport server, called the transport dumpster in Exchange Server 2007.

CCR uses two nodes joined in a single Windows Server 2008 failover cluster. The nodes in the failover cluster host a single clustered Mailbox server. The node that is currently running the clustered Mailbox server is the *active node*, and a node that is not running a clustered Mailbox server (but is part of the cluster) is the *passive node*.

CCR supports two-node clusters, and geographically dispersed Exchange Server 2007 clusters use the node-and-file-share-majority quorum configuration discussed in the [Quorum Considerations](#_Quorum_Considerations) section earlier in this document.

CCR is designed to provide high availability for Exchange Server 2007 Mailbox servers by providing a solution that:

* Has no single point of failure.
* Has no special hardware requirements.
* Has no shared storage requirements.
* Can be deployed in a two-site configuration.
* Can reduce full backup frequency, reduce total backed up data volume, and shorten the service level agreement (SLA) for recovery time from first failure.

CCR uses the database failure recovery functionality in Exchange Server 2007 to enable the continuous and asynchronous updating of a second copy of a database with the changes that have been made to the active copy of the database. During installation of the passive node in a CCR environment, each storage group and its database is copied from the active node to the passive node. This operation provides a baseline of the database for replication. After this initial database seeding operation is performed, log copying and replay are performed continuously.

In addition to providing data and service availability in the event of unplanned downtime, CCR also provides for scheduled outages (planned downtime). When updates need to be installed or when maintenance needs to be performed on a clustered mailbox server, an administrator can move the Mailbox server workload (called an Exchange Virtual Server in previous versions of Exchange Server) manually to the passive node. After the move operation is complete, the administrator can then perform the needed maintenance.

The key benefits of CCR in a multi-site cluster scenario are the following:

* **End-to-end multi-site cluster solution.** Exchange Server 2007 CCR has built-in support for data replication across geographically dispersed failover clusters. This means that you do not need a third-party replication solution.
* **Continuous replication is asynchronous.** Logs are not copied until they are closed and no longer in use by the Mailbox server. This means that the passive node usually does not have a copy of every log file that exists on the active node. (The one exception is when the administrator has initiated a scheduled outage of the active node to apply an update or perform some other maintenance.)
* **Continuous replication places almost no CPU and input/output (I/O) load on the active node during normal operation.** CCR uses the passive node to copy and replay the logs. Logs are accessed by the passive node via a secured file share. In this way, the continuous replication has little impact on performance of the Mailbox server.
* **Active and passive node changes over the lifetime of the cluster are designated automatically.** For example, after a failover, the active and passive designation reverses. This means the direction of replication reverses. No administrative action is required to reverse the replication. The system manages the replication reversal automatically, which reduces administrative overhead and complexity.
* **Failover and scheduled outages are symmetric in function and performance.** It takes just as long to fail over from Node 1 to Node 2 as it does to fail over from Node 2 to Node 1. Typically, this would be under two minutes. On larger servers, scheduled outages typically would be less than four minutes. (The time difference between a failover and scheduled outages is associated with the time it takes to do a controlled shutdown of the active node on a scheduled outage.)
* **Volume Shadow Copy Service (VSS) backups on the passive node are supported.** This enables administrators to offload backups from the active node and extend the backup window. In addition, larger configurations are not obligated by performance requirements to have hardware VSS support to use VSS backups. The workload on the passive node is primarily log copying and log replay, neither of which is real-time constrained like the clustered Mailbox server on the active node. For example, the active node has to respond to client requests in a timely way. A longer backup window can be used: because the passive node has no real-time response constraints, this allows for larger databases and larger mailbox sizes.
* **Total data on backup media is reduced.** The CCR passive copy provides the first line of defense against corruption and data loss. Thus, a double failure is required before backups will be needed. Recovery from the first failure can have a relatively short SLA because no restore is required. Recovery from the second failure can have a much longer SLA. As a result, backups can be done on a weekly full cycle with a daily incremental backup strategy. This reduces the total volume of data that must be placed on the backup media.

## SQL Server 2005 and Other Cluster-Aware Server Workloads

Multi-site clustering in Windows Server 2008 Enterprise is transparent to SQL Server 2005. Because SLQ Server is already cluster aware, SQL Server is designed to work with Windows Server 2008 failover clustering. What is more, SQL Server 2005 does not see a multi-site cluster as being any different from a local cluster, and so you do not need to make any additional configurations to SQL Server 2005 to take advantage of the combined high-availability and disaster recovery benefits of multi-site clustering.

Other Windows Server 2008 services are cluster aware and, similarly to SQL Server, they require no special configuration to go from working on a local failover cluster to a multi-site failover cluster. These services include Dynamic Host Configuration Protocol (DHCP), Windows Internet Name Service (WINS), file sharing, and print sharing.

The one cardinal consideration to keep in mind with deploying all of these workloads on multi-site clustering is data replication. As opposed to local clusters, multi-site failover clusters share no central storage. Cluster data must be replicated and synchronized between all of the sites of a multi-site cluster. You will therefore need a third-party data replication solution. However, Windows Server 2008 takes care of the rest of the clustering needs (heartbeat monitoring, failover, etc.) in these deployment scenarios.

# Conclusion and Final Considerations

Multi-site clusters can provide tremendous business benefits. As with many powerful business solutions, there are some small trade-offs to take into account when considering that the extreme disaster resilience of geographically dispersed clusters comes at the price of increased cost and greater complexity than local clusters of servers.

Multi-site clusters do require a little more overhead than local clusters. As opposed to a local cluster, in which each node of the cluster is attached to the mass storage device, each site of a multi-site cluster must have comparable storage. In addition, you will also need to consider vendors to set up your data replication schemes between cluster sites, possibly pay for additional network bandwidth between sites, and develop the management wherewithal within your organization to efficiently administer your multi-site cluster.

However, the improvements to failover clustering in Windows Server 2008 make multi-site clusters more resilient and easier to set up. From a new quorum model to the ability to span VLANs, geographically dispersed clusters are more feasible in a wider variety of situations in Windows Server 2008. These improvements lower the costs of setting up and administering geographically dispersed failover clusters; they can also tip the trade-off between the increased overhead of multi-site clusters and the pain of losing an essential application in favor of multi-site clusters in more cases. This extends the benefits of this unique combination of high availability and disaster recovery solution for a broader array of server workloads and more for organizations.

Windows Server 2008 multi-site clusters provide Exchange Server 2007, SQL Server 2005, and other services with higher resilience in the face of disasters. Because multi-site clusters are geographically dispersed, they remove the close proximity of servers as a potential point of failure for Exchange Server 2007 Mailbox servers and SQL Server 2005 databases, as well as for DHCP, WINS, and file and print servers. Moreover, because each site of a multi-site cluster uses its own distinct local storage, Windows Server 2008 multi-site cluster for Exchange Server 2007 and SQL Server 2005 workloads does not require specialized storage hardware. Best of all, Exchange Server 2007 Cluster Continuous Replication on Windows Server 2008 Enterprise provides an end-to-end multi-site cluster solution that requires no third-party data replication solution to work.

# Related Links

For the latest information about failover clustering in Windows Server 2008, see  
[www.microsoft.com/windowsserver2008/failover-clusters.mspx](http://www.microsoft.com/windowsserver2008/failover-clusters.mspx)

For more information about cluster continuous replication in Exchange Server 2007, see  
[technet.microsoft.com/en-us/library/bb124521.aspx](http://technet.microsoft.com/en-us/library/bb124521.aspx)

For an overview of high availability in SQL Server 2005, see  
[www.microsoft.com/technet/prodtechnol/sql/2005/sqlydba.mspx#EED](http://www.microsoft.com/technet/prodtechnol/sql/2005/sqlydba.mspx#EED)

To download a white paper on failover clustering with SQL Server 2005, see  
[www.microsoft.com/downloads/details.aspx?FamilyID=818234DC-A17B-4F09-B282-C6830FEAD499&displaylang=en](http://www.microsoft.com/downloads/details.aspx?FamilyID=818234DC-A17B-4F09-B282-C6830FEAD499&displaylang=en)

This is a preliminary document and may be changed substantially prior to final commercial release of the software described herein.

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This white paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS   
DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in, or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2007 Microsoft Corporation. All rights reserved.

Microsoft, SQL Server, Windows, Windows Server, and the Windows logo are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.