CARBONITE

Carbonite Availability

GeoCluster User's Guide

Notices

Carbonite Availability GeoCluster User's Guide Version 8.1.1, Tuesday, September 26, 2017

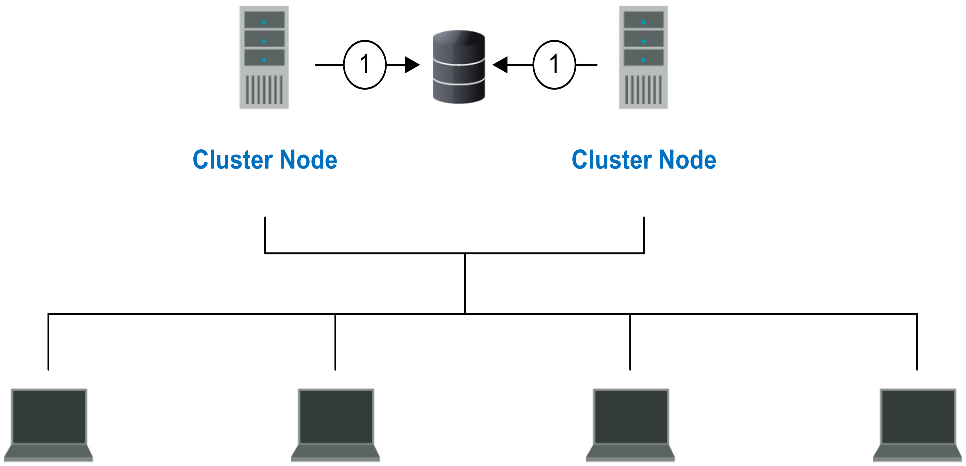
If you need technical assistance, you can contact CustomerCare. All basic configurations outlined in the online documentation will be supported through CustomerCare. Assistance and support for advanced configurations may be referred to a Pre-Sales Systems Engineer or to Professional Services.

Man pages are installed and available on Carbonite Availability and Carbonite Move Linux servers. These documents are bound by the same Carbonite license agreement as the software installation.

Contents

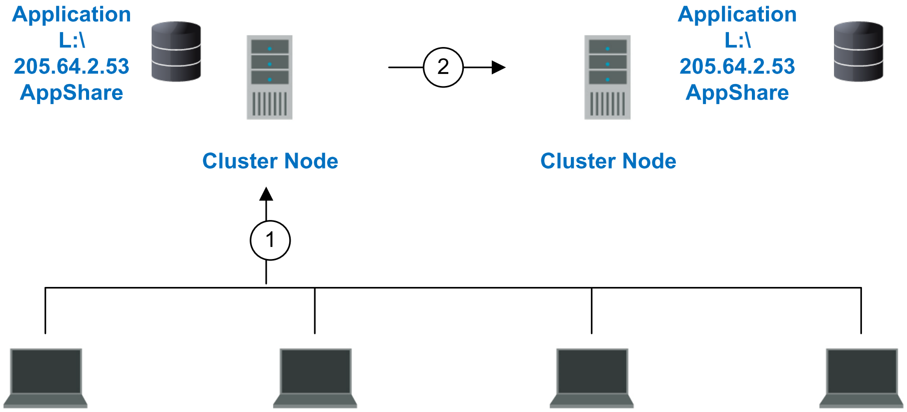
Chapter 1 GeoCluster overview

In a standard cluster configuration, a single copy of data resides on a SCSI disk that is shared between cluster nodes. Data is available without users knowing which node owns a cluster resource. MSCS handles failover between nodes of the cluster.

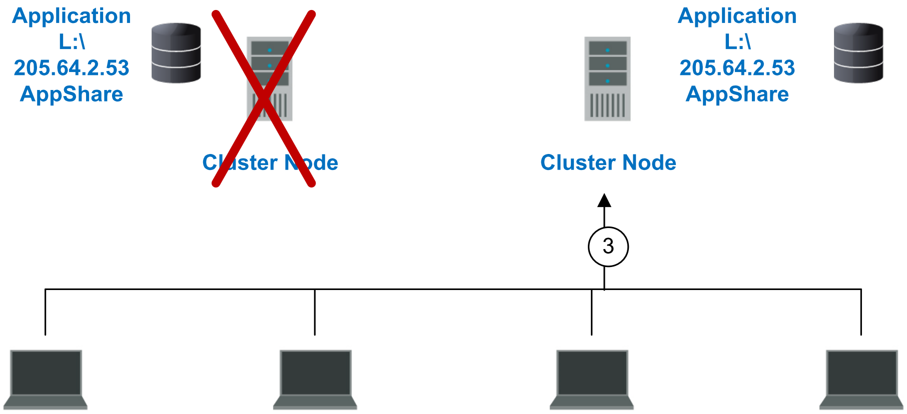


1. The source cluster nodes share data from a single SCSI disk.

In a GeoCluster configuration, data is stored on volumes local to each node and replicated to each node in the cluster using Carbonite Availability. This eliminates the single point of failure of a standard cluster (shared disk) configuration. With GeoCluster, resources and groups are handled in the same manner as a standard cluster. Instead of assigning one group by SCSI drive, you assign one group per logical volume. If a server, disk, group, or network interface should fail, MSCS relocates the failed group to another node, which contains the replicated copy of the data, thus maintaining availability.



1. Users access data from the owning node.
2. Data is mirrored and replicated between nodes of the cluster.



3. In the event the owning node changes, users access data from the new owning node.

Chapter 2 GeoCluster requirements

Your cluster must meet the minimum requirements below.

- **Operating system**—The following operating systems are supported
  - Windows 2008 R2 Service Pack 1 Enterprise and Datacenter editions
  - Windows 2012 and 2012 R2 Standard and Datacenter editions
- **Hardware**—Microsoft support for MSCS and MSCS-based Microsoft applications requires that the cluster configuration appear on the Microsoft Hardware Compatibility List under category Cluster.
- **Cluster Network Name**—Carbonite Availability does not handle dynamic changes to the cluster network names, the names assigned to the routes for network traffic. If a network name is changed for a network that is used by Carbonite Availability, the GeoCluster Replicated Disk resource must be taken offline, the resource's network property must be changed, and then the resource must be brought back online.
- **Server name**—Carbonite Availability includes Unicode file system support, but your server name must still be in ASCII format. If you have the need to use a server's fully-qualified domain name, your server cannot start with a numeric character because that will be interpreted as an IP address. Additionally, all Carbonite Availability servers must have a unique server name.

  If you need to rename a server that already has a Carbonite Availability license applied to it, you should deactivate that license before changing the server name. That includes rebuilding a server or changing the case (capitalization) of the server name (upper or lower case or any combination of case). If you have already rebuilt the server or changed the server name or case, you will have to perform a host-transfer to continue using that license. See the *Carbonite Availability and Carbonite Move Installation, Licensing, and Activation* document for complete details.

- **File system**—Carbonite Availability supports the NTFS file system. FAT, FAT32, and ReFS are not supported. For detailed information on other file system capabilities, see *Mirroring and replication capabilities* on page 9.
- **System memory**—The minimum system memory on each server is 1 GB.
- **Disk space for program files**—This is the amount of disk space needed for the Carbonite Availability program files. The amount depends on your operating system version and ranges from 350-500 MB.

  The program files can be installed to any volume while the Microsoft Windows Installer files are automatically installed to the operating system boot volume.

  Make sure you have additional disk space for Carbonite Availability queuing, logging, and so on.

- **Disk queuing**—The Carbonite Availability disk queue, configured during installation, should use a local volume for each node in the cluster.
- **Protocols and networking**—Your servers must meet the following protocol and networking

requirements.

  - Your servers must have TCP/IP with static IP addressing.
  - IPv4 only configurations are supported, IPv4 and IPv6 are supported in combination, however IPv6 only configurations are not supported
  - By default, Carbonite Availability is configured for IPv6 and IPv4 environments, but the service will automatically check the server at service startup and modify the appropriate setting if the server is only configured for IPv4.
  - If you are using IPv6 on your servers, your clients must be run from an IPv6 capable machine.
  - In order to properly resolve IPv6 addresses to a hostname, a reverse lookup entry should be made in DNS.
  - Multiple networks are recommended to isolate public and private traffic.
  - The private network should be a unique subnet so that Carbonite Availability will not attempt to use an unreachable private network.
  - Your network can contain direct LAN connections or VLAN technology.
  - If you are using Carbonite Availability over a WAN and do not have DNS name resolution, you will need to add the host names to the local hosts file on each server running Carbonite Availability.
- **Domain**—The cluster nodes must be members of the same domain.
- **DNS**—Forward and reverse lookups must be implemented on the primary DNS server for the cluster name and individual nodes.
- **Volumes**—The source and target should have identical drive mappings.
- **Third-party storage**—Third-party storage resources are not supported.
- **Resource registration**—In some cases, the Carbonite Availability cluster resources may not be registered automatically when Carbonite Availability is installed. You can manually register the resources by running DTResUtility.exe, which is installed in the \Windows\Cluster directory.
- **Licensing**—Each node in the cluster must have a valid Carbonite Availability license key.
- **Time**—The clock on your Carbonite Availability servers must be within a few minutes of each other, relative to UTC. Large time skews (more than five minutes) will cause Carbonite Availability errors.
- **Windows firewall**—If you have Windows firewall enabled on your servers, there are two requirements for the Windows firewall configuration.
  - The Carbonite Availability installation program will automatically attempt to configure ports 6320, 6325, and 6326 for Carbonite Availability. If you cancel this step, you will have to configure the ports manually.
  - If you are using the Carbonite Replication Console to push installations out to your Windows servers, you will have to open firewall ports for WMI (Windows Management Instrumentation), which uses RPC (Remote Procedure Call). By default, RPC will use ports at random above 1024, and these ports must be open on your firewall. RPC ports can be configured to a specific range by specific registry changes and a reboot. See the Microsoft Knowledge Base article 154596 for instructions. Additionally, you will need to

open firewall ports for SMB (server message block) communications which uses ports 135-139 and port 445, and you will need to open File and Printer Sharing. As an alternative, you can disable the Windows firewall temporarily until the push installations are complete.

- **Anti-virus software**—You should configure your anti-virus software to delete or quarantine viruses because cleaning them can cause an access denied retrying operation error. Additionally, configuring virus software to scan outgoing traffic will lessen performance impacts.
- **Windows Management Instrumentation (WMI)**—Carbonite Availability is dependent on the WMI service. If you do not use this service in your environment, contact technical support.

Mirroring and replication capabilities

For Windows source servers, Carbonite Availability mirrors and replicates file and directory data stored on any NTFS Windows file system. Mirrored and replicated items also include Macintosh files, compressed files, NTFS attributes and ACLs (access control list), dynamic volumes, files with alternate data streams, sparse files, encrypted files, reparse points, and hard links. Files can be mirrored and replicated across mount points, although mount points are not created on the target.

Carbonite Availability does not mirror or replicate items that are not stored on the file system, such as physical volume data and registry based data. Additionally, Carbonite Availability does not mirror or replicate NTFS extended attributes, registry hive files, Windows or any system or driver pagefile, system metadata files ($LogFile, $Mft, $BitMap, $Extend\\$UsnJrnl, $Extend\\$Quota, and $Extend\\$ObjId), or the Carbonite Availability disk-based queue logs. The only exception to these exclusions is for the full server job types. If you are protecting your system state and data using full server protection, Carbonite Availability will automatically gather and replicate all necessary system state data, including files for the operating system and applications. Additionally, since Volume Shadow Copy snapshots are associated with the volume they belong to and Carbonite Availability mirrors and replicates the data on the volume and not the volume itself, snapshots taken on the source cannot be used on the target's volume. Therefore, snapshots taken on the source are not mirrored or replicated to the target.

Note the following replication caveats.

1. FAT and FAT32 are not supported.
2. ReFS is not supported.
3. You cannot replicate from or to a mapped drive.
4. If any directory or file contained in your job specifically denies permission to the system account or the account running the Double-Take service, the attributes of the file on the target will not be updated because of the lack of access. This also includes denying permission to the Everyone group because this group contains the system account.
5. If you select a dynamic volume and you increase the size of the volume, the target must be able to compensate for an increase in the size of the dynamic volume.
6. If you select files with alternate data streams, keep in mind the following.
    a. Alternate data streams are not included in the job size calculation. Therefore, you may see the mirror process at 99-100% complete while mirroring continues.
    b. The number of files and directories reported to be mirrored will be incorrect. It will be off by the number of alternate streams contained in the files and directories because the alternate streams are not counted. This is a reporting issue only. The streams will be mirrored correctly.
    c. Use the file attributes and data comparison option when performing a difference mirror or verification to ensure that all alternate data streams are compared correctly.
    d. If your alternate streams are read-only, the times may be flagged as different if you are creating a verification report only. Initiating a remirror with the verification will correct this issue.
7. If you select encrypted files, keep in mind the following.
    a. Only the data, not the attributes or security/ownership, is replicated. However, the encryption key is included. This means that only the person who created the encrypted file on the source will have access to it on the target.

   b. Only data changes cause replication to occur; changing security/ownership or attributes does not.

   c. Replication will not occur until the Windows Cache Manager has released the file. This may take awhile, but replication will occur when Carbonite Availability can access the file.

   d. When remirroring, the entire file is transmitted every time, regardless of the remirror settings.

   e. Verification cannot check encrypted files because of the encryption. If remirror is selected, the entire encrypted file will be remirrored to the target. Independent of the remirror option, all encrypted files will be identified in the verification log.

   f. Empty encrypted files will be mirrored to the target, but if you copy or create an empty encrypted file within the job after mirroring is complete, the empty file will not be created on the target. As data is added to the empty file on the source, it will then be replicated to the target.

   g. When you are replicating encrypted files, a temporary file is created on both the source and target servers. The temporary file is automatically created in the same directory as the Carbonite Availability disk queues. If there is not enough room to create the temporary file, an out of disk space message will be logged. This message may be misleading and indicate that the drive where the encrypted file is located is out of space, when it actually may be the location where the temporary file is trying to be created that is out of disk space.

   h. Carbonite Availability supports mirroring and replication of data stored on BitLocker enabled volumes when using the certificate based authentication method. Trusted Platform Module (TPM) is not supported because TPM uses a microchip that is built into the hardware to store the encryption keys. That same microchip would not be present on the target after failover.

8. If you are using mount points, keep in mind the following.

   a. By default, the mount point data will be stored in a directory on the target. You can create a mount point on the target to store the data or maintain the replicated data in a directory. If you use a directory, it must be able to handle the amount of data contained in the mount point.

   b. Recursive mount points are not supported. If you select data stored on a recursive mount point, mirroring will never finish.

9. Carbonite Availability supports transactional NTFS (TxF) write operations, with the exception of TxF SavePoints (intermediate rollback points).

   a. With transactional NTFS and Carbonite Availability mirroring, data that is in a pending transaction is in what is called a transacted view. If the pending transaction is committed, it is written to disk. If the pending transaction is aborted (rolled back), it is not written to disk.

      During a Carbonite Availability mirror, the transacted view of the data on the source is used. This means the data on the target will be the same as the transacted view of the data on the source. If there are pending transactions, the Carbonite Availability **Target Data State** will indicate **Transactions Pending**. As the pending transactions are committed or aborted, Carbonite Availability mirrors any necessary changes to the target. Once all pending transactions are completed, the **Target Data State** will update to **OK**.

      If you see the pending transactions state, you can check the Carbonite Availability log file for a list of files with pending transactions. As transactions are committed or aborted, the list is updated until all transactions are complete, and the **Target Data State** is **OK**.

    b. During replication, transactional operations will be processed on the target identically as they are on the source. If a transaction is committed on the source, it will be committed on the target. If a transaction is aborted on the source, it will be aborted on the target.
    c. When failover occurs any pending transactions on the target will be aborted.
10. Carbonite Availability supports Windows symbolic links and junction points. A symbolic link is a link (pointer) to a directory or file. Junction points are links to directories and volumes.
    a. If the link and the file/directory/volume are both in your job, both the link and the file/directory/volume are mirrored and replicated to the target.
    b. If the link is in the job, but the file/directory/volume it points to is not, only the link is mirrored and replicated to the target. The file/directory/volume that the link points to is not mirrored or replicated to the target. A message is logged to the Carbonite Availability log identifying this situation.
    c. If the file/directory/volume is in the job, but the link pointing to it is not, only the file/directory/volume is mirrored and replicated to the target. The link pointing to the file/directory/volume is not mirrored or replicated to the target.
    d. Junction points that are orphans (no counterpart on the source) will be processed for orphan files, however, the contents of a junction point (where it redirects you) will not be processed for orphan files.
11. If you have the Windows NtfsDisable8dot3NameCreation setting enabled on the source but disabled on the target, there is a potential that you could overwrite and lose data on the target because of the difference in how long file names will be associated with short files names on the two servers. This is only an issue if there are like named files in the same directory (for example, longfilename.doc and longfi~1.doc in the same directory). To avoid the potential for any data loss, the NtfsDisable8dot3NameCreation setting should be the same on both the source and target.
12. Carbonite Availability can replicate paths up to 32,760 characters, although each individual component (file or directory name) is limited to 259 characters. Paths longer than 32760 characters will be skipped and logged.
13. If you rename the root folder of a job, Carbonite Availability interprets this operation as a move from inside the job to outside the job. Therefore, since all of the files under that directory have been moved outside the job and are no longer a part of the job, those files will be deleted from the target replica copy. This, in essence, will delete all of your replicated data on the target. If you have to rename the root directory of your job, make sure that the job is not connected.
14. Keep in mind the following caveats when including and excluding data for replication.
    a. Do not exclude Microsoft Word temporary files from your job. When a user opens a Microsoft Word file, a temporary copy of the file is opened. When the user closes the file, the temporary file is renamed to the original file and the original file is deleted. Carbonite Availability needs to replicate both the rename and the delete. If you have excluded the temporary files from your job, the rename operation will not be replicated, but the delete operation will be replicated. Therefore, you will have missing files on your target.
    b. When Microsoft SQL Server databases are being replicated, you should always include the tempdb files, unless you can determine that they are not being used by any application. Some applications, such as PeopleSoft and BizTalk, write data to the tempdb file. You can, most likely, exclude temporary databases for other database applications, but you should consult the product documentation or other support resources before doing so.
    c. Some applications create temporary files that are used to store information that may not be necessary to replicate. If user profiles and home directories are stored on a server and replicated, this could result in a significant amount of unnecessary data replication on large

file servers. Additionally, the \Local Settings\Temporary Internet Files directory can easily reach a few thousand files and dozens of megabytes. When this is multiplied by a hundred users it can quickly add up to several gigabytes of data that do not need to be replicated.

   d. Creating jobs that only contain one file may cause unexpected results. If you need to replicate just one file, add a second file to the job to ensure the data is replicated to the correct location. (The second file can be a zero byte file if desired.)

15. Carbonite Availability does not replicate the last access time if it is the only thing that has changed. Therefore, if you are performing incremental or differential backups on your target machine, you need to make sure that your backup software is using an appropriate flag to identify what files have been updated since the last backup. You may want to use the last modified date on the file rather than the date of the last backup.
16. Keep in mind the following caveats when using anti-virus protection.
    a. Virus protection software on the target should not scan replicated data. If the data is protected on the source, operations that clean, delete, or quarantine infected files will be replicated to the target by Carbonite Availability. If the replicated data on the target must be scanned for viruses, configure the virus protection software on both the source and target to delete or quarantine infected files to a different directory that is not in the job. If the virus software denies access to the file because it is infected, Carbonite Availability will continually attempt to commit operations to that file until it is successful, and will not commit any other data until it can write to that file.
    b. You may want to set anti-virus exclusions on your source to improve replication performance. There are risks associated with making exclusions, so implement them carefully. For more information, see the Microsoft article 822158 Virus scanning recommendations for Enterprise computers that are running currently supported versions of Windows.
17. SQL Server 2005 or later may not initialize empty space when the database size increases due to the auto grow feature. Therefore, there is nothing for Carbonite Availability to replicate when this empty space is created. When the empty space is populated with data, the data is replicated to the target. A verification report will report unsynchronized bytes between the source and target due to the empty space. Since the space is empty, the data on the source and target is identical. In the event of a failure, the SQL database will start without errors on the target.
18. If you have reparse points in your data set, Carbonite Availability will replicate the tag, unless it is a known driver. If it is a known driver, for example Microsoft SIS, Carbonite Availability will open the file allowing the reparse driver to execute the file. In this case, the entire file will be replicated to the target (meaning the file is no longer a reparse point on the target and has all the data).
19. Keep in mind if you have reparse points in your data set, the reparse driver cannot be loaded on the target during protection. You must load the reparse driver on the target after failover in order to access the data. Additionally, you cannot have reparse points in your data set if you are using same server protection because the server is functioning as both a source and target.
20. If you are using an archiving solution, do not archive any files after failover. Archiving files after failover could cause corruption.
21. If you are using the Microsoft Windows Update feature, keep in mind the following caveats.
    a. Schedule your Windows Update outside the times when a mirroring operation (initial mirror, remirror, or a restoration mirror) is running. In some cases, Windows Update may perform an NTFS transactional rollback before displaying the dialog box to reboot the computer. This rollback will cause a mirror. If that mirror completes before the reboot, the reboot will trigger another mirror, unless you have configured Carbonite Availability to only

mirror changed files on reboot.

b. You must resolve any Windows Update incomplete operations or errors before failover or failback. (Check the windowsupdate.log file.) Also, do not failover or failback if the target is waiting on a Windows Update reboot. If failover occurs before the required Windows Update reboot, the target may not operate properly or it may not boot. You could also get into a situation where the reboot repeats indefinitely. One possible workaround for the reboot loop condition is to access a command prompt through the Windows Recovery Environment and delete the file \Windows\winsxs\pending.xml file. You may need to take ownership of the file to delete it. Contact technical support for assistance with this process or to evaluate other alternatives. Before you contact technical support, you should use the Microsoft System Update Readiness Tool as discussed in Microsoft article 947821. This tool verifies and addresses many Windows Update problems.

22. If you are using Windows deduplication, keep in mind the following caveats.
    a. Deduplicated data on the source will be expanded to its original size on the target when mirrored. Therefore, you must have enough space on the target for this expansion, even if you have deduplication enabled on the target.
    b. If you have deduplicated data on the target, mirroring and replication (like any other write process) will create a new file or new blocks of data. Existing blocks of deduplicated data will remain as they were until the next garbage collection.
    c. If you are protecting an entire server, you must have the deduplication feature installed on both the source and target. It can be enabled or disabled independently on the two servers, but it must at least be installed on both of the servers.
    d. After failover, the amount of disk space on the failed over server will be incorrect until you run the deduplication garbage collection which will synchronize the disk space statistics.
23. Replication is not case-sensitive. For example, if you rename the file Test.txt to test.txt, that change will not be replicated to the target. You will have to delete the file on the target and when it is remirrored, the new case of the file name will be used.
24. If you are using Windows 2008 R2, virtual hard disks can be mounted and dismounted reusing the same drive letter. However, once you have established a job, you cannot mount a different virtual hard disk to the same drive letter used in your job. This could cause errors, orphan files, or possibly data corruption. If you must change drive letters associated with a virtual hard disk, delete the job, change the mounting, and then re-create the job.

Chapter 3 Getting started

Review the *GeoCluster requirements* on page 6 and then proceed with your GeoCluster protection using the following steps, in order.

1. Configure a cluster that does not require shared storage. This configuration includes:
    - Creating the cluster
    - Adding nodes to the cluster
    - Setting the cluster quorum
    - Installing Carbonite Availability

    See *Configuring a cluster for GeoCluster* on page 15 for the correct order of these cluster configuration steps, which is dependent on your operating system.
2. Create a new cluster group using the option **Create an empty service or application**. This will be the group name for the application.
3. Create a GeoCluster Replicated Disk resource in your application group. See *Creating a GeoCluster Replicated Disk resource* on page 17.
4. Install your application specifying the application group for the server, database instance, or name that your application requires. For example, if you were using Microsoft SQL Server 2008 R2, you would use the following installation procedure.
    a. SQL 2008 installation on the first node
        1. Select **New SQL Server failover cluster installation**.
        2. Select **Features to install**.
        3. Provide the SQL Server Network Name and Instance ID.
        4. Select the disk resource, which is the GeoCluster Replicated Disk resource that you created.
        5. Provide an IP address for the network resource.
        6. Complete the remaining installation steps using your SQL Server documentation.

    b. SQL 2008 installation on additional node(s)
        a. Select Add node to SQL Server failover cluster.
        b. For the cluster configuration, select the SQL Server instance name from the installation on the first node.
        c. Complete the remaining installation steps using your SQL Server documentation.
5. After the application installation is complete, edit the properties of the resources for your application and make them dependent on the GeoCluster Replicated Disk resource, if needed. This will ensure that the replicated data (as opposed to shared storage) is available before your application starts. In the SQL example, the installation will automatically set the dependencies.

Chapter 4 Configuring a cluster for GeoCluster

The default quorum resource will vary depending on your configuration (number of nodes, shared disks, and so on). The recommended quorum resource for GeoCluster is the Node and File Share Majority. There are other quorum types available. Review the following list to determine which quorum is appropriate for your environment.

- **Node Majority**—This quorum is recommended for clusters with an odd number of nodes. The cluster can handle failures of half of the nodes (rounding up) minus one and still stay online.
- **Node and Disk Majority**—This quorum is recommended for clusters with an even number of nodes. The cluster can handle failures of half of the nodes (rounding up), as long as the witness disk remains online, and still stay online. If the witness disk fails, the cluster can handle failures of only half of the nodes (rounding up) minus one and still stay online.
- **Node and File Share Majority**—This quorum is recommended for clusters with special configurations, such as GeoCluster. The cluster can handle failures of half of the nodes (rounding up), as long as the witness share remains online, and still stay online. If the witness share fails, the cluster can handle failures of only half of the nodes (rounding up) minus one and still stay online.
- **No Majority: Disk Only**—This quorum is not usually recommended. The cluster can handle failures of all nodes except one and still stay online.

Use the following instructions as a guideline for configuring your Windows 2008 or 2012 cluster. See your Windows cluster documentation as a complete reference.

1. Login with an account that has administrative rights on the domain and the local machine.
2. Create the cluster, if it is not already created. See your Windows documentation for instructions on how to create a cluster.
3. Configure a Node and File Share Majority quorum. See your Windows documentation for instructions on how to configure the quorum.
4. If you are going to be using Hyper-V, install the Hyper-V server role on all nodes in the cluster. Make sure that you have the required Microsoft hotfixes applied, including KB958065 which is a failover clustering hotfix and KB950050.
5. Install Carbonite Availability on each node of the cluster. For complete installation details, see the *Carbonite Availability and Carbonite Move Installation, Licensing, and Activation* document.
6. If desired, you can install Carbonite Availability on non-clustered client machines if you want to use Cluster Administrator to control the GeoCluster resources. Install Carbonite Availability, selecting the **Client Components Only** installation option.
7. If you are going to be using Hyper-V, create your virtual machine from within Hyper-V. Be sure to leave the virtual machine off.
8. From Failover Cluster Management, create your application group or role.

   If you are creating a file server using clustered file shares, the path for the file share in the Failover Cluster Management wizard is case-sensitive. If the drive letter is uppercase, the path in the clustered file share wizard must also be uppercase. If the case does not match, the wizard will fail stating the path does not exist.

If your application requires a disk before installation can begin, create an Empty Service or Application or Empty Role. After your GeoCluster Replicated Disk resource is created, you can delete the empty item and Carbonite Availability will automatically move the GeoCluster Replicated Disk resource to available storage for your application installation.

9. If you are using Hyper-V, add your virtual machine resource to the group or role. Any warnings about storage may be disregarded because the GeoCluster Replicated Disk will alleviate storage requirements.

Chapter 5 Creating a GeoCluster Replicated Disk resource

The GeoCluster Replicated Disk resource allows for the real-time copy of data to be available on other nodes in the cluster. In the event of a failure and another node takes ownership, the GeoCluster Replicated Disk resource is also moved to the other node and it continues to replicate data, in real-time, to the remaining nodes in the cluster.

The instructions for creating this resource are different depending on your operating system.

- *Creating the GeoCluster Replicated Disk Resource on Windows 2008 or 2012* on page 17
- *Creating the GeoCluster Replicated Disk Resource on Windows 2008 or 2012 Hyper-V* on page 19
- *Bringing the resource online* on page 21
- *Taking the resource offline* on page 21

Creating the GeoCluster Replicated Disk Resource on Windows 2008 or 2012

1. From the Failover Cluster Manager, right-click the application group or role where you want to add a replicated disk to and select **Add a resource**, **More resources**, **Add GeoCluster Replicated Disk** (for Windows 2008) or **GeoCluster Replicated Disk** (for Windows 2012).
2. Right-click on the new resource and select **Properties**.
3. On the **General** tab, specify a name that identifies which application, file set, disk, and so on that you are protecting. This name must be unique within the cluster.
4. On the **Connection Parameters** tab, specify the GeoCluster Replicated Disk connection parameters using the settings below.
    - **Disk to replicate**—Select a disk to replicate from the available volumes. The only volumes that will be displayed are those that meet the following criteria.
        - NTFS volumes
        - Volumes which are not already being replicated by another GeoCluster Replicated Disk resource
        - Volumes that are not physical disk resources
        - Volumes that do not contain system files (The volume that you booted Windows from will not be displayed.)
        - Volumes that exist on all nodes of the cluster
    - **Network to route Double-Take mirroring and replication traffic over**—Select the network that you want to use for Carbonite Availability mirroring and replication traffic. If you do not have multiple networks established, you will only be able to select the one network that does exist. If you do not select a network, Carbonite Availability will use DNS to determine a network route to use. Ideally, the networks used for various traffic should be separated. This is dependent on the number of networks that you established when you created the cluster and the priority assigned to each network. For example, if you have two network routes, separate Carbonite Availability and your public traffic. If you have three

routes, separate the public traffic and then separate Carbonite Availability from the cluster heartbeat.

- **Interval to check unresponsive nodes**—Specify how much time, in seconds, between checks of nodes to see if a Carbonite Availability connection can be made.
- **Delay connection until resources dependent on this one are online**—This option allows you to delay a Carbonite Availability connection until any resources that have the GeoCluster Replicated Disk resource as a dependency are online. By ensuring that all resources that are dependent on the GeoCluster Replicated disk resource are online before starting the connection, the chance of a conflict occurring because application resources are attempting to open files exclusively while Carbonite Availability is mirroring those files is removed.

5. On the **Orphans** tab, you will see the option to delete orphan files. An orphan is a file that exists in the target location but is not in the source location. You can select to delete orphans during a mirror.
6. On the **Compression** tab, you will see your compression configuration. If you want to configure compression, verify that **Enable Compression** is selected. Depending on the compression algorithms available for your operating system, you may see a slider bar indicating different compression levels. Set the level from minimum to maximum compression to suit your needs.
7. On the **Mirror Properties** tab, specify your Carbonite Availability mirroring settings.

   - **Mirror Options**—Choose a comparison method and whether to mirror the entire file or only the bytes that differ in each file.

     > If you are using a database application, do not use the compare file attributes only options unless you know for certain that you need it. With database applications, it is critical that all files, not just some of the files, are mirrored. In this case, you should compare both the attributes and the data.

     - **Do not compare files. Send the entire file.**—Carbonite Availability will not perform any comparisons between the files on the source and target. All files will be mirrored to the target, sending the entire file.
     - **Compare file attributes. Send the entire file.**—Carbonite Availability will compare file attributes and will mirror those files that have different attributes, sending the entire file.
     - **Compare file attributes. Send the attributes and bytes that differ.**—Carbonite Availability will compare file attributes and will mirror only the attributes and bytes that are different.
     - **Compare file attributes and data. Send the attributes and bytes that differ.**—Carbonite Availability will compare file attributes and the file data and will mirror only the attributes and bytes that are different.
   - **Calculate size of protected data upon connection**—Determines the size of the protected data set prior to starting the mirror. The mirroring status will update the percentage complete if the data set size is calculated.
8. No other settings are required for the GeoCluster Replicated Disk resource, although there are optional settings available. See *GeoCluster Replicated Disk resource properties* on page 25 for details. Click **OK** to save the GeoCluster Replicated Disk configuration changes that you made.

9. To control the resource, you can bring it online and take it offline. Neither of these actions trigger failover. They just control the activity of the resource.

Creating the GeoCluster Replicated Disk Resource on Windows 2008 or 2012 Hyper-V

1. Create a virtual machine using the Hyper-V Manager. For details, see your Hyper-V documentation.
2. From the Failover Cluster Manager, cluster the virtual machine using the High Availability Wizard.
3. Take the virtual machine cluster group offline by right-clicking on it and selecting **Take this resource offline** (for Windows 2008) or **Take Offline** (for Windows 2012).
4. Right-click on the virtual machine cluster group and select select **Add a resource**, **More resources**, **Add GeoCluster Replicated Disk** (for Windows 2008) or **GeoCluster Replicated Disk** (for Windows 2012).
5. Right-click on the resource and select **Properties**.
6. On the **General** tab, specify a name that identifies which application, file set, disk, and so on that you are protecting. This name must be unique within the cluster.
7. On the **Connection Parameters** tab, specify the GeoCluster Replicated Disk connection parameters using the settings below.
   - **Disk to replicate**—Select the volume where the virtual machine .vhd or .vhdx file is stored.
   - **Network to route Double-Take mirroring and replication traffic over**—Select the network that you want to use for Carbonite Availability mirroring and replication traffic. If you do not have multiple networks established, you will only be able to select the one network that does exist. If you do not select a network, Carbonite Availability will use DNS to determine a network route to use. Ideally, the networks used for various traffic should be separated. This is dependent on the number of networks that you established when you created the cluster and the priority assigned to each network. For example, if you have two network routes, separate Carbonite Availability and your public traffic. If you have three routes, separate the public traffic and then separate Carbonite Availability from the cluster heartbeat.
   - **Interval to check unresponsive nodes**—Specify how much time, in seconds, between checks of nodes to see if a Carbonite Availability connection can be made.
   - **Delay connection until resources dependent on this one are online**—This option allows you to delay a Carbonite Availability connection until any resources that have the GeoCluster Replicated Disk resource as a dependency are online. By ensuring that all resources that are dependent on the GeoCluster Replicated disk resource are online before starting the connection, the chance of a conflict occurring because application resources are attempting to open files exclusively while Carbonite Availability is mirroring those files is removed.
8. On the **Orphans** tab, you will see the option to delete orphan files. An orphan is a file that exists in the target location but is not in the source location. You can select to delete orphans during a mirror.
9. On the **Compression** tab, you will see your compression configuration. If you want to configure compression, verify that **Enable Compression** is selected. Depending on the compression algorithms available for your operating system, you may see a slider bar indicating different compression levels. Set the level from minimum to maximum compression to suit your needs.

10. On the **Mirror Properties** tab, specify your Carbonite Availability mirroring settings.
    - **Mirror Options**—Choose a comparison method and whether to mirror the entire file or only the bytes that differ in each file.

      If you are using a database application, do not use the compare file attributes only options unless you know for certain that you need it. With database applications, it is critical that all files, not just some of the files, are mirrored. In this case, you should compare both the attributes and the data.

      - **Do not compare files. Send the entire file.**—Carbonite Availability will not perform any comparisons between the files on the source and target. All files will be mirrored to the target, sending the entire file.
      - **Compare file attributes. Send the entire file.**—Carbonite Availability will compare file attributes and will mirror those files that have different attributes, sending the entire file.
      - **Compare file attributes. Send the attributes and bytes that differ.**—Carbonite Availability will compare file attributes and will mirror only the attributes and bytes that are different.
      - **Compare file attributes and data. Send the attributes and bytes that differ.**—Carbonite Availability will compare file attributes and the file data and will mirror only the attributes and bytes that are different.
    - **Calculate size of protected data upon connection—**Determines the size of the protected data set prior to starting the mirror. The mirroring status will update the percentage complete if the data set size is calculated.
11. No other settings are required for the GeoCluster Replicated Disk resource, although there are optional settings available. See *GeoCluster Replicated Disk resource properties* on page 25 for details. Click **OK** to save the GeoCluster Replicated Disk configuration changes that you made.
12. Modify the virtual machine and virtual machine configuration resources and add the GeoCluster Replicated Disk resource (not the GeoCluster Replicated Disk Status resource) as a dependency.

Bringing the resource online

The GeoCluster Replicated Disk resource will appear offline after it is created. When you bring it online, the following actions occur.

1. A Carbonite Availability job is created.
2. The job is connected to all of the possible owners specified in the resource (except the active node which is the source).
3. A mirror is initiated to create the baseline copy of data from the active node to all of the possible owners.
4. The drive where the mirrored data is located on each of the possible owners is made read-only to all other applications except Carbonite Availability.
5. Real-time replication from the active node to all of the possible owners begins.

If you are using Windows 2008, right-click the resource and select **Bring this resource online**.

If you are using Windows 2012, right-click the resource and select **Bring Online**.

Taking the resource offline

When you take the GeoCluster Replicated Disk resource offline, the following actions occur.

1. Real-time replication from the active node to the possible owners stops.
2. The read-only limitation is removed from the corresponding drive letters on the possible owners.
3. The job is disconnected from all of the possible owners.
4. The job is deleted.

If you are using Windows 2008, right-click the resource and select **Take this resource offline**.

If you are using Windows 2012, right-click the resource and select **Take Offline**.

If the GeoCluster Replicated Disk Resource is offline, it will impact any application depending on it. Data integrity cannot be guaranteed on the other nodes in the cluster.

Chapter 6 Monitoring and controlling a GeoCluster Replicated Disk resource

The Carbonite Availability job created by the GeoCluster will be displayed as an **Legacy Job** on the **Jobs** page of the Carbonite Replication Console. You will have limited control over the job, including taking and managing snapshots, viewing the job log, initiating mirrors and verifications, setting bandwidth limitations, deleting orphan files, and pausing and resuming the target.

If you delete the job, the GeoCluster Replicated Disk resource will not be deleted. You will have to remove and readd the passive nodes to the resource to re-create the job without impacting production. If you are not worried about impacting production, you can take the resource offline and then bring it back online to re-create the job.

Ideally, you should use the standard Windows cluster tools to monitor the status of the resource. See your cluster documentation for details on monitoring a cluster resource.

You can also use the following information to help you monitor the GeoCluster Replicated Disk resource.

- Do not use the **Initiate Failure** feature of Cluster Administrator to test failover of GeoCluster resources. Use other test methods, such as manually moving the group or unplugging the owning nodes network cable.
- Do not use the **Automatic Failback** feature of Cluster Administrator. If you need to return ownership to the original node, wait until GeoCluster has completed mirroring from the new owning node back to the original owning node and then manually move the group.
- If you change an IP address on any node of the cluster, you must stop and restart the cluster service on all of the nodes in the cluster in order for GeoCluster to detect the new IP address.
- If you must reboot the owning node, you should move all of your cluster groups to another node before the reboot. This will ensure data integrity and allow you to make sure the applications come online before the node is rebooted.

Resolving an online pending GeoCluster Replicated Disk resource

When the GeoCluster Replicated Disk resource is in an online pending state, you are protected from possible data corruption. You can see the online pending status directly in the description of the GeoCluster Replicated Disk resource. If the pending state were bypassed, the node where you are trying to bring the resource online would have incomplete data, which would then be replicated to the other nodes in the cluster. This state safeguards you from corrupting your data.

There are different options for resolving an online pending state, depending on whether your operating system supports snapshots. Therefore, some of the following options may not be displayed or may be disabled if they are not valid for your configuration.

Right-click the online pending resource, select **Properties**, select the **Online Pending** tab, and click the desired control.

- **Revert Snapshot**—If you have a snapshot of the target data available, you can revert to that data. If you revert to a snapshot, any data changes made after the snapshot's specified date and time will be lost. A Carbonite Availability connection will be established to replicate the node's data (at the snapshot point in time) to the other nodes.
- **Discard Queue**—If you have data in the target queue, you can discard that data. If you discard the queued data, you will lose the changes associated with that data made on the previously owning node. A Carbonite Availability connection will be established to replicate the node's data (without the data that was in queue) to the other nodes.
- **Fail Resource**—You can fail the resource and no Carbonite Availability connection will be established.
- **Verify Group**—With this option and snapshot capability, you can test the data on the node before deciding whether to use it. If you select this option, a snapshot of the node's current Carbonite Availability data will be taken, the disk will come online, but the GeoCluster Replicated Disk resource will not come online, allowing you to check the data. (This means there is no Carbonite Availability connection established at this time.) Once the snapshot is taken, you can test the data on the node to see if it is viable. Make sure you prevent user access while you are verifying the data. Once you have tested the data, you need to right-click on the online pending resource again and accept or reject the data.
- **Accept**—If you accept the data, the current data on the node will be used, and a Carbonite Availability connection will be established to replicate the current node's data to the other nodes. If any other nodes in the cluster contain more recent data, this node will overwrite that data and it will be lost.
- **Reject**—If you reject the data, the node will be reverted to the snapshot that was taken when you selected the **Verify Group** option. Any changes made on the node after that snapshot was created will be lost. This option essentially takes you back to where you were, allowing you the opportunity to check other nodes for more recent data.If you have multiple GeoCluster Replicated Disks in the same group and have selected to reject the data after verifying the group, the rejection processing may take several minutes.

GeoCluster Replicated Disk Status Resource

The GeoCluster Replicated Disk Status resource (also displayed as GRD Status) is automatically created when the first GeoCluster Replicated Disk resource is created in a group. Once the status resource is created, it will exist as long as there is a GeoCluster Replicated Disk resource in the group. When the last GeoCluster Replicated Disk resource in a group is deleted, the status resource will be deleted. Only one status resource is created per group. If the resource is deleted, it will automatically be re-created.

The description of the status resource corresponds to various states of your Carbonite Availability data. These status descriptions are seen directly in the GeoCluster Replicated Disk resource description. For example, you may see the status "The status of all targets is OK." This indicates the data on each target node is in a good state. Another message may be "Target target_name is queuing. Data in queue on target." This indicates the data on the specified target is not up-to-date. Because there is data in queue on the target, that has not been written to disk yet, the target data is out-of-date. Or you may see either of the following status descriptions.

- Target target_name is pending. Data integrity not guaranteed.
- Target target_name is suspect. Data integrity not guaranteed.

These messages indicate the data on the specified target node is not in a good state. This could be because a mirror is in progress, an operation has been dropped on the target, or another Carbonite Availability processing issue.As long as the status is pending, data integrity cannot be guaranteed on the specified target node. Check the Carbonite Availability logs for more information.

Another function of the status resource, for all Windows versions, is to keep you from moving the GeoCluster Replicated Disk resource to another node at the wrong time and potentially corrupting your data. If the GeoCluster Replicated Disk resource was moved while the status resource is in a pending or queuing state, the new node would have incomplete data, which would then be replicated to the other nodes in the cluster. This resource safeguards you from corrupting your data. This happens by removing passive nodes as possible owners and discarding any manual changes made to the possible owners list.

GeoCluster Replicated Disk resource properties

Right-click the resource and select **Properties**, when you want to view or modify the resource properties. There are nine properties tabs for the GeoCluster Replicated Disk resource on Windows 2008. If you are using Windows 2012, there are the same nine properties tab, plus an additional tab which shows the private properties of the resources. These are equivalent to the properties available in the other tabs.

1. **General**—This tab identifies the **Name** and **Resource type** of the resource. It also displays the current state of the resource and an additional detailed status message.
2. **Dependencies—**By default, the GeoCluster Replicated Disk resource is not dependent on any other resources.
3. **Policies**—This tab controls how and when MSCS handles a failure of the resource. For more information on **Policies** options, see your Windows documentation.
    - **If resource fails, do no restart**—Select this option if you do not want cluster service to restart the resource if it fails.
    - **If resource fails, attempt restart on current node**—Select this option if you want cluster service to restart the resource if it fails. Specify the length of time to attempt restarts and the number of restarts to attempt during that period of time.
    - **If restart is unsuccessful, fail over all resources in this service or application**—If this option is enabled, the failure of the group will cause the resource to move to another node. If this option is disabled, the failure of the resource will not cause the resource to move to another node.
    - **If all the restart attempts fail, begin restarting again after the specified period**—If this option is enabled, the cluster will delay the length of time specified before trying to restart the resource again.
    - **Pending timeout**—This value determines how long the resource is allowed to remain in a pending state before it fails. If the resource takes longer than the time specified to be brought online or taken offline, the resource fails.
4. **Advanced Policies**—This tab controls resource specific settings. For more information on **Advanced Policies** options, see your Windows documentation.
    - **Possible owners**—All nodes of the cluster are listed. Select or deselect the nodes that you want to be possible owners.
        - If you add additional owners, the GeoCluster Replicated Disk resource will connect the resource's replication set to the new owners and begin a mirror to each.
        - If you remove owners, the GeoCluster Replicated Disk resource will disconnect the resource's replication set from each owner removed.

          The GeoCluster Replicated Disk resource must have at least two possible owners to function properly.
    - **Basic resource health check interval—**This setting is formerly known as the Looks Alive poll interval. It specifies how often the resource is polled to determine whether it is still running on the active node. You can choose the standard time period of 5 seconds, or you can specify your own value.
    - **Thorough resource health check interval**—This setting is formerly known as the Is Alive poll interval. It designates how often the possible owners are polled to determine

whether the specified disk on each node can be written to and read from. You can choose the standard time period of 1 minute, or you can specify your own value.

- **Run this resource in a separate Resource Monitor**—You should enable this option so that each GeoCluster Replicated Disk resource runs in its own monitor.

5. **Connection parameters**—This tab controls disk replication, network routing, and orphan files for Carbonite Availability.
    - **Disk to replicate**—The volume to replicate
    - **Network to route Double-Take mirroring and replication traffic over**—The network to use for Carbonite Availability mirroring and replication traffic. If you do not have multiple networks established, you will only be able to select the one network that does exist. If you do not select a network, GeoCluster will use DNS to determine a network route to use.

      Ideally, the networks used for various traffic should be separated. This is dependent on the number of networks that you established when you created the cluster and the priority assigned to each network. For example, if you have two network routes, separate GeoCluster and your public traffic. If you have three routes, separate the public traffic and then separate GeoCluster from the cluster heartbeat.

      Modifications to either of the first two settings will not take effect until the next time the resource is brought online.
    - **Interval to check unresponsive nodes**—The frequency to determine how often an unresponsive node is checked to see if a Carbonite Availability connection can be made
    - **Delay connection until resources dependent on this one are online**—This option allows you to delay a Carbonite Availability connection until any resources that have the GeoCluster Replicated Disk resource as a dependency are online. By ensuring that all resources that are dependent on the GeoCluster Replicated disk resource are online before starting the connection, the chance of a conflict occurring because application resources are attempting to open files exclusively while GeoCluster is mirroring those files is removed.
6. **Orphans**—An orphan is a file that exists in the target location but is not in the source location. You can select to delete orphans during a mirror.
7. **Compression**—If you want to configure Carbonite Availability compression, verify that **Enable Compression** is selected. Depending on the compression algorithms available for your operating system, you may see a slider bar indicating different compression levels. Set the level from minimum to maximum compression to suit your needs.
8. **Online Pending**—Because context-sensitive, right-click menus are not available in the Windows 2008 Failover Cluster Administrator, GeoCluster processing controls have been added to a properties tab. For details on this tab, see *Monitoring and controlling a GeoCluster Replicated Disk resource* on page 22.
9. **Mirror Properties**—This tab controls the Carbonite Availability mirroring process.
    - **Full Mirror**—All files in the replication set will be sent from the source to the target.
    - **File differences**—Only those files that are different based size or date and time will be sent from the source to the target.
        - **Send data only if Source is newer than Target**—Only those files that are newer on the source are sent to the target.

> If you are using a database application, do not use the newer option unless you know for certain you need it. With database applications, it is critical that all files, not just some of them that might be newer, get mirrored.

  - **Use block checksum**—For those files flagged as different, the mirror performs a checksum comparison and only sends those blocks that are different.
- **Calculate Replication Set size prior to mirror**—Determines the size of the replication set prior to starting the mirror. The mirroring status will update the percentage complete if the replication set size is calculated.