# NUTS ABOUT CLUSTERS

**Linking Linux: Novell Open Enterprise Server, GroupWise and Novell Cluster Services**
**Non-Stop Collaboration Service**

BY KEN BAKER

PHOTOGRAPH BY NEIL BROWN

Have you ever met a business traveler that really enjoys airport layovers? A quick stop over might give you a chance to make a few calls, check your e-mail, and catch-up on the latest headline news. But after more than thirty to sixty thrilling minutes of twiddling your thumbs or trying to take a power nap in a less-than-comfortable airport chair, you usually wish you had booked a direct flight. Unfortunately, you don't always have the option to choose non-stop service and you have to settle for something considerably less desirable than a direct flight.

Unlike airline service, you can't settle when it comes to your critical business services. You demand non-stop service from the key processes that drive your profitability. Downtime kills productivity and does serious damage to your bottom line. Chief among these vital processes are your organization's messaging and collaboration services. Breakdowns in communication between your customers, suppliers, partners or employees can initiate business disasters that put the viability of your business at significant risk.

Recently released, Novell GroupWise 7 combines with Novell Cluster Services, which is bundled in Open Enterprise Server, to provide your Linux environment the high availability messaging and collaboration you need to ensure constant communication between all of your business' key players.

### Start Me Up

The availability of enterprise-class services has been a key factor in organizations' decisions to take advantage of the low cost and reliability of the Linux platform. Those anxious for an enterprise-class collaboration solution welcomed the release of GroupWise 6.5 for Linux with open arms. The recent release of GroupWise 7 has given these businesses even more cause to celebrate with the solution's wide array of new enhancements and features. However, little press space and attention has been given to a seemingly minor addition to the latest GroupWise offering: GroupWise High Availability Service. While this new addition might seem minor, it can actually have a huge impact on your ability to protect your bottom line, ensure around-the-clock productivity and keep your lines of communication open.

Appropriately named, the GroupWise High Availability Service makes sure that if key GroupWise services running in your Linux environment go down for any reason they will be automatically restarted. The specific GroupWise 7 services that this feature affects includes the Post Office Agents (POAs), Message Transfer Agents (MTAs), GroupWise Internet Agents (GWIAs) and Messenger agents.

Unlike in NetWare and Windows environments where these agents run in protected address space, these agents run in user space on Linux, which doesn't have the built-in ability to detect if they have stopped, let alone restart them. So, the engineers at Novell leveraged the enhanced capabilities of the monitor in GroupWise 7 to poll these services for signs of life. If the GroupWise Monitor detects that any of these services have stopped, the GroupWise High Availability service will automatically restart them to make sure you continue to get uninterrupted service for your communication needs.

But because this powerful new feature is not yet included in the standard installation of GroupWise, you'll need to first do some set-up work to take advantage of it, including the following processes:

- Verify Agent Operation
- Configure the High Availability Files
- Enable High Availability in YaST
- Start the Agents as Daemons
- Enable the Monitor
- Test the High Availability Service

### Verify Agent Operation

Before you configure the GroupWise High Availability service, make sure that your agents are operating properly by starting and stopping them. To do this, first open a terminal window, become root by entering sux and the root password. (The sux command enables the X Windows System, which is required for running the GUI interface for the GroupWise agents.) Then do the following:

GROUPWISE HIGH AVAILABILITY SERVICE MAKES SURE THAT IF KEY GROUPWISE SERVICES RUNNING IN YOUR LINUX ENVIRONMENT GO DOWN FOR ANY REASON, THEY WILL AUTOMATICALLY BE RESTARTED.

1 On your GroupWise Linux server, change to the root directory.
2 As shown below, start the POA, MTA and Internet Agents by providing the full path to the executables. Include the —show switch to make sure the agents are properly set up and configured.

```
/opt/novell/groupwise/agents/bin/gwpoa
—show @post.poa
/opt/novell/groupwise/agents/bin/gwmta
—show @dom.mta
/opt/novell/groupwise/agents/bin/gwia
—show @gwia.cfg
```

**Configure the High Availability Files**

Once you have verified the agents start and run as expected, shut them down so you can properly configure the GroupWise High Availability service file (gwha) as follows:

1 On your GroupWise Linux server, change to the /etc/xinetd.d directory.
2 Open the gwha file in a text editor.
3 Specify a unique port number such as 8400 for the port = field.

Because the GroupWise High Availability credentials you will be configuring later will utilize clear text, Novell recommends you use SSL. To use SSL, edit the GroupWise High Availability configuration file using the following steps:

1 Change to the /etc/opt/novell/groupwise directory.
2 Open the gwha.conf file in a text editor.
3 Under the [gwha] section, complete the fields as follows:
   [gwha]
   ssl = yes
   key = filename.key
   cert = filename.crt
   password = password

Be aware that if you are using SSL, you won't be able to use telnet to start, stop or check the status of your GroupWise agents.

## HIGH AVAILABILITY DESIGN CONSIDERATIONS

To obtain the highest levels of availability from your critical business systems, you need more than just a cluster of shared resources and the ability to monitor and restart specific server processes. You need to address a number of key services and issues when designing and planning a high availability collaboration system.

- Environment—Continuous power, proper ventilation and cooling, and protection from flooding and catastrophic emergencies are essential elements for the design of a highly available system.
- Single Points of Failure—Replicating the potential single points of failure that hardware and software components introduce in the system is critical.
- Network Connectivity—High availability in a network environment means avoiding the loss of service to network clients. You must plan for redundant network paths and hardware protection.
- Data Redundancy—Protect your data through redundant disk drive technologies such as RAID drives and offsite backups.
- Redundant Server Components—Servers that participate as cluster nodes in a high availability system must have redundant features and components to ensure continuous service to network clients.
- Performance Needs—Adding layers of redundancy and other high availability protection might cause the performance of the system to degrade. Technologies such as load balancing and sharing help counter this type of performance loss.
- Proactive Management—Quick notification of failures and even proactive notification of pending failures reduces the mean time to repair (MTTR) time, keeping outages to a minimum.
- Contingency Plans—Contingency plans and recovery procedures must be in place before a system failure. Hardware service agreements can help ensure proper service in the case of an outage.

# POWERFUL ENHANCEMENTS HAVE BEEN INTRODUCED INTO THE MONITOR AGENT IN GROUPWISE 7.

### Enable High Availability in YaST

Once you have properly configured the GroupWise High Availability files, you can enable the service using the YaST management interface in SUSE LINUX Enterprise Server 9:

1. In YaST, select Network Services | Network Services (inetd).
   **Note:** You might need to select Enable to activate the list of services.
2. Scroll down to the gwha line and highlight it. (See Figure 1.)
3. Choose On as the Toggle Status.
4. Click Finish.

### Start the Agents as Daemons

To start the GroupWise High Availability agents as Daemons and to verify they are operating properly, do the following:

1. On your GroupWise Linux server, change to the /etc/init.d directory.
2. Enter the command `./grpwise start` to start the POA, MTA, GWIA and Messenger agents.
3. Enter the command `./grpwise status` to verify the status of the agents.

Notice the status of the agents are displayed in terms of the names of the post offices and domains associated with each POA, MTA and GWIA similar to the following format:

[domain]
[post_office.domain]
[domain.GWIA]

---

Use the YaST management interface to enable the GroupWise High Availability service.

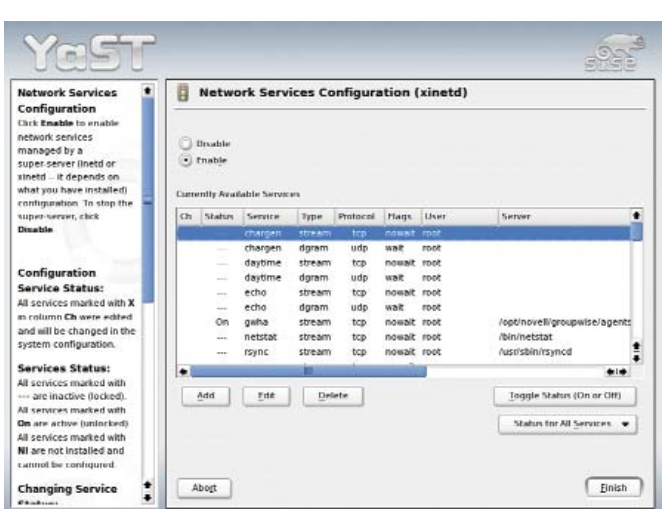


FIGURE 1

From the Windows Monitor agent you can configure the GroupWise Monitor to monitor the GroupWise Messenger agents.

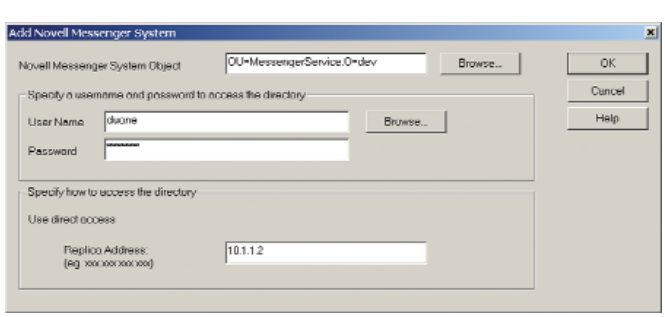


FIGURE 2

TO GET NON-STOP SERVICE FROM YOUR LINUX COLLABORATION SYSTEM, TAKE YOUR HIGH-AVAILABILITY EFFORTS UP A FEW NOTCHES BY ADDING SERVER CLUSTERING INTO THE EQUATION.

Use the `grpwise` command to control the agents as daemons. To control a specific agent, enter its location name (do not include brackets), not the name of the agent's executables. The location name is the name shown when you execute the status command.

Use the commands in Table 1 in the /etc/init.d directory to control the agents as daemons.

**Enable the Monitor**

Powerful enhancements have been introduced into the Monitor agent in GroupWise 7. One of these enhancements is its interaction with the GroupWise High Availability service to monitor the different agents and restart them if they happen to stop running. Before this interaction can take place you must create a username and password on your Linux server to represent the High Availability service. This username should have very limited rights and be quite obvious, such as "gwha" for GroupWise High Availability. This username and password will be startup switches for the monitor as well as be the necessary credentials it will supply for the High Availability service to operate properly.

To start the GroupWise Monitor, use either the grpwise-ma startup script or the `./ gwmon` command. Because you want to make sure the monitor always loads, Novell recommends using the startup script. Regardless of how you start the GroupWise Monitor, you must use the `—hauser` and `—hapassword` startup switches followed by the username and password you defined for the GroupWise High Availability service to enable its high availability monitoring. For example:

```
—hauser [your high availability username]
—hapassword [your high availability password]
```

If you are using GroupWise Messenger, enable the GroupWise Monitor to monitor the messaging agents by selecting Add Novell Messenger System from the File menu in the Windows Monitor agent. (See Figure 2.)

When the GroupWise Monitor is running, it polls the agents every two minutes. In most instances, the default polling intervals for the MTA, POA and GWIA will be sufficient. However, because GroupWise Messenger users have near-immediate communicate-and-respond expectations, you might want to consider adjusting its polling interval down to every fifteen seconds. You can adjust the polling interval by using the `—hapoll` startup switch.

**Test the High Availability Service**

After you set up and configure your GroupWise High Availability service, do the following to make sure it works properly:

1. On your GroupWise Linux server, change to the /etc/init.d directory.
2. Stop one of the agents using the `./grpwise stop agent_location_name` command.
3. To make sure the agent actually stopped, check it with the `./grpwise status agent_location_name` command.
4. Wait two minutes, or the amount of time you configured for the monitor polling interval, and then check the status of the agent again.

If the agent is running when you check its status again, then your GroupWise High Availability service is working.

**Take it up a Few Notches**

The GroupWise High Availability service is an excellent first step in helping you achieve uninterrupted service from the critical collaboration pieces that run in your Linux environment. But it's just a first step because it only protects you from software-related failures of individual service agents. What happens when you experience a hardware failure that brings your whole server down? To get non-stop service from your Linux collaboration system, you need to take your high-availability efforts up a few notches by adding server clustering into the equation.

When you use clustering with GroupWise, your collaboration services are not dedicated to a single server. Rather, your GroupWise services become shared resources within a cluster of

IF YOU'RE FAMILIAR WITH HOW CLUSTERS OPERATE IN A NETWARE ENVIRONMENT, THEN IMPLEMENTING A GROUPWISE CLUSTER IN A LINUX ENVIRONMENT WON'T BE DAUNTING.

servers. All or a specified set of server nodes within the cluster have responsibility for the GroupWise services and data stored on the cluster's Storage Area Network (SAN) or other type of shared disk system. (See Figure 3.) For example, if a server goes down that was running the GroupWise POA, another server node in the cluster automatically reloads and takes responsibility for that agent.

This transfer of collaboration services from a failed server to another server node in the cluster is referred to as failover. Failover happens transparently so none of your users are even aware a failure occurred. Additionally, you can work on reviving the failed node without loss of service to your users. As a result, users enjoy continual access to all their clustered collaboration services.

**Distributed Failover and Failback**

GroupWise should work with any standard Linux clustering solution, including the open source two-node cluster service called Heartbeat, which is packaged with SUSE LINUX Enterprise Server 9. But the obvious cluster solution of choice for the Linux environment and GroupWise is Novell Cluster Services, which ships with Novell Open Enterprise Server. Novell Cluster Services provides true multinode clustering by supporting up to 32 nodes in a cluster. (The distribution includes a two-node license and you can purchase licenses for additional nodes.) This level of multinode support not only allows for higher levels of availability, but also the ability to distribute resources from a failed node to multiple surviving nodes.

## NCS CONFIGURATION PARAMETERS

Whether you're running Novell Cluster Services in a Linux or NetWare environment, the solution provides you the flexibility to customize, as you see fit, the parameters that govern how your cluster system will behave. The following are the main iManager configurable settings for your cluster:

- Membership—The Membership, or quorum trigger, specifies how many nodes must be active within the cluster before any cluster resources will start to load. Generally, this number is set to greater than 1 so that all cluster resources don't automatically load on the first server that is brought up in the cluster.
- Timeout—Timeout specifies the amount of time to wait for the number of servers defined in the Membership field to be up and running. If the timeout period elapses before the quorum membership reaches its specified number, resources will automatically start loading on the servers that are currently up and running in the cluster. For example, if you specify a Membership value of 4 and a timeout value of 30 seconds, if after 30 seconds only two servers become available in the cluster, then resources will begin to load on those two servers.
- Master Node—The master node determines the true status of the cluster membership.
- Heartbeat Setting—Each node in the cluster must transmit a signal, or heartbeat, to the master node to let the master know that the node is still active. The heartbeat setting specifies how often each node must transmit this signal.
- Tolerance Setting—The tolerance setting is how long the master should wait for a heartbeat before assuming the node has gone down and initiate the group membership protocols to verify whether the node really is down.
- Watchdog Settings—Similar to the Heartbeat setting, the Master Watchdog and Slave Watchdog settings specify how often the master node must transmit its "alive" status to all the other nodes and how long the nodes should wait for that signal before initiating the group membership protocols to verify whether the master has actually stopped running.
- Resource Priority—The Resource Priority allows you to control the order in which multiple resources start on a given node when the cluster is brought up or during a failover or failback. This is useful for ensuring that the most critical resources load first and are available to users before less critical resources.

Distributing resources from a single node to multiple nodes is referred to as either multinode distributed failover or Fan-out Failover. This ability to move applications and services from a failed node to multiple surviving nodes allows you to make sure you don't overload any single node in your cluster. For example, if you have a four-node cluster and the node responsible for your GroupWise services fails, you can configure your POA and MTA to migrate to a second node, your GWIA to a third and the GroupWise Messenger agent to the fourth remaining node. But what happens if multiple nodes fail? Where do the agents go? Novell Cluster Services allows you to specify the preferred node order for a service to try to failover to. If the preferred node is also down, the service will failover to the next surviving node in the order.

Once you bring a failed node back into the cluster, you can move the services it originally hosted back to that restarted node. This process is known as failback. With Novell Cluster Services you have the option to have these services failback automatically or to failback through manual intervention.

To ensure that your services failover successfully, you need to make sure that all the nodes in your cluster have sufficient capacity to handle the additional resources when other nodes in the cluster fail. In other words, you need a firm understanding of the operational characteristics of all the server applications configured to failover, including memory requirements, processor and cache requirements, network access to and from the server, how to start and stop the application, how the application recovers from errors, where the applications install files, and any server name or file path dependencies.

### Deploying Novell Cluster Services

If you are already familiar with the concepts and principles of how clusters operate in a NetWare environment, then implementing a GroupWise cluster in a Linux environment should not be a daunting task. It works in a similar way. Still, you need to pay attention to some particulars when deploying GroupWise and Novell Cluster Services in a Linux environment. Those issues can't be completely covered in an

Most high availability collaborations systems use a shared-disk subsystem connected to a cluster of servers that share responsibilty for providing users continuous access to the collaboration resources they need.

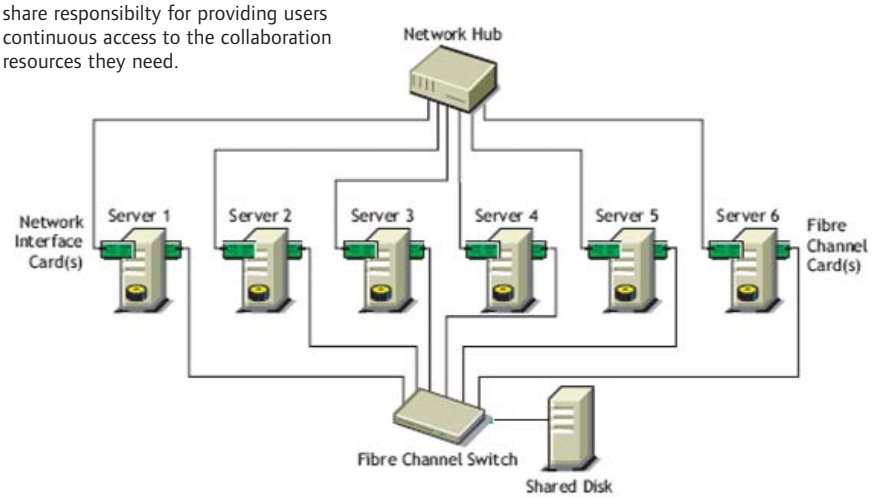


FIGURE 3

article of this size. But the following lists a few specifics that will give you an idea of what to watch out for.

**General Novell Cluster Services Requirements**

Novell Open Enterprise Server must be installed and running on each node in the cluster. Novell Cluster Services will automatically be installed as part of the Open Enterprise Server installation.

- All servers in the cluster must be configured with a static IP address and be on the same IP subnet.
- There must be an additional IP address for the cluster and for each cluster resource and cluster-enabled pool.
- All servers in the cluster must be in the same Novell eDirectory tree.
- A shared-disk system is required for each cluster with at least 20 MB of free disk space on the shared-disk system for creating a special cluster partition.
- The disks contained in the shared-disk system should be configured to use mirroring or RAID to add fault tolerance to the shared-disk system.
- If you use iSCSI for shared-disk system access, ensure you have configured iSCSI intiators and targets prior to installing Novell Cluster Services.

**GroupWise Clustering Specific Requirements**

The GroupWise Interoperability Guide includes a section on setting up and Configuring GroupWise to work with Novell Cluster Services. The guide also includes a System Clustering Worksheet that will help you make critical decisions regarding your GroupWise clustering implementation, such as whether or not to cluster-enable shared volumes used by GroupWise, how and where to install agent software in the cluster, determining appropriate failover paths for agents, and more.

The following are a few configuration considerations contained in the guide:

- GroupWise and Novell Cluster Services can run on NSS, ext3 or Reiser file systems. However, if you want to migrate an existing cluster from NetWare to Linux you will need to use NSS because NetWare clusters require NSS.
- GroupWise must be configured in Client/Server mode. GroupWise will not failover or failback in a cluster environment unless it is using Client/Server mode.
- The grpwise startup script must exist in the /etc/init.d directory of each node in the cluster and the gwha.conf file must exist in the /etc/opt/novell/groupwise directory of each

**TABLE 1: Use the following commands in the /etc/init.d directory to control the agents as daemons**

| DESCRIPTION | COMMAND |
|---|---|
| Start the POA, MTA, GWIA and Messenger agents | ./grpwise start |
| Start a specific POA, MTA or GWIA agent | ./grpwise start agent_location_name |
| Start the Messenger agents | ./novell-nmma start |
| | ./novell-nmaa start |
| Display the status of the POA, MTA and GWIA | ./grpwise status |
| Display status for a specific agent | ./grpwise status agent_location_name |
| Stop the POA, MTA, GWIA and Messenger agents | ./grpwise stop |
| Stop a specific POA, MTA or GWIA agent | ./grpwise stop agent_location_name |
| Stop the Messenger agents | ./novell-nmma stop |
| | ./novell-nmaa stop |

node. These files are created when GroupWise is installed.

- Each gwha.conf file should contain every agent that could possibly be loaded on that node.
- Even though some GroupWise daemons might load on all the nodes in your cluster because of the way GroupWise operates, you don't want your GroupWise agents to automatically load when the node or server boots up. For failover and failback to work properly, rely on the cluster scripts to load the appropriate agents on the appropriate nodes. To do this, manually remove the S99grpwise links in the rc3.d and rc5.d directories of each server so the agents won't automatically load during startup.
- The POA, MTA, GWIA and Messenger agents should be installed on each node in the cluster. By default, these agent startup files are stored in the /opt/novell/groupwise/agents/share directory. You can modify the gwha.conf file to specify a shared location for these agents.
- Secondary IP addresses are required for each GroupWise cluster resource. These IP addresses float with the resource when a failover or failback occurs. They are not bound to a specific server node in the cluster. If your MTA and POA are configured as the same cluster resource, they will share the same secondary IP address. If they are configured as different resources, they will require unique secondary IP addresses. You assign only one secondary IP address for each shared cluster resource.
- All domain links must be TCP/IP, not file path.
- Novell Cluster Services uses Postfix to send e-mail alerts. If you have a cluster resource that uses SMTP, that resource might not work in the cluster unless you change the Postfix configuration files to use a different port than the SMTP port used by GroupWise.
- Once GroupWise is installed and configured to run in a cluster environment, you must create and configure a GroupWise resource in Novell Cluster Services. This includes configuring GroupWise load and unload scripts; setting GroupWise Start, Failover and Failback modes; and assigning the GroupWise resource to specific nodes in the cluster.

**Enterprise Class Non-Stop Service**

Whether you want enterprise class collaboration services on a single Linux server or in a cluster of Linux servers, GroupWise 7 and Novell Cluster Services offer you the level of high availability that best fits your business needs. After all, when it comes to the critical business services that drive your productivity and profitability, you can't afford to settle for anything less than enterprise class, non-stop service. N

## SPLIT BRAIN DETECTION

Only Novell Cluster Services can use the Novell patented Split Brain Detector to avoid data corruption that can occur during split-brain conditions. A split-brain condition exists when a disruption in LAN communication makes it impossible for normal internode communication to occur within the cluster. In this event, certain nodes might become isolated from the others such that the separate nodes think they are the only surviving nodes. This creates a dangerous situation because all nodes might still have access to the shared data storage system. In this instance, if two separate nodes access the same volume, data corruption might occur. The Split Brain Detector in Novell Cluster Services detects and resolves these conditions and ensures that no data corruption occurs.