



Lotus Domino Configuration for
High Availability
Using
SteelEye LifeKeeper for Linux

This document details the process by which you can build a High Availability configuration for Lotus Domino using SteelEye Technology LifeKeeper for Linux. It is adapted from a white paper originally written by H-P's Lotus Domino Solutions team to document work which was performed in the construction of such a solution.

LifeKeeper Installation

This section provides a brief overview of the LifeKeeper for Linux installation process. For more detailed installation instructions, refer to the LifeKeeper product documentation available at www.steeleye.com.

First install the LifeKeeper core platform files. This installation will perform several tasks:

- Check the Kernel version
- Upgrade the kernel to the supported version
- Check for required versions ncurses and glibc packages
- Check the host-bus adapter (HBA) and offer to upgrade the HBA driver if needed

Second, install the platform core files on each of the remaining cluster nodes.

Once the platform installation is complete, install the LifeKeeper for Linux packages, these packages include:

- LifeKeeper core product
- LifeKeeper Application Recovery Kits
- LifeKeeper GUI manager
- LifeKeeper Documentation

Repeat the installation of the LifeKeeper packages on each of the remaining cluster nodes.

You are now ready to configure the LifeKeeper cluster.

Configuring the LifeKeeper Cluster

LifeKeeper is configured and managed through the LifeKeeper JAVA-based Graphical User Interface (GUI). The GUI is used to build communication paths between servers, create resources to protect, and define resource dependencies.

The LifeKeeper GUI can be started by launching `/opt/LifeKeeper/bin/lkGUIapp`, or if you have selected Gnome as your desktop manager select **Start/System/LifeKeeperGUI**. Start the LifeKeeper GUI and select **Edit/Server/ Create comm. path** as shown in Figure 1.

Communication paths must be created between each cluster node to tie the four nodes into a single clustered environment.

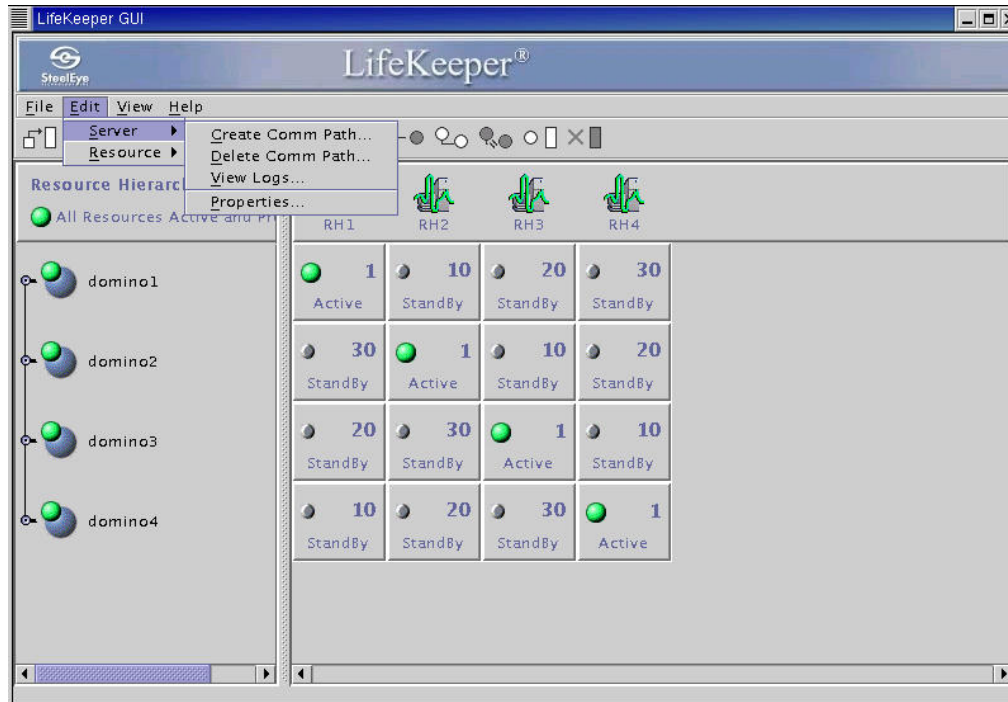


Figure 1. Creating Comm Path

Communication paths should be created using both the public and private networks to provide a level of redundancy as protection against a single down communication path becoming a single point of failure for your environment.

Figure 1 shows the communication paths for cluster node RH1. Communication paths are created between RH1, RH2, RH3, and RH4 on Networks 10.0.0.0 and 129.2.0.0. Continue creating communication paths on the remaining cluster nodes until there are two communication paths between each server as shown in Figure 2.

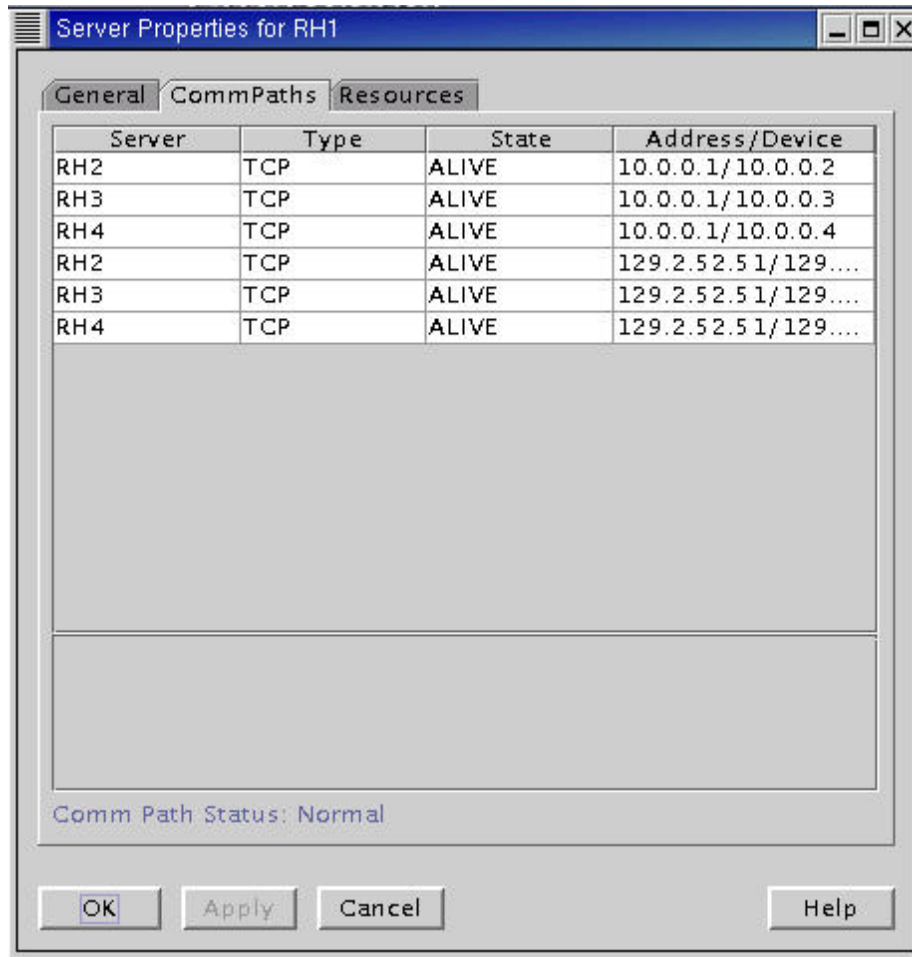


Figure 2. Create communication paths on the remaining cluster nodes until there are two communication paths between each server

Creating cluster resources

Cluster resources are system entities that are protected by the cluster nodes through failover. If a cluster node fails, the protected resource will be started on a designated alternate cluster node. A Domino server partition requires three cluster resources in order for it to be protected:

- IP resource – allows persistent client connectivity to Domino server after failover.
- File System resource – Shared logical volume that hosts the Domino server data directory.
- Generic Application resource – Scripts that start and stop Domino server partitions on cluster nodes. The Generic Application resource type is used to allow protection to be added for any application or system entity for which LifeKeeper does not have built-in protection. The Generic Application resources will be created after the Domino server partitions are installed on each cluster node.

IP resource

The IP address cluster resource is one or more virtual IP addresses that clients will use to connect to the Domino server. The IP address that clients use to access the Domino server partition remains constant regardless of which cluster node the Domino server is running on. This mapping of IP address and Domino server partitions is accomplished through a Parent / Child relationship formed by creating a resource dependency between the IP address resource and the Domino server partition (Generic Application) resource. To create the IP address resource, use the LifeKeeper GUI and select **Edit /Resource / Create Resource Hierarchy** as shown in Figure 3. This will invoke the Create Resource Wizard shown in Figure 4; select **IP Recovery Kit**. In the example, we created four IP address cluster resources that will bind virtual IP addresses 129.2.52.61 through 129.2.52.64, one IP address will be bound to each Domino server partition. These virtual IP addresses will move between cluster nodes along with the Domino server partition in the event of a server failure. Using the LifeKeeper GUI an administrator can manually move the cluster resources to an alternate cluster node in order to perform system maintenance.

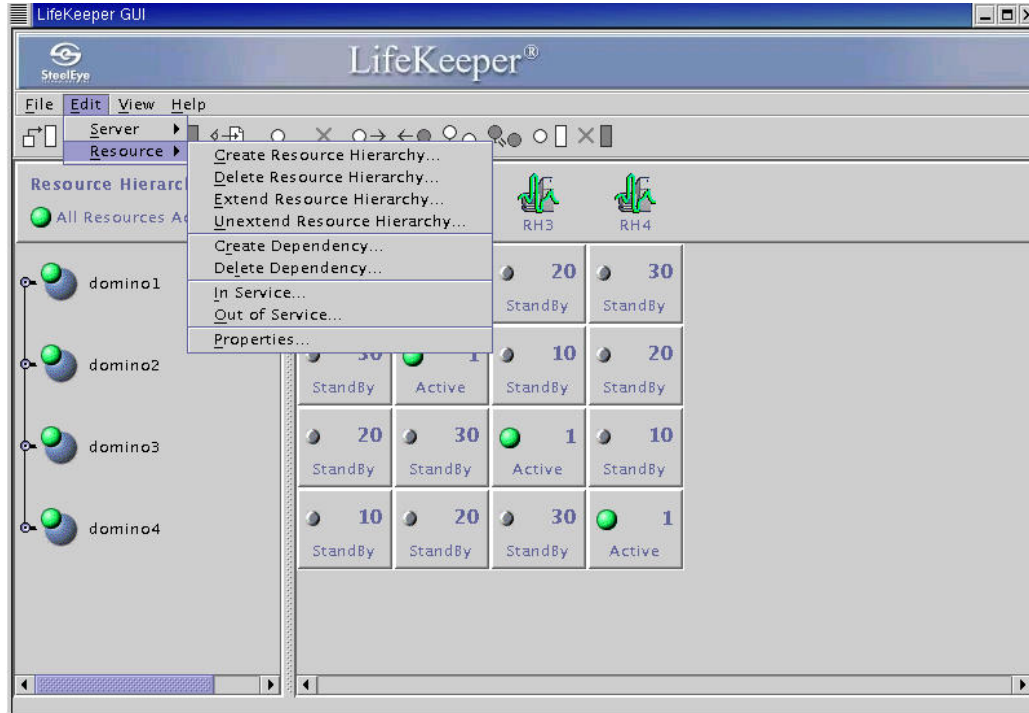


Figure 3. Create Resource Hierarchy

File System Resource

The File System resource is the cluster resource that will host the Domino data directory. The Domino data directory contains all the Domino databases, Domino directory, and configuration information specific to each Domino server partition. Domino server partitions each share a common set of program files and have unique Domino data directories. The file system that hosts the Domino data directory will be moved between cluster nodes along with the IP address and Domino server Generic Application resource. The Domino directory must be online and accessible before launching the Domino Server Generic Application resource script. This is accomplished by creating a dependency relationship between cluster resources, discussed later in this document.

Using the same procedure that was used to create the IP address resource, launch the LifeKeeper GUI and select **Edit / Resource / Create Resource Hierarchy** as shown above in Figure 6. This will invoke the Create Resource Wizard as shown in Figure 4; select **File System Recovery Kit**. Use the File System recovery kit wizard to create a File System cluster resource for each of the four file systems mounted on mount points /dsk1, /dsk2, /dsk3, and /dsk4.

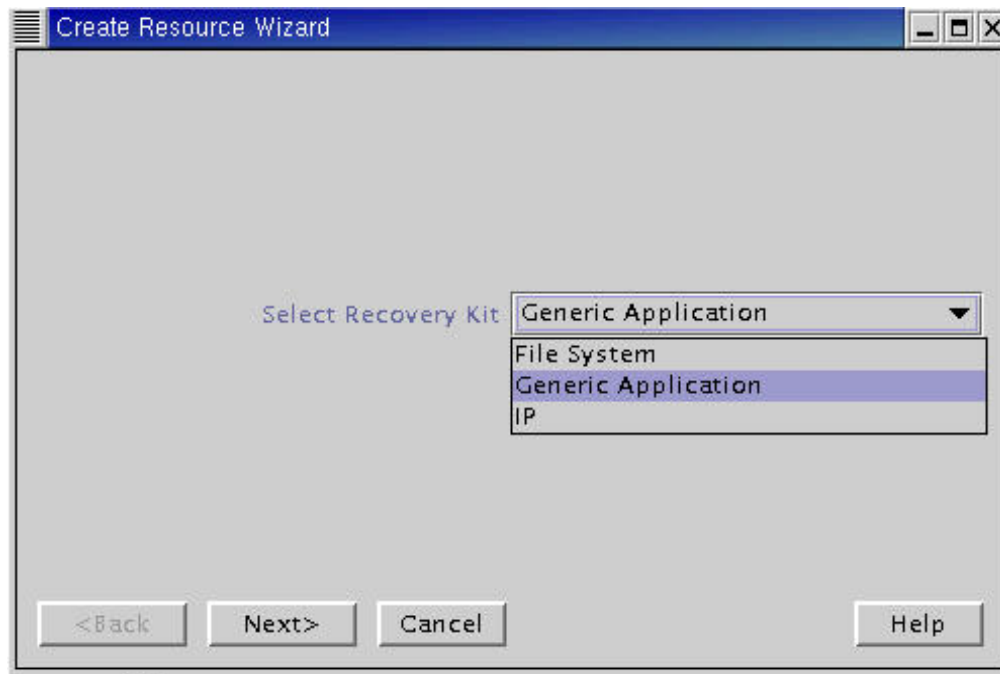


Figure 4. Create Resource Wizard

Extending Resource Hierarchy

Once a resource is created on the first node of a cluster, it can be extended to other cluster nodes capable of hosting the resource. Availability of a cluster resource is increased by extending the resource to other cluster nodes. This allows the resource to be protected by other nodes in the cluster. The cluster resource hierarchies do not need to be manually recreated on each cluster node. By extending the cluster resources to additional cluster nodes the resource hierarchies are created on the additional cluster nodes.

Domino Partition Server installation

The Domino partition server feature allows you to run separate autonomous instances of Domino on a single computer using a shared set of Domino program files. The Domino partition server architecture allows the flexibility to install Domino in a clustered server environment with minimal modification to a default Domino partitioned server installation.

In the previous sections, we created cluster resources for the file systems and IP addresses. The File system resources will host the Domino data directories and the IP address resources will provide client access to the Domino servers. The Domino server partitions must be installed on each cluster node before creating the Generic Application resource used to start and stop each Domino server partition. During the installation process, install the Domino server program files on local non-shared storage attached to each of the four cluster nodes. The Domino data directories for each partition will be installed on the four shared-file-system cluster resources created in the preceding section.

The default installation path for the Domino program files is `/opt/lotus/`; you must ensure that there is sufficient space in the root file system for the Domino server program files. If there is insufficient space, the default location can be changed during the installation process. Verify all resources are "In Service" by opening a terminal window and executing the `df` command to display the shared file systems `/dsk1`, `/dsk2`, `/dsk3`, `/dsk4` and the `ifconfig` command to display the virtual IP addresses.

Using the `tar -xvf` command, extract the Domino installation kit to a temporary installation directory. From the installation directory, execute `/Linux/install` command to launch the Domino server installation script.

Domino Server Type

After agreeing to the Domino server licensing agreement, you will be presented with several options used to configure the installation process. The first option shown in Figure 5 prompts the user for type of installation to be performed. In this example, we chose to install the Domino Mail Server option.

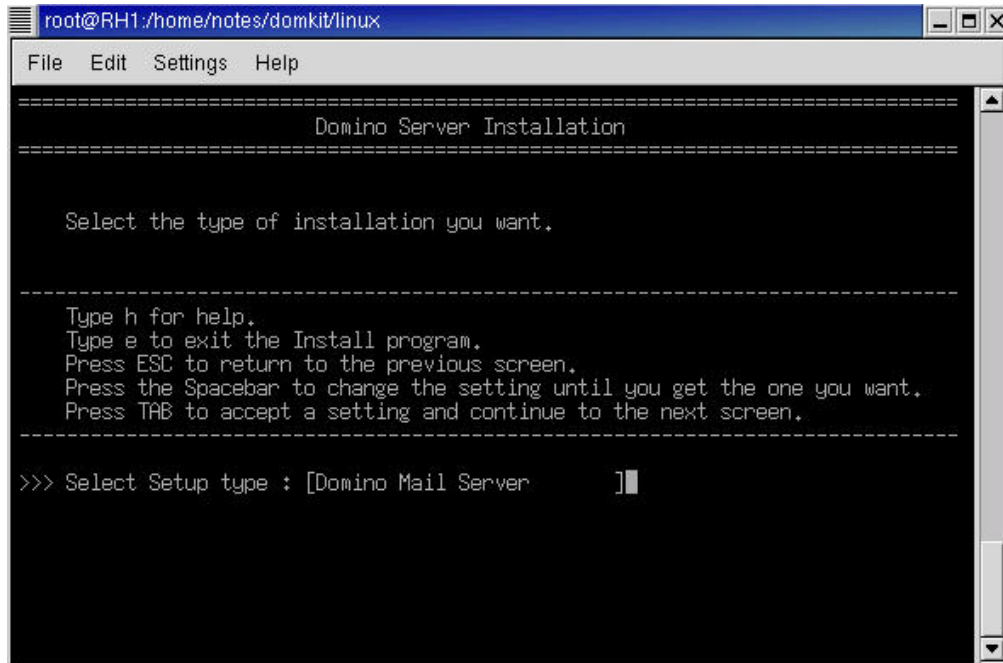


Figure 5. Configuration Options

The next screen presented during the Domino Server Installation process asks for the location of the Domino Server Program files as shown in Figure 6. Press **tab** to select the default location of **/opt/lotus** or press **enter** to edit and type in an alternate location. Provided there is sufficient free disk space on the root file system, use the default location **/opt/lotus** for the Domino Server Program files.

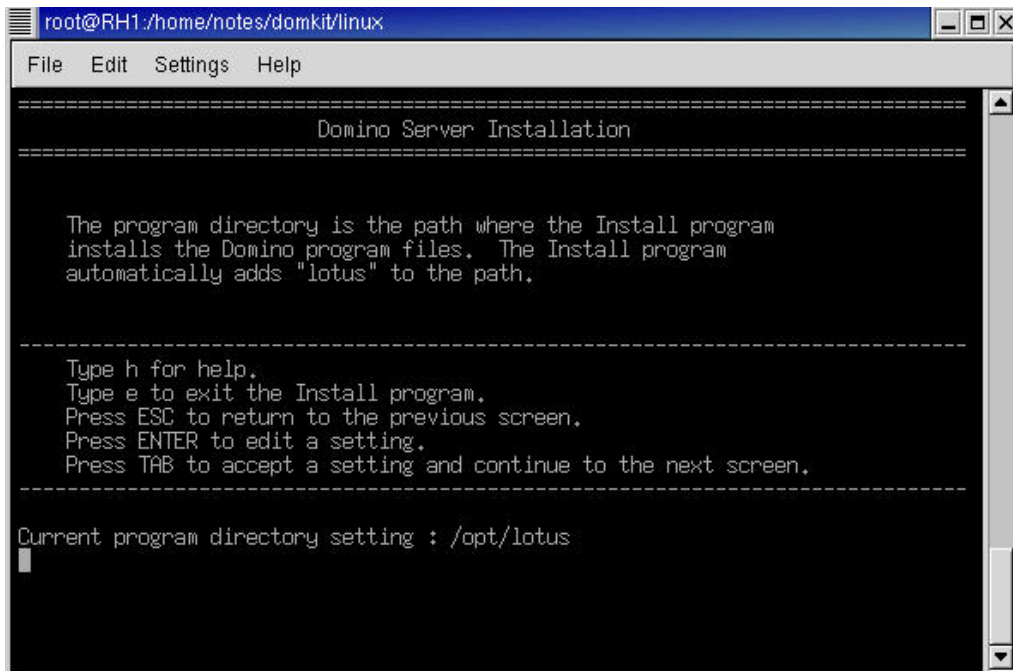
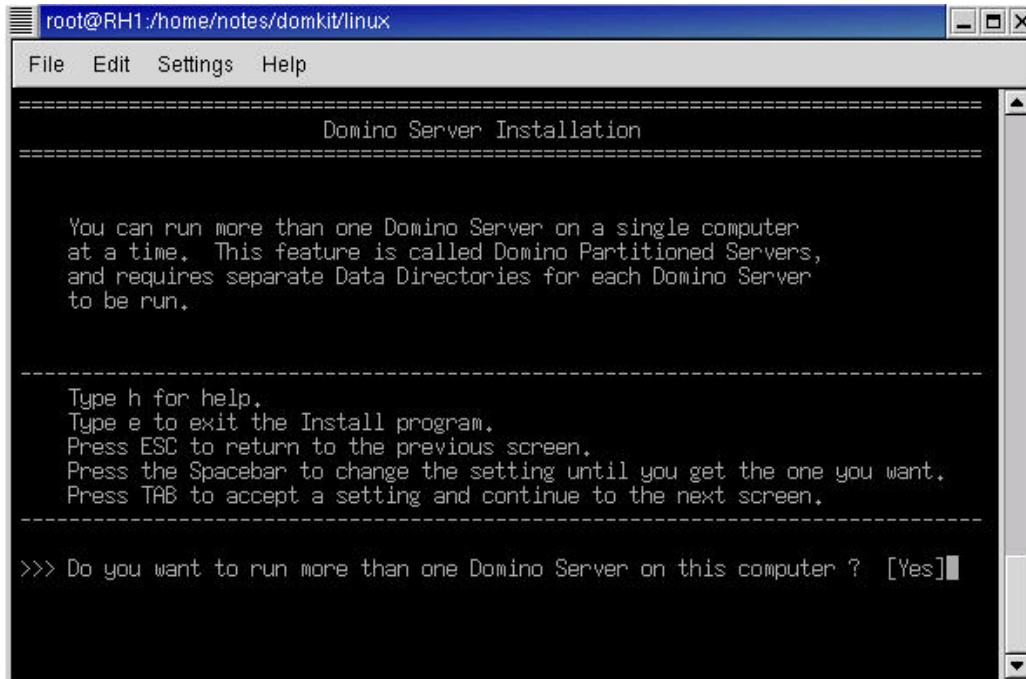


Figure 6. Location of the Domino Server Program files

Once a location for the program files has been selected, the installation program asks the user if they would like to run more than one Domino server on this computer. This is the Domino Partitioned Server feature; at the top of the screen there is a brief description of the Domino Partitioned Server feature shown in Figure 7. Press **tab** to edit and select **yes** for this option. This will invoke the Domino Partition Server installation shown in Figure 8.

Domino Partition Installation



```
root@RH1:/home/notes/domkit/linux
File Edit Settings Help
=====
Domino Server Installation
=====

You can run more than one Domino Server on a single computer
at a time. This feature is called Domino Partitioned Servers,
and requires separate Data Directories for each Domino Server
to be run.

-----
Type h for help.
Type e to exit the Install program.
Press ESC to return to the previous screen.
Press the Spacebar to change the setting until you get the one you want.
Press TAB to accept a setting and continue to the next screen.
-----

>>> Do you want to run more than one Domino Server on this computer? [Yes]
```

Figure 7. Domino Partitioned Server feature

Each Domino server partition uses a common set of program files located by default in **/opt/lotus** on a local file system that is not configured as a cluster resource. Each Domino partition has a unique Domino data directory. The Domino data directory contains the notes databases files *.nsf;

These files contain user mail files, application databases, and configuration information for the Domino server partition. In Figure 8, you are asked how many Domino data directories you would like to install. In this example, we are going to create one Domino server partition for each node in our cluster. There is no rule or restrictions that require us to map the number of Domino partitions to the number of cluster nodes. We can have one Domino server partition and four cluster nodes to protect it or we can have eight Domino server partitions on two cluster nodes. When the Domino Partitioned Server installation is selected. The default option for the number of data directories is two. In this example, we are going to change this to four Domino data directories, one for each of our cluster nodes.

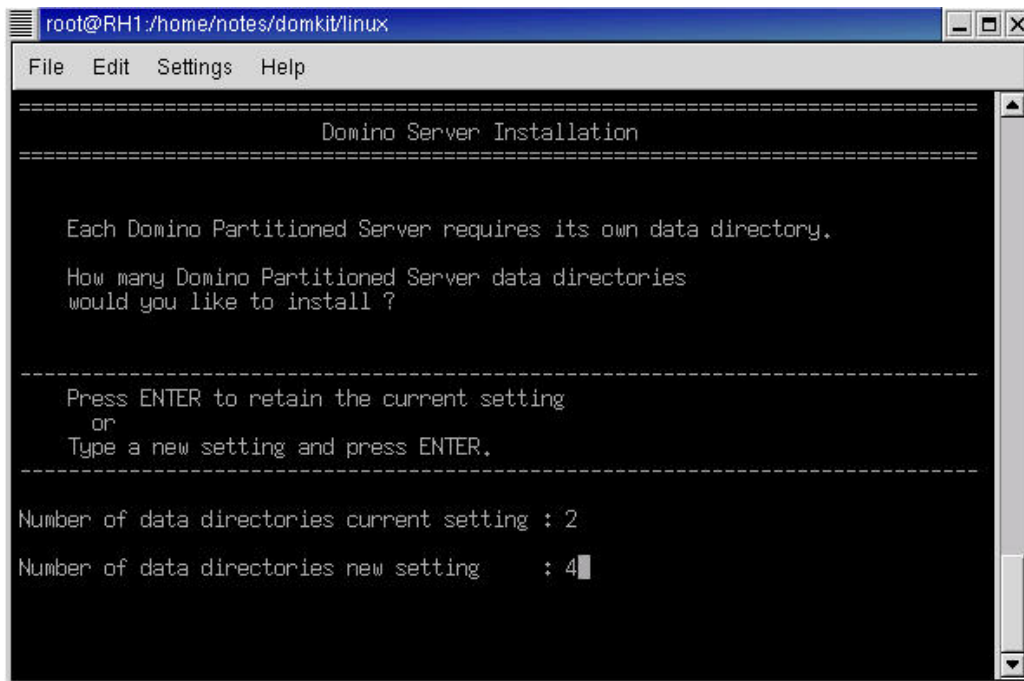


Figure 8. Number of Domino data directories desired

The installation process prompts for a location of the Domino Data directory as illustrated in Figure 9. The Domino Data directories will be located on the File Systems that are protected by the File System Cluster resources created when configuring the LifeKeeper cluster. These are the file systems that are located on the shared MSA1000 fibre Storage Array. In this example, we created four logical disks; these disks are presented to the operating system and are identified as the following devices: /dev/sda1, /dev/sdb1, /dev/sdc1, and /dev/sdd1. We formatted these partitions with the ext3 file system and enabled the journaling option to provide file system integrity. These formatted file systems are mounted on /dsk1, /dsk2, /dsk3, and /dsk4 on each cluster node. LifeKeeper controls access to the file systems and allows only one cluster node at a time to mount these file systems. During the Domino installation, we used the LifeKeeper GUI to move all four file systems and mount them onto one cluster node. We will now identify these file systems as the location for the Domino data directories. This screen will be repeated as the installation script is configured for each subsequent Domino server partition.

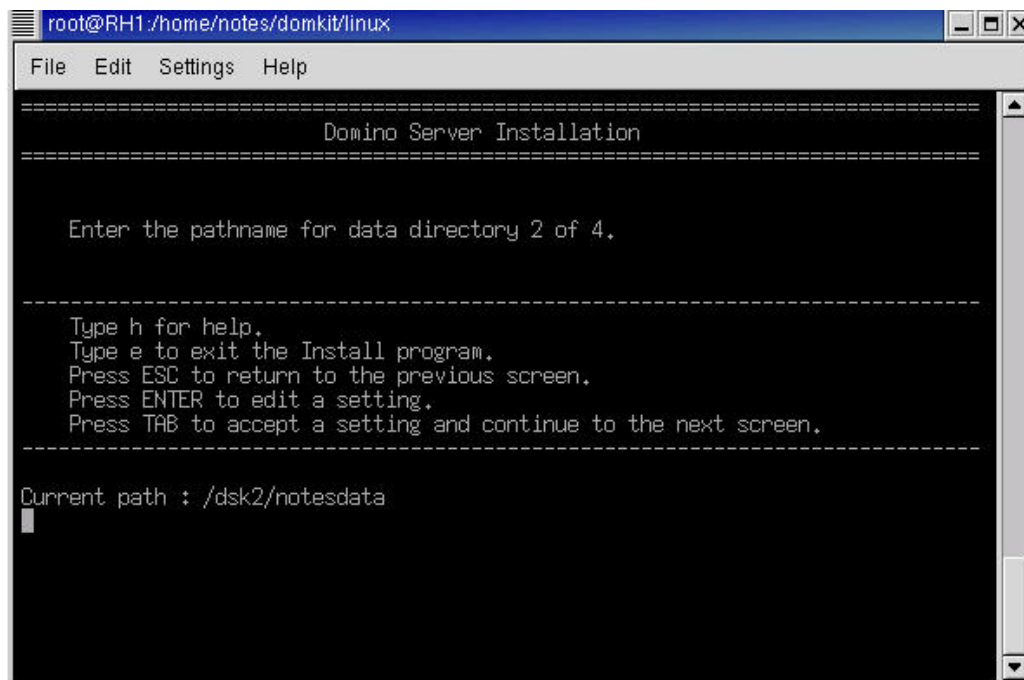


Figure 9. Prompt for a location of the Domino Data directory

Lotus Notes users and groups

Each Domino Server Partition's data directory is owned by a unique user account.

Lotus does not recommend starting the Domino server partitions as the root user account; instead, we chose notes1, notes2, notes3, and notes4 as user accounts for our Domino Server partitions. These accounts will own the processes and data directories for each Domino Server partition.

The four user accounts are all members of the operating system group notes. By using unique user accounts for each Domino server partition, one can group Domino Server files and processes by user name. This can be helpful when developing scripts to kill off orphaned processes that may exist after performing manual failovers of the Domino Server partitions.

Figures 10 and 11 illustrate the procedures for selecting the user and group accounts to be used with each Domino Server partition. These steps will be repeated for each Domino Server partition.

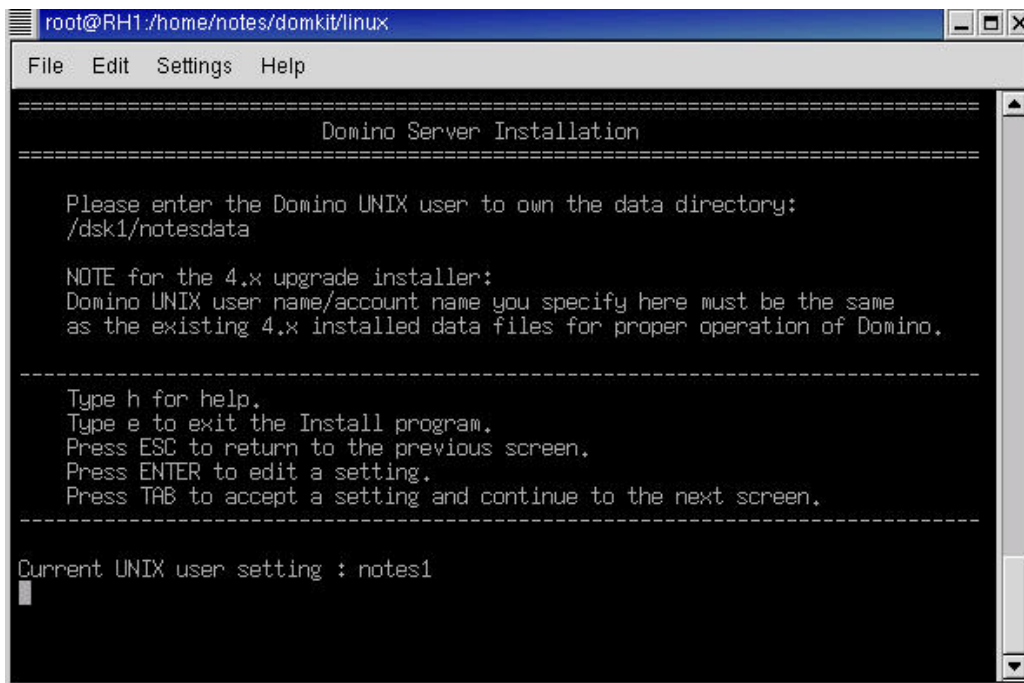


Figure 10. Selecting the user and group accounts to be used with each Domino Server partition

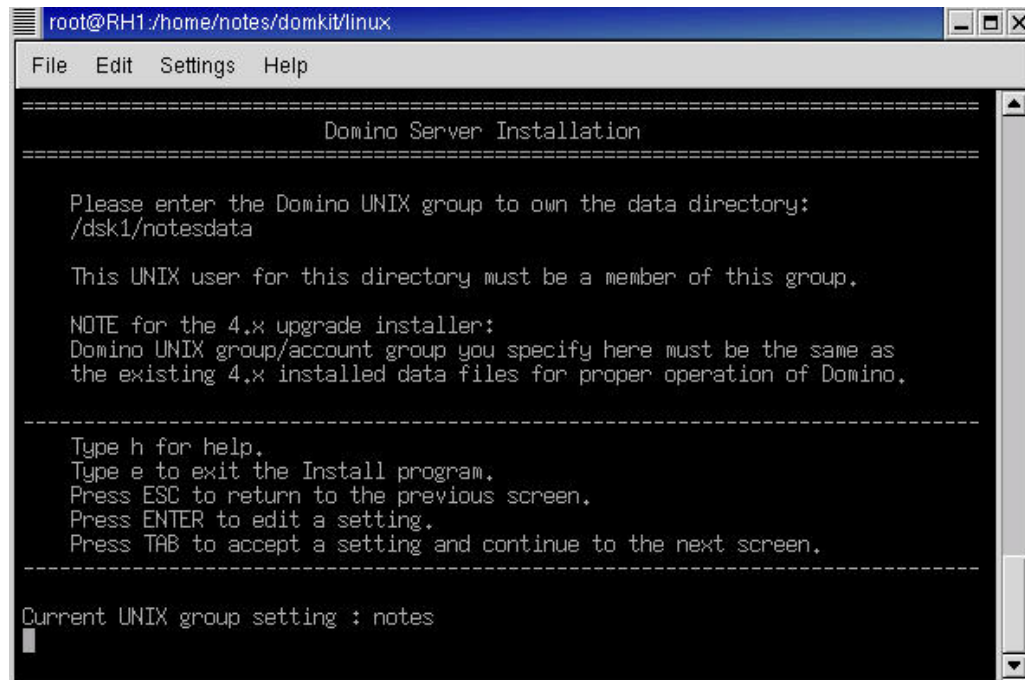
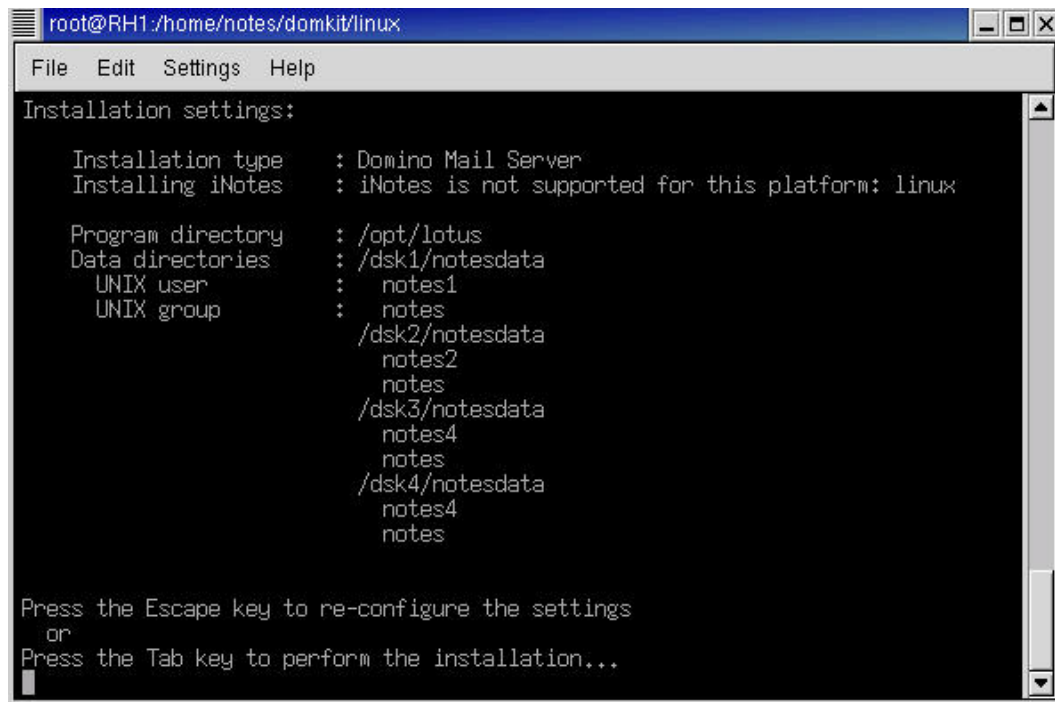


Figure 11. Selecting the user and group accounts to be used with each Domino Server partition – cont'd

Installation Summary

The user is presented with an installation summary screen as shown in Figure 12. The installation summary is a review of the options selected during the interactive portion of the installation. You now have the option of proceeding with the installation or making any changes before the Domino server partitions are installed onto the system.



```
root@RH1:/home/notes/domkit/linux
File Edit Settings Help
Installation settings:
Installation type      : Domino Mail Server
Installing iNotes     : iNotes is not supported for this platform: linux

Program directory    : /opt/lotus
Data directories     : /dsk1/notesdata
  UNIX user          : notes1
  UNIX group         : notes
                    /dsk2/notesdata
                    notes2
                    notes
                    /dsk3/notesdata
                    notes4
                    notes
                    /dsk4/notesdata
                    notes4
                    notes

Press the Escape key to re-configure the settings
or
Press the Tab key to perform the installation...
```

Figure 12. Installation summary screen

Completing the Domino Partition Server Installation

Domino partitions are now installed on only one node of the cluster. We must repeat the Domino partition server installation process for all Domino partitions on each of the remaining cluster nodes. This process will place the Domino server program files on a local file system on each of the cluster nodes.

Modifications to the Domino directory (**names.nsf**) and **notes.ini** file are required to bind the client protocols to specific IP addresses. These modifications are required to prevent conflicts resulting from network services attempting to bind to any available IP address. Use the LifeKeeper GUI to place all four IP address and File System cluster resources “In Service” on a single cluster node. Each Domino server partition must be configured to bind to a specific IP address. This is done by editing the **notes.ini** file found in the data directory for each Domino server partition. Using the addresses created as IP address cluster resources, modify the **notes.ini** file to include the following entry:

- TCP_TcpipAddress= 0,129.2.52.61

The **notes.ini** file can be found on the shared file system; each Domino server partition has its own **notes.ini** file and will be modified to specify the IP address created by the IP Address cluster resource. This is the IP address which Notes clients will use to access the specific Domino server partition.

In this solution, the Domino server partitions do not use port mapping and will not share IP addresses. Each Domino server partition will bind to a unique specific IP address created by the IP Address cluster resource and associated with the Domino server partition through a resource dependency. Use the Notes client to modify the Domino directory (**names.nsf**) server document settings to bind Internet Protocols to specific IP addresses.

Domino Partition Server (generic application) cluster resource

Earlier in this document, we described how to create the IP Address and File System cluster resources. We will now describe how to create the Generic Application resource used to start and stop the Domino server partitions. With the Domino server partitions installed on each cluster node, we can now create the Domino Partition Server Generic Application cluster resources. These resources will be used to control and monitor the Domino server partitions on each cluster node. In this example, the Generic Application cluster resource consists of a simple shell script used to start and stop a Domino server partition. To create the Generic Application Resource, launch the LifeKeeper GUI and select **Edit / Resource / Create Resource Hierarchy** and select the **Generic Application Recovery Kit** as shown previously in Figures 3 and 4.

The Generic Application Recovery Kit will allow you to select the path to the start and stop scripts. Before creating this resource, the Domino server partitions must be installed on each cluster node. In the start script below, we simply changed directory paths to the data directory for each Domino server partition, and changed the user to the user account that owns the Domino server partition and launch the server command as a background process by appending the **&** to the end of the process. We then indicated a successful completion of the script with an **exit 0** command.

Sample script to start a Domino server partition:

```
#!/bin/sh
cd /dsk1/notesdata
su notes -c /opt/lotus/bin/server&
exit 0
```

The stop script below is used to manually take a Domino server partition out of service for maintenance, upgrade, or repair. This script is executed whenever the Domino Partition Server resources is placed "Out of Service", or placed "In Service" on a different cluster node. The script changes the current directory to the Domino server partition data directory, then switches to the user account that owns the Domino server partition and launches the Domino server command with the **-q** (quit) option that initiates a graceful shutdown of the Domino server partition.

Sample script to stop a Domino server partition:

```
#!/bin/sh
cd /dsk1/notesdata
su notes -c '/opt/lotus/bin/server -q'
exit 0
```

A unique start and stop script must be created for each domino server partition running in the cluster. The scripts will contain reference to each of the Domino partition server data directories (/dsk1/notesdata, /dsk2/notesdata, /dsk3/notesdata, and /dsk4/notesdata) and references to each user account that owns the Domino server partition data directories (notes1, notes2, notes3, and notes4). When the Generic Application cluster resource is extended to additional cluster nodes, the scripts will be dynamically created on those nodes.

The sample scripts used in our example are basic scripts used to start and stop the Domino server partitions. More complex scripts that contain error checking should be created for production environments so that LifeKeeper can provide health monitoring and local recovery of the Domino services in addition to the ability to stop/restart in the event of a server failure.

Resource Dependency

Cluster resources must be started in a predetermined order to ensure a successful failover and initialization of the Domino server partition. The resource starting order is accomplished by creating a dependency between cluster resources. A resource dependency ensures that one cluster resource is available before attempting to start another resource.

In a Domino partition server environment, the Domino server will not initialize properly if the server does not have access to the Domino data directory or to a valid IP address. It can be said that the Domino server partition is dependant upon the IP address and the File System resources. The IP address and File System that hosts the Domino data directory are configured to failover between cluster nodes with the Domino server. If the Domino server was to attempt to start before the file system was mounted by the cluster node, the server process would fail. Similarly, if the Domino server started before the IP address was bound to the network adapter, the clients accessing the server via the virtual IP address would fail to reconnect after a failover.

In this example, we created two resource dependencies: one resource dependency between the Domino Server Generic Application resource and the IP resource, and another resource dependency between the Domino Server Generic Application resource and the File System resource. The Domino server Generic Application resource is now dependent upon both the IP address resource and the File System resource; thus it will not attempt to start until both resources are online. This resource dependency creates a Parent / Child relationship between the Domino server Generic Application resource and the IP address and File System resources. This Parent / Child relationship can be seen in Figures 13 and 14, and these figures show that dsk1 and ip – 129.2.52.61 are child resources for domino1. These resources are all active and in service on cluster node RH1, as illustrated by Figure 13.

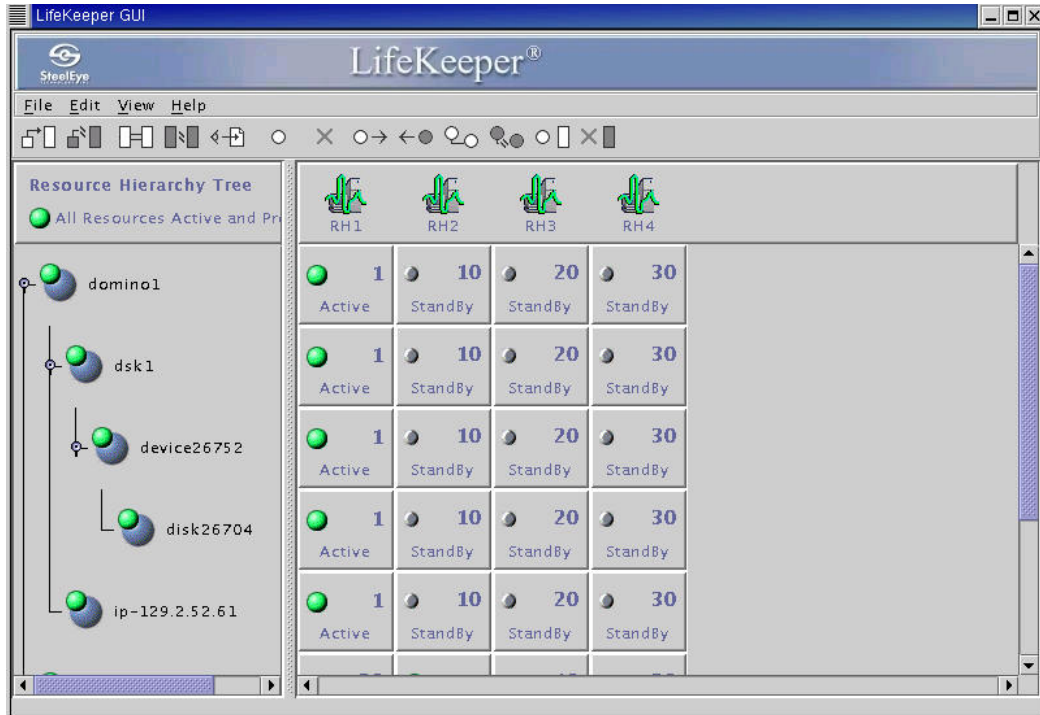


Figure 13. Resource Hierarchy

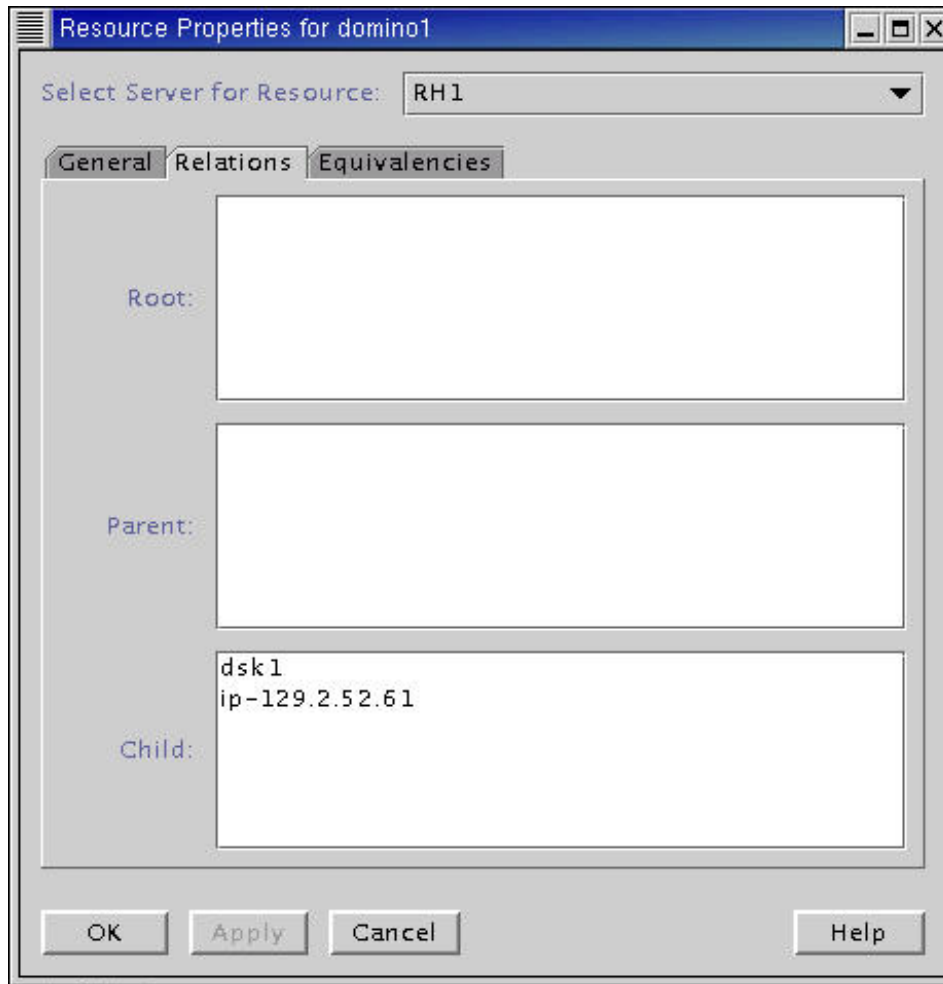


Figure 14. Parent / Child Relationship

The screen shot in Figure 15 shows a summary of a LifeKeeper four-node cluster with four Domino server partitions. The cluster nodes are shown in the top row as RH1 through RH4. The Domino server partitions are shown in the left column and are labeled domino1 through domino4. Figure 15 also shows that domino1 is active on node RH1, domino2 is active on node RH2, domino3 is active on node RH3, and domino4 is active on node RH4.

Controlling and Managing Resource Failover

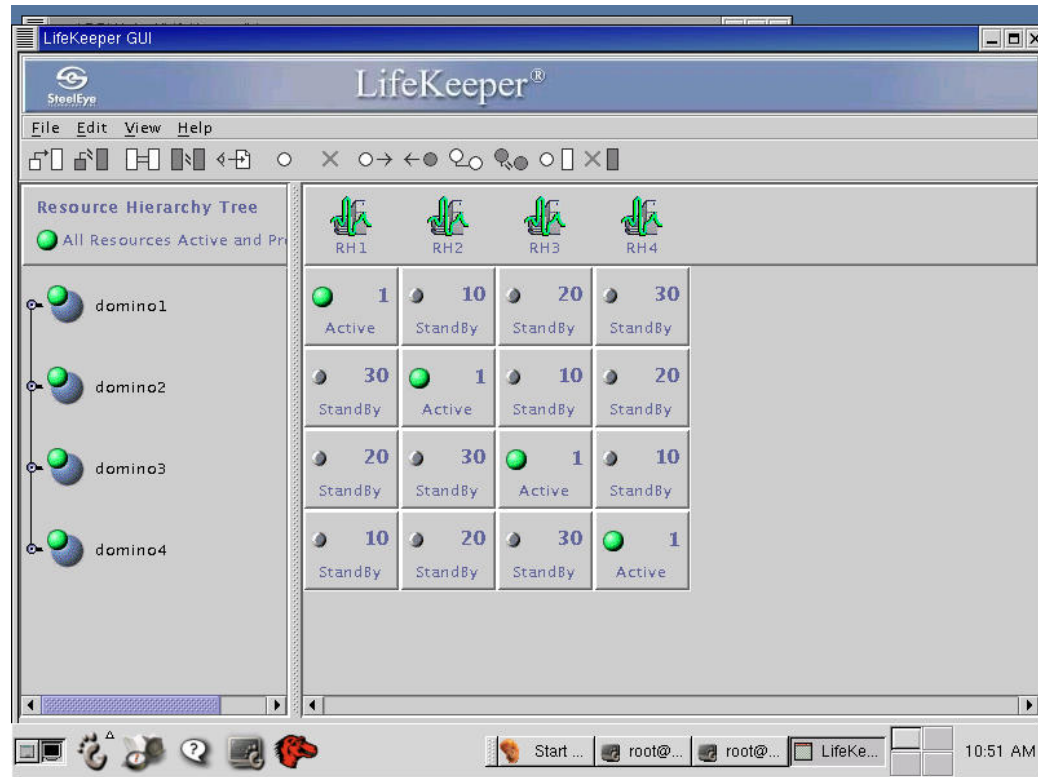


Figure 15.Cluster Nodes

In Figure 15, each status block indicates whether a resource is Active or Standby. Each status block has a number 1, 10, 20, or 30 associated with it. These numbers are used to determine failover priority for the cluster resource. A lower number indicates a higher priority for failover. If a cluster node fails, the surviving node with the lowest number (highest priority) for a protected resource will acquire and start that resource. In Figure 16, the failover priorities are shown for cluster resource domino1, similar property pages exist for all cluster resources. Dependent resources inherit failover priority settings from the parent resource, thus ensuring that the dependent resources failover to the correct cluster node and are in service when the parent resource attempts to start.

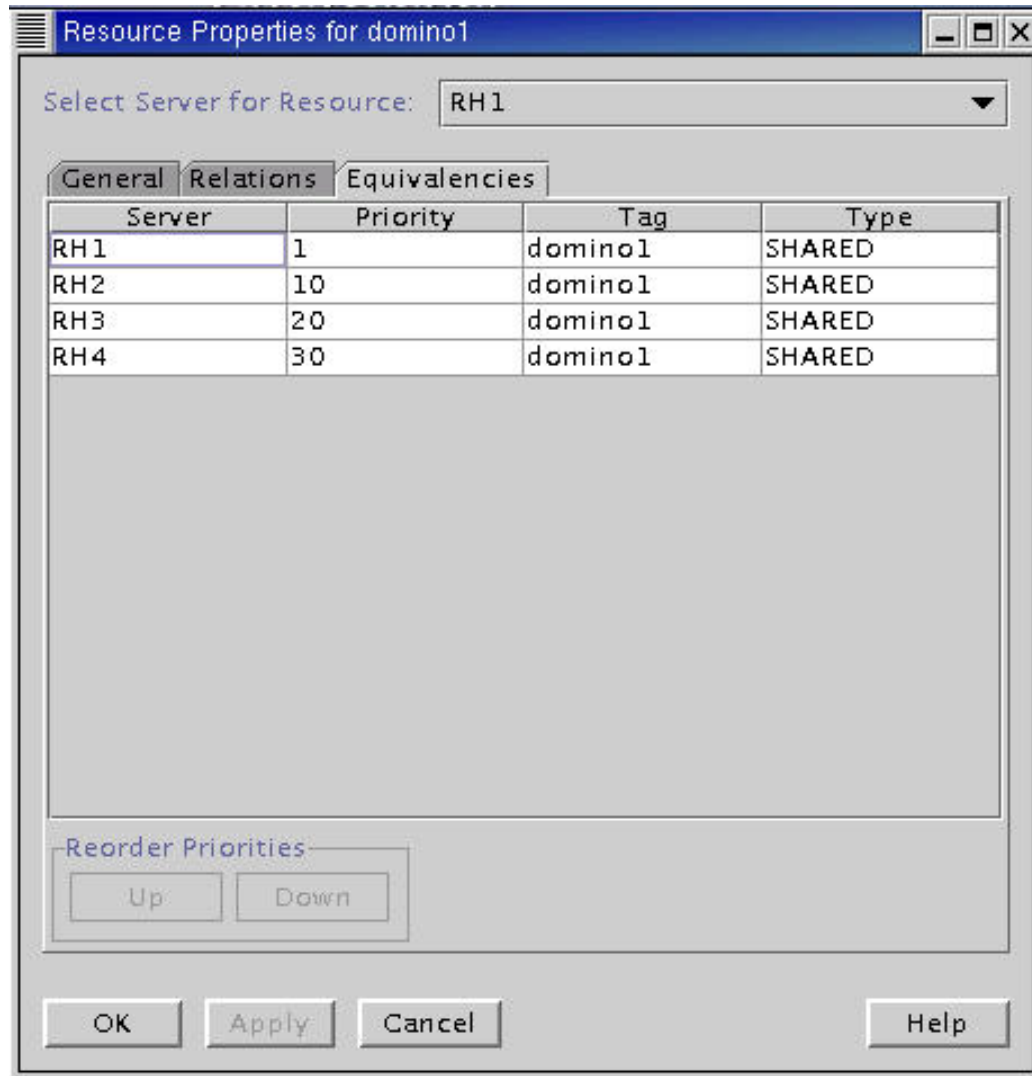


Figure 16. Failover priorities shown for cluster resource domino1

When a cluster node experiences a problem, the resources will fail to the cluster node with the lowest priority as shown in Figure 16. We want to be able to control the failback of the cluster resources. You want to avoid the situation where a server is going up and down repeatedly and disrupting service to the users by switching resources back and forth between cluster nodes. This is accomplished by setting the switch back property shown in Figure 17 to **OFF** so that the cluster resources will remain on the secondary cluster node until the primary node is repaired.

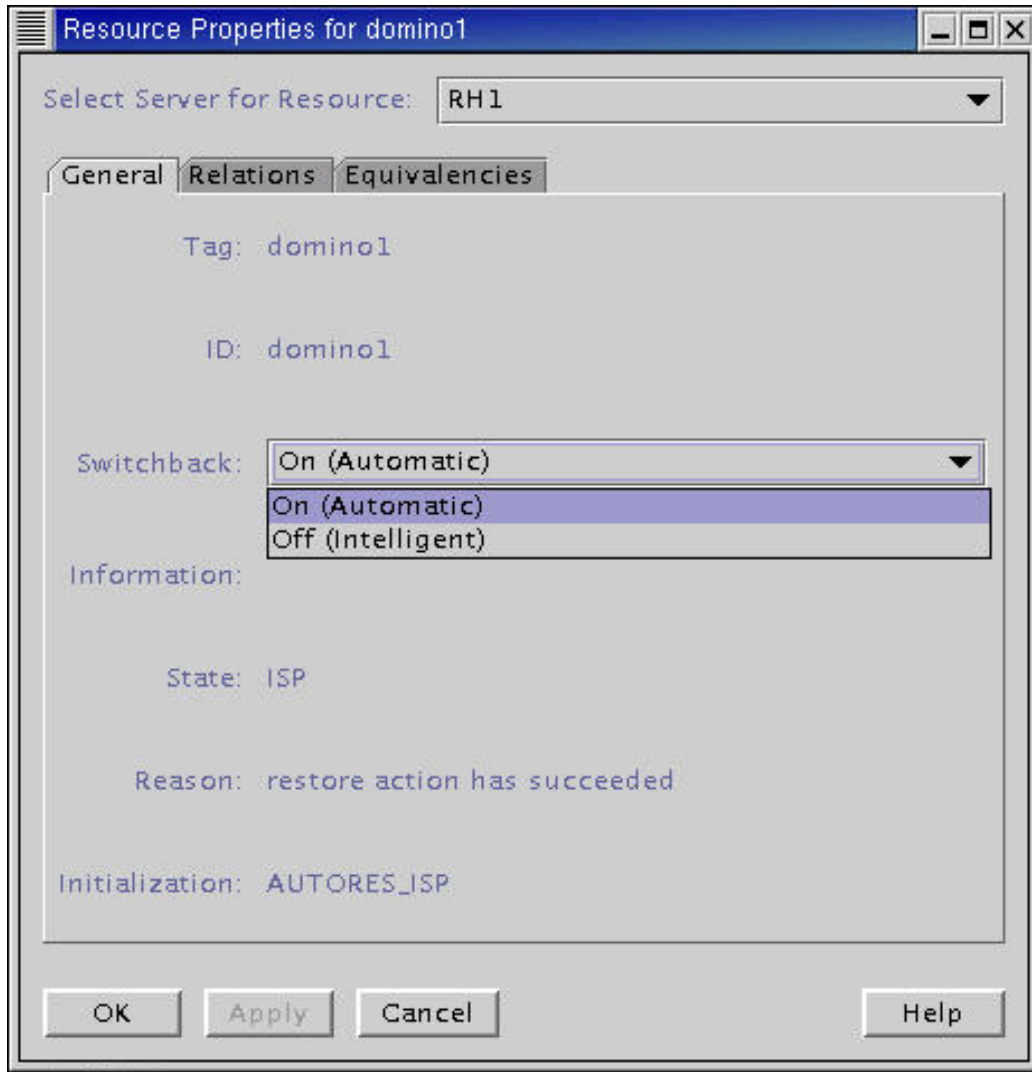


Figure 17. Set switchback property to "off."

You can also control how a cluster node handles a graceful shutdown. If the server is shutdown, the Server Properties can be configured to either switch the resources to a cluster node with the next lowest priority, or not to switch the resources at all. By not switching the resources to another cluster node, the resources would shutdown with the existing cluster node and be placed "Out of Service". Figure 18 shows the property page for controlling cluster resources during a server shutdown.

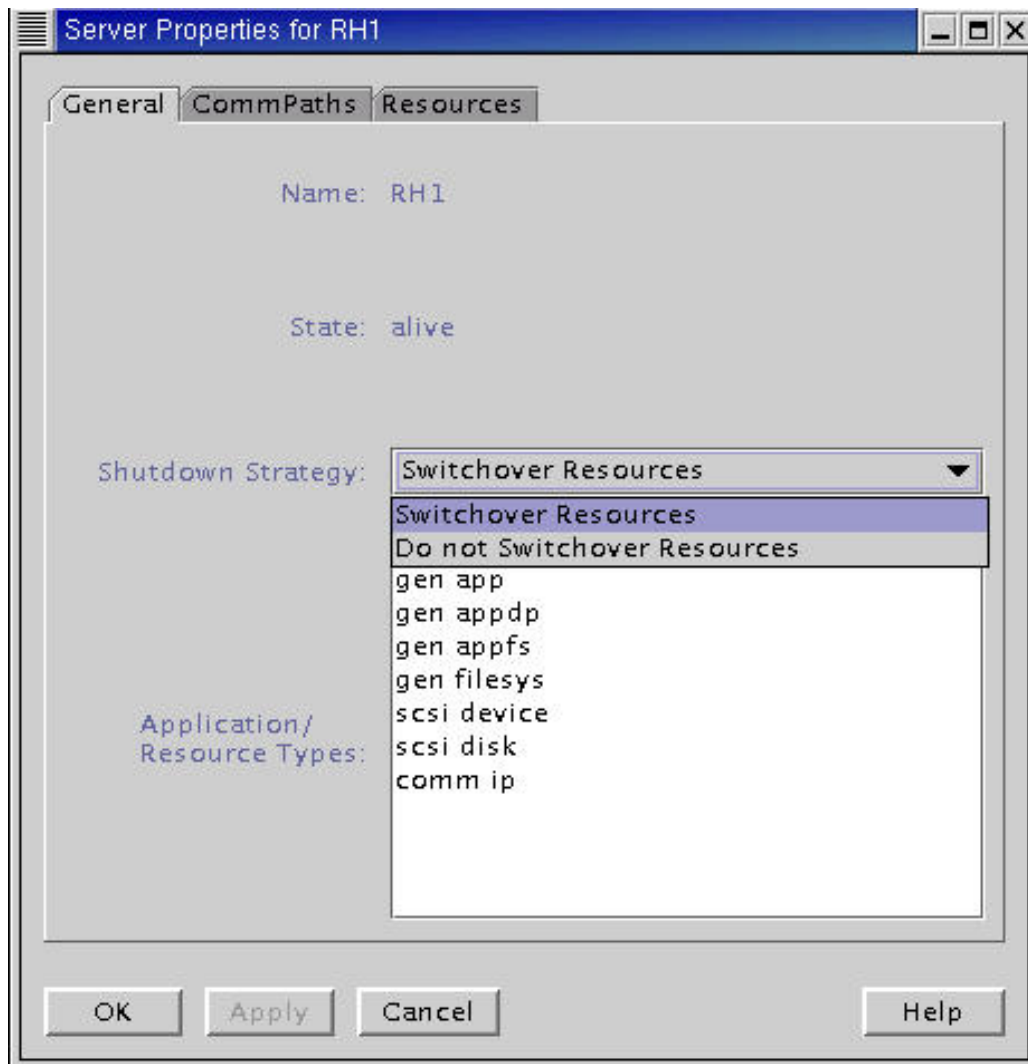


Figure 18. Property page for controlling cluster resources during a server shutdown

Example Cluster Configurations

The next three figures illustrate what happens to our Domino Server partitions in a four node cluster when we suffer a one, two, and three node server failure. In the beginning of this installation guide, Figure 1 shows a four-node cluster running normally with a Domino Server partition running on each cluster node. If node 1 were to fail, the IP address resource, the File System resource, and Domino Server 1 resource would be started on node 2 as shown in Figure 19. Node 2 would now be running Domino Server 1 and Domino Server 2 concurrently.

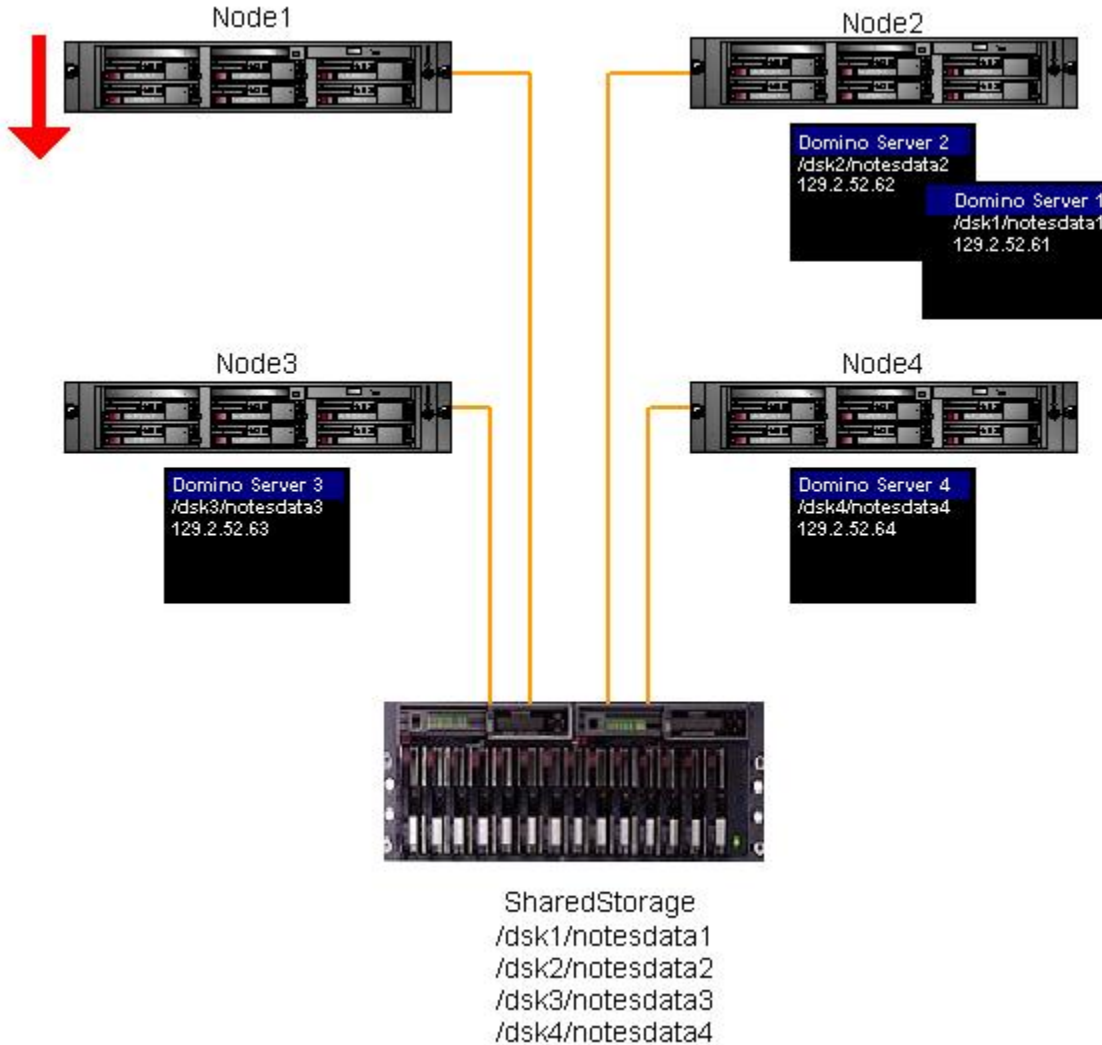


Figure 19. Failure of Node1. Start resource on Node2.

If this cluster suffers from a (simultaneous) two-node server failure, the cluster could be configured to dynamically distribute the cluster resources equally between cluster node 2 and cluster node 4 as shown in Figure 20. We have now lost two out of four cluster nodes and are still able to provide services to the end users.

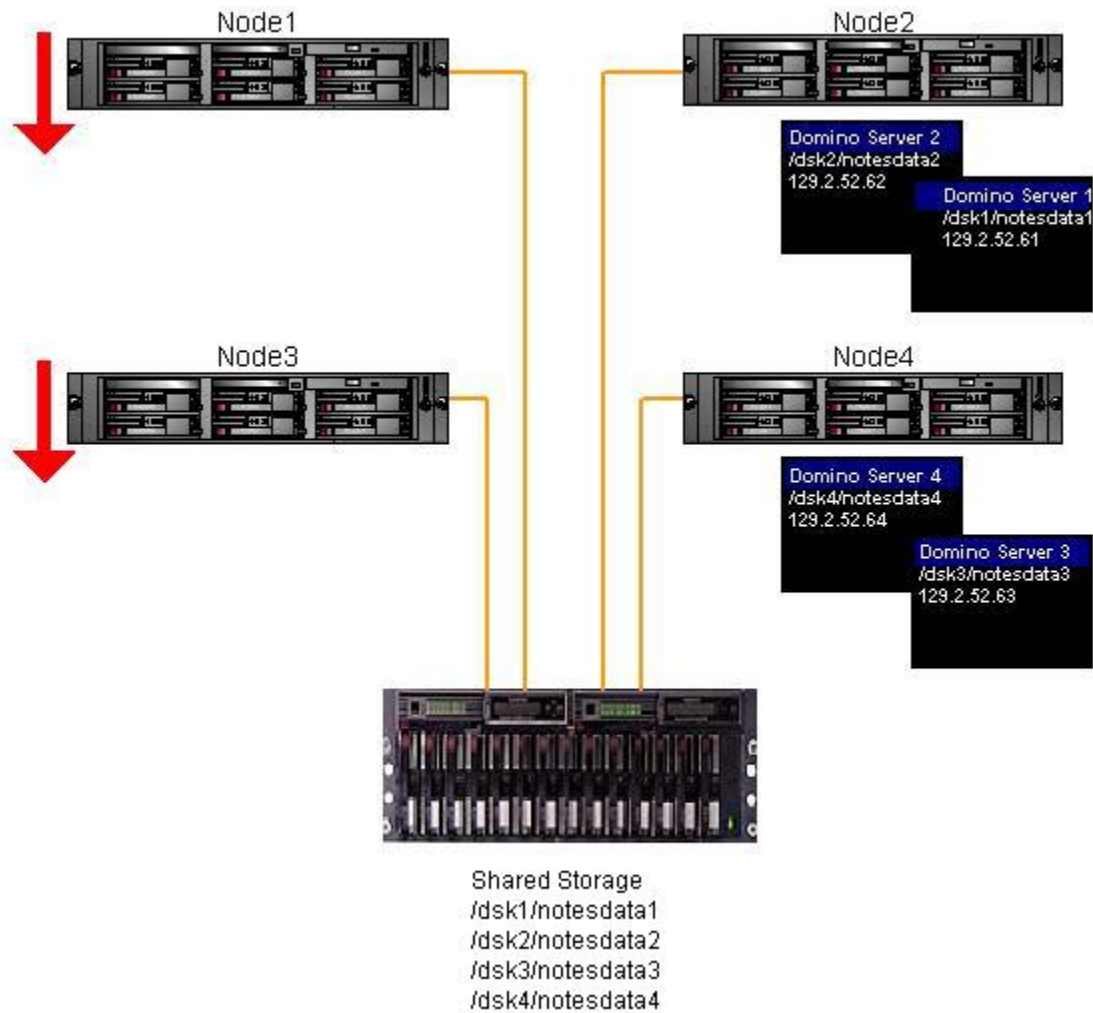


Figure 20. Failure of Two Server Nodes

In an extreme example of a catastrophic failure of three server nodes, as shown in Figure 21, the surviving cluster node can run all four Domino Server partitions concurrently to provide services to the user community. Proper server sizing and planning are required to ensure the surviving cluster node(s) can handle the additional load placed on the system resources and maintain acceptable performance levels. Proper server sizing applies to all of the cluster failover examples shown above.

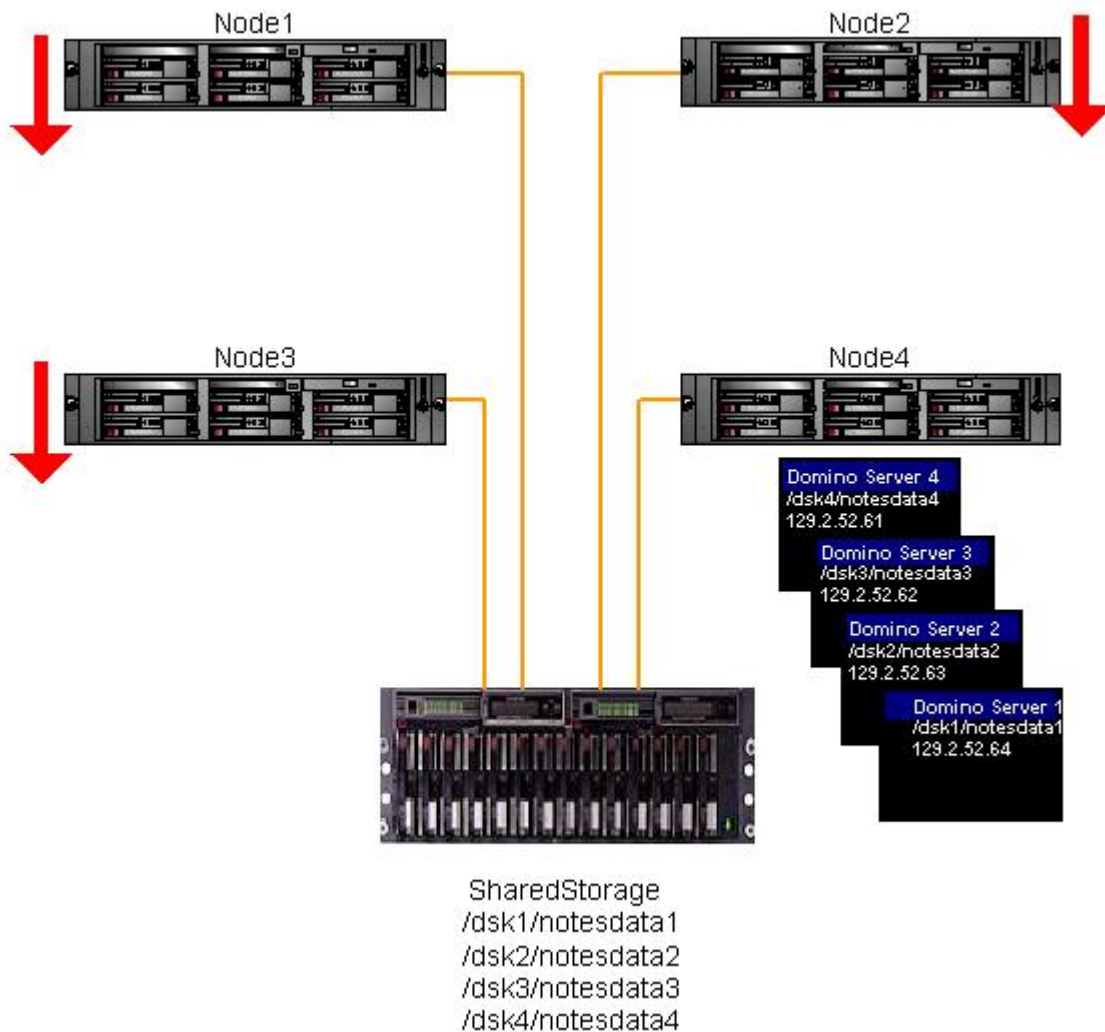


Figure 21. Failure of Three Server Nodes

Conclusion

Deploying Lotus Domino in a SteelEye LifeKeeper cluster can significantly enhance the availability of Domino server. Installing Domino into a LifeKeeper cluster is a straight-forward and simple procedure that leverages Domino Partitioned Servers. The LifeKeeper Generic Application resource allows you to create custom scripts that can be enhanced to control and monitor the Domino server partition. The LifeKeeper GUI provides a rich management interface to the cluster, allowing you to monitor cluster status, configure the cluster, and control resource availability. LifeKeeper allows system engineers and administrators to increase server availability for many of today's mission critical applications.

About Steeleye Technology, Inc.

SteelEye is the leading provider of enterprise IT reliability solutions for data protection, business continuity and disaster recovery on Linux and Windows 2000. The SteelEye LifeKeeper family of high availability clustering, data protection and disaster recovery software products are easy to deploy and operate, and enable enterprises of all sizes to ensure continuous availability of business-critical applications, servers and data.

For more information, go to <http://www.steeleye.com/>.