Understanding IBM Lotus Domino server clustering

Reetu Sharma
Software Engineer,
IBM Software Group
Pune, India

Ranjit Rai
Software Engineer
IBM Software Group
Pune, India

August 2009

Summary: This white paper explains how to set up, manage, and monitor an IBM® Lotus® Domino® server cluster. Learn what clustering is, how it works, its benefits and requirements, and how to configure the Internet Cluster Manager (ICM). Also included is information about error messages and other problems that may arise when using a cluster.

Contents

1 Introduction

Clustering is one of the most important features provided with the IBM® Lotus® Domino® Server. In every organization where Domino servers are used, it is essential that users have continuous access to their databases. If thousands of users are accessing a server and the server goes down, users' productivity may be critically hampered, in turn negatively impacting the business.

This white paper intends to help you understand clustering and how you can avail the benefits provided by the cluster, such as high availability of important databases, workload balancing, scalability, and data synchronization. We explain how to set up and manage the cluster and tips about error messages and other issues you may encounter when using a cluster.

Note that the material in this paper is largely excerpted from the [IBM Lotus Domino 8.5 Administrator information center](#), which you can always refer to for further information.

2 What is a Lotus Domino cluster?

A Domino cluster is a group of two or more servers that provides users with constant access to data, balances the workload between servers, improves server performance, and maintains performance when you increase the size of your enterprise.

The servers in a cluster contain replicas of databases that you want to be readily available to users at all times. If a user tries to access a database on a cluster server that is not available, Lotus Domino opens a replica of that database on a different cluster server, if a replica is available. Lotus Domino continuously synchronizes databases so that whichever replica a user opens, the information is always the same.

Lotus Notes clients can access all Domino cluster servers, while HTTP clients (Internet browsers) can access only Domino Web servers in a Domino cluster.

2.1 Benefits of a Domino cluster

Let's discuss more about the main benefits of Domino clusters:

High availability of important databases. When a hardware or software problem occurs, clustered servers redirect database open requests to other servers in the cluster, to provide users with uninterrupted access to important databases. This process is called failover.

Clusters provide failover for business-critical databases and servers, including passthru server failover to other servers in the cluster. Failover also lets you perform server maintenance, such as hardware and software upgrades, with little negative effect on users.

Workload balancing. When users try to access databases on heavily used servers, Lotus Domino can redirect the user requests to other cluster servers that aren't as busy, so that the workload is evenly distributed across the cluster.

Workload balancing of cluster servers helps your system achieve optimum performance, leading to faster data access.

Data synchronization. A key to effective clustering is setting up replicas on two or more cluster servers so that users have access to data when a server is down or is being used heavily. Cluster replication ensures that all changes, whether to databases or to the cluster membership itself, are immediately passed to other databases or servers in the cluster. Thus, databases are continuously synchronized to provide high availability of information.

Ease of changing operating systems, hardware, or Lotus Domino versions. When you want to change your hardware, operating system, or Domino release, you can mark the clustered server as RESTRICTED so that requests to access a database on the server fail over to other cluster servers that contain replicas. This lets you make changes without interrupting the productivity of your users.

2.2 Server requirements

When planning to set up a cluster, it is important to consider the performance and ability of the hardware. The cluster must have enough CPU power, memory, and disk space to handle the cluster traffic and the number of databases and replicas required.

Adding servers to the cluster increases the cluster's ability to balance the workload so that no server becomes overloaded and performance stays high. However, if servers use too many server tasks, CPU-intensive applications, and replicas, then adding servers can decrease performance because of the additional amount of cluster traffic required to keep databases synchronized on all servers.

The server requirements are as follows:

- All servers in a cluster must run on the Lotus Domino Enterprise server or the Domino Utility server.

- All servers in a cluster must be connected via a high-speed local area network (LAN) or a high-speed wide area network (WAN). You can also set up a private LAN for cluster traffic.

- All servers in a cluster must use TCP/IP and be on the same IBM Lotus Notes named network.

- All servers in a cluster must be in the same Domino domain and share a common Domino Directory.

- You must specify an administration server for the Domino Directory in the domain that contains the cluster. If you do not specify an administration server, the Administration Process (AdminP) cannot change cluster membership. The administration server does not have to be a member of a cluster.

- Each server in the cluster must have a hierarchical server ID. If any servers have flat IDs, you must convert them to hierarchical IDs to use them in a cluster.

- A server can be a member of only one cluster at a time.

- Each server must have adequate disk space to function as a cluster member. Because clusters usually require more database replicas, servers in clusters require more disk space than unclustered servers.

- Each server must have adequate processing power and memory capacity. In general, clustered servers require more computer power than unclustered servers.

3 Setting up a cluster

In this section we discuss how to create a cluster and how to verify that it is created correctly. You can choose to add the servers to the cluster immediately, or to use AdminP to add the servers to the cluster.

3.1 Using AdminP

If you use AdminP to add the servers to the cluster, Lotus Domino submits a request to the Administration Requests database to add the name of the cluster to the Server document of each server in the cluster.

If you use the administration server to create the cluster, AdminP runs immediately and adds the cluster name to the Server documents.

3.2 Adding clusters immediately

If you choose to add the servers to the cluster immediately, Lotus Domino immediately makes the changes to the Server documents in the Domino Directory on the server you use to create the cluster.

If that server is part of the cluster, the server immediately starts the cluster processes listed below and replicates its Domino Directory with another server in the cluster, so that the other cluster servers are informed that they are part of the cluster.

If you did not use a cluster member to create the cluster, this process starts when the Domino Directory of the server you used replicates with the Domino Directory of a server in the cluster.

3.3 Creating a cluster

To create a cluster, use these steps (note that you cannot use the Web Administrator to create a cluster):

1. From the Domino Administrator, make sure the administration server or another server is current.

2. Click the Configuration tab and in the Tasks pane, expand Server, and select All Server Documents (see figure 1).

Figure 1. All Server Documents
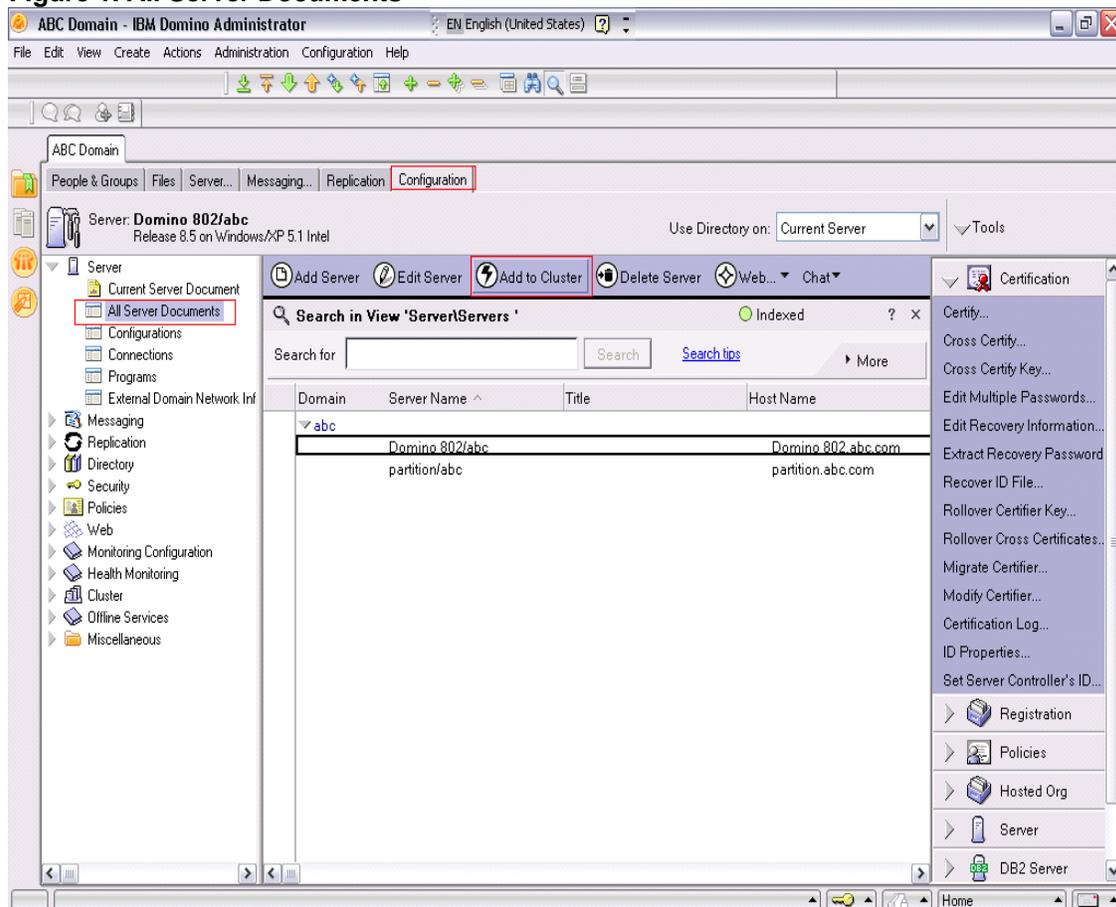
3. In the Results pane, select the servers that you want to add to the cluster.

4. Click Add to Cluster; a verification dialog will display (see figure 2).

Figure 2. Add to Cluster verification
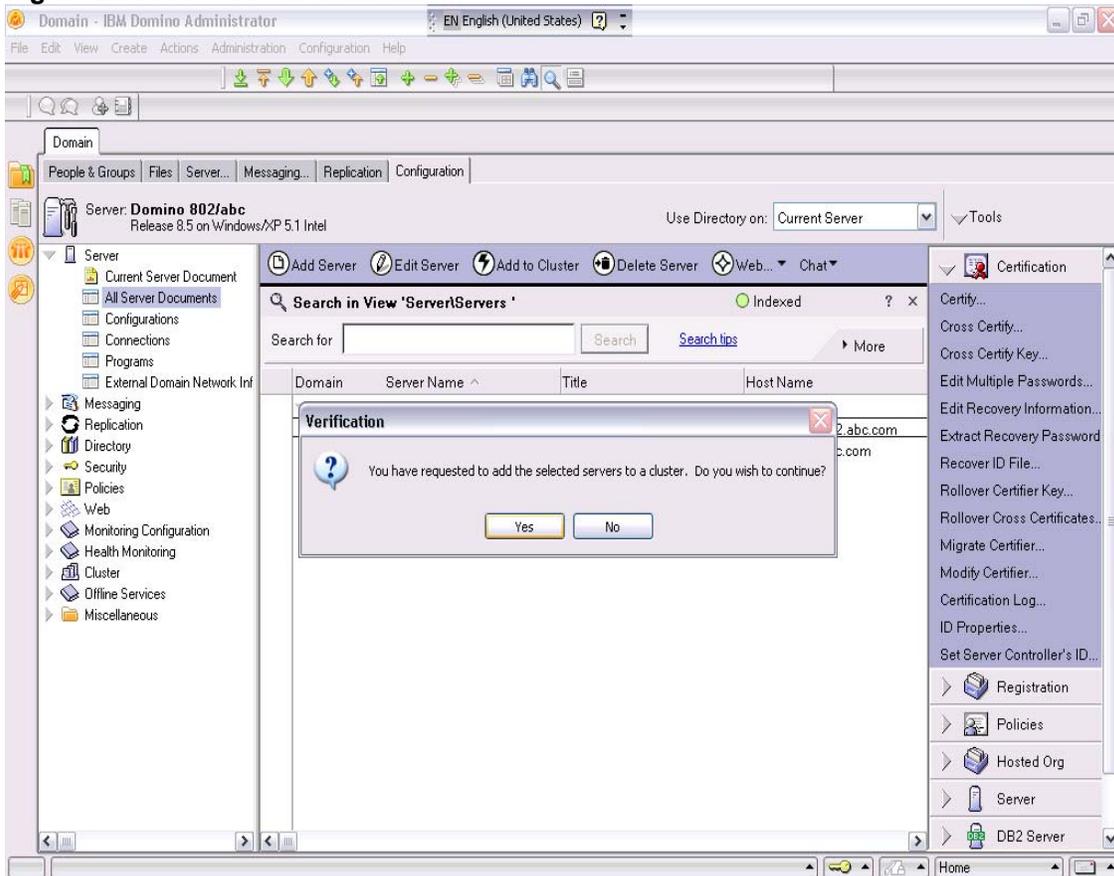
5. In the Cluster Name dialog box, choose Create New Cluster, and click OK.

6. Type the name of the new cluster, and then click OK (see figure 3).

Figure 3. New Cluster Name dialog
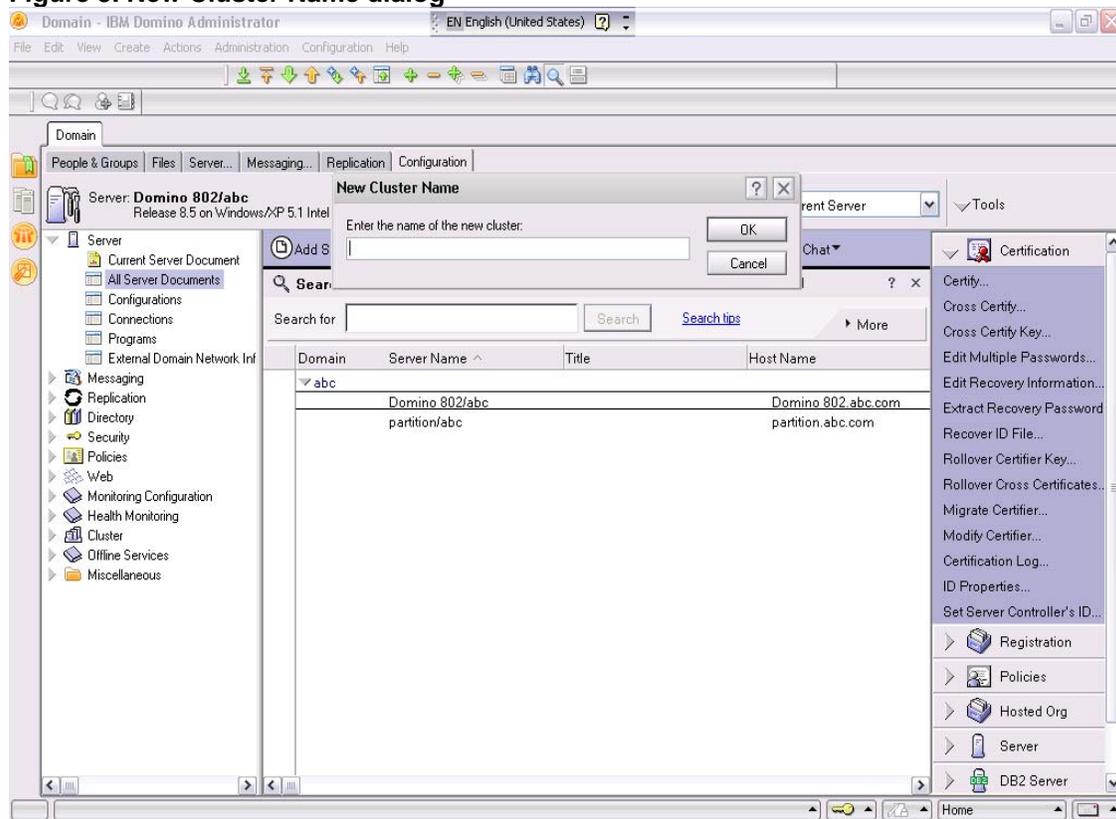
7. Choose Yes to add the servers to the cluster immediately, or choose No to submit a request to the Administration Process to add the servers to the cluster (see figures 4–6).

Figure 4. Immediate or via Administration Process option
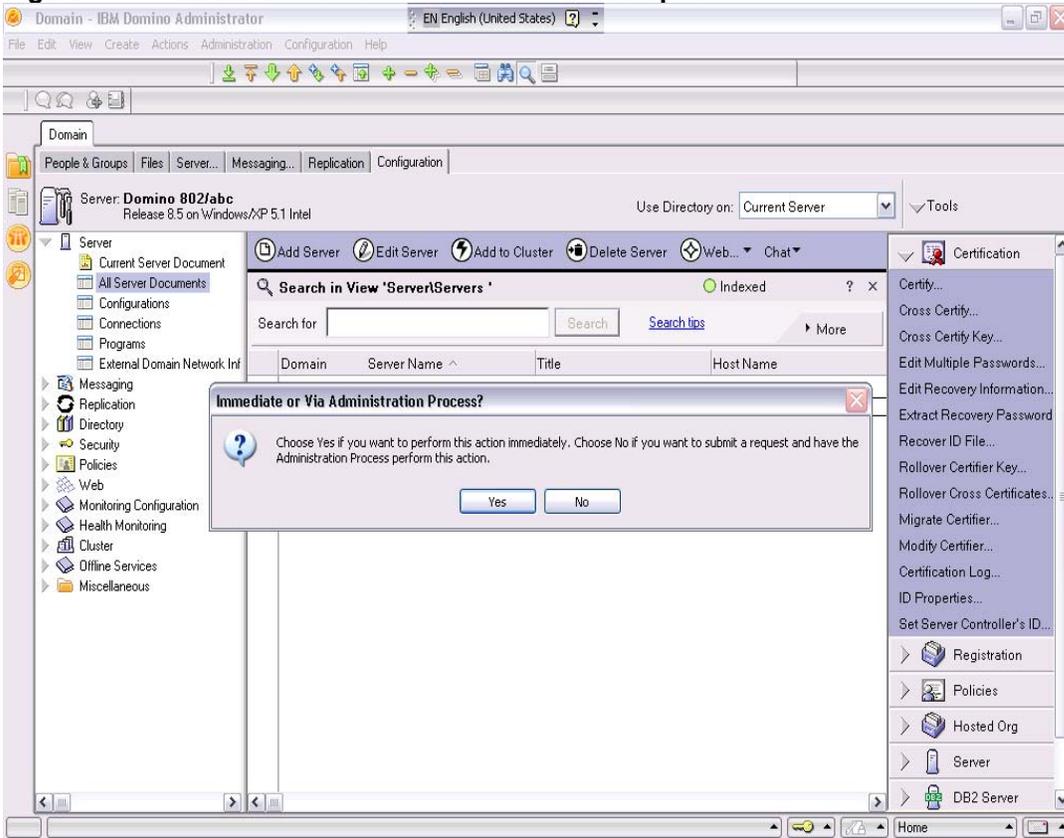
Figure 5. Choosing Yes to add the servers to the cluster immediately
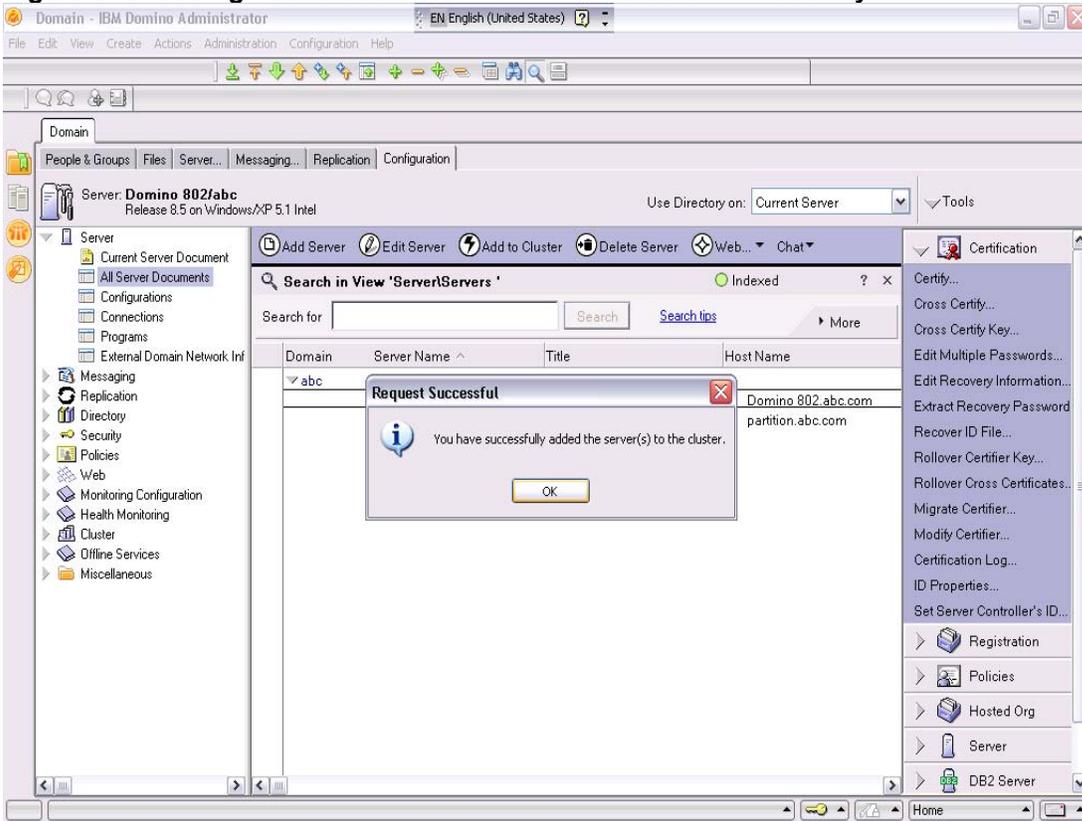
Figure 6. Choosing No to submit a request to the Administration Process
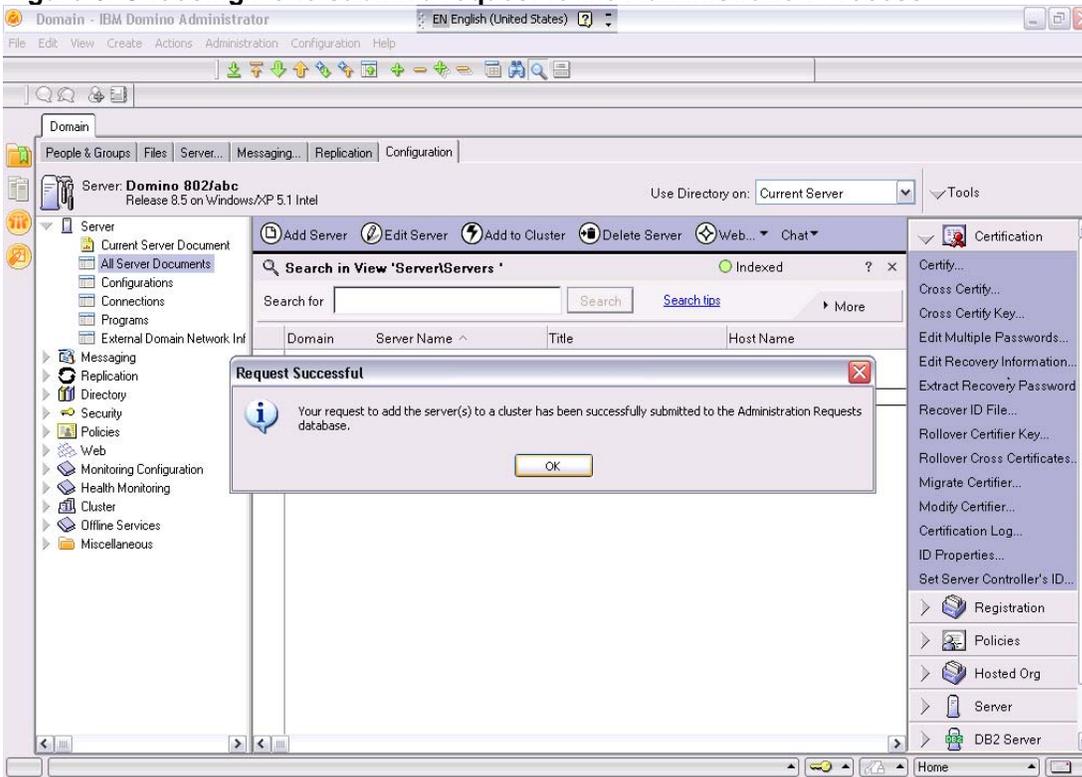
8. (Optional) If you chose No in Step 8 and you did not add the servers on the administration server, force replication between the server you used and the administration server so that the administration server receives the requested changes sooner.

9. (Optional) If you chose No in Step 8, force replication between the administration server and the cluster servers so the cluster servers receive all the changes sooner.

10. (Optional) If you chose Yes in Step 8, the cluster information is added immediately to the Domino Directory on the server you used to create the cluster. If this server is not part of the new cluster, replicate the changes to one of the servers you added to the cluster.

**NOTE:** If possible, use the administration server when creating a cluster, to make the creation process faster. The administration server does not need to be part of the cluster. **If a server belongs to a different cluster, you do not need to remove the server from that cluster before you add it to the new cluster. The Cluster Administration Process removes the server from the original cluster and then adds it to the new cluster.**

4 Setting up a private LAN for a cluster

Setting up a private LAN for your cluster separates the server probes and cluster replication from the rest of your network traffic. This prevents the cluster traffic from slowing down your primary network.

Follow these steps to set up a private network for your cluster:

1. Install an additional network interface card in each server in the cluster. These cards should be connected through a private hub or switch.

2. Configure the LAN to use TCP/IP.

3. Assign a second IP address to each server. In a cluster using two servers, for example, you could use the addresses 192.168.64.1 and 192.168.64.2.

4. Assign host names for the new IP addresses, and place entries in the HOSTS file or the DNS. In a cluster using two servers, for example, you could assign the host names Acme_clu.acme.com and Acme2_clu.acme.com. The entries in the HOSTS file might then be:

   Acme_clu.acme.com
   Acme2_clu.acme.com

5. Verify that the private LAN is operational by making sure that each cluster server can ping both the IP address and the host name of the other cluster servers. Also, be sure that other servers can still ping the cluster servers at their regular LAN addresses.

6. On each server in the cluster, add a new network port for the private LAN, and edit the Server document to enable the port. For example, add a port named CLUSTER, and then add the information in table 1 to the Ports - Notes Network Ports tab in the Server document to enable the port.

Table 1. Notes Network Ports tab info

Field name	Example
Port	CLUSTER
Protocol	TCP
Notes Network	Cluster Network
Net Address	Acme_clu.acme.com
Enabled	ENABLED

7. Assign each port an IP address from the corresponding subnets, and place this information in the Notes.ini file in the following form:

    *PORT1*_TcpIPAddress=0,*a.b.c.d*:1352
    *PORT2*_TcpIPAddress=0,*e.f.g.h*:1352

    where PORT1 and PORT2 are the port names, and *a.b.c.d* and *e.f.g.h* are the IP addresses for the ports.

    If you have ports named TCPIP and CLUSTER, for example, these lines might be:

    TCPIP_TcpIPAddress=0,192.114.32.5:1352
    CLUSTER_TcpIPAddress=0,192.168.64.1:1352

8. Now do one of the following:

    • Reorder the ports so that the cluster port is first. This ensures that all cluster traffic uses this port for cluster traffic. Be sure that all other traffic is assigned to use other ports.
    • Add the following line to the NOTES.INI file:

        Server_Cluster_Default_Port=*Cluster Port*

    where *Cluster Port* is the port you created for the cluster. In this example, this line would be:

        Server_Cluster_Default_Port=CLUSTER

    This ensures that all cluster traffic uses this port for cluster communications no matter what order the ports are in.

    NOTE: There is a disadvantage to using the Server_Cluster_Default_Port setting to assign a port to the private LAN for cluster traffic. **If a cluster server encounters a problem connecting over this port, it will not try another port.** Therefore, the

server will not be able to communicate or replicate with other cluster servers. You will need to resolve the network problem or remove this setting from the Notes.ini file before the server will be able to communicate with the cluster again.

9. Finally, restart the server.

4.1 Ensuring Lotus Domino is using the private LAN

To verify that Lotus Domino is using the private LAN, you can do the following:

1. At the server console, type `show cluster`.

2. In the results, look for the line "Server cluster default port: *Cluster Port*", where *Cluster Port* is one of the following:

   - If you did not use the Server_Cluster_Default_Port setting to specify a port for the private LAN, *Cluster Port* should be * (asterisk), meaning that the private LAN can use any port (see figure 7).

Figure 7. Server cluster default port: *


```
>
> sh cluster
[15F0:0007-1230]  Cluster information:
[15F0:0007-1230]  Cluster name: Cluster, Server name: Domino 802/abc
[15F0:0007-1230]  Server cluster probe timeout: 1 minute(s)
[15F0:0007-1230]  Server cluster probe count: 1120
[15F0:0007-1230]  Server cluster default port: *
[15F0:0007-1230]  Server cluster auxiliary ports:
[15F0:0007-1230]  Server availability threshold: 0
[15F0:0007-1230]  Server availability index: 100 (state: AVAILABLE)
[15F0:0007-1230]  Server availability default minimum transaction time: 3000
[15F0:0007-1230]  Cluster members (2):
[15F0:0007-1230]          Server: Domino 802/abc, availability index: 100
[15F0:0007-1230]          Server: partition/abc, availability index: 100
```

   - If you used the Server_Cluster_Default_Port setting to specify a port for the private LAN, *Cluster Port* should be the name of the port you specified, such as CLUSTER (see figure 8).

Figure 8. Server cluster default port: cluster


```
> sh cluster
[14FC:0007-1594]  Cluster information:
[14FC:0007-1594]  Cluster name: Cluster, Server name: Domino 802/abc
[14FC:0007-1594]  Server cluster probe timeout: 1 minute(s)
[14FC:0007-1594]  Server cluster probe count: 0
[14FC:0007-1594]  Server cluster default port: cluster
[14FC:0007-1594]  Server cluster auxiliary ports:
[14FC:0007-1594]  Server availability threshold: 0
[14FC:0007-1594]  Server availability index: 100 (state: AVAILABLE)
[14FC:0007-1594]  Server availability default minimum transaction time: 3000
```

4.2 Ensuring that cluster replication is using the private LAN

It is also a good idea to verify that cluster replication is using the private LAN. To do this, you can look at some of the cluster statistics:

1. From the Domino Administrator or the Web Administrator, click the Server -

Statistics tab.

2. In the list of statistics, expand NET.

3. Expand the port name you gave to the cluster.

4. Look for the following statistics for the private LAN port:

   BytesReceived
   BytesSent

5. Expand Replica, and then expand Cluster.

6. Expand SessionBytes, and look for the following replication statistics for the cluster:

   In
   Out

7. Compare the NET.*portname*.BytesReceived value with the Replica.Cluster.SessionBytes.In value. These values should be fairly close, although they will not be the same.

9. Compare the NET.*portname*.BytesSent value with the Replica.Cluster.SessionBytes.Out value. These values should also be similar, though they won't match exactly because the private network is used for more than just cluster replication.

4.3 Verifying the cluster was created correctly

Use the actions listed in table 2 to verify that the cluster was created correctly.

Table 2. Verification points

Action	What you should see
From the Lotus Domino Administrator or the Web Administrator, expand Clusters in the Server pane.	The name of the cluster followed by the names of the cluster servers.
From the Domino Administrator or the Web Administrator, click the Configuration tab. In the Task pane, expand Cluster, and then click Clusters.	The name of the cluster followed by the names of the cluster servers displayed in the Results pane.
In the Results pane, open the Server documents of the servers you added to the cluster.	The name of the cluster in the Cluster name field on the Basics tab.
From the Domino Administrator or the Web Administrator, click a cluster server in the Server pane, and then click the Files tab.	The title "Cluster Directory" in the Tasks pane. This shows that Lotus Domino created the Cluster Database Directory.
Compare the replica IDs of the Cluster Database Directories on each cluster server.	The same replica ID on each server.
From the server console, send the following command: show cluster	The name of the cluster, some statistics for the current server, and the names of all the cluster servers.

You can also use Cluster Analysis to generate reports that show if there are any configuration problems in the cluster. For more details, refer to the topics "Using Cluster Analysis to check the cluster configuration" and "Running Cluster Analysis" from the Domino 8.5 Administration Help database.

5 Cluster replication

Let's now examine the two types of cluster replication, namely, scheduled and streaming.

5.1 Scheduled replication

Scheduled replication is used in a cluster to replicate:

- **Changes that may be lost when a server in a cluster fails.**
  Because cluster replication events are stored in memory only, these events can be lost when a server fails before replicating the events. No data is lost, but the cluster replication events that were pending won't be replicated until you run standard Lotus Domino replication.

- **Databases for which you have disabled cluster replication.**
  You may have databases that you want to replicate but not every time they are updated. You can disable cluster replication for these specific databases.

  TIP: To see whether cluster replication is disabled for a database, view information in the Cluster Database Directory. Databases with the letter "X" in the left column have cluster replication disabled. You can also check this by looking in the Cluster Replication field in the document for each database in the Cluster Database Directory.

- **Based on selective replication formulas.**
  The Cluster Replicator leaves the processing of replication formulas to the standard replicator. Before using replication formulas in a cluster, you should be aware of how replication works in a cluster.

- **Replicas that are on the same server.**
  The Cluster Replicator pushes changes to other servers that contain replicas but does not update other replicas on its own server.

  NOTE: If there are multiple replicas on a server, the Cluster Manager uses failover by path to select the replica for a user to open during failover. If you put multiple replicas on a server, be sure that all replicas in the cluster that have the same path use the same selective replication formulas. Otherwise, the replica to which users fail over may contain different data than they expect.

You should run standard replication on a regular basis. In most cases, once or twice per day is sufficient, but the number of times per day you run standard replication depends on how important it is for you to keep all replicas synchronized. If it is absolutely critical to keep data synchronized at all times, you may want to replicate every hour or two.

In addition, you should replicate whenever you start the server, to be sure that all databases are up to date. You can create a Program document in the Domino Directory to accomplish this.

5.1 Streaming cluster replication

Streaming Cluster Replication (SCR) is a new feature in Lotus Domino 8. It takes advantage of the fully connected nature of clusters and data streaming to produce a low-overhead cluster replicator. SCR decreases the scheduled replicator overhead for clusters and provides a significant reduction in cluster replicator latency.

As changes occur, such as note changes, unread changes, and folder changes, they are captured and immediately queued to other replicas in the same cluster. The result is a more efficient cluster replication.

To use SCR, you must have at least two Domino 8 servers in a cluster. If your cluster contains any combination of Domino 8 servers and pre-Domino 8 servers, regular cluster replication is used to replicate between the version 8.0 and pre-version-8.0 servers.

Domino enables SCR when the following sequence of events occurs:

1. A database is opened.
2. Cluster replication replicates a change to another Domino 8 clustered replica.
3. All references to the database are closed.
4. On future changes in the database that needs to replicate, SCR replicates the change to the other Domino 8 servers.

If SCR encounters any errors during replication, it returns control of the database to the standard cluster replicator to replicate the change and bring the database in sync.

When a change occurs to a clustered server's database, the specific change is "captured" and is immediately queued for replication to the source server's Domino 8 cluster-mates that have a replica of the database. When the destination cluster-mates receive the change, they apply it to their replicas.

The performance benefits are as follows:

- SCR does not perform operations, such as database opens and note opens, across the network; instead, database changes are sent to the replicas with the assumption that the database replicas need the change.

- SCR does not need to replicate one-to-one with each cluster-mate for a database that has changed.

- The latency time for replication is immediate.

6 Failover in a Domino clustered environment

All the servers in a Domino cluster continually communicate with each other, to keep updated on the status of each server and to keep database replicas synchronized. A cluster's ability to redirect requests from one server to another is called failover.

When a user tries to access a database on a server that is unavailable or in heavy use, Lotus Domino directs the user to a replica of the database on another server in the cluster. To take advantage of failover and workload balancing, the databases and replicas should be distributed throughout the cluster.

The number of replicas you create for a database depends on how busy the database is and how important it is for users to have constant access to that database. For some databases, you may not need to create any replicas; for others, you may need to create multiple replicas.

When a user tries to open a database that is not available, the Cluster Manager looks in the Cluster Database Directory for a replica of that database. To find the replica, the Cluster Manager looks for a database that has the same replica ID as the original database. It also makes sure that the replica is not marked as Out of Service or Pending Delete.

If there is more than one replica of a database on a server, the Cluster Manager assumes that selective replication is used to replicate these databases. To be sure it fails over to the correct replica, the Cluster Manager selects a replica that has the same path as the original database.

Therefore, if you put multiple replicas on a server, be sure that all replicas in the cluster that use the same selective replication formulas have the same path. Otherwise, users may fail over to a different replica.

6.1 Failover example scenario

This example describes the process that Domino uses when it fails over. This cluster contains three servers. Server 1 is currently unavailable, and the Cluster Managers on Server 2 and Server 3 are aware that Server 1 is unavailable (see figure 9).

Figure 9. Three-server cluster example
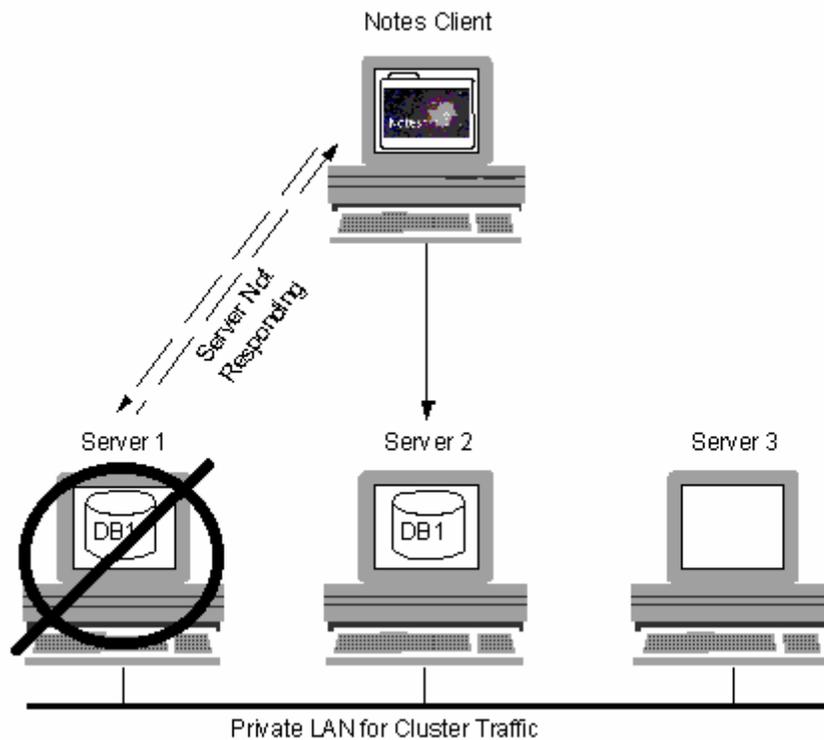
So let's look at the process in this scenario:

1. A Lotus Notes user attempts to open a database on Server 1.

2. Lotus Notes realizes that Server 1 is not responding.

3. Instead of displaying a message stating the server is not responding, Lotus Notes looks in its cluster cache to see if this server is a member of a cluster and to find the names of the other servers in the cluster.

   (When a Notes client first accesses a server in a cluster, the names of all the servers in the cluster are added to the cluster cache on the client. This cache is updated every 15 minutes.)

4. Notes accesses the Cluster Manager on the next server listed in the cluster cache.

5. The Cluster Manager looks in the Cluster Database Directory to find which servers in the cluster contain a replica of the desired database.

6. The Cluster Manager looks in its server cluster cache to determine the availability of each server that contains a replica. (The server cluster cache contains information about all the servers in the cluster. Cluster servers obtain this information when they send probes to the other cluster servers.)

7. The Cluster Manager creates a list of the servers in the cluster that contain a replica of the database, sorts the list in order of availability, and sends the list to Lotus Notes.

8. Lotus Notes opens the replica on the first server in the list (the most available server). If that server is no longer available, it opens the replica on the next server in the list. In this example, Server 2 is the most available server.

When the Notes client shuts down, it stores the contents of the cluster cache in the file CLUSTER.NCF. Each time the client starts, it populates the cluster cache from the information in CLUSTER.NCF.

7 Workload balancing in a Domino cluster

Clustering in Lotus Domino also provides active load balancing for optimal use of the resource. We can achieve workload balancing among Domino servers by:

- Limiting the workload on a server (by changing the server availability threshold)
- Changing the setting for the maximum number of users who can use a server at the same time
- Moving (distributing) databases to other servers
- Planning the number and placement of replicas in a clusters to create more replicas of busy databases
- Adding one or more servers to the cluster
- Using the Activity Trends tool to analyze and make recommendations about the conditions in a cluster

Keep in mind that workload balancing is not the solution for a general lack of capacity in your enterprise. If your Domino servers are struggling with their current workload, and there are no additional servers to handle the excess load, enabling workload balancing will not solve the problem.

To balance the workload, there must be somewhere to send the overflow from the overworked servers. If there is nowhere to send these requests, they will be handled by the original busy servers.

Note, however, that the process of looking for another available server for each request increases the workload on the server. If you do not have enough capacity in your enterprise, consider adding more memory or processors, or otherwise upgrading your equipment to handle a larger load.

If the workload in your cluster is normally well distributed, consider configuring the cluster for failover only, not for workload balancing.

7.1 Server availability index

The server availability index is an important setting for managing the workload balance; hence, to achieve efficient load balancing we need to understand what it is and how it is calculated.

Each server in a cluster periodically determines its own workload based on the response time of the requests the server has processed recently. The workload is expressed as a number from zero to 100, where zero indicates a heavily loaded server and 100 indicates a lightly loaded server. This number is called the server availability index. As response times increase, the server availability index decreases.

The server availability index is approximately equal to the percentage of the total server capacity that is still available. For example, if the server availability index is 65, you still have approximately 65% of the capacity of your server available.

Although the servers in your enterprise may vary in power and resources, the server availability index represents the same thing on each server—the amount of total availability of that server that is still available.

7.2 Expansion factor

The server availability index is based on the expansion factor, which indicates the current workload on a server. The expansion factor is determined by comparing recent response times for specific types of transactions to the minimum time in which the server has ever completed the same types of transactions.

For example, if the server is currently averaging 12 microseconds to perform Database Open transactions, but the minimum time the server has ever performed a Database Open transaction is 3 microseconds, the expansion factor for Database Open transactions would be 4 (the current time of 12 microseconds divided by the fastest time of 3 microseconds).

In other words, the expansion factor determines how many times longer it takes for a transaction to complete currently than it takes under optimal conditions.

Lotus Domino stores the minimum time for each type of transaction in memory and in the LOADMON.NCF file, which the server reads each time it starts. When the server shuts down, Lotus Domino updates the LOADMON.NCF file with the latest information.

To determine the current expansion factor, Lotus Domino tracks the most-commonly-used types of Domino transactions for specified periods of time, by default tracking these transactions for 5 periods of 15 seconds each.

Lotus Domino then determines the average time it took to complete each type of transaction and divides that time by the minimum time it ever took to complete that same type of transaction. This determines an expansion factor for each type of transaction.

To determine the expansion factor for the entire server, Lotus Domino averages the expansion factors for all the types of transactions, assigning a heavier weighting to the most-frequently-used types of transactions.

As the server gets busier, adding more load has an increasingly greater effect on performance and availability. In other words, adding more load to a busy server increases the expansion factor faster than adding more load to a less busy server.

Because servers differ in speed, capacity, and power, servers also differ in the workload they can handle. Therefore, the same expansion factor on two different servers does not necessarily indicate the same workload relative to the ability of the servers.

For example, on a small server that takes a long time to perform transactions when the server is otherwise idle, an expansion factor of 40 might indicate that users are waiting many seconds for responses. On a very large, fast server, however, an expansion factor of 400 might indicate that users are waiting less than a second for responses.

7.2.1 Comparing the availability index to the expansion factor

To determine the server availability index, Lotus Domino uses a formula that converts the expansion factor into an approximation of the percentage of the total server capacity that is still available.

Table 3 shows a few examples of expansion factors converted to availability indexes (note that these values are based on the default expansion factor of 64 indicating a fully loaded server).

Table 3. Expansion factors converted to availability indexes

Expansion factor	Availability index
1	100
2	83
4	67
8	50
16	33
32	17
64	0

NOTE: The expansion factor and the availability index measure only the response time of the server, which is usually only a small portion of the response time that clients experience. For example, the network response time between a client and a server often accounts for a significant portion of the response time the client experiences.

7.2.2 Changing the value of the expansion factor that indicates a fully loaded server

To use Domino workload balancing effectively, you must adjust the relationship between the expansion factor and the availability index so that servers fail over when they reach the workload at which you want them to fail over. You do this by specifying the expansion factor value that you want to represent a fully loaded server.

The default value in Lotus Domino is 64. When the expansion factor reaches that value, the server is considered to be fully loaded, and the availability index drops to 0 (zero).

TIP: You can use the `Show AI` server command to view a suggested availability index setting.

If your server is particularly powerful and fast, you might want to increase the value of the expansion factor that is considered fully loaded. On some very fast servers, you might want to make this value several hundred or higher. If your server is particularly slow, you might want to decrease this value.

To change the expansion factor value that indicates a fully loaded server, follow these steps:

1. Add the following setting to your Notes.ini file, and then restart the server.

   SERVER_TRANSINFO_RANGE=n

2. For the value of n, choose a number such that 2 raised to the power of n equals the expansion factor value that you want to indicate a fully loaded server. The default value for n is 6, which leads to an expansion factor value of 64 (because 2 raised to the 6th power is 64).

   If you set SERVER_TRANSINFO_RANGE to 7, then the expansion factor value that indicates a fully loaded server becomes 128. If you set the parameter to 8, the value becomes 256.

3. To determine the optimal value for SERVER_TRANSINFO_RANGE, do the following:

   a. During a period of heavy usage, monitor the expansion factor on your server, using the console command `show stat server.expansionfactor`. You can also monitor performance statistics during these periods. Record enough values for the expansion factor during heavy use so that you can determine the expansion factor value you want to indicate a fully loaded server.

   b. Determine a value for SERVER_TRANSINFO_RANGE such that 2 raised to the power of that value results in the expansion factor value you chose in step 1. When you change the expansion factor value that indicates a fully loaded server, the relationship between the expansion factor and the availability index changes.

   Table 4 shows a few examples of expansion factors converted to availability indexes when the value of SERVER_TRANSINFO_RANGE is 8. The maximum expansion factor in this example is 256 because 2 raised to the power of 8 is 256.

Table 4. Expansion factors converted to availability indexes

Expansion factor	Availability index
1	100
2	88
4	75
8	63
16	50
32	38
64	25
128	13
256	0

c. Use the server console command `Show AI` to obtain an appropriate value for this variable. Type the command after running the server under load for a while, and it will display the history of the expansion factor and AI for the server.

7.2.3 Changing the amount of data used to compute the expansion factor

Although it is not usually necessary, you can use the Notes.ini settings below to change the amount of data that Lotus Domino collects in order to figure the expansion factor.

- To change the number of data collection periods that Domino uses, use the setting

    Server_Transinfo_Max=x

    where x is the number of collection periods you want Lotus Domino to use.

- To change the length of each data collection period, use the setting

    Server_Transinfo_Update_Interval=x
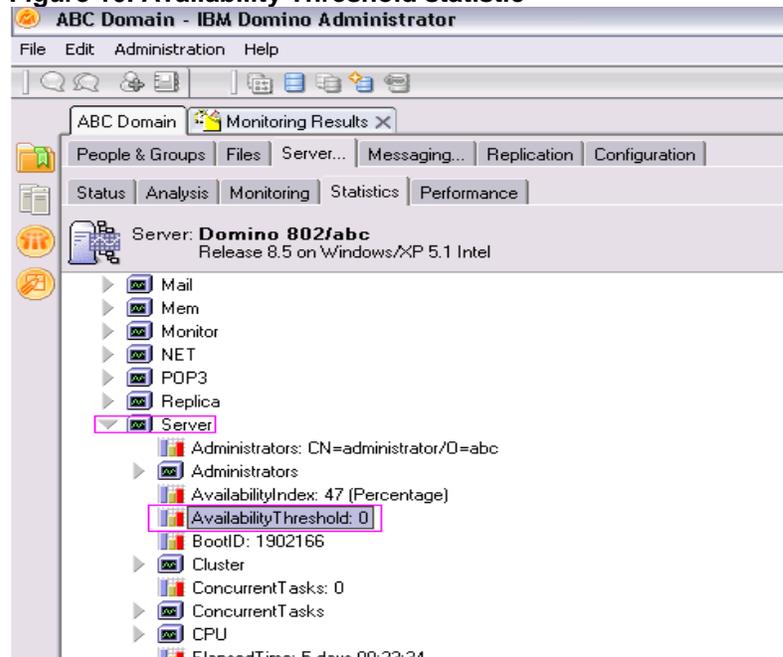
    where x is the length of each period in seconds.

7.3 Setting the server availability threshold

Setting the server availability threshold on each server is a key factor in balancing the workload in the cluster. Setting this value too high can result in failover occurring unnecessarily. Setting it too low can result in poor performance for users who could have received better performance from a different server.

To determine the proper value for the server availability threshold, do the following:

1. During periods of normal-to-heavy load, use one of the following bulleted methods to observe the server availability index:

    - From the Lotus Domino Administrator or the Web Administrator, make the server you want current. Then click the Server tab, then the Statistics tab, and in the statistics list, expand Server to see the Availability Threshold stat (see figure 10).

Figure 10. Availability Threshold statistic

- At the server console, type `show cluster`:

- At the server console, type `show stat server`:

2. Set an initial availability threshold based on the results of your observation. Consider the following when setting this value:

- The value should be near the lower end of the values you observed.
- Add some extra capacity (that is, decrease the availability threshold number) to accommodate servers that may fail over to this server.

NOTE: When a server fails, the workload fails over to other servers in the cluster. If there are only two servers in the cluster, each with the same workload, this would result in approximately a 100% increase in the workload of the remaining server. If

there are six servers in the cluster, this would result in approximately a 20% increase in the workload of the remaining servers. Therefore, set the availability threshold of each cluster server low enough to allow the server to absorb an adequate portion of the workload if another cluster server fails.

3. Track other cluster statistics to see if the workload is reasonably balanced. Table 5 lists some of the statistics that are helpful in determining this. (These statistics are cumulative from when the server was started.)

Table 5. Statistics relevant for workload balancing

Statistic name	Description
Server.AvailabilityIndex	The current value of the server availability index. The values range from 0 to 100. A value of 0 indicates that there are no resources available on the server. A value of 100 means that the server is completely available.
Server.ExpansionFactor	The current expansion factor. This value is used to compute the availability index. A value of 1 indicates that the server is completing transactions at the minimum time for that server. A value of 64 indicates that it is taking 64 times longer than the minimum time to complete transactions. By default, an expansion factor of 64 indicates a fully loaded server, and results in an availability index of 0 (zero). You can use the Notes.ini setting Server_Transinfo_Range to change the expansion factor value that indicates a fully loaded server. You can use the Show AI server command to obtain a suggested availability index.
Server.Cluster.OpenRedirects.LoadBalance.Successful	The number of times a BUSY server successfully redirects a client to another cluster member.
Server.Cluster.OpenRedirects.LoadBalance.Unsuccessful	The number of times a BUSY server is unsuccessful in redirecting a client to another cluster member. A server will be unsuccessful if the target server is also in a BUSY state or otherwise unavailable.
Server.Cluster.OpenRequest.ClusterBusy	The number of times a BUSY server tries to redirect a client request when all other cluster servers were also BUSY.
Server.Cluster.OpenRequest.LoadBalanced	The number of times a user tried to open a database on this server when the server was BUSY.

4. Compare these statistics for all the servers in the cluster, and adjust the server availability threshold on any servers that do not seem well balanced.

TIP: Workload balancing is not a substitute for having adequate computer resources for your enterprise. If your servers are already overloaded, workload balancing merely increases the problem because there is no place for a BUSY server to send client requests. Moreover, looking for an available server decreases the performance on an already busy server.

7.4 Setting the maximum number of users on a server

You can also balance the workload in a cluster by using the Notes.ini setting:

Server_MaxUsers

This setting specifies the maximum number of active users allowed on a server at one time. When the server reaches this limit, the server goes into the MAXUSERS state and rejects any additional requests until the number of active users falls below the Server_MaxUsers limit.

When Lotus Domino rejects an access request because of a MAXUSERS state, the Cluster Manager attempts to redirect the request to another cluster server that contains the appropriate replica. If no other server is available, Lotus Domino rejects the access request and displays an explanatory message.

Note that the Server_MaxUsers setting does not affect replication. Replication occurs even when a server is in a MAXUSERS state.

Further, you can monitor the statistics in table 6 to manage the cluster better. Other than AvailabilityIndex, AvailabilityThreshold, and ExpansionFactor, each statistic name includes the prefix Server.Cluster, which is not included in the names in this table.

AvailabilityIndex, AvailabilityThreshold, and ExpansionFactor include the prefix "Server."

Table 6. More statistics for cluster management

Statistic name	Description
Member	Shows the names and the availability indexes of the servers in the cluster
AvailabilityIndex	A measure of a server's availability. 0 (zero) indicates no available resources; 100 indicates complete server availability
AvailabilityThreshold	The current availability threshold of the server. When the availability index is less than or equal to the availability threshold, the server is put into the BUSY state.
ExpansionFactor	The current expansion factor. This value is used to compute the availability index. A value of 1 indicates that the server is completing transactions at the minimum time for that server. A value of 64 indicates that it is taking 64 times longer than the minimum time to complete transactions. By default, an expansion factor of 64 indicates a fully loaded server, and results in an availability index of 0 (zero).
	You can use the Notes.ini setting Server_Transinfo_Range to change the expansion factor value that indicates a fully loaded server.
OpenRedirects.Failover.Successful	Total times that server successfully redirects a client to another cluster member after the client fails to open a database by replica ID
OpenRedirects.Failover.Unsuccessful	Total times that server is unable to redirect a client to another cluster member after the client fails to open a database by replica ID
OpenRedirects.FailoverByPath.Successful	Total times server successfully redirects a client to another cluster member after the client fails to open a database by path name
OpenRedirects.FailoverByPath.Unsuccessful	Total times server is unable to redirect a client to another cluster member after the client fails to open a database by path name
OpenRedirects.LoadBalance.Successful	Total times server successfully redirects a client to another cluster member after the client tries to open a database by replica ID when the server is BUSY
OpenRedirects.LoadBalance.Unsuccessful	Total times server is unable to redirect a client to another cluster member after the client tries to open a database by replica ID when the server is BUSY
OpenRedirects.LoadBalanceByPath.Unsuccessful	Total times that server is unable to redirect a client to another cluster member after the client tries to open a database by path name when the server is BUSY

OpenRedirects.LoadBalanceByPath.Successful	Total times that server successfully redirects a client to another cluster member after the client tries to open a database by path name when the server is BUSY
OpenRequest.ClusterBusy	Total client requests when all servers are BUSY
OpenRequest.DatabaseOutOfService	Total times a client tries to open a database that is marked out-of-service on the server
OpenRequest.LoadBalanced	Total times a client tries to open a database on the server when the server is BUSY
PortName	Default port used for intra-cluster network traffic or an asterisk, which indicates there is no default port and any available active port can be used
ProbeCount	Total times that a server completes a probe of the other cluster members
ProbeError	Total times that a server receives an error when probing another server
ProbeTimeout(mins) *	Shows the interval at which an intracluster probe occurs

* This statistic does not appear in the Cluster statistics report, but you can use the Show Stat command to view it, or you can add it to the Cluster statistics report form.

Table 7 lists the cluster replicator statistics. Each statistic name includes the prefix Replica.Cluster, which is not included in the names in this table.

Table 7. Cluster Replicator statistics

Statistic name	Description
Servers	Total servers in the cluster that are receiving replications from this server
Successful	Total successful replications since the server started
Failed	Total failed replications since the server started
Docs.Added	Total documents added by the Cluster Replicator
Docs.Updated	Total documents updated by the Cluster Replicator
Docs.Deleted	Total documents deleted by the Cluster Replicator
Files.Local	Number of databases on the current server for which there are replicas on other servers in the cluster
Files.Remote	Number of databases on other servers to which the Cluster Replicator pushes changes
Retry.Skipped	Total times the Cluster Replicator did not attempt to replicate a database. The retry is skipped when the destination server is known to be unreachable or the database is waiting for another retry attempt.
Retry.Waiting	Total number of replicas that are waiting for retry attempts
SecondsOnQueue	Total time, in seconds, that the last database replicated spent on the work queue
SecondsOnQueue.Avg	Average time, in seconds, that a database spent on the work queue
SecondsOnQueue.Max	Maximum time, in seconds, that a database spent on the work queue
SessionBytes.In	Total bytes received during cluster replication
SessionBytes.Out	Total bytes sent during cluster replication
WorkQueueDepth	Current number of databases awaiting replication by the Cluster Replicator
WorkQueueDepth.Avg	Average work queue depth since the server started
WorkQueueDepth.Max	Maximum work queue depth since the server started

8 Clustering Domino servers that run Internet protocols

You can configure and use the Internet Cluster Manager (ICM) to extend the benefits of Domino clusters to HTTP clients and Domino Web servers. You can also employ an alternative method of using failover with HTTP servers, as well as methods to use failover with POP3, IMAP, and LDAP servers.

8.1 How the Internet Cluster Manager works

The ICM lets you use Domino clusters to provide failover and workload balancing to HTTP clients (Internet browsers) when they access Domino Web servers. This makes your Web servers and databases highly available to clients.

You can run the ICM on a Lotus Domino Enterprise server or a Lotus Domino Utility server, and you install and configure Domino clusters as you normally would, then configuring the ICM.

The ICM does the following:

- Supports the HTTP and HTTPS protocols, acting as an intermediary between HTTP clients and the Domino Web servers in a cluster. When Domino Web servers are running in a cluster, they generate URLs that direct HTTP client requests to the ICM.

- Maintains information about the availability of servers and databases in the cluster. When the ICM receives a client request, it redirects the client to the most available server that contains a replica of the requested database.

- Sends periodic probes to the Web servers in the cluster to determine their status and availability. When the ICM receives a client request, it looks at the information in the Cluster Database Directory to find a server that contains the requested database.

- Determines the most available server that contains the requested database and then redirects the client to that server, causing the client to close the session with the ICM and open a new session with the selected server.

  The user may see this as a change in the host name in the URL. The user may also see the path to the database change in the URL because the database may have a different path on the target server.

If the page that a Web server displays to a client includes links to other databases, the Web server includes the host name of the ICM in the URLs to those databases, when generating URLs to databases:

- on the same server as the original database
- on different servers, if there are replicas of those databases on the server that contains the original database

This ensures that users accessing those links go through the ICM.

8.2 Configuring the ICM

You configure the ICM by making entries in the Internet Cluster Manager section of the Server document. You can also set up a separate IP address for the ICM. You can then start the ICM.

You can configure the ICM settings on one server and have more than one ICM access these settings. This lets ICMs on different servers share a common configuration. You should include the ICM configuration information on every Web server in the cluster, not just the server on which you run the ICM.

This is done because each Web server uses its own Server document to determine how to generate URLs that refer to the ICM. The Web server obtains the host name of the ICM from the Server document and then uses that host name to generate URLs that reference the ICM.

To configure the ICM, follow these steps:

1. From the Domino Administrator or the Web Administrator, click the Configuration tab.

2. In the Tasks pane, expand Server, and click All Server Documents.

3. Do one of the following:

    - In the Results pane of the *Domino* Administrator, select the Server document for the server on which you want to run the ICM. Then click Edit Server.

    - In the Results pane of the *Web* Administrator, open the Server document for the server on which you want to run the ICM. Then click Edit Server.

4. Click the Server Tasks tab and then the Internet Cluster Manager tab (see figure 11).

Figure 11. Internet Cluster Manager tab

5. Complete the fields in this tab as shown in table 8 and save the Server document.

Table 8. Fields for Server Tasks > Internet Cluster Manager tab

Section	Field name	What it is
Basics	Cluster name	The name of the cluster the ICM will service. If this field is blank, which is the default, Domino uses the name of the cluster that contains this server.
	ICM Notes port	The name of the Domino port the ICM will use to communicate with HTTP clients. If this field is blank, which is the default, the ICM can use any Domino TCP/IP port to communicate with HTTP clients. Enter a port name only if you want to restrict ICM communication to one specific port.
	ICM SSL keyfile	The name of the SSL key file that contains certificates to identify the ICM when communicating with HTTP clients. If this field is blank, the ICM uses the key file specified on the Ports - Internet Ports tab of the Server document.
	Allow users to browse databases in the cluster over HTTP	Lets HTTP clients view a list of the databases in a cluster. When you enable this field, users can enter http://*icmhostname*/?OpenServer as the URL to access. Entering this URL displays a list of databases on the servers in the cluster associated with the ICM named in *icmhostname*.
Configuration	Get configuration from	Lets you specify a different Server document from which to get configuration information. This field lets multiple ICMs share the same configuration.
	Obtain ICM configuration from	This field appears when you select "another server document" in the field "Get configuration from." Enter the name of the server whose Server document contains the configuration you want to use.
	ICM hostname	The fully qualified name of the host that clients should use to communicate with the ICM. This can be the registered DNS name or the IP address. The Domino Web server uses this field to create URLs that reference the ICM. If this field is blank, the Web server will not be able to generate URLs that refer to the ICM.
ICM HTTP Port Settings	TCP/IP port number	Enter the port number for the ICM to use. If you are running the ICM on the same server as the Web server, you must avoid address and port conflicts. If you do not give the ICM its own IP address, be sure the port number the ICM uses is different from any of the other port numbers you use on the server.
	TCP/IP port status	To enable HTTP communication with the ICM, choose enabled. To disable HTTP communication with the ICM, choose Disabled.
	SSL port number	Enter the port number to use for SSL. If you are running the ICM on the same server as the Web server and you do not give the ICM its own IP address, be sure the SSL port number is different from any of the other port numbers you use on the server.
	SSL port status	To enable HTTPS communication with the ICM, choose enabled. To disable HTTPS communication with the ICM, choose Disabled.

When the ICM starts, it looks at the Server document on the server on which it is running to find the ICM cluster name and its network address. It then obtains the host name and port settings from the same Server document or from the Server document specified in the field "Obtain ICM configuration from."

If you run the ICM on the same system as a Domino Web server, you must avoid IP address or port number conflicts. The best approach is to assign the ICM its own IP address. You can also have the ICM share an IP address with the Web server, if you specify different port numbers for the ICM and the other protocols on the Web server.

8.3 Failover and workload balancing

When using the ICM, failover and workload balancing work the same as in standard Domino clusters. Lotus Domino computes the server availability index based on all open sessions, whether they are from Lotus Notes clients, HTTP clients, or other Domino services.

To limit the workload of a server and manage failover, you use the same settings, such as SERVER_RESTRICTED and SERVER_AVAILABILITY_THRESHOLD, as in a standard cluster. For database availability, you also use the same settings, such as marking a database "Out of service" or "Pending delete."

NOTE: Unlike in a standard Domino cluster, the ICM can direct a client to a server that is in the MAXUSERS state, if no other server is available. The Notes.ini setting Server_MaxUsers applies only to NRPC; it does not apply to HTTP.

The ICM maintains information about the following so that it can find a replica when a client asks for one:

- Which databases are available in the cluster and where they are stored. The ICM obtains this information from the Cluster Database Directory.
- The availability of each server. The ICM obtains this information each time it probes the servers in the cluster.
- Which Web servers are configured for HTTP and which are configured for HTTPS. The ICM obtains this information from the Server documents of each server in the cluster.

To determine which replica of a database to open, the ICM does the following:

- Determines where replicas reside and whether they are marked out of service or pending delete.
- Checks the server availability index of each server that contains a replica.
- Checks the availability of the server by pinging the HTTP port or the HTTPS port, depending on the client request.
- Eliminates any servers that are not reachable or are RESTRICTED.
- Eliminates any servers that are BUSY or in the MAXUSERS state.
- Selects a server from those remaining. If there are no servers remaining, the ICM chooses a server that is BUSY or in the MAXUSERS state, if one is available. If there are multiple servers remaining, the ICM chooses the server with the lightest current workload.

After choosing the server to access, the ICM looks at the Server document to determine which port to use to access the server.

8.4 When a server fails

When an HTTP client is connected to a server that fails, the client receives a message stating that the server is not responding. To fail over to a different replica, if one is available, the user must contact the ICM again, in one of the following ways:

- Click the Back button in the browser one or more times to connect to a page through the ICM
- Use a bookmark
- Type the URL

The user may or may not need to re-authenticate with the new server. This is determined by the following factors:

- If the user already authenticated with the new server during this session, no authentication is necessary.
- If the HTTP client and the server both support SSL3, re-authentication occurs automatically.

9 Failover scenarios

Failover occurs when a user cannot access a database in a cluster. This occurs either because they cannot access the database server or they cannot access the database itself.

Failover will occur when a user:

(1) Attempts to open a database on a server that is unavailable by:

- double-clicking a database icon on the workspace.

- selecting a database icon on the workspace and selecting File > Open.

- selecting a database icon on the workspace and selecting View - <database view> or Create - <form name>.

(2) Attempts to open a database that is unavailable due to:

- the database being marked Out of Service.

- the database being marked Pending Delete.

- trying to access their home server, but the server is unavailable.

- trying to open their mail file, but the server is unavailable.

- a mail router request to transfer and deposit mail to a user's mail file not completing because the server is unavailable.

(3) Attempts to access a Web Navigator server that is unavailable by:

- selecting the Open URL icon.

- double-clicking a URL hotspot (HTTP address).

- replicating a local database with a database in the cluster, and the server is unavailable.

- activating any field, action, or button that contains an @Command (FileOpenDatabase), and either the server or the database is unavailable. attempting to launch a doc link, view link, or database link that is connected to a database or server that is unavailable.

- running a LotusScript routine that includes the DB.OPENWITHFAILOVER call, which opens a database with failover if either the database or server are unavailable.

Failover will *not* occur when:

- a server becomes unavailable while a user has a database open. (Note that the user can open the database again, which causes failover to a different replica, if one exists in the cluster. If the user was editing a document when the server became unavailable, the user can copy the document to the replica.)

- a user selects File > Application > Properties or File > Open > Lotus Notes Application.

- the router attempts to deliver mail while MailClusterFailover is set to zero.

- the template server is unavailable while creating a new database.

- running agents, other than the mail pre-delivery agent.

- replicating with a server that is restricted by the administrator, or has reached the maximum number of users or the maximum usage level set by the administrator. Also, when replicating with a database marked Out of Service; replication occurs regardless of such restrictions, so there is no need for failover to occur.

10 Conclusion

You should now understand how to take advantage of the benefits provided by Domino server clustering, and how it can be set up and managed. We discussed how and when failover occurs, how to configure and manage work load balancing, scheduled vs. streaming replication, the server availability index, and ICM, which extends the benefits of Domino clusters to HTTP clients and Domino Web servers.

Included in the Resources section below are some documents with more information about clusters and common issues that may occur after setup.

11 Resources

- [IBM Lotus Domino 8.5 Administrator information center](#) (search on "clusters")

- [developerWorks Lotus Notes and Domino product page](#)

  IBM Lotus Support Technotes:

  - "[How the cluster name cache works in Notes/Domino](#)"

  - "[Removing all servers from a cluster](#)"

  - "[What is failover and when does it happen in a clustered Domino server environment?](#)"

  - "[Cluster Analysis - Replicas exist within cluster results not what is expected](#)"

  - "[Domino Mail routing does not fail over within clustered servers](#)"

  - "[Can mixed versions of Domino servers be in the same cluster?](#)"

  - "[How to test cluster failover on one Domino database (application)](#)"

  - "[Cluster failover works in one direction but not the other](#)"

  - "[Can users automatically fail back to the primary home mail server when the server becomes available?](#)"

  - "[Users are not failing over in a clustered environment](#)"

  - "[How is the Server Availability Index (SAI) calculated](#)"

About the authors

Reetu Sharma is Software Engineer working on the Domino Server team at IBM's Technical Support Center in Pune, India. She is an IBM Certified System Administrator for Lotus Notes and Domino 6/6.5.

Ranjit Rai is Technical Advisor for the IBM Technical Support Center in Pune, India. He is a Lotus Certified Professional (CLP), with more than eight years of experience working on Lotus Domino.