Red Hat Reference Architecture Series
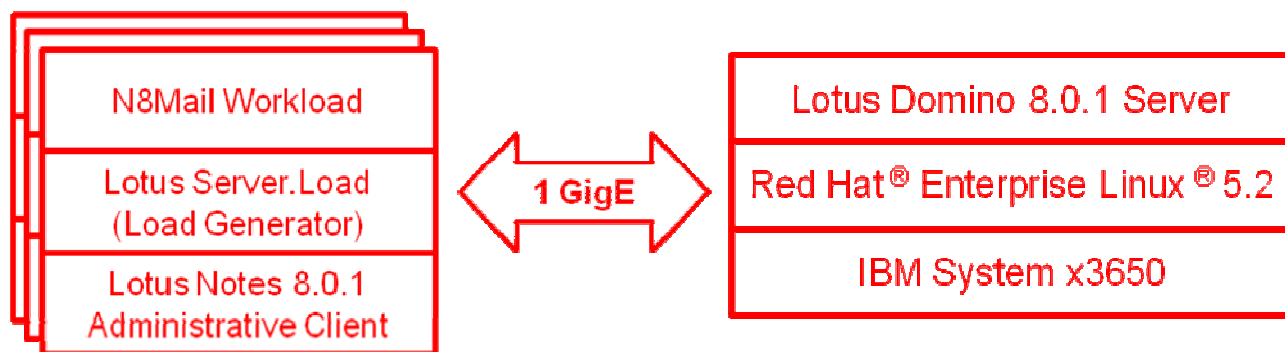
IBM® Lotus® Domino® Server Performance on Red Hat® Enterprise Linux® 5

N8Mail Workload

Lotus Server.Load (Load Generator)

Lotus Notes 8.0.1 Administrative Client

1 GigE

Lotus Domino 8.0.1 Server

Red Hat® Enterprise Linux ® 5.2

IBM System x3650

Version 1.0

November 2008

redhat.

IBM®

IBM® Lotus® Domino® Server Performance
on Red Hat® Enterprise Linux® 5

1801 Varsity Drive
Raleigh NC 27606-2072 USA
Phone: +1 919 754 3700
Phone: 888 733 4281
Fax: +1 919 754 3701
PO Box 13588
Research Triangle Park NC 27709 USA

The following terms used in this publication are trademarks of other companies as follows:

- IBM, the IBM logo, Lotus Notes, Domino, Lotus, and Notes and all other IBM products and services mentioned herein are trademarks of International Business Machines Corporation in the United States and other countries
- Linux is a registered trademark of Linus Torvalds
- Red Hat, Red Hat Enterprise Linux and the Red Hat "Shadowman" logo are registered trademarks of Red Hat, Inc. in the United States and other countries
- Intel and Xeon are registered trademarks of Intel Corporation.

All other trademarks referenced herein are the property of their respective owners.

Table of Contents

1. Executive Summary

The purpose of this paper is to evaluate the performance of Lotus Domino 8.0 Server running on Red Hat Enterprise Linux (RHEL) 5.2.

First, this paper gives a brief introduction to the Lotus Domino performance benchmarking concepts and methodology. Second, it outlines the detailed procedures for running the Server.Load Utility with the N8Mail Workload. Finally, it presents the performance results of Lotus Domino Server on Red Hat Enterprise Linux (RHEL) for simulated workloads of:
- 600 users and
- 1200 users

Server.Load and NotesBench are two load drivers developed for the performance analysis of Domino server systems. They both run on Notes client systems and use workload scripts to control simulated user activity against a server.

NotesBench and Server.Load are both built using the same server performance testing engine that has been expanded over the years to include Internet mail, discussion databases, and calendar and scheduling. Server.Load was created as a simpler, more flexible alternative to NotesBench.

The NotesMark rating, analogous to a TPC-C rating for OLTP workloads, allows you to compare Domino scalability on different hardware platforms. Using NotesBench to get a NotesMark rating requires more setup and a stricter set of guidelines than running workloads with Server.Load to expose system bottlenecks.

However, the fastest way to get started with Domino performance analysis is to use the Server.Load load driver.

In addition to a load driver, one needs a workload that simulates user activity to drive the Domino server, i.e., System Under Test (SUT). One of the ways to achieve a close-to-real-world view of a Lotus Notes user is to closely mimic the API calls by the Lotus Notes V8 clients. While there are older workloads like N7Mail and R6Mail, the N8Mail workload is the latest workload. The N8Mail workload is used to reproduce Lotus Notes V8 client calls. It is a completely new workload with heavier transaction rates. The N8Mail workload can be used to benchmark Lotus Domino V8.0.

For the performance analysis presented in this paper, the following were used:
- Load Driver = Domino Server.Load
- Workload = N8Mail

2. Domino Performance Benchmarking Concepts

This section introduces the basics of analyzing Lotus Notes/Domino performance to identify bottlenecks at the OS and application levels. It includes analyzing and tuning your system hardware and software for optimal performance of all major components including memory, CPUs, disk drives, and the network. It discusses performance testing tools and analyzing test results, and lists resources for tuning and capacity planning.

Performance tuning has one overall goal: To get the most from your hardware. But how can you make sure that you are getting maximum utilization of your Domino server's resources?

You should start with a top-down approach to performance tuning. Before tuning your Domino servers for performance, you have to make sure that your network components are operating efficiently. Network bottlenecks can come from many sources and can often be tricky to isolate. Routers, gateways, firewalls, and network collisions could all be a source of performance bottlenecks. You may also need to use a sniffer device to look for problems with network traffic.

You should then make sure that your Domino systems are using the network efficiently, Check the Network Utilization and Total Bytes sent/second.

Once the network is tuned, you are ready to look at the performance of your Domino servers. How involved or complex Domino server performance tuning is depends on the size and complexity of your Domino network. If you have just one Domino server with one disk drive and one central processing unit (CPU), your performance tuning options are fairly basic:

- Design applications, forms, views, and pages for maximum performance
- Use redirection URLs
- Use faster alternatives to formulas, such as computed subforms and JavaScript
- Turn off unneeded Domino server tasks
- Optimize LotusScript and Java code

Performance tuning options on Domino servers with more than one CPU and multiple hard drives is obviously more involved. All of the above techniques apply to such systems, but in addition, you need to make sure that the Domino workload is efficiently distributed and balanced among the physical disk drives. Because the Domino server can be very disk intensive, configuration of disk drives, and arrays of hard drives called RAID drives, can greatly enhance server performance.

One of the keys to Domino server performance on anything but the most simple system can be summed up as follows:

- Spread out the work over the hard disks and processors

  Performance tuning of these systems involves setting up the disk drive partitions, and arranging the system, program, and data directories to distribute the workload across the system as evenly as possible. These systems often require some experimenting with and adjustment of hardware and software before you reach optimal performance. Performance testing tools let you experiment with different system configurations, such as RAID array stripe size and hard disk file format, to see which combinations result in better performance.

2.1 Domino performance testing tools

Many performance bottlenecks will only reveal themselves when the server is under high load, so server performance tools are used to simulate a load of mail messages or database activity. Both the Server.Load and NotesBench performance tools that are developed with Domino run on Notes client systems and use workload scripts to control simulated user activity against a server.

NotesBench and Server.Load are both built using the same server performance testing engine that has been expanded over the years to include Internet mail, discussion databases, and calendar and scheduling. Server.Load was created as a simpler, more flexible alternative to NotesBench.

2.1.1 Server.Load

Server.Load uses a graphical user interface (GUI) to control workload operation. Server.Load is best for small and mid-sized evaluations and is included on the Domino server CD ROM. It:

- Collects 150 performance metrics from the system under test.
- Runs built-in and customized workload scripts.

2.1.2 NotesBench

NotesBench is a powerful command line based tool available only to members of the NotesBench Consortium and others who are individuals trained in performance analysis. It requires more Domino system knowledge and setup time than Server.Load and is not designed for small-scale evaluations. It:

- Generates a number of users-per-second system rating with a response time.
- Runs a standard set of built-in workloads.
- Gets all workload control information from the NOTES.INI file.
- Synchronizes multiple test drivers in a parent/child relationship.

2.1.3 Which is right for you?

Both NotesBench and Server.Load can be used to run workloads that will expose the bottlenecks in your server. However, the fastest way to get started with Domino performance analysis is to use Server.Load. You already have a copy of the program and documentation on your Domino server CD!

If you are planning a large-scale deployment, we recommend you enlist the services of a NotesBench Consortium member. They have experience configuring and running NotesBench and can customize a NotesBench run on their platforms that will conform to your requirements.

The NotesMark rating, analogous to a TPC-C rating for OLTP workloads, allows you to compare Domino scalability on different hardware platforms. Using NotesBench to get a NotesMark rating requires more setup and a stricter set of guidelines than running workloads with Server.Load to expose system bottlenecks.

2.2 Understanding the Role of Benchmark Servers (= System Under Test or SUT)

A benchmark server is a Domino server set up for testing with Server.Load, NotesBench, or third party tools. This server is created in an isolated Notes domain to contain the heavy load placed on the testing systems.

Hundreds, or even thousands, of mail files are created along with related Person documents by setup scripts and agents provided with the performance testing tools. This lets you better understand how your system will behave under high load, to get an idea of system limitations, and to head off problems in production servers before those servers are overloaded.

After the tests are run and the information is analyzed, the benchmark server can be converted back to a production server or can become a testing server with another focus.

2.3 What is steady state?

Steady state means the system under test has reached equilibrium and that the system could sustain the test load for as long as needed because memory, CPU, and disk I/O (input/output) are not increasing and are not near critical limits.

The graph below shows a server running a Server.Load workload overnight that has reached steady state. Note how activity for the major system components (in this case the CPU) have leveled off after the initial ramp-up. If activity of any system component is increasing throughout the test run, the system has not reached steady state.
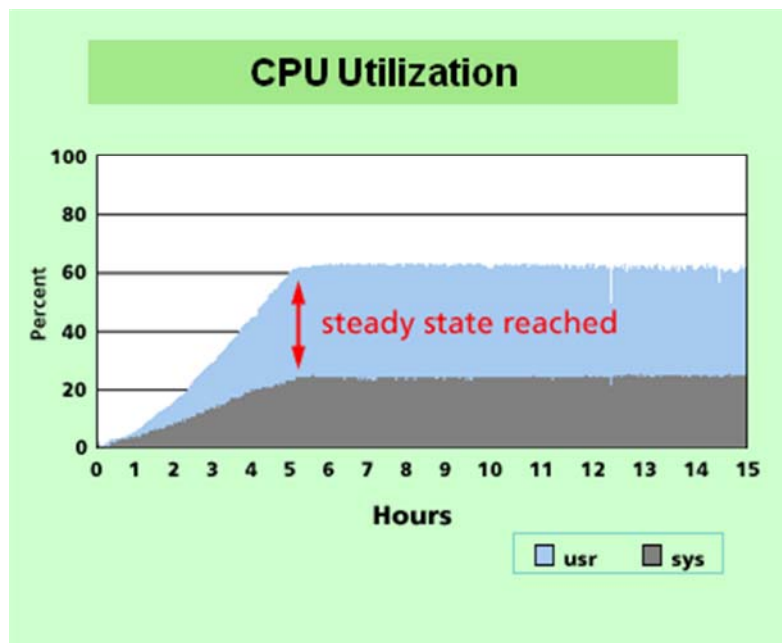

Figure 1

2.4 Collecting performance data with system monitoring tools

The operating systems that Domino runs on keep track of everything that is going on inside the system. Programs called monitors gather the requested data from the statistic counters and record it to a text file or display it as a graph. These data will show which system components are at or near maximum capacity and allow you to measure how adjustments to the system have affected performance. Unix and Linux platforms use command line utilities like IOSTAT and VMSTAT. Other tools and methods are available on the other Domino server platforms.

2.5 How Domino performance testing works

The illustration below shows how the data flows between test drivers and the system under test. The blue arrow represents the requests that simulate users on the server. The orange arrow represents the statistics retrieved from the server that are captured in the Server.Load metrics file. Server.Load also captures the text from the Script Output Monitor window into a results file, which can be searched for timeouts and error messages.
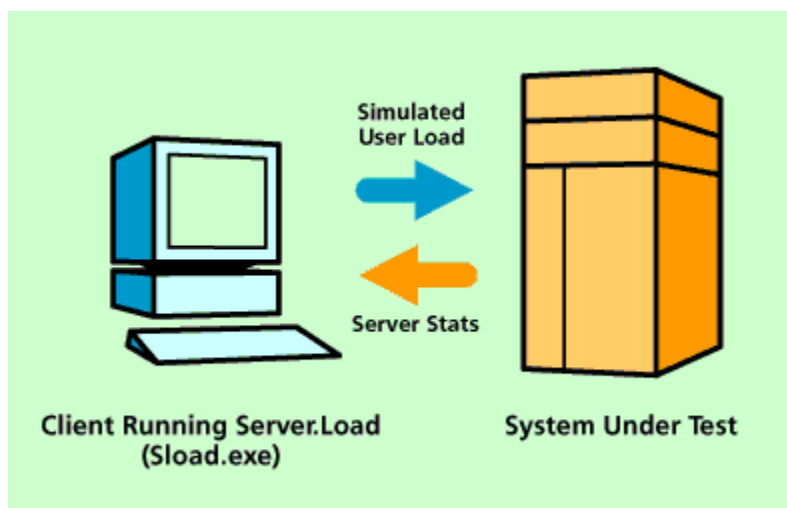
Figure 2

To run performance tests and collect data with monitoring tools:

- Install Server.Load on the test clients.
- Set up performance tests using special workloads and agents to create mail files and Person documents for test users.
- Run the performance workload to create a heavy workload on the system under test until it reaches steady state.
- View and graph the server statistics collected from Platform Stats on Unix/Linux platforms in the Statrep.nsf database.
- Collect the response time data from the test driver client metrics file.
- See the section below to learn how to interpret the collected data.

2.6 Analyzing test results: The most likely suspects

When looking for data bottlenecks, you should start at the central system components. Just as traffic backs up at some intersections only in heavy traffic, data can back up in your system in critical areas only when under heavy load. Performance testing clogs the weak points in your system by creating an e-mail "rush hour." Then the operating system performance statistics are used like traffic webcams to spy on the data flowing through the system.

The key thing to remember is that every subsystem is associated with a queue, and it is the queue length that needs to be monitored. In the case of the CPU, it is the System Processor Queue Length, for Disk it is the Average Queue Length, and so on. You can look for bottlenecks in a logical order.

2.6.1 The server CPU

The first place to check for bottlenecks is the server CPU. Even systems with infinite disk space and RAM are limited by the amount of data that can be pumped through the CPU. If the percent busy is close to or at 100 percent, then the CPU could be limiting the system. However, you can be at 100 percent CPU, and the CPU does not have to be the bottleneck. Note that you can have a server hit 100 percent CPU, but it should not be pegged at 100 percent CPU for longer than one minute.

Ideally, a server should have some room for growth and once it is pushed to 80 percent processor time, you should not add any more registered users or server tasks.

2.6.2 Available memory

The next stat to check is available memory. If available RAM is approaching zero at any time during load, your system performance is limited by RAM. Look at the physical RAM and memory committed bytes.

2.6.3 Disk input and output

Next look at the disk input and output. This is where it gets interesting! If you have plenty of RAM and CPU cycles available, but the system is still taking a long time to open databases, look at the Logical Disk Percent Time and Logical Disk Service Time.

2.7 Lotus Notes V8 Workload

One of the ways to achieve a close-to-real-world view of a Lotus Notes user is to closely mimic the API calls by the Lotus Notes V8 clients. The **N8Mail workload** used to reproduce those calls is a completely new workload with heavier transaction rates. The N8Mail workload can be used to benchmark Lotus Domino V8.0.

2.7.1 NotesBench and Server.Load Workload

Server.Load, a GUI version of the NotesBench workloads, is widely available to all customers. Server.Load is the Lotus Domino feature-based load generation tool that you can use to measure and characterize various Lotus Domino server capacity and response metrics. The workload is available in Server.Load for Lotus Notes and Domino V8. The workloads (also called tests or scripts) simulate the behavior of Lotus Domino client-to-server operations. The workload allows you to use a common tool to apply a consistent, repeatable load against the Lotus Domino server so that you can assess the effects of various operating system, hardware, and configuration changes. Server.Load is an installation option in the Lotus Domino Administrator client.

The Lotus Domino Performance Team continues to add features and enhancements to the Server.Load utility that ships with the Lotus Domino Administrator client. With each major release of Lotus Notes and Lotus Domino, close attention is paid to the changes in the Lotus Notes V8 client to represent workload user actions accurately. This accuracy stems from the team making sure the workload's transactions with the Lotus Domino server matches that of the Lotus Notes V8 client. Also, with this release, setup components of the Lotus Domino environment that were not involved in previous workload versions have been included in the workload. The new components consist of all default server tasks, mail journaling, transaction logging, Lotus Domino Domain Monitoring (DDM) probes, mail rules, and larger mail files. All of this enables the N8Mail workload to provide a closer-to-real world view of a Lotus Notes user interacting with a Lotus Domino server in a more typical environment.

The users shown in tables 1-4 are only instances of the benchmark script running, and they do not necessarily correlate to the number of actual users deployed on a server. The goal of the improvements in the N8Mail workload is to bring the user model closer to a typical production user. Because representing every customer usage scenario is difficult, we emphasize that you should contact your hardware vendors for the best capacity planning and deployment advice for your configuration and environment.

2.7.2 Workload Specifications

N8Mail simulates full Lotus Notes V8 client users. The new features of the Lotus Notes and Domino V8 workloads include those shown in Table 1.

Table 1. More realistic mail database profile for N8Mail

	N8Mail
Mail database size	Approximately 250 MB
Average document size	100 KB
Initial document count	3,000

2.7.3 Messages Generated

The frequency of creating/sending memos has increased. Fifty total documents are generated in an average six-hour run, one per iteration, and an additional one every other iteration. One iteration is approximately 15 minutes. Memo recipients now vary between one and three recipients. There are now fewer invitations and appointments, one each every six hours. Overall, you find 50 memos, invitations,

and RSVPs to about 78 recipients, per user for a six-hour run. A new feature is the generation of replies to received mail, one every other iteration. See Table 2 for more details about memo profiles.

Table 2. Memo profile for N8Mail

Message body size in KB	Attachment size in KB (if any)	Percent
500	N/A	10
10,000	N/A	30
50,000	N/A	40
50,000	50,000	10
150,000	N/A	9.5
1,000	10,000,000	0.5

Table 3, which shows data from the perspective of recipients of mail and the message distribution therein.

Table 3. Message distribution in N8Mail

Message size distribution (using the profile option)	Percent of messages to total number of recipients	Attachment size (if any)
0 < size <= 1K	32.0%	N/A
1K < size <= 10K	3.6%	N/A
10K < size <= 100K	57%	50 KB
100K < size <1M	6.8%	N/A
1M < size <= 10M	0.4%	10 MB

NOTE: The percentages do not match the percentages defined in the message profile discussed previously because the profile is used only for memos. It is not used for replies, invitations, or RSVPs, which are accounted for in this table.

N8Mail new user actions include the following:

- Move random notes from the inbox to the NotesBench folder
- Read the newest inbox items
- Reply to a read message
- Save all sent messages in the Sent folder
- Conduct new mail polling every 15 minutes (that is, every iteration)

Table 4. User actions: N8Mail (six-hour period, 24 iterations)

Workload actions	N8Mail workload action count
Refresh inbox	24
Read message	120 (reads 5 per iteration)
Reply to all	12
Send message to one recipient	24
Send message to three recipients	12
Create appointment	1

Send invitation	1
Send RSVP	1
Move to folder NotesBench	24
New mail poll	24
Delete two documents	24
Stamp documents (Release 6)	N/A
Add documents to the inbox (Release 6)	N/A
Total transactions (per user in a six-hour period, 24 iterations)	267

NOTE: One iteration is approximately 15 minutes. The workload action counts are:

- Send 3,000 mail notes (from distinct users) to populate the inbox
- Create necessary profile documents with appropriate fields
- Create a NotesBench folder
- Generate five replies from what is newest in the inbox
- Put ten invitations in the database
- Optionally add the blocksender rule
- Add ten users not in the range of valid workload users to the rule
- Optionally add the color profile
- Run Updall -R <maildb> to update the maildb

2.7.4 Server configuration

In addition to offering improved user actions, the N8Mail workload uses server features not used by previous workloads. Using these features brings the benchmark environment closer to your deployment environment. This increases the Lotus Domino server CPU utilization.

Server features include the following:

- Transaction logging. Disk space is approximately 4 GB, Circular, Standard, or favor run time.
- Mail journaling. Disk space is approximately 4 GB, and all messages are journaled.
- DDM probes. All operating system and messages with default values (events4.nsf) are enabled.
- Mail filtering. Each mail file has 10 users in the block mail list. The 10 users are not in the active user list in the workload run. The goal is measuring the overhead of having filtering active, not the overhead of doing the filtering.
- Use default server tasks.
  ServerTask=Update,Replica,Router,AMgr,AdminP,CalConn,Sched,RnRMgr,LDAP (from the server's Notes.ini file)

3. Hardware/Software Configuration

Our Domino Server environment under test consisted of a single Domino 8.0.1 Server on an IBM System x3650 configured with one Dual Core Intel Xeon processor and 4 GB of RAM running Red Hat Enterprise Linux (RHEL) 5.2.

Three Windows XP Clients were configured with the Domino Administrator Client to run the N8Mail Initialization Workload, N8Mail Workload, the Data Collection Script and The Data Rollup Workload included with the Server.Load utility to simulate a workload of up to 1200 users.

Note: Results published in this paper are from benchmarks executed in this limited and controlled environment. They do not represent recommendations for a production environment. For assistance with Domino Capacity Planning it is recommended that you consult your hardware vendor and IBM Techline which handles capacity planning for new hardware.

3.1 Domino 8.0.1 Server (1)

- IBM System x3650
- 4 GB PC2-5300 Memory
- 4 73.4 GB IBM SAS 15K hard drives
- 2 146.8 GB IBM SAS 15K hard drives
- 1 Intel Xeon Dual-Core 3.0 GHz
- 1 ServeRAID 8k-l SAS Controller
- 2 full-duplex Broadcom 5721 Gigabit Ethernet PCI controller
- 32-bit RHEL 5.2 Server
- 32-bit Domino 8.0.1 for Linux

3.2 Lotus Notes 8 Administrator Client Systems (3)

- IBM System x325
- 4 GB PC2700 ECC DDR SDRAM memory
- 2 18.2 GB IBM SAS 15K hard drives
- 1 2 GHz AMD Opteron Processor
- Integrated dual Gigabit Ethernet ports
- Windows XP
- Lotus Notes 8.0.1 Administrator Client

3.3 Preparing the Domino Server for Performance Measurements

Prior to tuning your Domino server for performance, ensure that your network components are operating efficiently. Next, make sure that your Domino systems are using the network efficiently. You can check the network utilization and Total Bytes sent/second to see how much processor time is spent servicing the network.

Plan your Domino Server installation so that work will be distributed across hard disks and processors as evenly as possible. At a minimum place the main operating system files and the Domino application files on a separate disk than your Domino data files. If using transaction logging, the log files should be on your fastest drive. The Domino server can be very IO intensive. How you configure your disk drives can greatly enhance server performance.

When installing Domino on Linux it is important to consider which file system is best for your environment. Several options are available including: xfs, ext3, jfs, reiserfs and ext2. Journaled file systems will perform slower than non-journaling file systems; but, recover from errors faster. Domino mail databases, like any mail files, can get quite large; you will want to avoid file systems which are optimized for working with small files.

Finally, you will want to reduce unnecessary load on your server. Do this by disabling or uninstalling any operating system and features that are not required by your Domino Server. You can also optimize Domino by not running Domino server tasks which are not required in your Domino environment.

3.4 Domino Server Settings and Configuration

Prior to running Server.Load against the Server Under Test (SUT). The following Domino Server Optimization settings were applied to the Domino Server under test.

- Two Raid 0 Arrays were created. 1 73 GB partition to store the operating system and Lotus Domino Server Binaries. The second, consisted of the 5 remaining disks and was used to store the Domino Data Directory
- Ext2 was chosen as the Linux File System
    - To further enhance performance of the ext2 filesystem on Linux, the Domino data partition was mounted with the noatime option. The noatime option turns off ext2 filesystem access time tracking.
- In the Configuration Settings document on the Router/SMTP Basics tab, set the field "Number of mailboxes" to 2. .
- The following notes.ini settings were set on the Linux Server Under Test
    - SERVER_SHOW_PERFORMANCE=1
    - platform_statistics_enabled=1
        - *Note: There is a known issue with Domino 7.x/8.x related to enabling platform statistics. Prior to running any performance testing complete the procedure outlined in* Technote: 1292478

- SCHEDULE_NO_VALIDATE=1
- NSF_buffer_pool_size_MB=512
  Note: 512 is the default value for NSF_buffer_pool_size_MB with 32-bit Domino 8.0.1 and does not need to be explicitly set. For Recommendations for setting NSF_BUFFER_POOL_SIZE_MB visit [Technote: 21286171](#)
  Server_Max_Concurrent_trans=100
- server_pool_tasks=80
- Servertasks=Router,Update,AMgr,Adminp,Sched,CalConn,RnRMgr,HTTP,LDAP

Lotus Domino includes a utility called tunekrnl that will attempt to tune the Linux environment. The memory information returned by the Linux operating system in not always accurate; therefore, it is sometimes necessary to tell Domino the amount of memory it has available. We do this by setting additional notes.ini variables. For the purpose of this paper, in addition to the notes.ini variables set above the following notes.ini variables were also set:
- MEM_AddressableMemSizeMB=2560
- ConstrainedSHMSizeMB=2048
- MEM_EnablePreAlloc=1

Note: These Notes.ini parameters deal with the allocation of memory for the Lotus Domino server. Prior to setting these variables, you should attempt to run Domino using the default settings. These setting should be used with caution. Granting to much or to little memory to Domino can cause performance to degrade, always test your server performance before and after changing these values.

The Notes.ini parameter MEM_AddressableMemSizeMB=2560 tells Lotus Domino that it has 2560 MB of memory to use. The 32-bit Linux operating system reserves 1 GB of Memory for kernel process. Since, Domino Server is the main application running on the OS, I chose to allocate most of the remaining memory for Domino. We must leave some memory for other user space applications. Usually, 500 MB is ample memory for the remaining user space applications.

The next Notes.ini parameter, ConstrainedSHMSizeMB=2048, tells Lotus Domino how much shared memory it can use. Because most of the memory Lotus Domino requires is shared memory, we give it 2 GB for these tests.

The third Notes.ini parameter, MEM_EnablePreAlloc=1, tells Lotus Domino to pre-allocate the shared memory defined in the Notes.ini parameters. This helps prevent Lotus Domino from crashing with an insufficient shared memory error, but it is a balancing act because locking down more shared memory than needed is not wise. Doing so can cause a shortage of local memory for applications to use and cause insufficient memory errors and application failure.

4. Running Domino Server.Load Experiments

Lotus Server.Load is the IBM Lotus Domino load generation tool that can be used to measure and characterize various Lotus Domino server capacity and response time metrics.   This chapter will detail how to install and configure Domino Server.Load to run the N8Mail Workload against a Domino Server under test.

4.1 Installing and Running Server.Load

The Server.Load utility and its associated workloads are included as part of the Domino Administrator Client. To install Server.Load you will need to select this feature during installation. Domino Designer is also required to deploy and run the setup agents required by the N8Mail Workload.

4.2 Installing and Configuring Lotus Domino Administrator, Domino Designer and Server.Load

In this section we will install the Lotus Domino Administrator

Change to the client running Microsoft Windows XP.

1. Login as user *Administrator*.

2. Open a Windows Explorer by selecting **Start** -> **Programs** -> **Accessories** -> **Windows Explorer**

3. Change to location of the Notes Installer executable and double click **notes8basicallclient_w32_prod.exe** to start the installation.

4. You will be asked where to save files. Keep the default path:
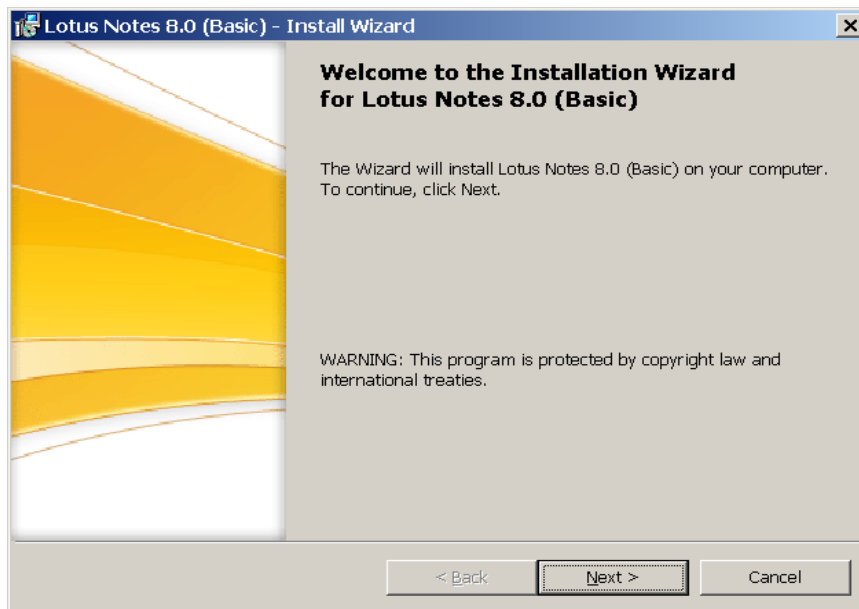   **C:\Documents and Settings\Administrator\Local Settings\Temp\Lotus Notes**
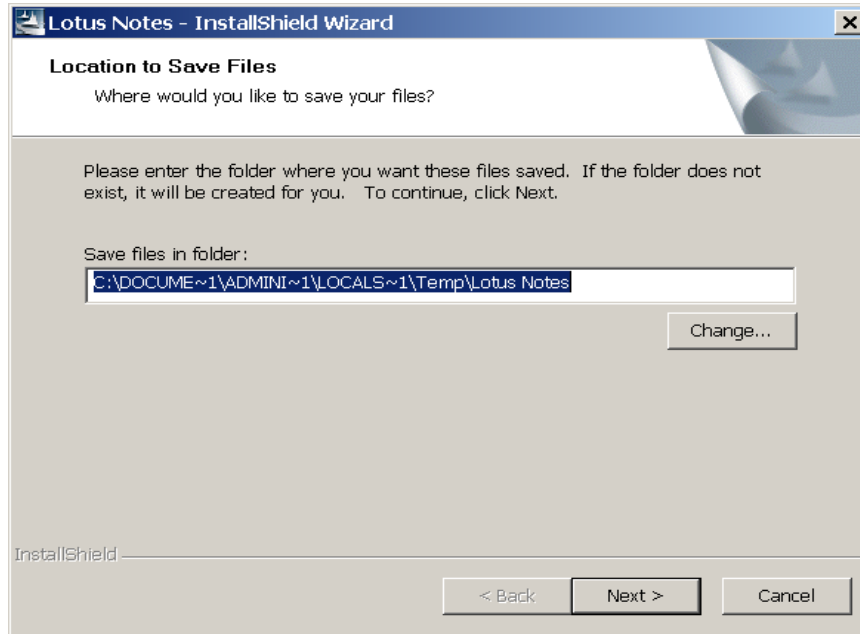
Click **Next**
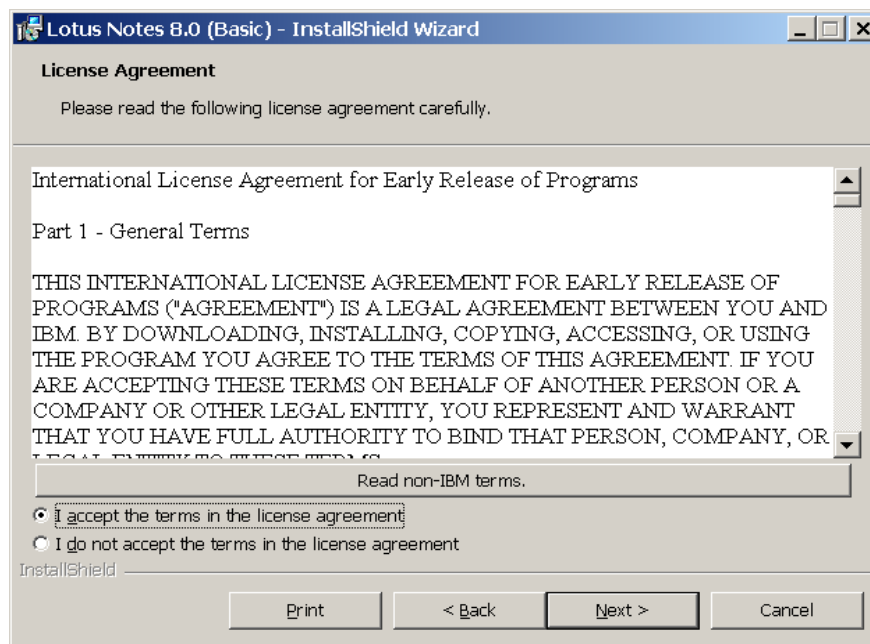
5. The files are extracted.
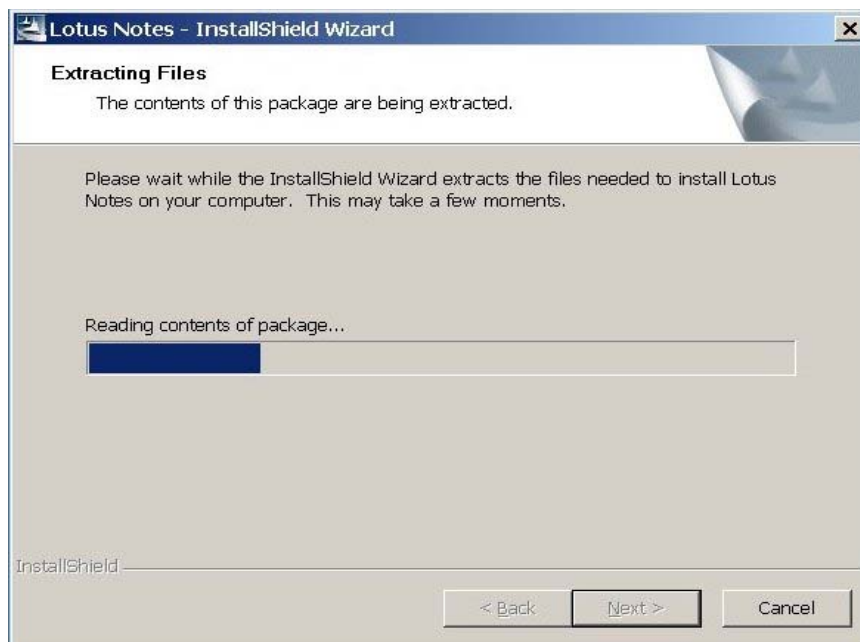
Please be patient. This may take a few minutes.

6. When the file extraction is finished, the *Welcome* screen will be displayed.

Click **Next**

7. In the *License Agreement* screen, select: **I accept the terms in the license agreement**
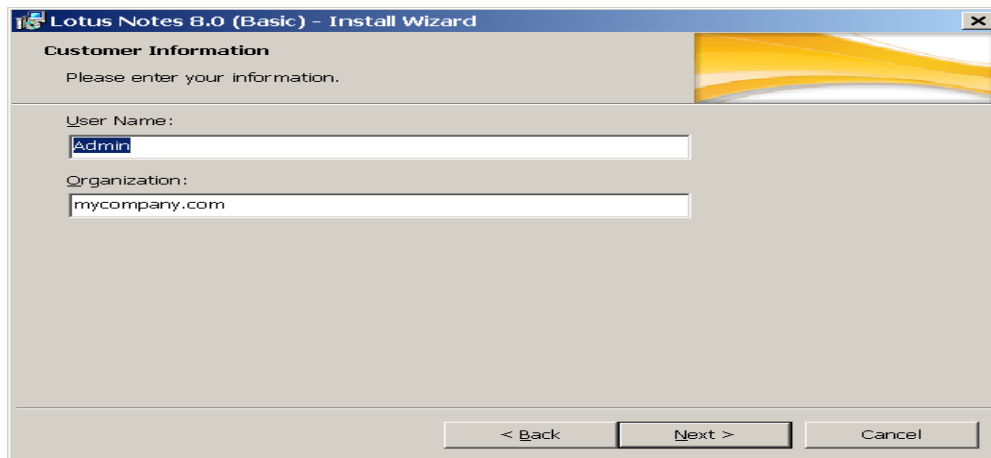

Click **Next**

8. In the Customer Information screen, provide the information requested. Record your values below.

Field	Entry
User Name:	
Organization:	

Click **Next**

9. Keep the default values in the *Installation Path Selection* screen.
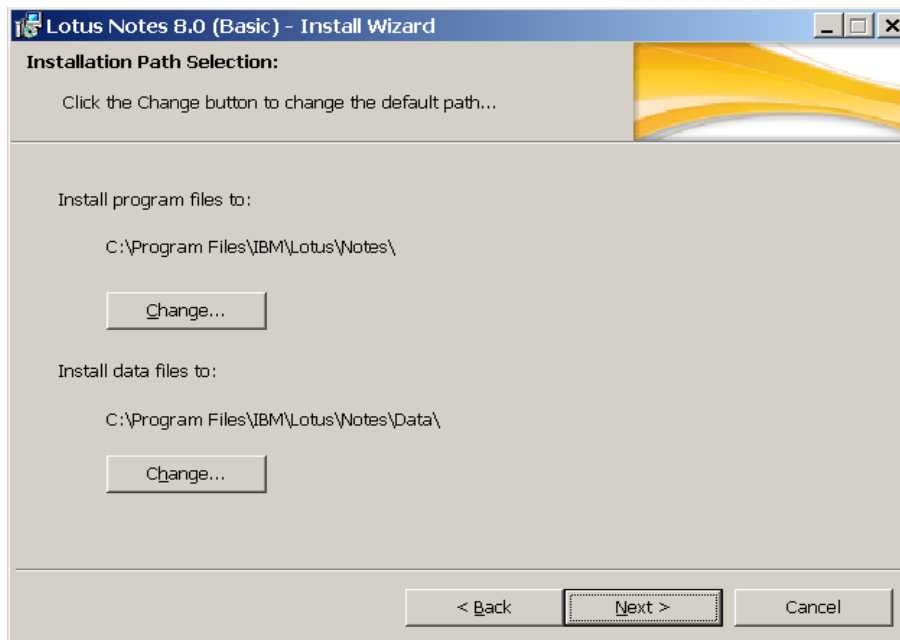
Click **Next**

10. The *Custom Setup* screen appears. Ensure that you select **all features except "*Client Single Logon Feature*"**. You can select a feature by clicking the icon next to the feature and selecting **This feature will be installed on local hard drive**.
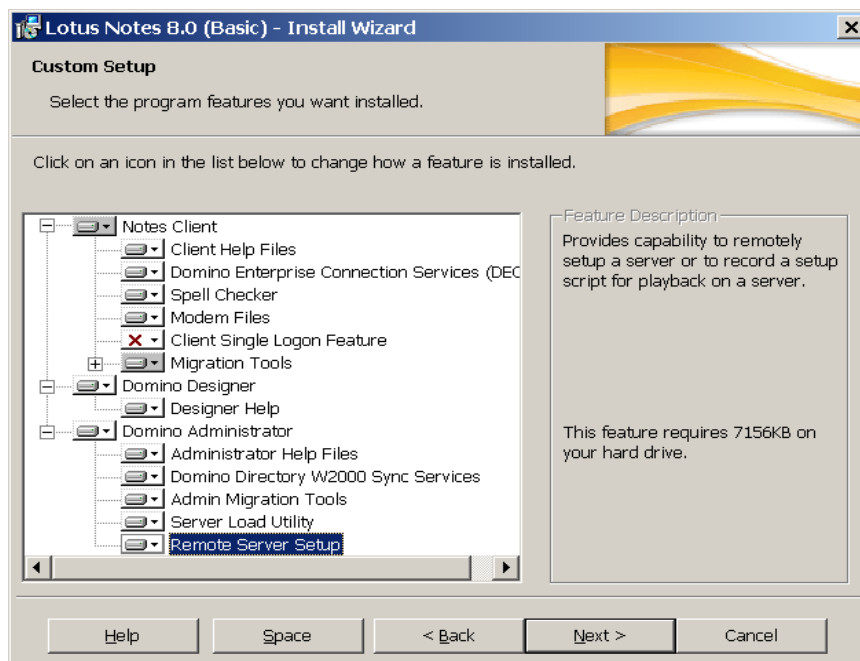
Click **Next**

11. The installation wizard has all required information and is ready to install.
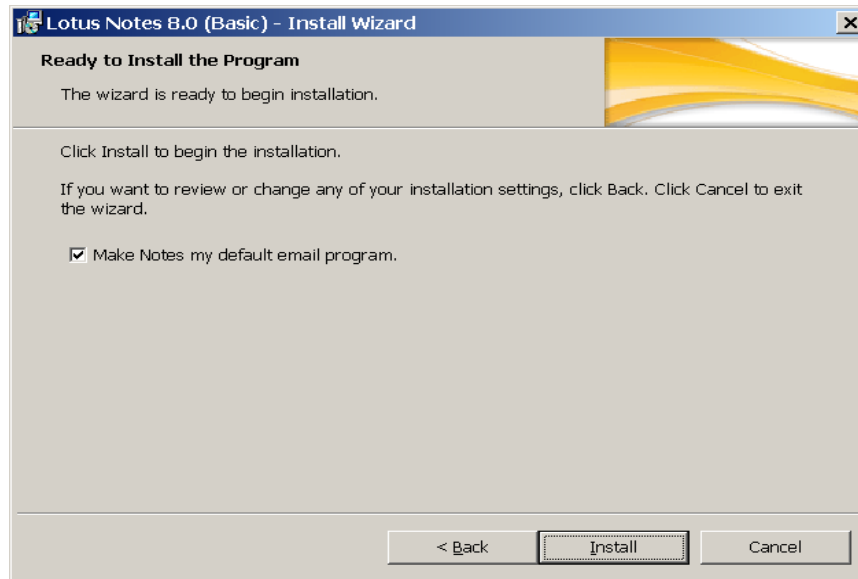
Click **Install**

12. The Lotus Domino Administrator installation begins
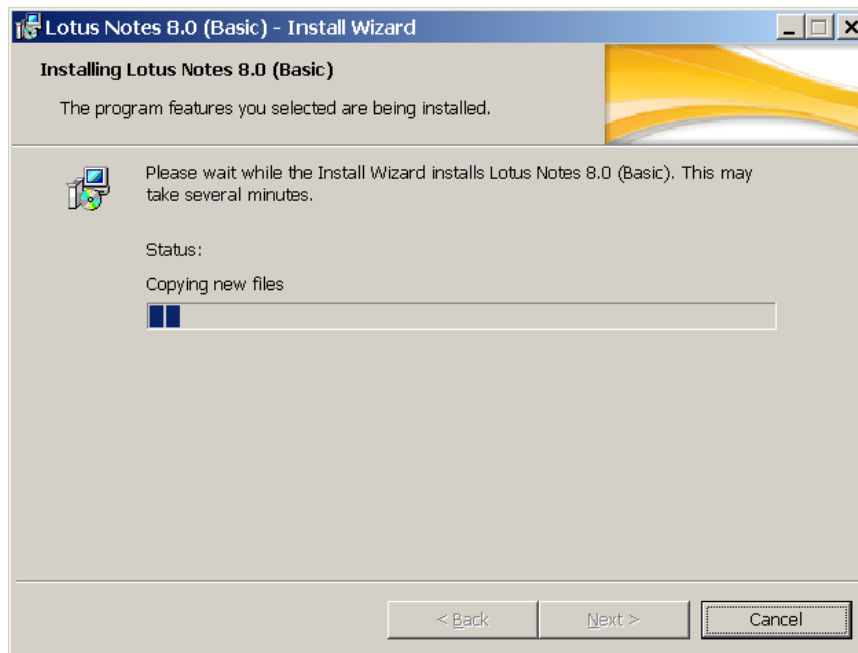
This may take a few minutes.

13. The *Install Wizard Completed* screen is displayed.
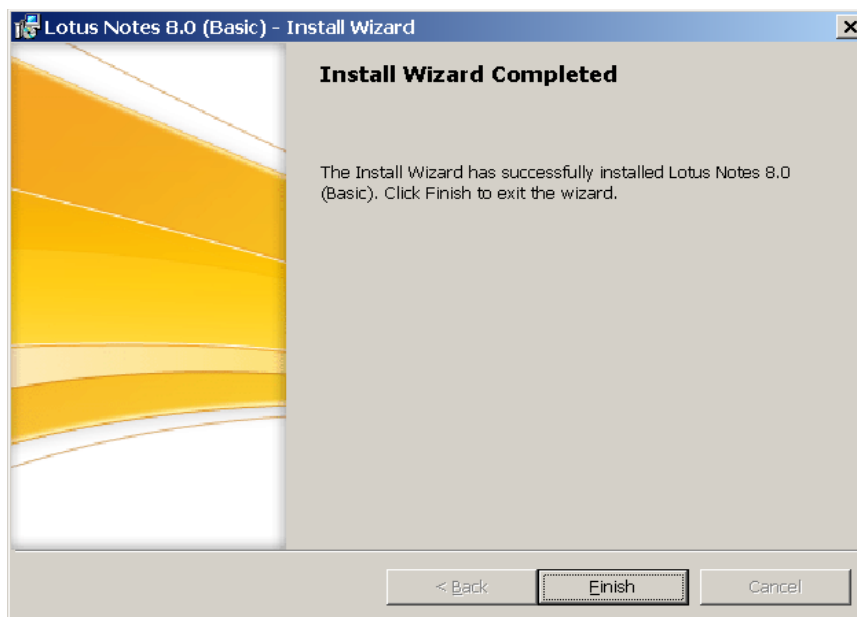
Click **Finish**

14. You should three new icons on the desktop:
    - Lotus Notes 8
    - Domino Designer 8
    - Domino Admin 8
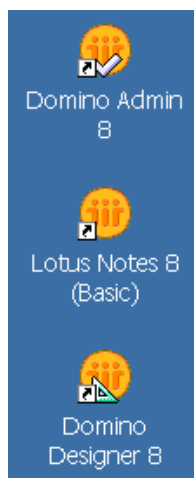
4.2.1 Configuring Lotus Domino Administrator

In this section we will configure the Lotus Domino Administrator

1. Double click the icon **Domino Admin 8** located on the desktop. The Lotus Domino Administrator is started.

2. The *Welcome* screen is displayed.
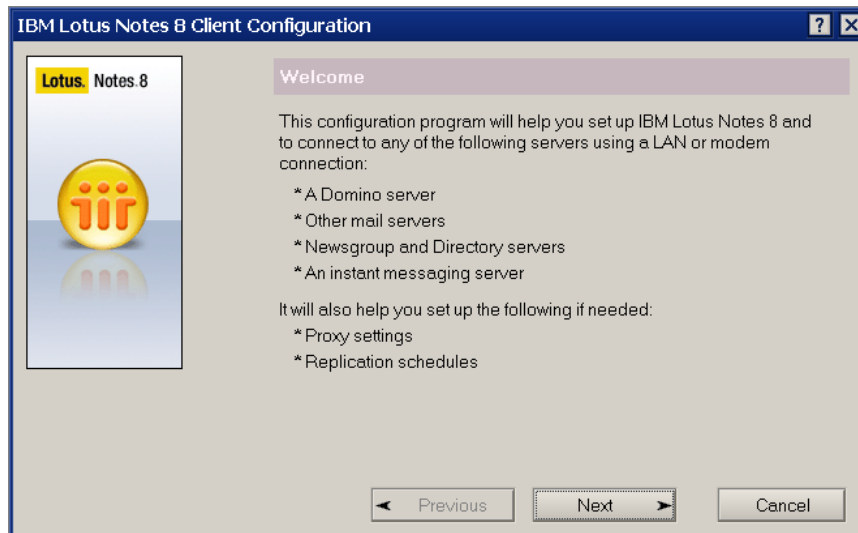
   Click **Next**

3. Provide information about the user and the Domino server you will be administering. This is the Server that will be the target of your server.load test.   In the your name field enter the name of the Domino Administrator used to Install Domino.

   ### Table 1. User Information

Field	Entry
Your name	
Domino server	

Keep the **I want to connect to a Domino** server selected.
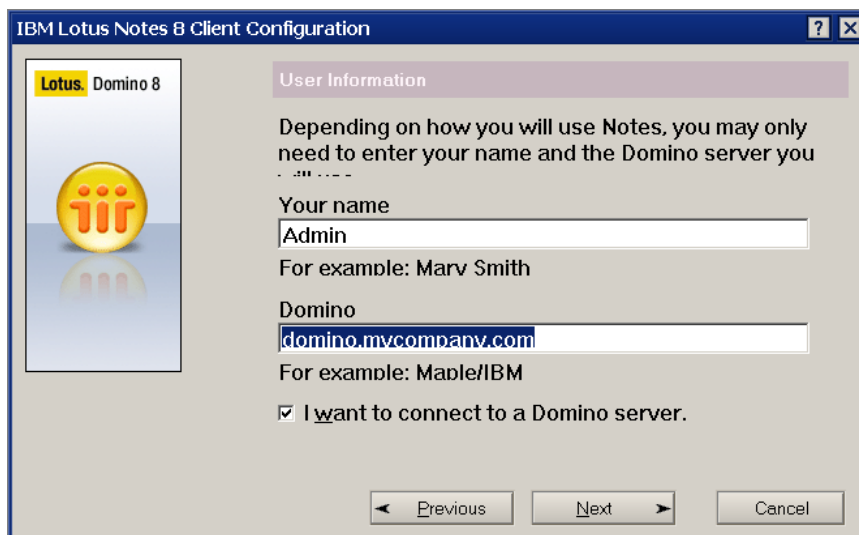
Click **Next**

4. The configuration wizard attempts to find the ID-file for this user in the Domino Directory.
If the wizard cannot find your ID-file, a window will be displayed where you have to provide the location: **C:\Documents and Settings\Administrator\My Documents\admin.id**
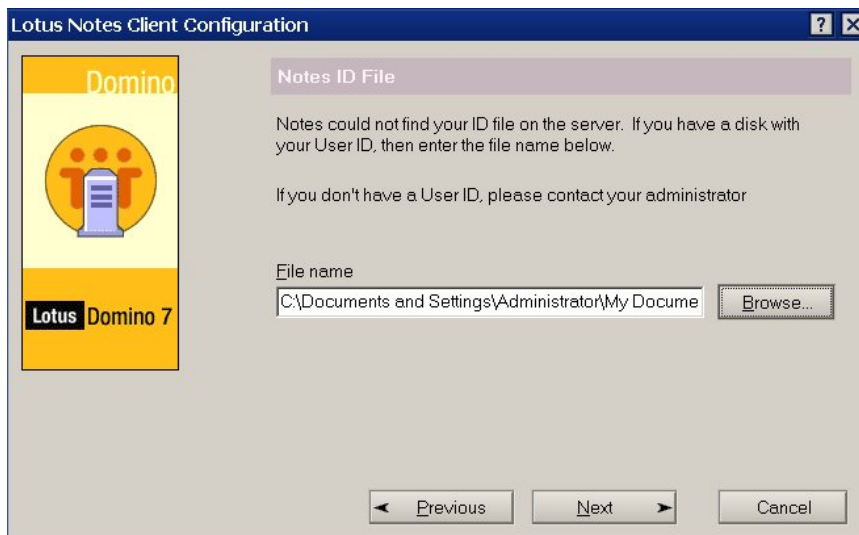
Click **Next**

5. You will be asked if you want to copy the ID file to the data directory of Lotus Notes.
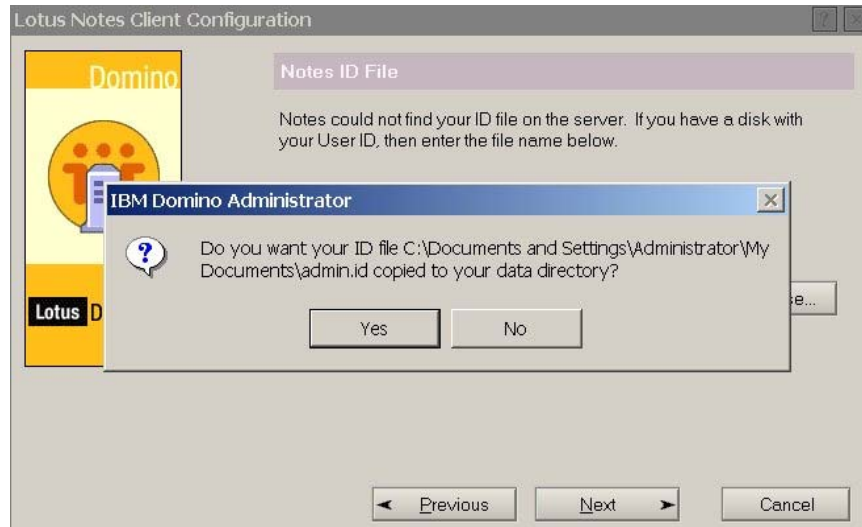
Click **Yes**

6. The password prompt appears. Type in the password for the Domino user *admin.*
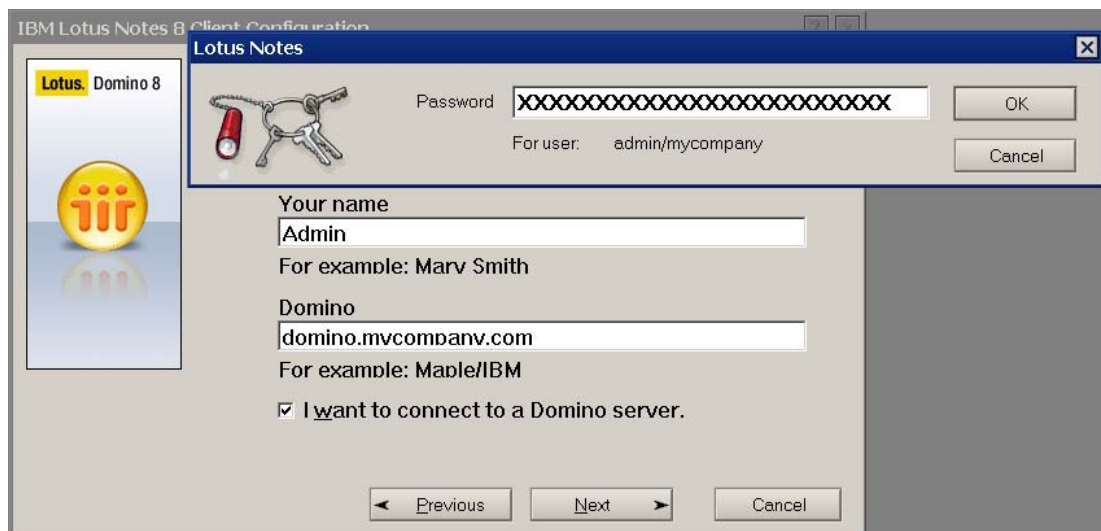
Click **OK**

7. In the *Instant Messaging Setup* screen, deselect **Setup instant messaging** as we do not have a Lotus Sametime server configured.
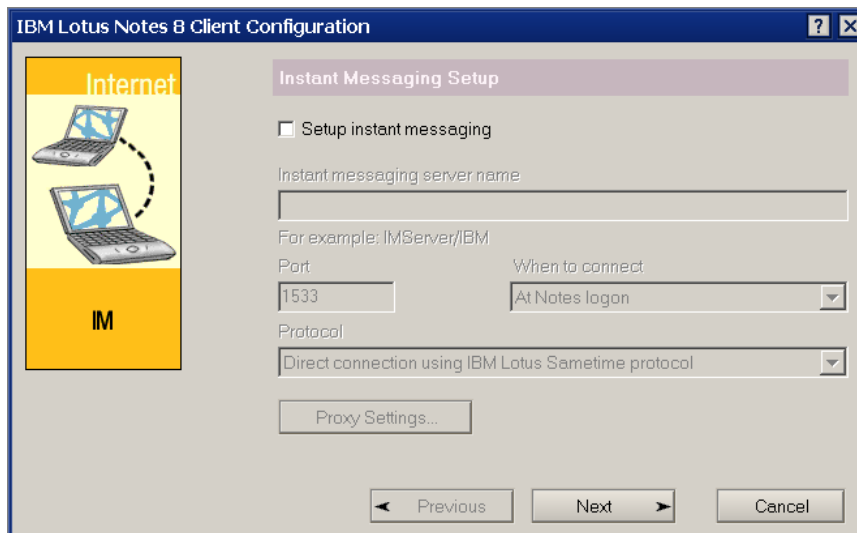
Click **Next**

8. In the *Additional Services* screen, leave everything unselected.
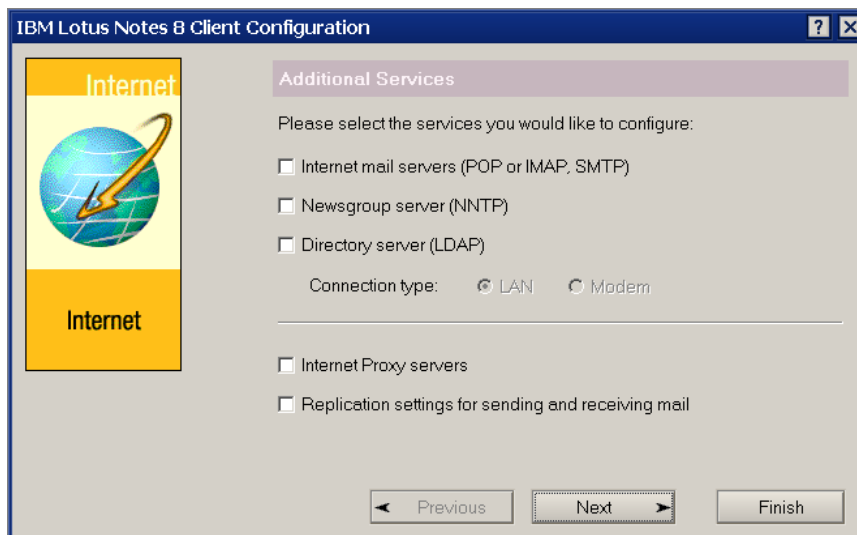
Click **Finish**

9. The Setup is completed.

Click **OK**

10. The IBM Domino Administrator will be started.
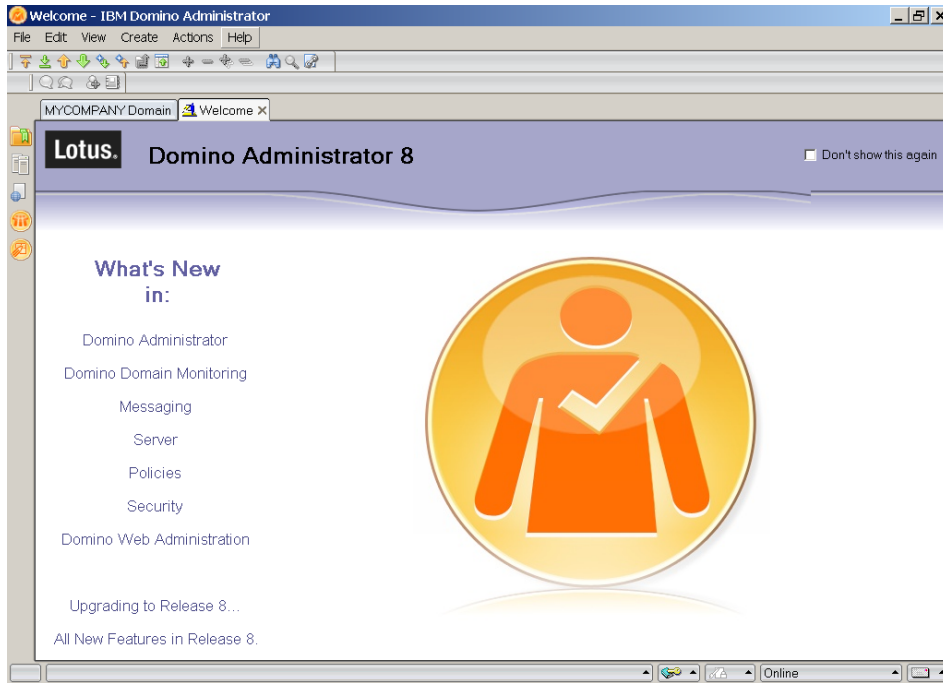
4.2.2 Testing Lotus Domino Administrator

For testing the installation of Lotus Domino Administrator we will access the Server Console to see the runtime messages generated by the Lotus Domino server.

1. If not started, start *IBM Domino Administrator* by double clicking the icon **Domino Admin 8** located on the desktop.

2. Click on the **Domain** tab
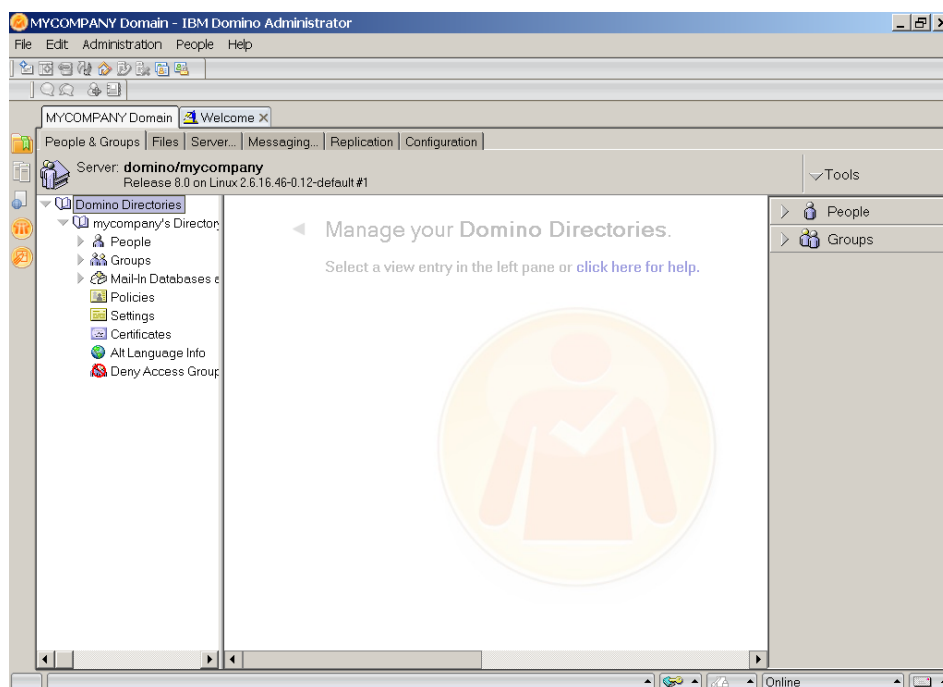
3. Within the **Domain** tab, click on the **Server** tab
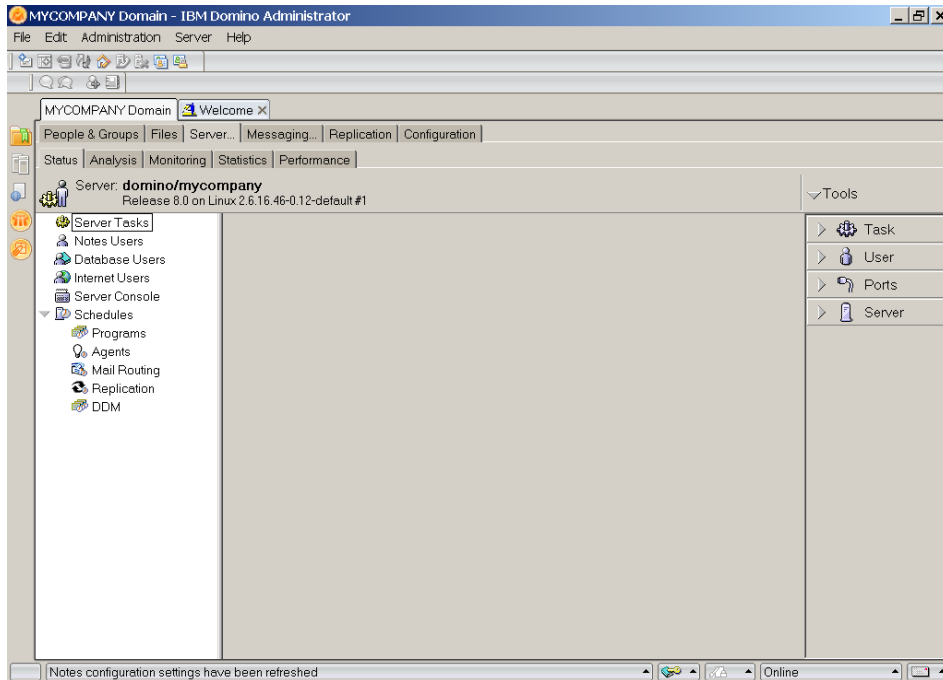
4. In the left navigation bar, select **Server Console**
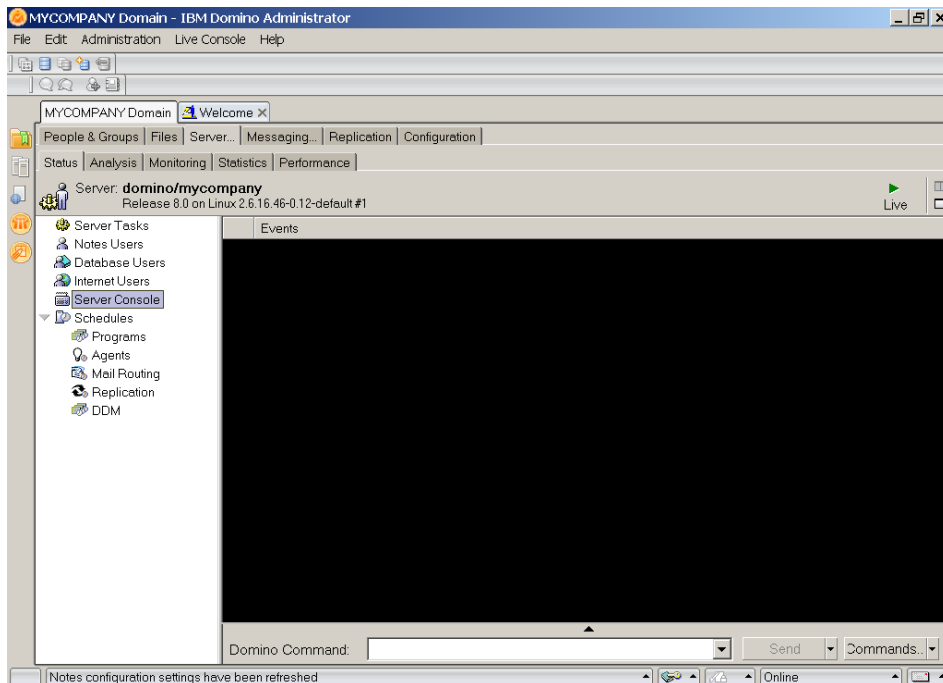
5. Click the **Live** button in the right upper corner to connect to the Lotus Domino server.
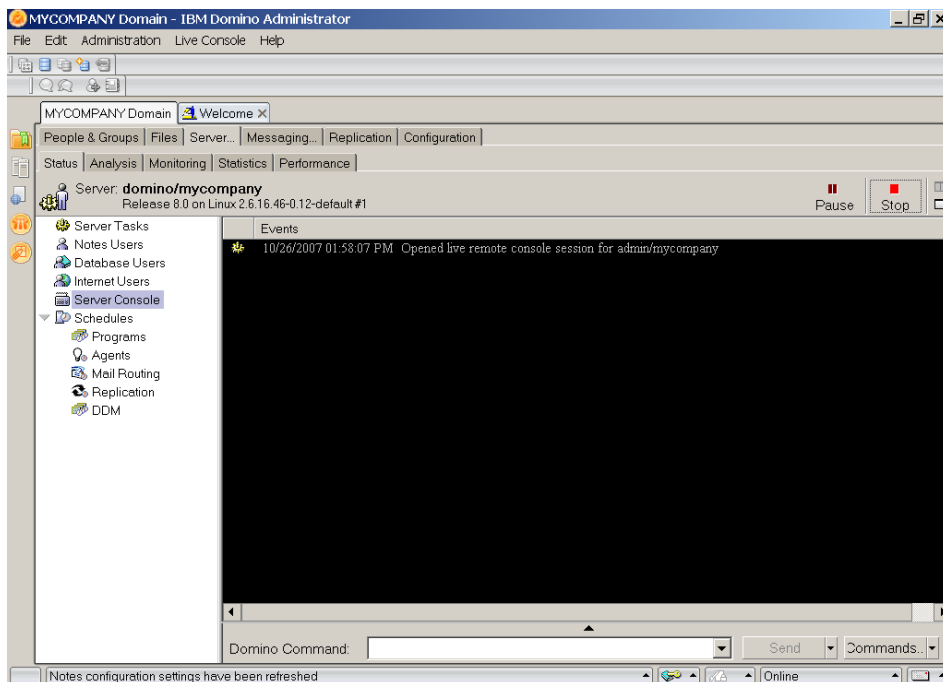
6. Type following command in the **Domino Command:** field.
   ```
   show server
   ```
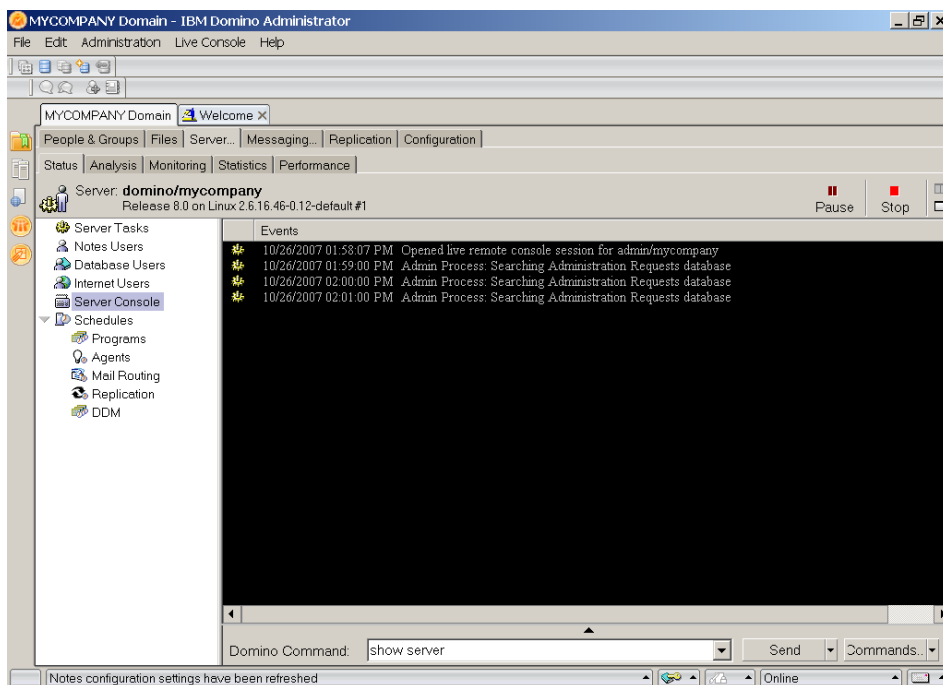

   Click **Send**

7. You should see information about the server scroll by.
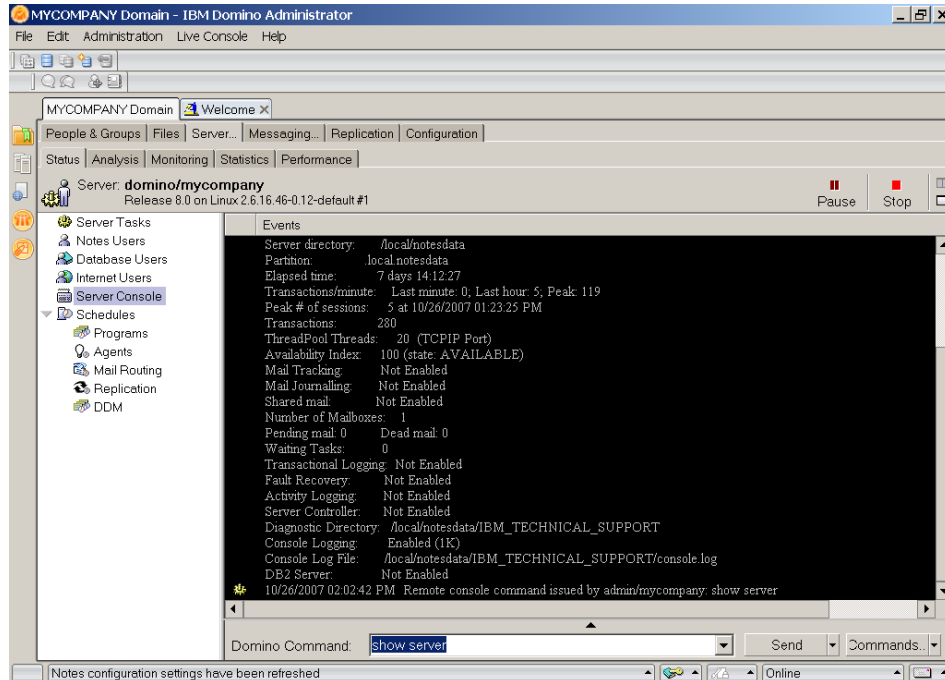
You have successfully installed and configured Lotus Domino Administrator.

4.3 Copy Agents Required by Server.Load to the Server Under Test

Namagent.nsf is provided by the Server.Load feature it includes the agents required to setup the test environment for running the server.load workloads. These agents must be copied to and run on the Server under test (SUT).

1. Launch Domino Designer.

2. Open Namagent.nsf. From the Menu bar select, **File -> Application -> Open ...** and then Select **Server.Load Setup Agents** (namagent.nsf)

Click **Open**.

3. Open the Domino Directory on the Server Under Test. From the Menu bar select, **File -> Application -> Open ...**, Click the **Look in: Drop down menu** and select the **SUT,** then Select **SUT's Directory** (names.nsf)

4. Navigate to view **Server.Load Setup Agents -> Shared Code -> Agents.** Select all the agents in the right pane and press **Ctrl + C** to copy these agents.

5. Navigate to view **SUT's Directory -> Shared Code -> Agents** and press Ctrl + V to paste the agents into the Server Under Tests Directory (names.nsf).

4.4 Register the Simulated Users on the Domino Server

In the previous section we copied the agents required to run a workload to the Domino Server under test (SUT). In this section we will run the Create NotesBench mail Person Document agent on the SUT to create 1200 Mail Users.

1. Launch Domino Designer.
2. Navigate to SUT's Directory -> Shared Code -> Agents view.
3. Select agent "Create notes Bench Mail Person Documents" then right click and select **Run** from the context menu.
4. You are presented with a series of Dialog boxes. Enter and record your values here.

Dialog	Value
Starting Values to Create Mail Users:	1

Number of Users to Create	1200
Number of Mailn.NSF Files to Use:	1200
Starting Mailn.NSF File	1
Location of Mail Databases (relative to Notes Data directory, blank = root)	mail\
Mail Domain (example: mycompany)	Enter the name of SUT's Domain.
Number of Mail Servers	1
Mail Server #1 (example: domino/mycompany)	Enter the name of the SUT plus its Domain.
Internet Host Name #1?	Enter the fully qualified TCP/IP Hostname of the SUT.
Message Storage Format (0=Notes, 1=Keep in Sender's Format, 2=MIME)	0
Mail System (1=Notes, 6=POP or IMAP)	1
Percentage of These Users with Local Replicas (0 – 100)	0

4.5 Verify Creation of Person Documents

Prior to Running the Mail Initialization work load. Verify that the necessary Person Documents have been created on the Domino Server under test.

1. Launch the **Domino Administrator Client** and log in using the ID of the Domino Administrator for the SUT.

2. Open the **SUT Domain** and navigate to the **People & Groups** tab.

3. Navigate to view: **Domino Directories -> SUT's Directory -> People**

4. Press **Ctrl + A**, to select all the document in this view. You should see 1201 documents selected. The 1200 Mailn.NSF documents you created plus your administrators person document.
    *Note: If you don't see these documents, Press **F9**, to refresh the view. If these documents are still not available and error occurred. Retrace your steps and correct the problem before you continue.*

4.6 Run the Mail8 Initialization Workload

The N8 Mail Initialization is used to prepare for the N8 Mail workload and the N7 Mail workload. This workload creates and populates the mail databases needed to run both workloads. As with all workloads you may collect performance metrics with this workload; but, these will not be meaningful as this workload is only designed to create the mail databases required by the N8 Mail Workload. You should not use the metrics gathered with this workload as a guideline for tuning Domino.

Note: Prior to running any workload ensure that the SUT and the Driver Client systems can communicate, are synchronized with respect to date/time and the admin.id and cert.id are correct.

1. Set notes.ini settings on each client driver that will run Mail8 Initialization Workload.

notes.ini Setting	Description
Domain=mycompany	SUT's Domain

2. Launch the Server.Load Utility. Select **Start->Run** from the menu and enter C:\Program Files\IBM\Lotus\Notes\sload.exe
3. The Lotus Server.Load Graphical User interface opens.
4. In the **Test Type** section, Select **Built-In** and the Select **N8Mail Initialization Workload** from the drop down menu.
5. Click On the **Test Parameters** tab, and complete the following fields.

Field	Value
Number of Users/Threads	0
Script Loop Count	1
Thread Creation Interval	5
Starting Thread No.	1
Test Time Parameter	No time limit
Build Recipient List using Name and Address Book (example: domino/mycompany!!NAMES.NSF)	sut/domain!!names.nsf
Storage test output to	C:\results\n8initworkload1.txt

6. Click on the **Script Variables** tab, and complete the following fields

Value	Variable Name
CN=domino/O=mycompany	MailServer (example: domino/mycompany)
mail\	nb_dbdir
domino\mycompany!!mail8.ntf	MailTemplate
3000	NumMailNotesPerUser
1	BlockSenderOFF
0	ColorProfile

7. Click **File -> Save Setting** to save these settings.
8. Click Execute to open the Server.Load Metrics Window.

9. Determine which metrics you would like to collect. For each metric you would like to collect, Select the **Metric** from the **Script Metrics** or **Server Stat Metrics** pull down list and then click **Add Metric** button. To delete a metric, Select the Metric and then click the Delete Selected Metric button.

10. Verify that the Server to receive Console Commands field matches the SUT. (example: CN=domino/O=mycompany), if it doesn't enter the correct value.

11. Enter the location of the file to store metrics in. (example: c:\results\n8initworkload1metrics.csv)

12. Click **Start Test** to initiate the workload.

13. The N8Mail Initialization Workload may take several hours to complete. To verify databases are being created, log into the SUT and change to the mail directory located in the Domino Data Directory on your server. The default location on Linux is `/local/notesdata/mail`. Issue the command to list the contents of this directory. You should see a series of mailn.NSF where n is a number between 1 and 300.
Note: You should not issue this test for more than 300 users at a time, if the script fails for any reason you will have to clean up the databases already created and restart the test.

14. Once the Initialization Workload completes. Close all open Server Load windows to exit from the program.

15. Repeat Steps 1 – 13, replacing the **Starting Thread No** in step 5 with the next mail box number. (i.e, 301, 601, 901), then change the output file names accordingly until you have created the desired number of databases.

4.7 Gathering Operating System Statistics

The workload included with the Server Load Utility simulate user load and query statistics from the Domino Server Under Test. To gather additional operating systems statistics you must run external performance monitoring tools like nmon on Linux. Prior to running the N8Mail Workload you will need to setup these tools, as they should be run concurrently with the N8Mail Workload.

There are many performance monitoring utilities available for Linux. This section will explain how to use nmon to gather Linux operating system statistics. `nmon` is a free tool designed for AIX and Linux performance specialists to use for monitoring and analyzing performance data, including: CPU utilization, Memory use, Disks I/O rates, transfers, and read/write ratios, Network I/O rates, transfers, and read/write ratio and much more. For more information on nmon see the following IBM developer works article: nmon performance: A free tool to analyze AIX and Linux performance

1. Download the correct version of nmon for your platform: nmon

2. Log on as `root` and extract the nmon executable:
```
unzip nmon_x86_11f.zip
```

3. Ensure that the file permissions are set to executable
```
chmod +x nmon_x86_fedora5
```

4. Start nmon just prior to launching the Domino Server Load Workload and set parameters to gather information so that it completes shortly after your last workload. *NOTE: The example below will*

run for 4 hours and 1 minute collecting data once per minute.

```
./nmon_x86_fedora5 -ft -s 60 -c 251

-f instructs nmon to start in data collection mode. This mode
    launches nmon and a background processes and writes results
    to a file with the following name: hostname_yymmdd_hhmm.nmon

-t include top output

-s interval to capture statistics in seconds (60 = once/minute)

-c number of times to collect data
```

5. The file produced by initiating nmon in data collection mode is a csv file. It can be opened with most spreadsheet editors. You many need to rename the file to have a .csv extension. As an alternative you may use the nmon analyzer tool for analyzing performance data captured using nmon. Nmon analyzer requires Microsoft Excel 2002 or greater; but allows you to: View the data in spreadsheet form, and produce graphs for presentation based on the nmon output. For more information on using nmon analyzer see the following article: nmon analyzer -- A free tool to produce AIX performance reports

4.8 Workload Data Collection

The Workload Data Collection custom script, SHSTAT.SCR allows test data from a group of test drivers and a SUT to be combined into a single data file for analysis. While a workload runs, the Workload Data Collection custom script collects performance data from the SUT. When the workload is complete, the Workload Data Rollup script rolls up the performance data.   These script require a system other than the one that contains the test drivers that put the load on the SUT.

4.8.1 Setup the Workload Data Collection Script

The Workload Data Collection script should be started just prior to starting the first Workload driver and should not stop until all the Workload drivers complete their test. To setup the client that will run for workload data collection perform the following steps.

1. Create a folder to use as a results directory and then set sharing privileges on that folder to allow the client drivers to write to the results directory.

2. On each test driver, map a drive to the Results directory on a system that will collect the data from all drivers.

3. In the Workload Data Collection driver's NOTES.INI include the following settings.

notes.ini Setting	Description

ResultsDirectory=Z:\	Where Z:\ is mapped to the results directory created in step 1
NB_SaveCMDConsole=1	Tells the script to save results to a file. When we run the rollup script we will change this setting to 0

4. Launch Domino Designer.

5. Open Nameagent.nsf. From the Menu bar select, **File -> Application -> Open ...** and then Select **Server.Load Setup Agents** (nameagent.nsf)
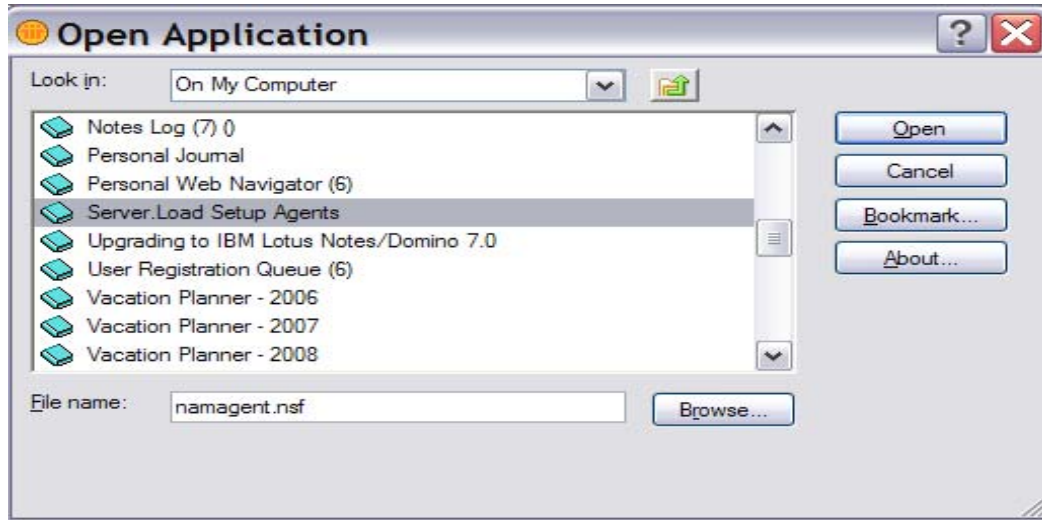
   Click **Open**.

6. Select **Help -> Using This Application** from the Domino Designer Menu.

7. Scroll down to section: **Show Stats script used to collect server data during workload for Workload Data Rollup script,** Right Click on the **showstat.scr** attachment and Select **Save...**.

8. The Save attachment window opens. Navigate to C:\Program Files\IBM\Lotus\Notes and Click **Save**, to save showstat.scr to your notes directory.

9. Launch the Server.Load Utility. Select **Start->Run** from the menu and enter C:\Program Files\IBM\Lotus\Notes\sload.exe

10. The Lotus Server.Load Graphical User interface opens.

11. In the **Test Type** section, Select **Custom** and then Click **Browse**. The Open Window displays, navigate to C:\Program Files\IBM\Lotus\Notes, Select **showstat.scr** and Click **Open**.

12. Click On the **Test Parameters** tab, and complete the following fields.

Field	Value
Number of Users/Threads	1

Script Loop Count	1
Thread Creation Interval	1
Starting Thread No.	1
Test Time Parameter	Specify total test time (in minutes)
Total Test Time (in minutes)	Set to be greater than or equal to the total run of the workload drivers.
Build Recipient List using Name and Address Book (example: domino/mycompany!!NAMES.NSF)	sut/domain!!names.nsf
Storage test output to	Z:\results\datacollection.txt

13. Click **Execute** to access the Metrics window

14. Set the "Server to Receive Console Commands" field in the Metrics window to blank. Server.Load does not need to authenticate with a server for this script.

15. Wait to Click **Start Test**. Finish preparing the workload drivers before continuing. The data collection script should be started just prior to starting the first workload driver, and it should not be stopped until all workloads are completed.

4.9 The N8Mail Workload

The N8 Mail workloads model an active user on an IBM® Lotus® Notes® 8 client acting on an IBM® Lotus® Domino™ 8 server performing the same set of calendaring and mail operations.   To Learn more about this workload see section 2.7.

4.9.1 Prepare to Run the N8Mail Workload

Prior to running this workload you should have run the N8Mail Initialization Workload to create the total number of users that you want to simulate and test. Each Client driver can handle 500 to 1500 simulated users. If you will be testing a larger number of users, you will need to set up additional client driver systems, prior to proceeding. Ensure that you have synchronized the time on all servers and clients before running this workload.

4.9.2 Run the N8Mail Workload

1.   On the SUT, do the following:
   - Start the Calendar Connector task (Calconn).
   - In the Configuration Settings document on the Router/SMTP Basics tab, set the field "Number of mailboxes" to 1 or Higher. *Note: The Server Must be restarted for this change to take effect. I used 2 mailboxes. One mailbox may be enough, it depends upon how many mail users you have and how high the mail waiting statistic gets.*

2. Set notes.ini settings on each client driver that will run the N8Mail Workload.
   ***Note:** MailRecipientBeginNumber and Mail RecipientEndNumber should correspond to the Starting Thread Number and Number of Users/Threads in step 6 below.*

notes.ini Setting	Description
Domain=mycompany	SUT's Domain
NB_PERCENT_ATTACHEMENTS_READ=25	What percentage should be read out of all attachments encountered.
NABENTRIES=1	Number of valid users on the server.
NB_MailRealFromField=1	The Notes ID file.
MailRecipientBeginNumber=1	Beginning range of users from which to choose a random recipient.
MailRecipientEndNumber=1200	End of the range of users from which to choose a random recipient.
ContextIteration1=24	Iteration interval to add a calendar appointment (24 = every 6 hours).
ContextIteration2=24	Iteration interval to send a meeting invitation (24=every 6 hours).
ContextIteration3=24	Iteration interval to respond to a calendar invite (24=every 6 hours).
RUNTIME=9999	How Long over all the entire test should run (9999=forever)
NB_MAIL_SAVE_MESSAGE=1	When sending a memo, save to the Sent folder.

3. Launch the Server.Load Utility. Select **Start->Run** from the menu and enter **C**:\Program **Files\IBM\Lotus\Notes\sload.exe**

4. The Lotus Server.Load Graphical User interface opens.

5. In the **Test Type** section, Select **Built-In** and the Select **N8Mail Workload** from the drop down menu.

6. Click On the **Test Parameters** tab. If you are running the test on multiple clients, increment the value of the Starting Thread No. parameter when you run the test on each client.

Field	Value
Number of Users/Threads	600
Script Loop Count	9999
Thread Creation Interval	1
Starting Thread No.	1
Test Time Parameter	Specify total test time (in minutes)
Total Test Time (in minutes)	240
Build Recipient List using Name and Address Book (example: domino/mycompany!!NAMES.NSF)	sut/domain!!names.nsf
Storage test output to	C:\results\n8workloadDrivern.txt

7. Click on the **Script Variables** tab, and enter the following values for each variable listed.

Value	Variable Name
CN=domino/O=mycompany	MailServer (example: domino/mycompany)
mail\	nb_dbdir
domino\mycompany!!mail8.ntf	MailTemplate
50	NB_N8_NUM_READ_ENTRIES
0	%NUMCALENTRIES%
1	N8_ENABLE_V801_TRANS
2	NB_ReplyNthIteration
1	NB_Memo1NumRecipients
1	NB_Memo1NthIteration
3	NB_Memo2NumRecipients
2	NB_Memo2NthIteration
3	NB_InvtNumRecipients
9999	ScriptIterationLimit

8. Click **File -> Save Setting** to save these settings.

9. Click **Execute** to open the Server.Load Metrics Window.

10. Determine which metrics you would like to collect. For each metric you would like to collect, Select the **Metric** form the **Script Metrics**, or **Server Stat Metrics** pull down list and then click **Add Metric** button. To delete a metric, Select the Metric and then click the Delete Selected Metric button.
    Note: Do not select any additional metrics if you are running this workload in conjunction with the Workload Data Collection and Workload Data Rollup Scripts.

11. Verify that the Server to receive Console Commands field matches the SUT. (example: CN=domino/O=mycompany), if it doesn't enter the correct value.

12. Enter the location of the file to store metrics in, i.e. Z:\results\RES1

Note: If you will be using the workload data collection and rollup scripts these must be stored in files with the following name RESn, where n is the number of the corresponding workload driver.

13. Wait to Click **Start Test** to initiate the workload. Prior to initiating the first workload you should prepare all of your drivers, initiate any operating system statistics collections using tools such as nmon and start the Workload Data Collection test.

4.10 Workload Data Rollup

The Server.Load Workload Data Rollup feature is adapted from the NotesBench data rollup feature and allows test data from a group of test drivers and an SUT to be combined into a single data file for analysis. While a workload runs, the Workload Data Collection custom script, SHSTAT.SCR, collects performance data from the SUT. When the workload is complete, the Workload Data Rollup script rolls up the performance data.

1. In the Workload Data Rollup driver's NOTES.INI include the following settings.

notes.ini Setting	Description
ResultsDirectory=Z:\	Where Z:\ is mapped to the results directory created prior to running the Workload Data Collection Script.
NB_SteadyStateTime=45	Enter the number of minutes that you want to disregard in order to ensure you are getting steady state data. If you do not set this variable, the default of 30 minutes applies.
NB_MeasureTime=60	Enter the number of minutes after steady state that the rollup should read data (length of the real test). If you do not set this variable, the default of 60 minutes is used. If you specify a test length value greater than that which is in the data file, Server.Load stops at the end of the file.
NB_SaveCMDConsole=0	Set this value to zero (0) prior to running the Workload Data Rollup script. *Note: If this value is not set then the default value of 1 is used, and the file will be overwritten during initialization.*
NB_NumOfClients=2	Enter the number of regular drivers in the run. The default is 20
NB_Rollup=1	Enter 1 to enable the Data Rollup Workload to run.

2. Launch the Server.Load Utility. Select **Start->Run** from the menu and enter C:\Program Files\IBM\Lotus\Notes\sload.exe

3. The Lotus Server.Load Graphical User interface opens.

4. In the **Test Type** section, Click **Built-in** and then select "Workload Data Rollup" from the Built-in scripts menu.

5. Click On the **Test Parameters** tab, and complete the following fields.

Field	Value
Number of Users/Threads	1
Script Loop Count	1
Thread Creation Interval	1
Starting Thread No.	1
Test Time Parameter	No time limit

Build Recipient List using Name and Address Book (example: domino/mycompany!!NAMES.NSF)	sut/domain!!names.nsf
Storage test output to	Z:\results\datarollup.txt

6. Click **Execute** to access the Metrics window

7. Set the "Server to Receive Console Commands" field in the Metrics window to blank.Server.Load does not need to authenticate with a server for this script.

8. Click **Start Test**. The data rollup output resides in the datafile **NB_Domstat.csv** containing the averaged client data, and any relevant SH STAT data. This file is location in the ResultsDirectory specified in the notes.ini setting.

5. Graphing and Charting Results

Significant statistics for Domino Server performance include CPU utilization, Disk I/O Operations per second, Network Utilization, Domino Server Response Time, and Domino Server Availability Index. In this chapter we will look at the results of these statistics for a Domino 8.0.1 Server running the N8Mail Workload at loads of 600 and 1200 users on Red Hat Enterprise Linux 5.2.

The main sources for graphing and charting results from the Red Hat Enterprise Linux 5.2 Server Load Test are the Domino Server Load Workload Data Rollup file NB_domstat.csv and operating system statistics collected by nmon in conjunction with the nmon analyzer tool.

Included below is a sample of the first few lines of NB_domstat.csv. As with the Windows testing, Each Client Driver simulated a load of 600 users. The first driver ran for 4 hours and the second driver, which was started 2 hours after the first driver, ran for 2 hours. A Start time of 9:11 corresponds to the initial steady state time. We actually started the N8Mail workload for the first client driver at 8:26 am, and set NB_SteadyStateTime=45 in the notes.ini file prior to running the Workload Data Rollup Script. These time frames are important as they represent the steady state time from which you should take and compare results from the operating system results.

NB_domstat.csv

Client Drivers		1	2
Active Users		600	1200
Transactions per Minute		59	118
Response Time		0.26661	0.182895
Warnings		0	0
Errors		0	0
Failures		0	0
Start Time		09:11:43 AM	11:19:03 AM
End Time		10:11:43 AM	12:19:03 PM
Date		10/10/2008	10/10/2008
Workload_Name			
Server_Name	lin		
Platform.LogicalDisk.1.AssignedName	sda	sda	
Platform.LogicalDisk.1.AvgQueLen		0.01	0.01
Platform.LogicalDisk.1.AvgQueLen.Avg		0.00	0.00

Included below in sample output from running nmon on the RHEL 5.2 based Domino Server. The output file lin_081010_0818.nmon was renamed lin_081010_0818.csv so that my spreadsheet editor would recognize it as a csv file. Notice nmon was started prior to starting the first Domino N8Mail Workload Driver. The data that will correspond to steady state time will correlate to the Start and Stop times in NB_domstat.csv above. To process this data we used nmon analyzer to sort and plot all the information produced by nmon.

lin_081010_0818.csv

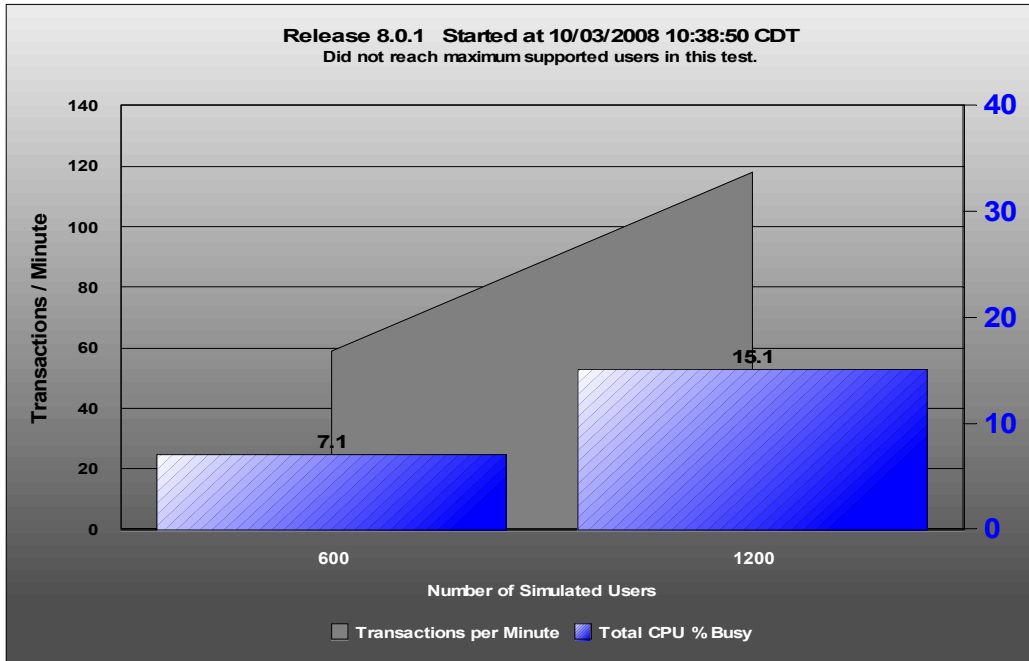| AAA	progname	nmon_x86_fedora5	
AAA	command	./nmon_x86_fedora5 -ft -s 60 -c 300	
AAA	version	11f	
AAA	disks_per_line		150
AAA	max_disks		256
AAA	disks		10
AAA	host	lin	
AAA	user	root	
AAA	OS	Linux	
AAA	runname	lin	
AAA	time		08:18.54
AAA	date		10-Oct-08
AAA	interval		60
AAA	snapshots		300
AAA	cpus		2
AAA	note0	Warning - use the UNIX sort command to order this file before loading into a spreadsheet	
AAA	note1	The First Column is simply to get the output sorted in the right order	
AAA	note2	The T0001-T9999 column is a snapshot number. To work out the actual time; see the ZZZ section at the end	

5.1 Linux Graphs and Charts

Highlighted in this section are some statistics that are significant to Domino Sever Performance. While it is possible to use any spreadsheet editor that recognizes csv format, nmon analyzer has a prerequisite of Microsoft Excel 2002 or greater.

Server Configuration Settings	
Workload_Name	N8Mail
Server_Name	lin
Server.Version.Notes	Release 8.0.1
Server.Time.Start	10/03/2008 10:38:50 CDT
Server.Version.OS	Linux 2.6.18-92.el5PAE #1 SMP Tue Apr 29
Server.CPU.Count	2
Server.MailBoxes	2
Database.Database.BufferPool.Maximum.Megabytes	512
Database.DbCache.MaxEntries	1536
Database.DBUCache.MaxEntries	2560
Database.RM.Sys.Logged	Disabled
Platform.Memory.RAM.TotalMBytes	4050.77
Domino.Config.ActiveThreads.Max	40

Results		
Active Users	600	1200
Transactions per Minute	59	118
Platform.System.PctCombinedCpuUtil	7.10	15.14
Response Time (milliseconds)	266.61	182.90
Platform.System.PctTotalPrivilegedCpuUtil	.96	2.08
Platform.System.PctTotalUserCpuUtil	6.14	13.06
Platform.Network.Total.NetworkBytesPerSec	352835.13	712073.21
Server.AvailabilityIndex	43	31
Server.ExpansionFactor	10.778579	17.649999
Platform.Memory.RAM.AvailMBytes	157.05	155.68
Platform.System.ContextSwitchesPerSec	1590.51	2502.54
Mail.Delivered	70792	100852
Mail.DeliveredSize.100KB_to_1MB	4062	5833
Mail.DeliveredSize.10KB_to_100KB	34586	49255
Mail.DeliveredSize.1KB_to_10KB	26971	38347
Mail.DeliveredSize.1MB_to_10MB	205	305
Database.NSFPool.Used	19749184	25261888
Database.Database.BufferPool.PerCentReadsInBuffer	98.9	98.91

Release 8.0.1   Started at 10/03/2008 10:38:50 CDT
Did not reach maximum supported users in this test.

The following graph was produced by nmon analyzer, It shows disk utilization statistics for the Domino Server. The boxes labeled 600 and 1200 respectively represent the corresponding steady state time when the N8Mail Workload is running with 600 and 1200 simulated users.
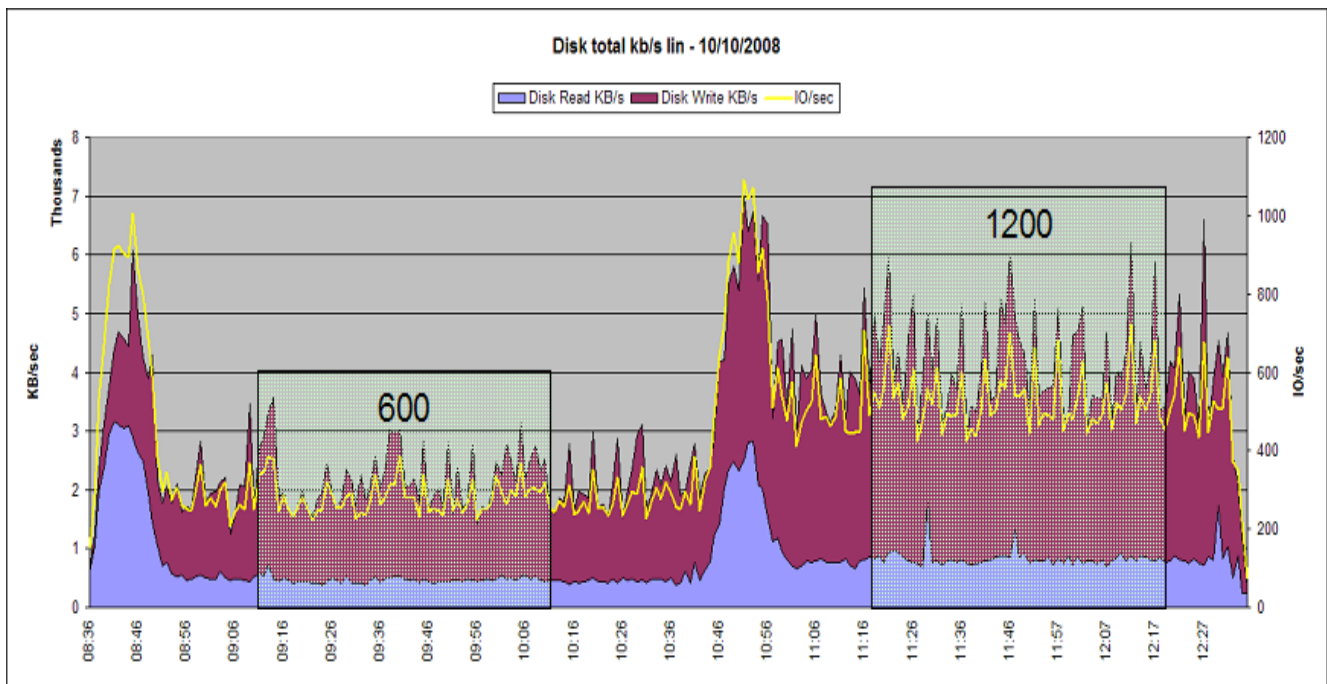
The following two charts present the Network Statistics pulled from nmon analyzer for the corresponding steady state time with 600 and 1200 simulated users.

Network I/O (KB/s)

Avg. WAvg. Max.

600 Simulated Users

Network I/O (KB/s)

Avg. WAvg. Max.

1200 Simulated Users
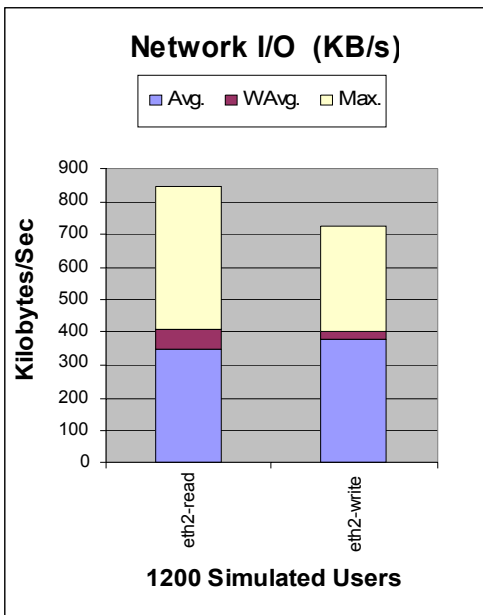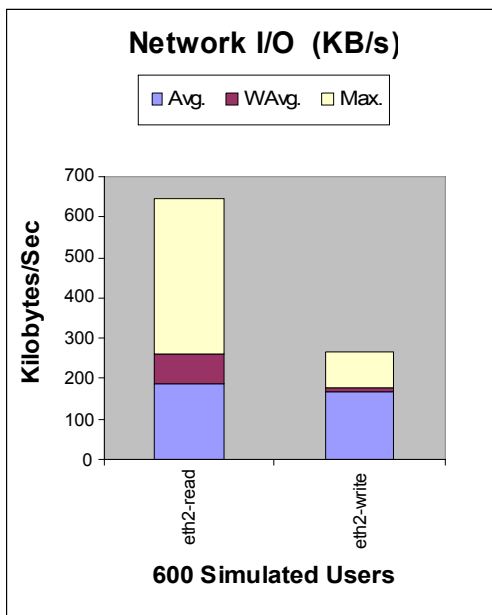
6. Conclusions

The purpose of this paper was to evaluate the performance of Lotus Domino 8.0 Server running on Red Hat Enterprise Linux (RHEL) 5.2.

First, this paper gives a brief introduction to the Lotus Domino performance benchmarking concepts and methodology. Second, it outlines the detailed procedures for running the Server.Load Utility with the N8Mail Workload. Finally, it presents the performance results of Lotus Domino Server on Red Hat Enterprise Linux (RHEL) for simulated workloads of:
- 600 users and
- 1200 users

Server.Load and NotesBench are two load drivers developed for the performance analysis of Domino server systems. They both run on Notes client systems and use workload scripts to control simulated user activity against a server.

NotesBench and Server.Load are both built using the same server performance testing engine that has been expanded over the years to include Internet mail, discussion databases, and calendar and scheduling. Server.Load was created as a simpler, more flexible alternative to NotesBench.

The NotesMark rating, analogous to a TPC-C rating for OLTP workloads, allows you to compare Domino scalability on different hardware platforms. Using NotesBench to get a NotesMark rating requires more setup and a stricter set of guidelines than running workloads with Server.Load to expose system bottlenecks.

However, the fastest way to get started with Domino performance analysis is to use the Server.Load load driver.

In addition to a load driver, one needs a workload that simulates user activity to drive the Domino server, i.e., System Under Test (SUT). One of the ways to achieve a close-to-real-world view of a Lotus Notes user is to closely mimic the API calls by the Lotus Notes V8 clients. While there are older workloads like N7Mail and R6Mail, the N8Mail workload is the latest workload. The N8Mail workload is used to reproduce Lotus Notes V8 client calls. It is a completely new workload with heavier transaction rates. The N8Mail workload can be used to benchmark Lotus Domino V8.0.

For the performance analysis presented in this paper, the following were used:
- Load Driver = Domino Server.Load
- Workload = N8Mail

Results published in this paper are from benchmarks executed in this limited and controlled environment. They do not represent recommendations for a production environment.

For assistance with Domino Capacity Planning it is recommended that you consult your hardware vendor and IBM Techline which handles capacity planning for new hardware.

7. References

1.   **Using Server.Load to Evaluate and Improve Domino Server Performance**
      Maureen P. Gerlofs
      21 October, 2008

2.   **Optimizing server performance: I/O subsystems**
      http://www.ibm.com/developerworks/lotus/library/ls-IO_subsystems/index.html
      Razeyah Stephen, Domino Performance Engineer, Iris Associates
      02 Aug 1999

3.   **Introduction to Domino performance tuning**
      http://www.ibm.com/developerworks/lotus/library/ls-perf_intro/index.html?S_TACT=105AGX13&S_CMP=ART
      Louis Bradbard, Software Engineer, Iris Associates
      01 Sep 2000

4.   **IBM Lotus Notes V8 workloads: Taking performance to a new level**
      http://www.ibm.com/developerworks/lotus/library/notes8-workloads/
      Razeyah Stephen, Manager, IBM
      Joseph Malek, Software Developer, IBM
      Leon Desy, Software Developer, IBM
      18 Sep 2007

5.   IBM Lotus Domino V8 server with the IBM Lotus Notes V8 client: Performance
      http://www.ibm.com/developerworks/lotus/library/ls-IO_subsystems/index.html
      IBM Lotus Notes/Domino Performance Team, Software Performance Analysts, IBM
      18 Jan 2008

6.   **Quick start guide to IBM Lotus Domino Server.Load V8**
      http://www.ibm.com/developerworks/lotus/library/domino8-serverload/index.html
      Daniel Dumouchel, Software Engineer, IBM
      Joseph Malek, Software Developer, IBM
      James Powers, Software Developer, IBM
      12 Dec 2007

7.   **IBM Lotus Domino V8 server with the IBM Lotus Notes V8 client: Performance**
      http://www.ibm.com/developerworks/lotus/library/domino8-performance/
      IBM Lotus Notes/Domino Performance Team, Software Performance Analysts, IBM
      09 ct 2007
      Updated 18 Jan 2008