iNotes Web Access
**performance
analysis**

by Razeyah Stephen
and Rich Buck

Level: Intermediate
Works with: Domino 5.0
Updated: 01/02/2002

Customers are so excited about deploying iNotes Web Access that two members of the Performance team decided to take a short break from their Rnext projects and fill you in on what they've learned about iNotes Web Access performance.

This article examines the peak benchmark numbers with the R5iNotes workload on some of the hardware configurations in the performance lab. The configurations discussed here were based on requests and comments that the Performance team received via e-mail and at conferences.

This article also discusses the load that an R5iNotes user creates for the Domino server, compares that to the load generated by an R5Mail user, and shows what happens when you mix both kinds of users on the same server. Finally, it provides some valuable performance deployment tips.

What is iNotes Web Access?

iNotes Web Access is our next generation Web client for Web-based access to Domino messaging and PIM (Personal Information Management) capabilities. iNotes has an elegant user interface, and it leverages Domino Off-Line Services (DOLS) to provide off-line access to information. iNotes Web Access was shipped in Q2 of 2001 on the Windows 2000, Windows NT SP4, Solaris, pSeries (AIX), iSeries (AS/400), and zSeries (S/390) platforms as part of Domino Release 5.0.8. iNotes Web Access will be supported on Linux and HP-UX in a later release of Domino. For an overview of iNotes Web Access, see the *Iris Today* interview "**Jason Dumont and Vinod Seraphin on iNotes Web Access**."

iNotes Web Access includes a large set of features, including customer's most frequently requested features for HTTP messaging. These features include:
- Support for unread marks
- A rich view management user interface
- The ability to spell check messages
- Unlimited attachments
- Virtual scrolling of views
- Improved calendar views
- Group calendar capabilities
- Rich calendar printing that generates PDF files
- Sametime chat integration
- New mail notification
- Archiving
- Out of Office agent support
- A ToDo Gantt chart view
- Name validation/resolution for Mail and Calendar and Scheduling addressing (in 5.0.9)

Because iNotes Web Access is a browser-based client, it reduces administration and user training costs and is easy to deploy.

Performance features
Specific iNotes Web Access performance features include:
- Server-side memory caching of forms and subforms

- Sharing of design elements across all users on the server
- Careful manipulation of HTTP headers to trigger optimal browser caching
- Compression of all JavaScript code sent to the browser (for example, stripping of comments and reducing the length of identifiers)
- Browser caching of static code in external script files for up to a year
- Browser caching of variable data in external script files for up to a year or until the data changes
- Use of dynamic HTML (DHTML) to minimize the need to reload entire pages and avoid certain page transitions
- Use of cascading style sheets to reduce the size of HTML pages
- Use of URLs to the shared forms file for static code that are the same for all users for more efficient caching by a proxy server

The NotesBench workload for iNotes Web Access

The **NotesBench Consortium** makes available a selection of workloads for vendors to measure Domino server performance on a variety of hardware configurations. These workloads cover a variety of messaging protocols such as IMAP, NRPC, and HTTP. Two of the workloads used by the NotesBench Consortium are R5Mail, which simulates Notes client users, and the new R5iNotes workload, which simulates browser-based users who access their Domino mail via the Web with iNotes Web Access.

The NotesBench R5iNotes workload was introduced in 5.0.8. Here's what the workload entails:
- Every 15 minutes a user will:
  Open the Inbox
  Read the first message
  Delete the first message
  Read the second message
  Read the third message
  Read the fourth message
  Read the fifth message
- On every sixth loop, the user will:
  Create a new message
  Add three recipients
  Submit the message

Note that the workload does not include Calendar and Scheduling.

Internal benchmark results

Below are the internal peak benchmark results on several different hardware configurations as measured in the performance lab. All tests were done using the R5iNotes workload. Keep in mind that these are not official, audited NotesBench numbers. For official published NotesBench reports, you can visit the **NotesBench Consortium** Web site. Also keep in mind that:
- All tests showed less than 1 second response time; that is, our acceptable response time is equal to or less than 1 second. (For a NotesBench officially published report, the acceptable response time is equal to or less than 5 seconds). So if we are adding users in groups of 250 at a time, we use the maximum number of users where the average response time was equal to or less than 1 second. This is more clearly demonstrated in the graphs below.
- The response time values that you see are the time it takes for all the data requested from the server to arrive at the client. This includes all the server processing time and network transmission time, but does not include any time it may take for the client to process and render the data.
- In the configurations presented, the disk subsystem and memory were not bottlenecked, so those statistics were not included.
- In these configurations, we use RAID 0 because we have a limited

number of disk drives at our disposal. In the official NotesBench reports published by the hardware vendors, they used RAID 0 + 1, RAID 5, and so on, for high availability. Please refer to these reports at the **NotesBench Consortium** Web site for useful configuration details.

When reviewing this peak benchmark information, remember that we are *not* comparing or recommending any particular platform. These are peak R5iNotes benchmark results on different hardware systems that are in the lab.

Also, remember that these are benchmark numbers where we are simulating users based on a standard workload, so you should consult your vendor for capacity planning information specific to your configuration and Domino usage. For more information, use the list of URLs at the **end** of this article.

IBM RS/6000 Server S80 running AIX 4.3.3
The first peak benchmark is for an RS/6000 Server S80:

Number of CPUs (400 MHz)	Memory (GB)	Number of Users	Response Time (seconds)	CPU Utilization
12	16	5,850	0.3	95%

Data was on two SSA loops with fifteen 9 GB drives per loop, each configured as one RAID 0 logical unit.
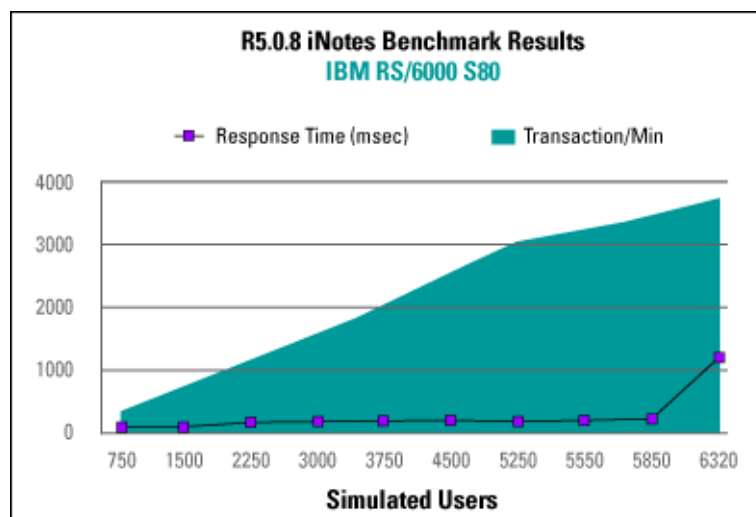

For this test, we had a well-balanced, high-end server with more than enough memory and disk bandwidth to handle the load. You can see from the graph that at around 6,000 users, CPU utilization was maxed out, and response time began to climb.

IBM RS/6000 Server B80 running AIX 4.3.3
This data is for an RS/6000 server with fewer resources:

Number of CPUs (400 MHz)	Memory (GB)	Number of Users	Response Time (seconds)	CPU Utilization
4	4	2,500	0.8	98%

Data was on two SSA loops with fifteen 9 GB drives per loop, each configured as one RAID 0 logical unit.
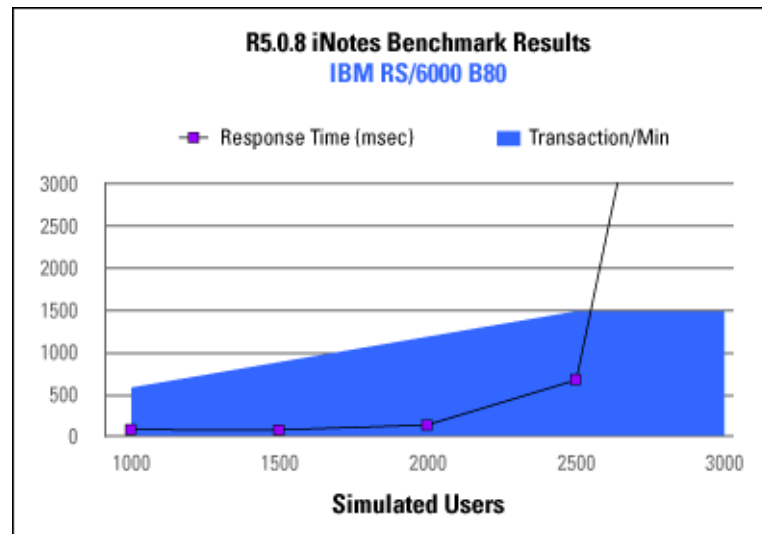
The B80 is a smaller AIX server, and the CPU was maxed out when we attained 2,500 users.

IBM Netfinity Server 8500R running Windows 2000 SP1
The 8500R is a high-end Netfinity server, and its benchmark reflects this:

Number of CPUs (550 MHz)	Memory (GB)	Number of Users	Response Time (seconds)	CPU Utilization
8	2.5	4,500	0.5	90%

Data was on three EXP 200s, each with nine 9 GB drives and configured as one RAID 0 logical unit.
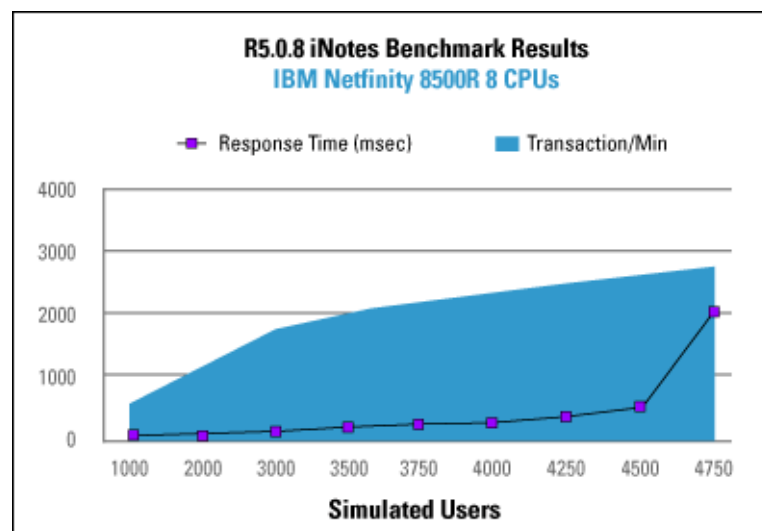
On this high-end server, we were able to attain 4,500 users comfortably with less that 1 second response time. At 4,500 users, the CPU was maxed out
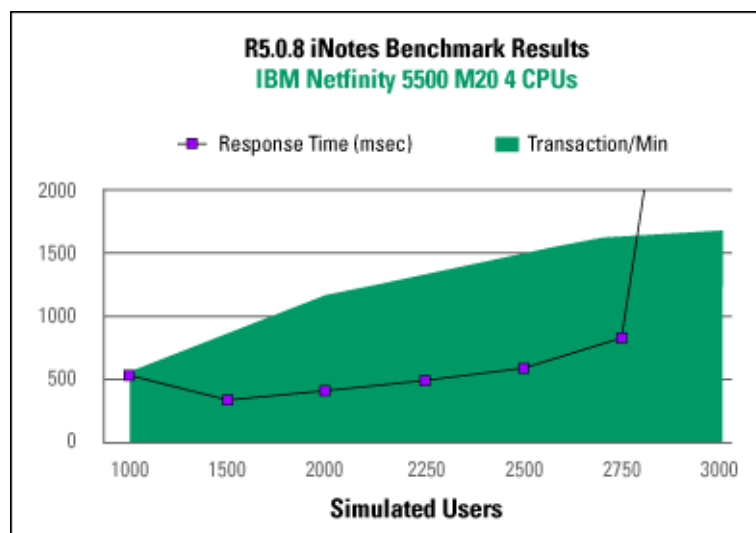
and response time increased.

IBM Netfinity Server 5500 M20 running Windows 2000 SP1
The 5500 M20 is a mid-range Netfinity server, which had these results:

Number of CPUs (500 MHz)	Memory (GB)	Number of Users	Response Time (seconds)	CPU Utilization
4	2.5	2,750	0.8	87%

Data was on two EXP 15s, each with nine 9 GB drives and configured as one RAID 0 logical unit.

On the 5500 M20, we were able to attain 2,750 users comfortably with less that 1 second response time. At 2,750 users, the CPU was maxed out and response time increased.

IBM Server zSeries Model 2064-116 running z/OS V1R1
These results were measured by the IBM zSeries performance team in their lab:

Number of CPUs (2650 total MIPs)	Memory (GB)	Number of Users	Response Time (seconds)	CPU Utilization
16	32	7,900	0.4	59%

The data disks were located on a Shark Enterprise Storage Server. There were two HFSs per volume. Each volume was a disk with 2.4 GB, and each HFS was 1.2 GB. The mail databases were spread evenly in 408 HFSs using 204 volumes.

IBM s/390 Server Turbo G5 running z/OS V1R1
Here are some early results that were measured by the IBM zSeries performance team in their lab. These tests were done on older hardware; we are including these results for customers who have this type of hardware and would like to deploy iNotes Web Access on it:

Number of CPUs (409	Memory (GB) (central	Number of Users	Response Time	CPU Utilization

total MIPs)	storage)		(seconds)	
3	2	1,200	0.2	75%

The data disks were located on a RAMAC Enterprise Storage Server. There were two HFSs per volume. Each volume is a disk with 2.4 GB, and each HFS is 1.2 GB. The mail databases were spread evenly in 50 HFSs using 25 volumes.

Practical variations and tuning configurations

Next, we'd like to share some of the test results we've seen while working with iNotes. We'll focus on the load that an R5iNotes user creates for the Domino server and then compare that to the load generated by the R5Mail user. We'll also show you what happens when you mix both kinds of users on the same server.

All of the results we discuss were achieved by using the NotesBench benchmarking tool to simulate various numbers of R5iNotes or R5Mail users. This allows us to apply a consistent, repeatable load against the server so that we can assess the effects of various hardware and configuration changes. The "users" shown in the following tables are only instances of the benchmark script running and do not necessarily correlate to the numbers of "real users" deployed on a server.

To simplify matters, all of the following numbers were measured while using the same Netfinity 4-way server. This is a mid-range server by today's standards. Also note that this was not a maximum or peak users test but instead was testing on a reasonably loaded server.

Netfinity 5500r M20 Server	
Processors	4x 500 MHz Pentium III Xeon
Memory	2.5 GB
Domino data disks	2x EXP 15 RAID 0 arrays with 9 disks each
Network	100 Mhz full duplex Ethernet
OS	Windows 2000 server
Domino	5.0.8

Domino was installed with the default settings in the NOTES.INI file unless otherwise noted. The one change we generally did make was to set 90 HTTP threads in the Server document; we'll discuss why we did this in the **Tuning considerations** section.

To generate the load against the server, we used a series of 800 MHz PIII drivers, each with 512 MB of memory. This class of driver can easily simulate upwards of 1,000 NotesBench users. For these tests, we used up to 10 of these drivers in a given test.

Observations

To start things off, here is some detail of the server resource usage as we increase the number of active R5iNotes users. This particular server was able to support 2,500 of our NotesBench simulated users with a response time of approximately 1 second.

Benchmark results using the R5iNotes workload					
Users	1,000	1,500	2,000	2,250	2,500
Response	0.5	0.5	0.5	0.6	1.0

time (seconds)					
% CPU busy	27.64	41.09	57.3	67.32	78.87
Available MBs	1,919	1,784	1,591	1,438	1,298
Disk bytes/sec	354,539	362,600	564,780	1,206,454	2,328,928
Network bytes/sec	332,238	410,688	598,482	670,016	753,381

iNotes uses a simple browser as the client, and consequently, a good deal of the processing that had previously been done in the Notes client is shifted to the Domino server. You can see that effect here where CPU increases in proportion to the number of active users and eventually becomes the limiting resource. Memory, network, and disk usage are all relatively low and not a problem on this server.

To put these numbers in context, here are some results on the same server, but with our R5Mail workload. This workload is very similar to the R5iNotes workload, but adds in some Calendar and Scheduling events. As expected, when dealing with the high-function Notes client, the number of users the Domino server can support is much higher. In this case, we see 6,000 R5Mail users with sub-second response time.

Benchmark results using the R5Mail NotesBench workload						
Users	1,000	2,000	3,000	4,000	5,000	6000
Response time (seconds)	0.1	0.1	0.1	0.1	0.2	0.3
% CPU busy	5.9	12.04	18.65	25.63	33.39	41.79
Available MBs	1,310	1,202	1,117	995	859	691
Disk bytes/sec	1,096,268	2,875,250	4,622,897	6,343,839	8,271,697	10,546,564
Network bytes/sec	115,319	239,785	358,773	478,997	557,688	667,086

From a server's perspective, increased CPU demand is the primary characteristic of an R5iNotes user when compared to an R5Mail user. Also, we see about double the amount of network traffic because iNotes Web Access needs to send more than just the raw data to the client; it needs to send the JavaScript necessary to display the data nicely in the browser. In addition, the data sent to the browser needs to be ASCII text, which is not as an efficient method as sending binary data (which the Notes client is able to use).

On the positive side, there is a reduction in the total memory usage when the clients are R5iNotes, and there is a reduction in the amount of data read from disk as well because iNotes Web Access users share common design elements and the Domino server is able to cache these more efficiently.

What should you plan for when you add iNotes Web Access users to a server that already has R5 mail users? Mainly additional CPU capacity and network bandwidth. The next tables show the server response to a workload that was a combination of both R5iNotes and R5Mail users. For the first test,

25 percent of the total users were running the R5iNotes workload, while the second test has 50 percent of the users running the R5iNotes workload and the remaining, the R5Mail workload.

Benchmark results for a mixed workload with 25% R5iNotes users and 75% R5Mail users					
Users	1,000	2,000	3,000	4,000	5,000
Response time (seconds)	0.1	0.1	0.2	0.3	0.5
R5iNotes users response time (seconds)	0.3	0.4	0.5	0.6	1.2
R5Mail users response time (seconds)	0.1	0.1	0.1	0.1	0.2
% CPU busy	11.53	24.98	37.67	55.63	71.81
Available MBs	1,420	1,159	1,018	876	741
Disk bytes/sec	1,000,064	2,693,818	4,561,753	6,634,726	8,374,760
Network bytes/sec	160,732	319,892	460,958	607,691	733,699

You can compare the data from these mixed workload runs to the R5Mail data above. For the 25 percent R5iNotes mix, looking at the 4,000 user case, we see that CPU usage has roughly doubled from 26 percent to 56 percent busy. We also see growth in the network utilization from 478,997 to 607,691 bytes per second. Also, with the increase in demand for server resources, we see that the maximum number of users supported (with approximately 1 second response time) would be around 5,000 users.

Benchmark results for a mixed workload with 50% R5iNotes users and 50% R5Mail users				
Users	1,000	2,000	3,000	4,000
Response time (seconds)	0.2	0.2	0.3	0.5
R5iNotes users response time (seconds)	0.3	0.4	0.5	0.9
R5Mail users response time (seconds)	0.1	0.1	0.1	0.1
% CPU busy	17	35.86	57.5	81.95
Available MBs	1,575	1,223	1,114	1,010
Disk bytes/sec	808,688	2,045,142	4,129,333	5,598,383
Network bytes/sec	206,385	371,177	560,131	755,129

Increasing the percentage of R5iNotes users in the mix to 50 percent, we now see the CPU at 82 percent busy in the 4,000 user case, and the network usage is up to 755,129 bytes per second. Supported users here would be slightly over 4,000, which is roughly two-thirds of what we saw when the server had purely R5Mail clients. Of course, this ratio is very configuration dependent. In our case, we started with an R5Mail server that had excess CPU capacity and was limited primarily by disk bandwidth.

How much does an R5iNotes user weigh?

Another way at looking at the data is to try and figure out what sorts of resource requirements are placed on the server by each incremental user. This can give us a crude idea of what kind of server capacity would be needed to support a given population of users and also shows the relative weight of the different workloads. If we were to divide the change in server resources by the increase in users for our R5Mail and R5iNotes workloads, we come up with the following:

Incremental, per user server resources used			
Average resources per user for the range	R5iNotes 2,500 to 1,000 users	R5Mail 6,000 to 1,000	Ratio R5iNotes to R5Mail
% CPU busy	0.034	0.009	3.8
Megabytes used	0.41	0.15	2.7
Disk bytes/sec	1,316	2,363	0.56
Network bytes/sec	281	138	2.03

For this specific environment, it seems that adding an R5iNotes user to an existing R5Mail server will require roughly four times the server CPU and three times the memory as a comparable Notes client. In the previous data, when running a purely R5iNotes-based server, we saw that the larger per user memory was offset by a lower initial memory expense. In addition, the increase in traffic on the network will be about double, but the disk usage will only be half.

Keep in mind that these are benchmark results and as such, are meant only as a general guideline for what to expect. In the real world, users do not run benchmarks, so results in production environments will no doubt be different than what we see in the lab.

Tuning considerations

Tuning is all about borrowing something from one part of the system to compensate for a weakness in another. If you have a well-balanced hardware platform, such as the one we used for these tests, there are probably very few changes to the default Domino configuration that you should need to make. Our testing has shown that CPU tends to be what an iNotes Web Access server needs most dearly. Therefore, actions that decrease the need for CPU, where and if you can find them, will be of the most use when tuning the iNotes Web Access server.

We did a series of experiments to see just what impact some of the commonly recommended tuning actions would have with an iNotes Web Access server. The results for 2,000 active R5iNotes users appear in the following table. About the only significant help came from adjusting the number of HTTP threads to the minimum number required for all users to reliably get connections. In our case, 90 threads seemed about right. With 40 HTTP threads and our workload, many of the users failed to get connections on the first try; while with 256 HTTP threads, everything ran fine but at an added 7 percent of CPU busy.

Effects of tuning with 2000 active NotesBench simulated R5iNotes users			
	Response time (seconds)	% CPU busy	Available megabytes	Disk bytes/sec
90 HTTP threads	0.5	57.3	1,591	564,780
256 HTTP threads	0.5	64.29	1,537	780,525

40 HTTP threads	0.5	52.95	1,683	608,621
Increase buffer pools 50% (1120)	0.5	56.22	1,529	505,633
Decrease buffer pools 50% (375)	0.5	59.54	1,714	542,716
NSF_DBcache_Max Entries = 3100	0.5	56.13	1,602	555,461
Web authentication on	0.5	61.14	1,595	540,011
All default server tasks enabled	0.5	58.72	1,313	1,142,466

We tried changing the NSF buffer pool size and the number of NSF_DBcache entries, but since our constraint really isn't memory, or disk bandwidth, these had little effect. Turning on Web-style authentication for the users databases caused a 4 percent increase in CPU use. Our previous numbers did not have this enabled.

Normally, for benchmarks, we disable all the server tasks except those absolutely needed for the measurement at hand. This is done to prevent spikes in the numbers, when something like update kicks in, and to show the maximum number possible. For R5iNotes measurements, we would have the router and HTTP tasks running; while for R5Mail, we would run only the router task.

Since we often get asked what the impact of the other tasks is, we've included one last run with all of the default tasks left enabled in the NOTES.INI file. The major impact in this scenario is a large reduction in the amount of available memory, and a consequential increase in disk activity due to less memory being available for disk caching by the operating system.

Summary and useful links

In this article, we shared the performance data we have obtained for iNotes Web Access R5.0.8. We presented the peak numbers achieved from a number of different configurations and gave you some more practical performance information on a mid-range system. You have also seen that very little tuning is required for deployment; the default settings are optimized for performance.

iNotes Web Access uses a simple browser as the client, and consequently, a good deal of the processing that had previously been done in the Notes client has now been shifted to the Domino server. The effect of this is that CPU usage increases in proportion to the number of active users. On the other hand, there is a reduction in the total memory usage when the clients are iNotes Web Access users, and there is a reduction in the amount of data read from disk as well.

For further performance-related information, check out these Web sites:
- **The NotesBench Consortium**
- The **Domino** page of the Lotus Web site
- The **iSeries Domino performance** page of the IBM Web site
- The **zSeries Domino performance** page of the IBM Web site
- The **xSeries Domino performance** page of the IBM Web site

ABOUT THE AUTHORS

Razeyah Stephen is the co-lead on the Domino Performance team. She has worked at Iris since 1998. She came to Iris from Digital Equipment Corporation, now Compaq, where she worked for five years in their StorageWorks division.

Rich Buck has had a long career working on performance within IBM. Rich works on the full spectrum of operating systems supported by Domino, with a current concentration on Solaris.