Notes.net

Iris Today

Putting the Right Spin on Domino Server Performance

Part 1

by Carol Zimmet
and Amy E. Smith

Level: Advanced
Works with: Domino 5.0
Updated: 08/01/2000

The Domino server comprises a complex and comprehensive set of technologies. It is not always easy for administrators to know all the correct options that are available -- the best ones to choose for a given server configuration -- or to appreciate the inter-relationships of Domino components and their effects on server scalability.

The Iris Domino Server Performance Team has become aware of misconceptions and erroneous information being promulgated in the user community about the Domino server. Concurrently, it is becoming increasingly apparent that many Domino user sites have not yet taken advantage of recent new features.

This article, the first in a multi-part series, identifies and clarifies issues and misconceptions with which Domino administrators, users, consultants, and Business Partners are often confronted. The issues discussed in this series have a performance-related slant; that is, their use (or non-use) has a direct impact on Domino server performance or deployment and capacity-planning decisions.

These articles will attempt to set the facts straight and make recommendations as to how to proceed. Special emphasis is placed on those server options that are both widely available to users and provide great performance benefits, but that are not always in widespread use.

Additionally, some philosophical issues around performance are discussed, which should help guide administrators in making future large-scale planning decisions.

Mem.Allocated -- Don't be fooled by the name

On a number of occasions, users have observed problems in their environment, and they use the **Mem.Allocated** server metric to support their case when reporting the problems to Lotus Support. At times, customers have also reported that the value associated with the same metric exceeds the amount of physical RAM on their machine.

Based on its name, it is reasonable to assume that Mem.Allocated (which is not yet documented) reports the total amount of memory that has been allocated by the Domino server at the current time. This assumption is not correct. Mem.Allocated actually reports on the *total amount of memory allocated by the Domino server, based on end user generated demand and on server processes, since the server was started.*

After a Domino server has been up and running for a period of time, the value reported by Mem.Allocated can exceed the amount of physical RAM, because it includes the amounts of physical and virtual memory that have been allocated to Domino. The value is cumulative; it increases as a function of time and activity.

For example:

© Copyright 2000 Iris Associates, Inc.

```
Mem.Allocated = 29,165,024
...
Mem.Free = 27,157,248
Mem.PhysicalRAM = 4,493,312
```

Mem.Allocated has been around since Release 4, and probably earlier. It was put in as a simple method to detect how much memory has been allocated by the Domino server. It is not used for any specific purpose within the Iris end user community and should not be considered as having a meaningful relationship to the NSF Buffer Pool metrics.

Similarly, Mem.Free, which is cumulative over time and shows the memory that has been released, doesn't report what users think. However, in conjunction with Mem.Allocated, Mem.Free can be used as an indicator, as the two metrics should be keeping approximate pace in growth.

We recommend the use of the platform statistic **Platform.Memory.KBFree** to obtain an overall estimate of the amount of memory available on a Domino system. Domino 5.0.2 (and higher) supports the metric on NT and Solaris platforms.

Note: Enable Platform.Memory.KBFree by adding Platform_Statistics_Enabled=1 in NOTES.INI.

The memory statistic **Mem.PhysicalRAM** (also not yet documented), accurately reports the amount of physical memory on the system.

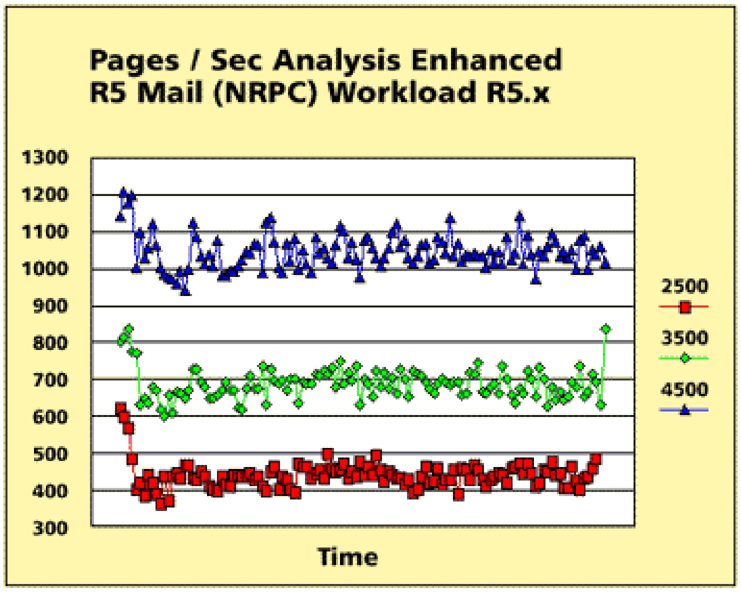
Pages/Sec counter -- What's the right value?

Platform vendors often specify metrics for their systems (such as CPU, memory, disk I/O, and network), as well as information for users about how to evaluate system performance with respect to those metrics. Sometimes, however, vendors underestimate values for these metrics, which can be misleading and administrators think that these are the only correct values for their systems.

One such metric is the Pages/Sec counter, which is a common system (not Domino) metric. **Pages/Sec**, which is part of the memory object, is officially defined as the number of times the Virtual Memory Manager has to page to disk to resolve a memory reference (that is, the number of times the disk was accessed, as current memory didn't have a referenced value on both the read and write paths). Many Domino system administrators take the vendor specification for this at face value, rather than see what their Domino systems can actually handle. Iris has observed performance variations in the Pages/Sec counter, and its value varies significantly from baseline platform recommendations.

Platform vendors often publish a range of 10 - 20 as a reasonable value for Pages/Sec. Actual performance observations for different production Domino servers include values for this metric that are often out of this range. This issue was discussed in detail in an earlier *Iris Today* article **Optimizing server performance: Handling the curves like a pro**. The article includes data to help capacity planners make better decisions. It is important to note that the vendor data analyzed in the article only included that which fell within valid system recommendations: including a workload response time of < 1 second; CPU utilization at 75% or less; and memory available for use at greater than 4 MB. The article stated that, if all other observed metrics fell in the reasonable and good range, it would not be a problem if the Pages/Sec metric exceeded the platform vendors' observations. More importantly, the article emphasizes that a single metric should not be used to evaluate a system's overall performance.

The data used in the "Handling the Curves" article was based on Domino Release 4.6. New data has been included for Domino Release 5, and to reflect different system profiles. See the following charts:

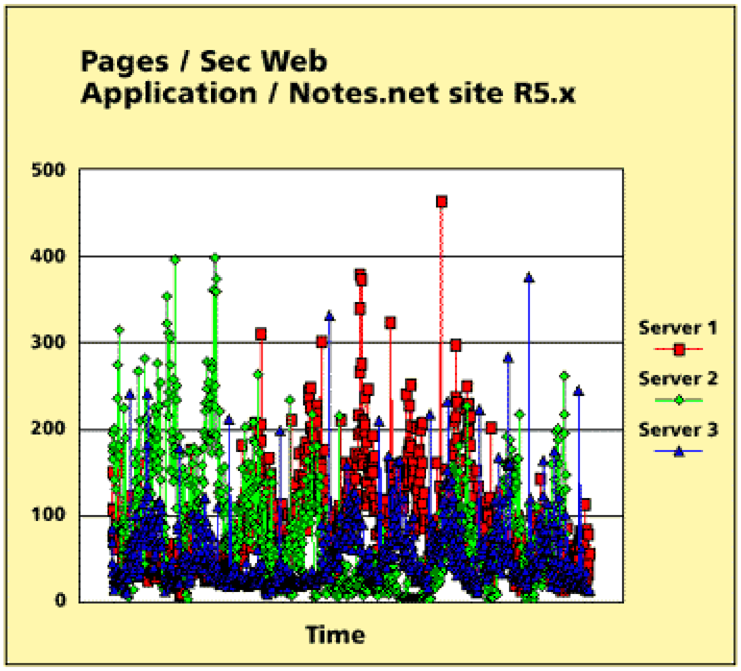


The chart above shows the results of three different benchmark tests; these tests simulated 2500, 3500, and 4500 users, over a 2-hour monitoring period. This workload is an NRPC mail and C&S workload. It should be emphasized that the workloads executed above are different than any of the workloads published to date at the **NotesBench Consortium**; however, it performs the same basic task of sending mail using the NRPC protocol.

Note: These results are included here for illustrative purposes only; they cannot be compared with published results.

For the data points associated with the 2500- and 3500-user workload runs, Pages/Sec is greater than the typical range recommended by the platform vendor. From the platform vendor's viewpoint, it is hard to estimate how different applications will operate on their systems; hence their recommendations tend to be very conservative.

To illustrate how different Domino server profiles exhibit different system utilization, see the following chart, which illustrates how a *production* system operates with a Web application profile (the servers are from Iris' Notes.net site):



The data points illustrate data captured over a few days for some of the Web servers that make up the Notes.net site. They are summarized below, for Pages/Sec:

	Server 1	Server 2	Server 3
avg	89.10	83.49	52.29
max	463.63	397.11	376.27
min	10.00	2.76	10.38

The average Pages/Sec for a Web server is in the < 100 Pages/Sec range for system performance. As this is a production system where data was captured at regular intervals, there is no specific probe turned on to capture the system response time, which due to the volume of activity would be pushing the < 1 second response time. CPU utilization and available memory fall within the ranges described above, as the goals for a production system.

The key point to take away is that workloads have different behaviors. See the *Iris Today* article **Optimizing server performance: Handling the curves like a pro**, which describes several different workloads in terms of their CPU, memory, and Pages/Sec metrics. These metrics can be used by Domino administrators to evaluate their production environments, in terms of the reasonable ranges for those values.

It is important to consider other memory metrics in order to better understand the amount of memory available and to determine if more is needed. We recommend using metrics **Available Bytes** and **Committed Bytes** to further pinpoint if there is a memory bottleneck.

Here's the summary data for the 2500 and 3500 simulated NRPC user run, with those additional metrics:

2500 users			
	Available bytes	**% Committed bytes in use**	**Committed bytes**
avg	2,629,185,433.60	21.48	1,304,857,634.13

max	2,729,472,000.00	21.87	1,328,902,144.00
min	2,601,111,552.00	20.13	1,222,811,648.00

3500 users			
	Available bytes	**% Committed bytes in use**	**Committed bytes**
avg	2,521,133,563.77	23.24	1,412,367,216.13
max	2,605,039,616.00	23.61	1,434,456,064.00
min	2,493,661,184.00	21.89	1,329,868,800.00

The metric Available Bytes refers to the amount of physical memory that is available to processes running on the computer. More specifically, this is the amount of memory remaining after the working sets of running processes and the cache have been served. Virtual memory is comprised of the real memory plus the paging space. If this value remains consistently under 4 MB (according to the vendors reference), more virtual memory should be allocated (both real memory and the associated paging space).

Note: For server configurations with more memory demands, threshold value rises to the 10 MB range.

Reviewing the metrics in the chart above, Available Bytes is well within the "acceptable" range. This is the main metric that the Performance Team uses for gauging memory utilization.

Committed Bytes refers to the size of the virtual memory that has been committed for use. Committed memory must have either hard disk storage to back it up, or it must be defined so as to never have to go to disk. Committed Bytes needs to be compared to the Commit Limit, which represents the maximum available memory. The Commit Limit is a constant value (1,781,063 KB) for the system configured above. When the committed bytes value approaches the committed limit, then more serious investigation is warranted.

Benchmark data isn't gospel

The Iris Domino Server Performance Team is often asked to consult with users about resource requirements for a proposed system configuration. While the Performance Team's first priority is to focus on specific feature areas and how they operate on multiple platforms, the results of this work enable the team to contribute information and knowledge to help formulate those requirements, and respond to those requests for information.

The Performance Team spends most of its time working on pre-release code, often in a non-production environment -- this means that much debugging information is included, which in turn means extra processing overhead, not to mention that all layers of the code base are constantly changing.

They generally don't use "typical" system configurations (one reason being is that it is nearly impossible to define what a typical system configuration is). The team also wants to remove the hardware as the limiting factor in Domino server performance.

While they can't always provide numbers for the "big picture" usage of the Domino server, the results of the team's efforts can support some of the "pieces to the puzzle" as well as help identify where the upper boundaries of the product exist.

It cannot be emphasized enough that, when numbers are published or announced from the Iris Domino Server Performance Team, these numbers are to be considered high-level benchmark numbers and, in most cases, cannot be applied as specified directly to a production environment. The team feels strongly enough about this to add disclaimers to their presentations and documentation, stating that benchmark numbers should not be applied to deployment and capacity planning activities. What benchmark numbers **are** useful for is understanding the upper boundaries of Notes and Domino, which are defined through a focused analysis process and an extended evaluation period.

There are a number of differences between a system used for performance analysis and benchmarking, and one deployed in a production environment. The following are typical conditions in which the Domino server performance team runs its benchmarking systems. These conditions are not normally found in most end-user installations. Benchmark servers are generally:

- Not fully loaded with all the supporting system processes. For example, when running a mail test effort, we would not have Adminp task loaded.
- Not always configured as typically done in the production environments, particularly with respect to the disk configuration and RAID architecture options.
- Configured without regard to system cost, meaning that our configuration(s) try to remove the physical configuration and supporting operating system as the bottleneck.
- Won't have supporting software installed, like backup or virus detection software, which is required for live system support. These software components would typically be running simultaneously during select time periods.

A common mistake many administrators make in capacity planning is to extrapolate performance data to the wrong areas. For example, it is common, but incorrect, to make the following generalizations:

- System configurations -- What worked for 512 MB memory should generalize for a 1 GB memory configuration.
- Platforms -- What worked for Solaris/UNIX should work the same way for NT.
- Domino server profiles -- What worked for a NRPC mail workload should work the same way for WebMail workload.

These points would be applied differently by the Performance Teams found at the various vendors sites, where server workloads are tailored to resemble end-user profiles. They are able to present information and recommendations closer to real production environments. Check out the **NotesBench Consortium** for the contact list of various vendors who are willing to assist in sharing their observations and knowledgebase. They also may be able to better supply detailed information more specific to your requirements.

Worker thread model improves scalability -- and so much more

Release 5 for Domino includes a new internal architecture for NRPC connections only (**not** HTTP or IMAP connections), which provides better support for server scalability.

Previous releases (R 4.x and earlier) of the Domino server supported a single thread per session model. This means for every user session established, a dedicated set of resources were allocated at the Domino application and supporting operating system level. As Domino supports more users in the form of session connections, the amount of resources tied up to support those additional users also increases, which in turn requires additional internal threads.

© Copyright 2000 Iris Associates, Inc.

The NOTES.INI parameter SERVER_MAX_SESSIONS gives users the option to "throttle" the number of active sessions. It should be noted that if the session has a low activity rate, those same Domino and operating system resources are tied up as much as for a session with a higher activity rate. Additionally, the resources are "reserved" and can't be used to help out to improve the responsiveness of an actively processed session.

The new architecture introduced in R5 is the "worker thread model" or **Input/Output Completion Ports (IOCP)**. These worker threads are defined to service multiple client sessions via a limited number of threads that execute. Domino ships with a default number of worker threads already configured; administrators do not have to enable it. The default configuration is 40 threads. IOCP is supported on the NT, Solaris, and AIX platforms. Additional improvements were also made for AIX in Domino Release 5.0.3 to better utilize the worker thread model.

Note: Domino Release 5.0.3 contains the fixes necessary to properly leverage the thread pool model on AIX. This Domino release works in conjunction with AIX release 4.3.3. Please verify the correct software patch level is applied on AIX to work with the Domino fix.

It should be noted that disabling the worker thread model causes the server to revert to R4 behavior, that is, the single thread per session model.

The worker thread model can be likened to the cashiers in a grocery store. There are far fewer cashiers than shoppers who need to use them. Shoppers queue up, as needed, to be serviced by a cashier. Sometimes all cashiers are busy; sometimes only a few are servicing customers. Like the number of worker threads, the number of cashiers is constant, regardless of workload. Depending on the number of customers and available cashier stations, more cashiers can be added to service increasing numbers of customers.

Administrators should see improvements in memory utilization in larger-scale and more fully utilized systems with IOCP, when the system is supporting a reasonable user workload. When the Domino server starts up, all worker threads are all allocated to do work. If there is a small number of active users connected, the memory reserved may be higher than that experienced in earlier Domino releases. But the more typical situation, where the Domino server supports an active user community, is that overall memory requirements decrease, as the worker threads model will take up a smaller footprint in system memory than the thread per session model.

Additionally, when addressing larger scale configurations (5000 active NRPC user range), administrators have the ability to set controls that better optimize Domino server performance to service those requests. Two NOTES.INI parameters enable more concurrent work to be completed if the system is heavily utilized:

- SERVER_POOL_TASKS (not yet documented within the Domino Administration Guide) should be adjusted to 100 worker threads.
- SERVER_MAX_CONCURRENT_TRANS should be set to 1000, which enables 1000 Domino server transactions to be processed simultaneously.

All supported platforms can benefit from this feature, and we encourage its use!

Client-level decisions DO have an impact upon server performance

Although this article focuses on server level performance tuning, the server doesn't work in isolation. Activities initiated by the client also have a big impact on server performance.

One example of how client activity affects server performance is the use of the Hide Design option. Hide Design is used in template development as a security measure. It protects the original template design from being seen or changed. Prior to R5, this option was just a bit setting. However, in R5, Hide Design actually encrypts the formulas. This increased security has a performance cost, as the formulas need to be decrypted before they are run. These overhead costs are incurred each time a template is invoked. If a template is heavily utilized, these costs add up rapidly.

Mail.* stats DO relate to multiple MAIL.BOX databases

One frequently-asked question is whether the mail.* stats have changed as a result of the support for multiple MAIL.BOX databases introduced in Domino R5.

To give you some background, the ability to have multiple MAIL.BOX databases is a performance and scalability enhancement introduced in R5 that reduces the contention for a single mail file. Administrators can now configure multiple mailboxes on a single Domino server.

It is assumed, wrongly, that the current set of stats just reports information on the first MAIL.BOX configured (for example., MAIL1.BOX). In reality, the Mail.* stats, including the Mail.Waiting stat, applies to all MAIL.BOX files configured on a Domino server.

When reviewing the mail.* stats, the information represents a report on what is going on for the whole Domino server; delivering the big picture for mail routing and delivery. (There are no Domino stats currently that specifically address the individual MAIL.BOX databases.)

To analyze the MAIL.BOX information at a lower level of granularity, the output from the Show Dbs console command should be reviewed. The output analysis, and associated columns of information, from this command was discussed in the *Iris Today* articles about Semaphores (**Semaphores Part 1** and **Semaphores Part 2**). The articles also include information about how to perform the necessary analysis to determine whether additional MAIL.BOX files are needed on a given server, utilizing the output from the Show Dbs command.

The following console output provides specific information about the access rate and time for multiple MAIL.BOX databases from one of Iris's production servers:

```
> sh dbs
Database   Refs  Mod FDs  LockWaits/ AvgWait #Waiters  MaxWaiters
mail2.box  7     Y   2    223        810     0         1
mail1.box  7     Y   2    71         107     0         2
```

Note that the average wait time is < 1 second. This is still one of the best metrics to monitor on a regular basis to determine mail routing efficiency.

Directory Catalog can't be optimized -- says who?

The Iris Domino Directory team and Lotus Professional Services contributed some helpful suggestions about directory optimization for this article.

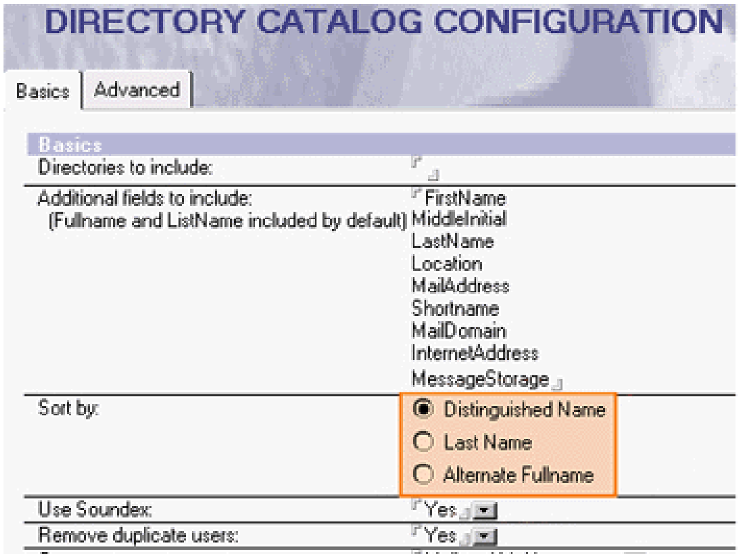
One suggestion addresses a problem that users typically experience when taking advantage of type-ahead addressing. Type-ahead addressing looks up names in a directory catalog, only if the order in which the user types the

© Copyright 2000 Iris Associates, Inc.

name corresponds to the "Sort by" format configured for the directory catalog.

The default "Sort by" option is "Distinguished Name," which means that the type-ahead logic looks up the name in the Directory catalog when a user types in the first name first, then the last name. The default makes sense as most people choose to type in names this way. Another sorting option is by "Last Name," which means the last name must be entered before the first name. This represents an additional way of performing the lookup, and when the option is changed, the expected format for name submission should also change.

The graphic belows highlights the configurations options on the Directory Catalog Configuration document:



DIRECTORY CATALOG CONFIGURATION

Basics

Directories to include:	
Additional fields to include: (Fullname and ListName included by default)	FirstName MiddleInitial LastName Location MailAddress Shortname MailDomain InternetAddress MessageStorage
Sort by:	● Distinguished Name ○ Last Name ○ Alternate Fullname
Use Soundex:	Yes
Remove duplicate users:	Yes

If you subscribe to *The View*, (The Technical Journal for Notes and Domino), check out the November/December 1998 edition. In it, Iris developer Mike O'Brien wrote "The New Domino R5 Directory Catalog: An Administrator's Guide," where he discusses the Sort by field options. He also outlines a method in which you can configure two versions of the Directory Catalog -- one sorted by last name and the other sorted by distinguished name (which means by first name).

You can also get information on setting up the Directory Catalog in **Domino 5 Administration Help** or the Domino R5 *Administering the Domino System* manual.

Note: This configuration option is only available on R5 clients.

No performance gains with transaction logging? Wrong again (but don't use it just for that)

One of the major features introduced in Domino R5 is support for transaction logging and recovery. Transaction logging is defined simply as a solution for reliable data storage. With transaction logging enabled, the Domino server captures all the database updates and writes them to the transaction log, which generally resides on the local server's data store. If a system or media failure occurs, you can use the transaction log and a third-party backup utility to recover your database. Transaction logging is discussed in greater detail in the *Iris Today* article **Optimizing server performance: Transaction**

logging.

Recent conversations with customers have revealed some hesitation among administrators in turning on the transaction logging. Our most recent data for R5 suggests that transaction logging may not help performance when the I/O is low to moderate. However, there is data that indicates that transaction logging may lower response time, % CPU utilization, and % disk utilization when the system has high disk usage. If the system I/O is maxed out, adding more disks and RAID sets may be the only solution.

Increased performance, however, is an incidental benefit. The primary goal behind the transaction logging implementation is to increase the reliability and availability of Domino databases. So, in production environments, where reliability and availability is the top priority and where service level agreements have stringent rules, the feature that contributes the most toward that success should naturally be fully utilized.

Domino administrators have found that another benefit of enabling transaction logging is that it takes much less time to restart a Domino server. This can save a considerable amount of time in larger, multi-server environments, where there are a large number of databases. While this is secondary to the feature's intended benefit, it is nonetheless an added perk that should not be overlooked.

All system memory is utilized by the Domino server? Maybe, maybe not -- what did you make available versus how much it will use

Making assumptions about server system memory has a couple of implications. For planning purposes, administrators need to know that the memory that is available on the system will be utilized by the operating system and the applications running on it.

The Domino server itself makes decisions about memory allocation for buffers and active processes, based on the amount of available memory. Generally, "more is better," but in this case, more is better only up to a point.

For certain operating systems, additional system memory provides no benefit at all, as the operating system can only address up to a certain range of memory. There is an upper limit to the available memory that can be effectively utilized by the operating system kernel. This section provides some guidelines that will enable administrators to plan for the maximum amount of memory that can be effectively used on their systems.

These calculations are based on the premise that the Domino server is a 32-bit application. The maximum amount of memory that a 32-bit operating system can address is 4 GB. This is typically divided into 2 x 2 GB areas; one 2 GB section is allocated to applications and the other 2 GB section is reserved for the operating system.

The use of additional memory is platform-dependent; it also depends on whether the operating system supports 32-bit or 64-bit addressing. For example, some varieties of UNIX can use more memory than 4 GB by running multiple instances of the operating system. 64-bit operating systems, such as Solaris, can address much more memory. Domino R5, while a 32-bit application, is currently supported on Solaris's 64-bit version.

Note: For all the platforms listed below, the analysis is based on a single partition configuration. Information about multiple partition configurations will be covered in a future installment of this series.

Windows NT
Standard Edition
Domino can address up to a 2 GB physical memory limit. If there is additional memory on the system (for example, if it is a 3 GB system), we have observed that Domino can use up to approximately 2.6 GB. In these cases, the .6 GB are used by the operating system, and 2 GB are used by the application.

Enterprise Edition
This version of NT supports the ability for Domino to access 3 GB of physical memory. To enable this, NT Enterprise Edition (NT EE) needs to be booted with the /3 GB switch. This changes how memory is allocated, from 2 GBs for Domino and 2 GB for NT, to 3 GBs for the application and 1 GB for NT. Domino needs to be specially built to take advantage of that option. However, internal tests run by Iris indicate that the extra memory does not provide any performance or scalability benefits for our different workloads. So for all intents and purposes, for the Enterprise Edition of NT, Domino can address up to a 2 GB physical memory limit.

Additionally, NT EE's 192 MB kernel paged memory is a limiting factor. See the Lotus Customer Support Technote #179781, **Domino R5 on NT Returns: "Insufficient System Resources Exist to Complete the Requested Service"** for more information.

Windows 2000
Server Edition
Domino can address up to a 2 GB physical memory limit, just as it can for NT. If there is additional memory on the system (for example, if it is a 3 GB system), we have observed that Domino can use up to approximately 2.6 GB. In these cases, the .6 GB are used by the operating system, and 2 GB are used by the application.

Advanced Server Edition
Domino can address up to a 2 GB physical memory limit, just as it can for NT. If there is additional memory on the system, we have observed that Domino can use up to approximately 2.6 GB. In these cases, the remaining .6 GB are used by the operating system on a 3 GB system. This edition also supports the /3 GB switch, where an application could take advantage of additional physical memory, if specially built. Again, internal tests do not show that extra memory provides any performance or scalability benefits for our different workloads. So, for this version of Windows 2000, at the current time, Domino can address up to a 2 GB physical memory limit.

Data Center Edition
Currently not commercially available.

Solaris/UNIX
Domino can address up to 4 GB of physical memory on the 32-bit version of the operating system. Some of this physical memory is also used by the kernel.

The Solaris kernel can address/use more than 4 GB physical memory (for example, for the file system).

AIX/UNIX
Domino can address up to 4 GB of physical memory on the 32-bit version of the operating system. Some of this physical memory is used by the kernel.

Starting with release versions AIX 4.3.3 and higher, the AIX kernel supports 64-bit addressing, and thus can address up to 64 GB easily. The upper limit for supported memory on AIX will be increased.

For more information on memory

- We've found the article by Mark Russinovich **Inside Win2K Scalability Enhancements Part 2** in *Windows NT Magazine* (December 1999) to be useful in understanding the different options available.
- Check out the **NotesBench Consortium**. The published reports contain information about the memory used in the benchmark efforts.

Conclusions

Factors affecting Domino performance run the gamut from physical hardware restrictions (such as memory) to simply not having certain options enabled (such as transaction logging). Just knowing about these factors isn't enough; it's important for administrators to understand their options and, in some cases, be able to distinguish reality from myth.

The Iris Domino Server Performance Team considers it especially important to clear up misconceptions about Domino, as this often provides immediate benefits to users. Look for future installments of "Putting the Spin on Domino Performance" in *Iris Today*, where the team will continue to pass along their considerable experience and acquired wisdom.

ACKNOWLEDGEMENTS
Special thanks to the various individuals and teams that have contributed their insights and datapoints to make this article more successful. In particular, James Grigsby, Maria Krylova, the Iris Performance Team, Lotus Professional Services Technology Team, the Mail Routing Team, and the Domino Directory team have all made valuable contributions to the points and the content in this article.

ABOUT THE AUTHORS

Carol Zimmet started working at Iris in 1994. She is the co-lead on the Domino Performance Team, and responsible for evaluating performance and performance tool development. Carol continues to search for the one-step solution to everyone's performance problems. She is also interested in a "white box" approach towards improving the quality of the product. Carol enjoys bicycling with her kids and playing racquetball. She has a longing to return to stained glass!

Amy E. Smith is a principal user assistance writer for Lotus. She writes and maintains functional specs for Domino and Notes. She also is a member of the Notes UA Web team.