Notes.net

Iris Today

Command Performance 3: The Domino Performance Team

by
Lynda
Urgotis

Level: Intermediate
Works with: All
Updated: 10/01/2001

What a day! The home team is hot; the opposing team is not. It's the top of the ninth. The pitcher shakes off the signal, then nods. Fast-ball. Outside corner. The last strike of a perfect game. What a performance! Part skill, part art, part luck. Change any one factor in the day's game, and the pitcher could just as easily have been yanked in the fifth inning. So what about your team? Are your servers star performers on their way to a perfect game? Or are you hanging out in the bush leagues with a flaky server that always seems to have problems?

One team that's on the ball is the Domino Performance Team. In this interview, we'll talk with them about their experiences in the world of server performance and tuning. And whether you're a performance rookie or a seasoned veteran, they'll tell you what it takes to turn a promising server into a Cy Young winner.

Let's start with a look at where the Domino Performance Team fits into the organization, and talk a little bit about its goals.
Carol Zimmet
There are actually several performance teams working throughout the company. Some of them line up with QE, and others with Marketing. Our Performance Team is an integral part of the Domino Server Development Team. We're in step with development, with a two-part set of goals. One part of our goals supports performance analysis on the Domino server. Razeyah Stephen heads this effort. The other part is focused on developing and quality engineering review of performance tools, which is my part of the team. And both parts are focused on supporting our users. Who are our users? For our purposes, we think of the "users" as administrators, capacity planners, analysts, and decision-makers.

The performance tool development goals aim to bring the user closer to understanding their server's performance through easier-to-use interfaces and easier-to-understand concepts. After analysis, the "next steps" should be as visible and available to the user as possible so that they know which steps to take to address their system performance and utilization. We want to head off trouble. We don't want you to have to roll out a whole production system before you see that you're in quicksand!

Part of our mission is also performance analysis of the Domino server. We want to identify and remove any obstacles that the Domino server operates under. We want to support more users through a variety of functional paths, with fast response time and low-resource consumption. In a nutshell, we make sure that Domino is not the bottleneck! We want to exploit the latest technologies and developments to Domino's advantage and share with the public how they will also benefit. So as faster processors, more memory, faster disks, faster network connections, and latest platform OS releases appear, we're there in-step, leveraging and exploiting.

Razeyah Stephen
One of the things I love the most about working on Domino is that everyone at Iris believes that performance is important! So we work very closely with the developers, personnel who interact with customers, and product management to ensure we have a very highly performing and scalable

product that meets the needs of our customers.

George Demetriou and Carol Zimmet

It sounds like you cover all the bases for performance.
Razeyah Stephen
The Performance Team is mainly responsible for the performance of the core Domino server. That includes NRPC, HTTP—both messaging and applications—IMAP, POP3, and SMTP protocols in both an Enterprise and ASP configuration. But we keep "performance tabs" on the performance of the total product. We start by defining performance goals for each release of Domino. These goals are based on customer needs and input. Then we work very closely with the developers doing performance analysis, working through each bottleneck until we achieve those goals. We also work very closely with each hardware vendor with the common goal of making Domino high performing.

Carol Zimmet
We also promote understanding of how the Domino server performs. We have a presence throughout the IBM Software Group and with our customers. We present information internally and externally, and we publish information, which you can find in the **Domino Performance Zone** and on **Notes.net**.

Razeyah Stephen, Rama Karedla, and Andy Nolet

What does it take to be part of the Performance Team. Is it broad product perspective? In-depth knowledge about particular areas? Both?
Carol Zimmet
I think we have the most interesting and varied team in the company! We have some former Lotus support people, like Andy Nolet and Jim Powers,

who bring the customer perspective and the customer challenges. We have a device-driver developer, Rama Karedla, which is a job that requires being very sensitive to performance. We have people like Mark Dowdy who's done application work elsewhere and now brings a new perspective to analyzing performance for Web applications. Actually, Mark acts as the champion of Domino application developers, making sure that we include the current points of vulnerability and the well-traveled paths in our workloads.

Other developers have worked closely with performance engineers, doing things like disk driver analysis, or, like Joe Malek, developing capacity planning tools. They appreciate the process and the results, and they know how to leverage the information that's achieved through performance analysis. We also have a great, extended team support network with product management and customer advocates. We even have a former team member who continues to help advance our issues with his benchmarking and tuning efforts on additional platforms. He's considered the "Performance Team, Western Division."

Razeyah Stephen
We have an excellent team of performance analysts and performance toolsmiths who takes pride in making Domino very scalable. The performance engineers have the broad system perspective that allows us to identify and resolve performance bottlenecks at the hardware, operating system, and application levels. The performance toolsmiths are very competent developers, who understand performance. They are the ones that develop the Domino performance tools. You end up with two basic areas of in-depth knowledge—Domino and performance.

Carol Zimmet
There's also an interesting relationship going on within the team, where the performance analysts may develop a technique or validate some new knowledge or advice, and then the performance developers take that advice and incorporate it into a product development effort.

The Domino Performance Team. Standing (left to right): Joey Malek, Mark Dowdy, Andy Nolet, Lou Bradbard, Nirmala Venkatrama. Kneeling: Jim Powers, Rich Buck, Harry Murray. Seated: Rama Karedla, Razeyah Stephen, Carol Zimmet, George Demetriou.

What's a typical day like for the Performance Team?
Razeyah Stephen
As we work toward shipping Rnext, a typical day for me looks like this: I

usually start by checking and answering e-mails from Iris staff who are working closely with beta customers, our hardware vendors, and team members to resolve any urgent performance issues. Then typically, I ensure that the performance engineers have all they need to continue doing Rnext performance analysis so that we can achieve our goals. We currently meet with developers on a biweekly basis to discuss progress, but typically e-mails are flying between the Performance Team and the developers throughout the day. Also, I work on various ways of keeping the field and product management informed of our progress with Rnext Domino performance. In addition, we constantly listen to customers' feedback on the performance of current Domino deployments. Our days are never boring! And we're so fortunate to work with developers who are eager to optimize Domino code.

George Demetriou
I don't believe that there is such a thing as a typical day for anyone on the Performance Team. For example, a performance engineer may spend one day setting up his systems to run a workload, the following day analyzing the results, and subsequent days working with the Domino developers to locate the performance bottlenecks. As a developer in the Performance group, I'm working on performance tools, and I'm not much more likely to have a clear-cut "typical day" than the engineers. As with the development of any product, there are various parts of the process: design, coding, testing, debugging. Depending on the phase of the development cycle, I may concentrate on one of these areas, but it's more likely that I'm involved with more than one, for instance, both coding and testing.

To extend your baseball analogy, you could look at a performance engineer as a star pitcher. Typically, a great starting pitcher takes the mound every five or six days and turns in an outstanding performance. What may not be as appreciated by most people are his pre-game preparations in the days leading up to his start, and his post-game analysis (lessons learned) in the days following his start. Similarly, to ensure a successful performance test, a performance engineer, like Andy "Pedro" Nolet, must be well prepared before launching his test and will always come away from the test with ideas that will be used in future tests. As for the performance tool developers, we may be like the everyday players who provide offensive and defensive support for the pitcher.

Andy Nolet
We currently spend a large percentage of our time on the next release of Notes, so we're trying to uncover every problem to scalability that we can. A "typical" dream day right now would be to have a test that completed five minutes before I got to work. I can then stop the Domino server and OS level data capture, copy all data to my data repository, delete the "used" mail files, and take a quick look to see if the test run was successful or not. If it was successful and could possibly go further, then I recreate the mail files and restart the test. If the test failed, I need to determine what failed and why and make the necessary corrections. For instance, if it was the Notes code that failed, then I have to find out why and what is needed to fix the problem. Or maybe it's something that can be changed via an OS or Notes configuration parameter. Then I would make those changes, reset the test, and start a run.

Once a new run is started, I'm free to analyze the previous night's data and post it to our Notes database for others to see, and to respond to e-mails that have been sent in about performance issues that customers are seeing. Oops—almost forgot the meetings!

Carol Zimmet
Andy, how could you forget the meetings! Meetings help us all stay connected and informed. We have internal meetings to discuss our product development progress. And we have a group meeting every other week to review recent developments, present technical concepts, discuss the latest hot issues, and hand out vegetables.

Hand out vegetables?
Carol Zimmet
One week I walked into the group meeting with two grocery bags. Towards the end of the meeting, someone made a great suggestion. So I handed over an oversized squash and said, "Great idea!" And we got through those two bags of "great ideas" in that one meeting.

From squash, let's move on to the mysteries of benchmarks and workloads. I say "mystery" because most of us don't see and use them, but I know that they're important. Can you enlighten us about creating benchmarks and how you use them?
Andy Nolet
Actually, it's more like generating benchmarks rather than creating them. We define a benchmark as the highest possible number of simulated users that we can attain on a given hardware platform and operating system. Each time we make any benchmark numbers available, it's the culmination of many weeks and many iterations of test runs to find the maximum number of simulated users possible. Our environment is set up to test the Domino code with no other resource restriction. Raz works her magic on vendors, and we have some very high-end test machines. This allows us to run until we exhaust a resource, which is usually the CPU.

We work very closely with the development project teams during each phase of generating benchmark results. For example, to improve the scalability of our IMAP offering in Rnext, for over a year now, we've had biweekly meetings with the members of the IMAP project team. During each meeting, we review the results of all runs from the previous two weeks, platform by platform, to see what kind of numbers we've achieved and how these numbers hold up with different daily builds. Whenever a problem is uncovered, we work directly with the developer to determine the cause of the problem and, more importantly, the solution to the problem. Then we start again to find the next roadblock.

Another example is the way that we worked with the iNotes development team prior to its release in R5.0.8. During the months leading up to shipment of iNotes, we worked on a daily basis with the developers of iNotes trying to find ways to improve the performance and scalability. The level of cooperation between teams is an amazing thing to see and experience.

What about workloads? How do you craft a workload? What are some of the considerations? And how do you know when a workload is successful?
George Demetriou
To borrow one of your adjectives, we'd like a workload to be as close as possible to a modeling a "typical" user. The workload is a script consisting of operations that are executed on a target server. These operations usually cover a 15-minute period and are repeated until the desired workload execution period is attained, usually over several hours.The operations included in the workload or script are what one would expect of a typical user. For example, in a mail workload, we would open a user's mailbox, read several items, delete a few, and generate new mail. Obviously, there's no such thing as a typical user, and we cannot include every possible user operation in the workload; but the objective is to come up with a benchmark—a measurement of performance based on a given load. If you define the success of a workload by whether it sufficiently captures the essence of the typical user, we're always looking to refine the workload by the changes of user activity. For example, between R5 and Rnext, we needed to beef-up our mail workloads to increase the message size and include message attachments, because that is what was happening in the real world.

Carol Zimmet
Workloads are often overlooked as part of the process. People often think

that you just go out and test. Performance analysis does have its testing side, but it also includes the workload definition phase. It's a challenge because it's hard to define and then it's hard to agree on. We base our knowledge on the current product deployment model and industry forecasts as well as our projections of where users will be three years from now. Our projections feed into how the workload performs, and also the quantity, such as the size of the mail item, how frequently a database transaction should execute, or in a Web application, how often a book order is placed.

So, for one of the Web applications that we were working on, we jokingly negotiated how many keyboard keys were being ordered at the fictional Web site over a given period, or what number of black and white keys should be in the "color frequency distribution" for the workload.

We can also have a little tug-of-war with development, where as part of the three-year forecasting effort, we're projecting work behaviors that are different than the current operating procedure. This often means that the workloads are heavier than the current ones, translating to more pressure on the development team. The amount of data being read and written seems to keep on growing. But we've found that enhancing the workloads is not presenting a problem for us right now.

So just like there is a methodology for software development, there's a methodology behind workload development. There are additional steps in the process, but this gives you an idea of the time investment and the basis for finalizing a workload.

What's the tool you use?
Carol Zimmet
We use NotesBench for our performance analysis and knowledge-gathering efforts. We're continuously updating the workloads to include our new clients and features of the product. NotesBench is a standard for Domino benchmarks and is used by all of our platform teams: SUN, Compaq, HP, IBM iSeries, IBM zSeries, IBM xSeries, IBM pSeries, Network Appliance, Hitachi, DELL, Unisys, and BMC.

We also use Server.Load within the team to put a quick workload on a system. We advocate its use within the development organization because it's easier to set-up and get going. Our quality engineering team member, Louis Bradbard, has found a variety of situations where Server.Load can be used to validate and generate a workload on a server.

Do you take requests?
Carol Zimmet
Sometimes we're asked to "just run a test." But before we can run that test, we have to explore the workloads, platforms, and configurations. Now the initial assumption of "three days" turns into a three month analysis.

What's the difference between performance analysis and capacity planning for code development?
Razeyah Stephen
Currently, we're focusing on performance analysis for Rnext, which is making Rnext Domino as scalable as possible without hardware or the operating systems being the limiting factor. In our test environments, we have high-end and mid-range systems configured in Enterprise and ASP configurations using directly attached SCSI, SAN, and NAS disk subsystems.

Capacity planning is the performance that customers would expect on a particular hardware and operating system configuration. We do get some of that information during our performance engineering analysis. But when we are close to shipping a product, we concentrate on getting that information for customers and the field. For example, we just completed some studies for R5.0.8 iNotes with various sized systems, with different configurations like all the Domino tasks enabled, and so on. These configurations are based on customer input. These results will be available on Notes.net shortly. We plan to do the same for Rnext when we ship. We're also constantly working with the hardware vendors to generate capacity planning data for customers.

Performance analysis seems to be part art and part science. What goes into the process? What assumptions do you make about what an analysis effort takes?
Andy Nolet
I've been working with customers on Notes performance-related issues for over six years now and "part art and part science" is a real good way to summarize performance analysis. The art is being able to almost sense what is causing a performance bottleneck, and the science is being able to logically and analytically prove and present it so that others see the same thing. When I was working in Lotus Customer Support handling performance issues at our largest customer sites, people would constantly come to me with "The Notes server slows down. What should I look at to find out why?" My answer was "everything." I try to look at every piece of data that is collected—OS level statistics, Notes statistics, network statistics, and log files. I approach each data collection by looking for "something that just doesn't look right." I call these items "flags." Each flag is noted, along with the time of the flag. Then I start looking for a pattern—cause and effect. The key point is to let the data lead you.

We're trying to quantify each of our test runs now, which is the science part, so that we can present folks with a summarized graph that depicts the behavior before and after "the knee." The "knee" is that point in time when the simulated user experienced greater-than-one-second response time. Our goal for every test is to ramp our simulated users up in such a way that we can measure both Domino busy and OS busy at every step of the way until we find the "knee." We have adopted a method that ramps each client driver up at a rate that is suited to the workload. We allow each driver to settle down after the ramp up, and then run at steady state for a certain amount of time. Again, this will depend on the workload. Then the process is repeated for the next client driver. We affectionately call this the RBuck method, after its biggest proponent on our team, Rich Buck. This way of running the test allows us to focus on the point in time when a problem occurred or identify when and what resource was exhausted. This is the knee of the curve. Additionally, we can answer questions about the behavior of any resource at almost any point in the run.

Rama Karedla
Yes, performance analysis is partly an art, in the sense that being able to identify and rectify the cause of certain system behaviors satisfies both the heart and the mind. But I choose to define success in performance analysis as being a science coupled with experience—lots of it. We're literally inundated with hundreds of statistics that describe the state of the system at any point in time. Collecting and analyzing all the hundreds of statistics is very time-consuming and can lead to "mental indigestion." Zeroing in on the relevant data is what experience is all about. A performance analyst is like an experienced auto mechanic. When you take your car to the garage, the mechanic seems to know exactly where to look. He doesn't go through a list and ask you hundreds of questions.That's why we pay the mechanic big bucks for 15 minutes of his time. Experience guides us as to where to look and to ignore the unrelated variables.

Also many times you may have multiple problem events on a server and a performance analyst has to be able to correlate the right cause with the appropriate event. So a performance analysis effort requires an ability to sift through data asking the right questions, as if one is conducting an experiment and slowly but surely eliminating irrelevant data to focus on and identify the correct causes.

Can you give us an example?
Rama Karedla
We had a situation recently in which the server was dropping packets over a network, while the number of users that should have been constant kept rising. We dived into the 50-odd network statistics suspecting either an intermittently working network card, an improperly configured network, or a bug in the Domino code. Some stats that describe the adapter showed odd behavior, and we were ready to change the card; but we suspected fragmented memory could be an issue. Bingo! We realized the system had no usable memory left. Apparently, the card was fine but being unable to use allocated buffers in a given time period led to time-outs leading to packets being dropped. Again, not being able to access memory in a timely fashion was cited as a possibility for the inability to drop connections when finished. Actually that wasn't the case—further examination led us to conclude that there was a bug in the operating system code that caused the problem.

In addition to having knowledge of the Domino server, a performance analyst must have thorough knowledge of the underlying server, which is both the hardware and the operating system. Metrics that were relevant on one version of an OS may not be relevant in a revised incarnation. For example, we're told that the scan rate, which is one of the indicators of memory usage on UNIX systems, could be of use on Solaris 2.7; but with the implementation of a new page scanning algorithm, we are told it is not as relevant under Solaris 2. 8. Again, we have to be aware of the subtle interaction between the Domino server and the underlying server.

So having said that performance analysis is largely a science, we're putting our efforts where our mouth is. We're trying to algorithmize parts of the performance analysis efforts that can be easily quantified. The result of our initial efforts is the Server Health Monitor.

That's a good segue to the next question. Some people who have a beta build of Rnext may have already seen or read about Server Health Monitoring. What's that all about?
George Demetriou
For Rnext, the Domino Administrator client will extend its server monitoring capabilities to where it can identify "unhealthy" servers, based on analysis of the servers' resource utilization, including CPU, memory, disk, and network. The Server Health Monitor will assign a health rating to each monitored server, and for those servers that are unhealthy, point out the component that has a problem and provide some recommended actions.

Carol Zimmet
One thing that's been fun is watching the deployment process of Server Health Monitoring. We work with various teams within Iris and outside. We want to see if the tool is helping them as much as we envisioned, and we also want to check the accuracy of the "health" of the server.

Here at Iris, we feel strongly that we should "eat our own dog food," meaning that our work is done off servers that are running pre-release code, for both the Rnext and R5 code streams. We've done this with Server Health Monitoring. When a server is not performing well or is having problems connecting, our operating procedure is to point the Server Health Monitoring tool to that server so that we can see what additional details can be gleaned. This helps relieve the pressure of the unknown, as we can better understand what's going on under the covers.

It's also been great working with different IBM teams. As part of their early deployment, they use the product and report back how it helped them. That helps us to verify the numbers observed.

You're also doing performance work with iNotes Web Access, which is a very hot product right now. What's that project like?

Razeyah Stephen
Working on iNotes Web Access has really been so exciting. iNotes Web Access has so many new features that customers have been asking for. The iNotes developers are excellent. They're committed to iNotes Web Access performance goals, and we worked jointly until these goals were achieved.

Andy Nolet
The iNotes project is very exciting for me—it's a new product and new ground. Working with Jeff Jablonowski's team has been great. Everyone has the common goal of providing a reliable, scalable product that suits our customers' needs. We've tried to create a workload that accurately simulates how a "typical" Web user would use the product and then we run the workload until our server runs out of resources, make changes to fix or work around the cause, and then go again.

The workload starts with our best-guess of how someone would use the product and it evolves from there. We start with how we would use the product to get a feel for it and progress to actual customers and beta testers. I feel that there are two kinds of workloads: one that generates benchmark numbers marketing can tout and one that customers can gain from.

The work that we are doing on iNotes is exactly like the other areas of Domino—it involves many iterations of testing/troubleshooting/problem resolving to get to our ship goals.

Carol Zimmet
We were thrilled recently when Jeff gave an internal presentation about iNotes Web Access and he consistently demonstrated that the investment and attention given to performance really paid off in results. What's even better is that the focus on performance will continue through into Rnext. I consider that a vote of confidence and success. To me, it shows a maturing in the product development process, with more attention paid to performance.

How will our customers benefit from your testing?
Razeyah Stephen
Customers will benefit on two fronts. First, we've done in-depth performance analysis to insure that iNotes is very scalable on each platform. Second, we have completed some iNotes capacity planning studies that we'll make available to customers via a Notes.net article. Customers are very interested in iNotes Web Access, so we are trying to provide as much information as possible to customers and the field.

What are you finding out about performance and tuning for a Web application?
Razeyah Stephen
As we find out more and more about Web application tuning, we try to make the defaults as accurate as possible so that customers have to do the least amount of tuning. Nirmala Venkatraman has done a lot of work with the new Domino Web application development models. She developed the Domino application server-based workload using JSPs and servlets for our evaluation. One example we've developed is a Web-based book ordering application that models many of the features found in some of the well-known public Web sites.

How do you figure out how people are using a product like iNotes so that you can translate that into your own testing plans?
Razeyah Stephen
In addition to our own use of the unreleased product, we get as much feedback as possible from customers and from field personnel who work closely with customers and product management.

Does the work on iNotes relate to other areas of Domino?
Razeyah Stephen

Yes, it helps us with HTTP applications and Webmail. Also, if we develop new performance analysis techniques, we implement them for Domino in general.

As the up-and-coming rookie, Rnext must get at least some of your attention.
Carol Zimmet
Rnext gets a lot of our attention! And customers will see benefits from this work for sure. Not only will they understand the limits and ceilings that the product operates under, but they can be guaranteed a certain level of optimization across the product and stability and availability through the performance monitoring evaluation period. We execute tasks and workloads on behalf of our end-users, to guarantee an operating standard.

What's the focus in performance analysis for Rnext?
Razeyah Stephen
A major performance focus for Rnext is HTTP performance. This includes HTTP messaging and applications. We have a re-architected HTTP stack that improves HTTP performance. And we will continue to work on improving iNotes performance. Plus, we're focused on the new Rnext application performance features such as servlets and JSPs, which will provide better performance than R5 Web agents.

Another major focus area for Rnext is IMAP. We're focused on making Rnext IMAP as scalable as the NRPC protocol. This area has been re-architected and re-implemented for Rnext. Core IMAP semantics have been implemented in NSF. A new thread model allows for streaming between NSF and the IMAP client. And MIME storage has been improved and streamlined.

Carol Zimmet
Harry Murray, who works in this area, has good reason to look like the cat who ate the canary, since the progress in this area has been very positive.The goals are aggressive, but there's been very encouraging progress.

Beginning with Beta 2 of Rnext, we've started seeing a lot of new performance metrics for the operating system. What is the value of these new platform statistics? How should people be using them?
Rama Karedla
As operating systems evolve, more statistics that measure the performance of the underlying OS are being provided. For example, Win2K has many more statistics available than NT has. What we've done is to select to display a set of core OS level statistics, out of the hundreds available, that can meaningfully represent the performance of the operating system. This hasn't been an easy job, as every metric tells you something about the behavior of the system and is therefore relevant. R5 has a handful of OS-level stats, known as platform stats, while Rnext has about 65 more. The new stats include information such as the behavior of the network, information about individual disk performance, and more information on the state of memory usage.

All these new statistics are provided for both the system and the Notes administrator so that they can better monitor the performance of the hardware and the OS of the underlying server. Administrators must use these statistics as general indicators of how their system is performing, since system-level characteristics tend to change often. For example, a sudden surge in memory usage or the disk queue length does not imply an impending memory shortage or a bottleneck in disk accesses, since computer usage tends to be very spiky. A constant increase in the value of a metric over a period of time, say a day or a week, should however, alert the user to pay more attention to this metric on a periodic basis. We've given details of a particular stat and its normal values in its definition in the Admin client, making it easier for the user to work with a particular stat and decide future action. For example, if you see that the Network Collision Rate is greater than 2 percent, we suggest that you take a closer look at the network card for possible hardware failures.

Carol Zimmet
Having platform stats working on AIX, which is coming out in the next beta build, is such a great win. Every public presentation that we've made always includes individuals coming from the pSeries systems asking when will platform stats be available. If you're a customer who uses platform stats, please contact us directly. We want to hear your feedback.

Will our customers see some benefit from this work?
Razeyah Stephen
All our customers will benefit from this work because our performance redesigns and optimizations for Rnext were based on input from customers as to their needs.

A year ago, you had access to what was then the ultimate in hardware for performance testing—8-way Intel servers. What's the most impressive machines you are working on today?
Razeyah Stephen
We do have many high-end test systems—8-way Intel servers and 12-way UNIX systems. We're trying to push the performance of Domino as high as possible without the hardware and operating systems being the limiting factor. But we do closely look at Domino resource utilization, and we try to make that as efficient as possible. We have many 4-way Intel and UNIX systems that we do performance analysis work on.

When you run into a problem with testing, how do you figure out what is a hardware limitation and what might be a Domino issue that could be tweaked.
Razeyah Stephen
Our performance engineers really have the "total system perspective," so they can quickly figure out if we're hitting a hardware limitation. That's why we have some high-end systems for testing—this eliminates the hardware as the possible bottleneck in many cases, and we can then concentrate on finding the bottleneck at the OS and Domino levels.

What's been your experience with resource limitations?
Andy Nolet
I mainly run tests on AIX and use a 12-way RS/6000. My development test machine is a 4-way RS/6000. I'm hoping that Raz can come through with a new Regatta machine from IBM for me. It would make a nice Christmas present.

History in the computer industry has a way of repeating itself—first software lagged behind hardware in terms of speed, then software improvements made it faster than the hardware. Since I work for a software development group, I feel that we should stay ahead of hardware. The piece that I think is most important is to always keep the customer in mind. Results that are generated on machines that no one could realistically use are fine to prove where the bottleneck is, and to make sure that we optimize the Notes code to take advantage of the maximum available resources on any hardware platform. But I try to also generate results on a machine that could be considered "typical" at a customer site. Running tests on the RS/6000 4-way adds to my personal challenge of trying to squeeze everything that I can out of the machine, the operating system, and Domino. I can then look for ways to tweak caches, memory pools, Domino NOTES.INI settings, and thread priorities to see what makes differences that customers can use on their systems. The way that we figure out what to change is by looking at everything possible. For example, today a run was finishing and it started generating screens of Timeouts, so I looked at what happened. The files that I used were the Notes Console Log, the Results files (the Debug_Outfile from each client), Notes statistics, our NSD utility, and an OS data capture script. After going through these files, I was able to determine the cause of the problem. The next step is to work with the developer responsible for the area of the code that had the problem, fix the code, and re-run the test to find the next hurdle. Like I said before, this is an iterative process.

Is there a trend towards using more storage area network (SAN) and NAS solutions? What does that mean for your team?
Razeyah Stephen
We do try to test in as many different types of environments possible so as to cover various customer configurations. We currently have configurations with direct attached SCSI, SAN, and NAS. We'll be sharing our test results on these configurations with customers at Lotusphere 2002 and in Notes.net articles.

How much contact do you have with people who are using Notes and Domino outside of the company. In addition to what you're providing to our customers, what are they asking from you?
Andy Nolet
I spent the last few years working directly with customers both on the phone and in person, so I still try to stay in touch with some of our largest customers, especially the ones who drive the performance envelope. My previous job had me on-site at customers sites quite frequently, so there are relationships that I value. Whenever we are discussing new ways to simulate customer users in our workloads, I like to check to make sure we're on the right track—kind of like a sanity check. Many of our long-time Notes users are awesome sources of information, and they usually aren't afraid to tell us exactly what they think.

What's new in tuning cross-platform?
Razeyah Stephen
As we find out more and more about Web application tuning, we try to make the defaults as accurate as possible so that customers have to do the least amount of tuning in their environments.

Carol Zimmet
Actually, we're finding more and more of a trend where Domino is doing its own self-tuning and doing it very successfully. This translates to easier administration and lower total cost of ownership. When we look at the iNotes Web Access results, we're seeing fewer options to explore. That's better from the end-user's viewpoint, but still the instinct is to say "give me something."

We know there are big decisions that need to be made. One great source that we use are the NotesBench reports that are posted on the **NotesBench Consortium** Web site. These postings are great! The individual platform vendors have spent a lot of time and energy optimizing and exploring different options to produce the most optimized solution. They are a help to us, as much as they are a help to our users. The platform vendors learn from each other and build upon the knowledge in succeeding reports and evaluation efforts.

Here are some Web sites to help you get started. This list is not complete but includes some recent efforts from their performance teams.

Compaq	You can start at the **NotesBench Consortium**. For the most recent postings, see the Compaq Proliant 8500/Webmail NotesBench Report.
Sun Microsystems	Start at the **Domino for Sun Solaris** page of the Lotus. Check out the most recent postings at Sun Sunfire 280R/Web NotesBench Report at the **NotesBench Consortium**.
zSeries (s390)	Refer to the IBM Redbook, ***Lotus Domino for***

	S/390 Release 5: Performance Tuning and Capacity.
iSeries (AS400)	Look at this **Lotus Domino iSeries** page and the IBM Redbook, ***Lotus Domino for AS400 R5: Implementation***.
pSeries (AIX)	Check out the **Implementation Guide for Lotus Domino R5 on IBM pSeries** off the **Domino Performance Zone** page of Lotus.com.
xSeries Netfinity	Try the **Benchmarks** page of the IBM Web site and IBM xSeries 370 Server/R5Mail at the **NotesBench Consortium**.
Network Appliance	Check out the **Network Appliance** Web site.

Although we support Unisys, HP, and Linux, there's no recent public information available at this time.

What recommendations can you give to people that would improve their server's performance today? Maybe even right this minute.
George Demetriou
Add memory—it's cheap!! Seriously, I don't believe that there is any one recommendation that can be applied to all situations. There are systems that will reap performance gains with additional memory. Other systems require upgrades on their disk. Still other systems are running with slow CPUs. There are systems that require a combination of these upgrades.

So the dream of a "one-step" solution to all your performance woes remains elusive. Any other recommendations that might help?
Carol Zimmet
My first recommendation is to get a handle on how your system is performing, beyond just avoiding irate users complaining about performance. You should know the operating system characteristics of your system. Once you have a baseline, you can understand when and how things are changing. You'll have facts to deal with the next batch of irate phone calls.

The second recommendation is to size that server correctly. You want to plan for current usage and then project for growth. Use the NotesBench reports. The capacity planning tools supported by the different vendors are also very helpful.

Unfortunately, it does take some time to plan and support. But balance it out with the number of users and the mission-critical role it takes within your enterprise. We haven't yet been able to develop a "go_faster=1" NOTES.INI parameter, but if it ever happens, you'll be the first to know. And we have a request for you—if you're conducting performance analysis on your systems today, we want to hear from you.

What are you looking forward to personally after you ship Rnext, and what's ahead for the Performance Team?
Razeyah Stephen
I'm going to try to remember how to relax, and catch up on the things I've been neglecting at home. Here at work, after we ship, we'll complete some Rnext capacity studies. We gather performance requirements from customers on future projects. And we work with development on new designs for improved Domino performance. We look at all our Rnext benchmarks and start designing new benchmarks to meet customers' future needs.

Andy Nolet

All my years in support must have killed some nerve endings—I look forward to working with customers who are running the new code! Hopefully after we ship this release, we start on the next one. We focus on the cause-effect-one-change-at-time tests that most of us are chomping at the bit to do.

Rama Karedla
After we ship, I plan to go mountain climbing and then research how the Internet can be used to teach music. Maybe I'll start an on-line music university.

George Demetriou
I still have Carol's oversized squash; I'm planning to save it for the Rnext release and then single-handedly indulge in the "mother of all stuffed zucchini casseroles." Seriously, after the Rnext release, I believe that I'll continue to work on post-Rnext performance tools, where, as Carol said, we can add and extend features and incorporate user feedback.

Carol Zimmet
Personally, I plan to clear out the stacks of papers, articles, and documents that I've been holding on to as "invaluable"—hopefully via a shredder, rather than by reading. And since I'm known for standing in front of my chair, keying into my laptop, after we ship, I will try and sit down!

We're already talking about work that can be done in the next major release—that's how excited we are about what we're doing, and how important it is. The features can all be built upon and expanded. That's when the user feedback becomes important, so we know how people use our stuff as well as what we should concentrate on. I hope that it's come across in this interview that we look at you, the users, as being partners in defining the future functionality.

ABOUT THE DOMINO PERFORMANCE TEAM
George Demetriou started working at Iris in 1997. His current areas of responsibility include performance tool development, including NotesBench and Server.Load. He also was involved in the effort to improve Webmail performance for Notes R5. George is an avid runner and has completed several marathons.

Rama Karedla, who realized that the planet is not yet ready for telecommuting from mountaintops, came back to the world again and joined Iris in May, 2001. Prior to Iris, he worked at Compaq in the area of advanced development of storage products, I/O performance, and device driver development. Rama is a developer on the Domino Performance Team and currently works on a Domino performance feature called platform stats and a forthcoming product yet to be named. In his spare time, he manages a not-for-profit music school and works on a dream to teach music in real-time, over the Internet.

Andrew Nolet joined the Domino Performance Team on October 1, 2000, after 5 1/2 years in Lotus Customer Support. In LCS he worked with large and small customers on enterprise performance issues. He was one of the charter members of the LCS Engineering Team who, at a moment's notice, could be on a plane to a customer site to resolve their problems. He enjoys anything to do with being outdoors.

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.

Carol Zimmet started working at Iris in 1994. She is the co-lead on the Domino Performance Team and is responsible for evaluating performance and performance tool development. Carol continues to search for the one-step solution to everyone's performance problems. Carol has found her best thinking occurs while jogging in the morning and surprising squirrels and chipmunks in the process. She has a longing to return to stained glass, (OK, so maybe stained glass will happen when Rnext ships!)

ABOUT LYNDA URGOTIS
Lynda Urgotis began her career during the Paleolithic era writing about chipped-stone tools. She has documented her way from Data Resources to PSDI to Software House.

She is rapidly approaching her fourteenth anniversary at Lotus where she contributed to Lotus Improv, SmartSuite, eSuite, and now, Domino Administrator. She wrote the Que book, *Quick Reference for Improv*, as well as chapters for the Improv Que book. Gardening delights her spirit. Her daughter, Megan, and husband, Michael, are just plain delightful.