Domino for IBM eServer xSeries and BladeCenter Sizing and Performance Tuning

Describes server sizing tips and techniques

Quick start to performance, monitoring, and tuning

Provides access to reference material and research

Darren Belford
Darryl Miles

ibm.com/redbooks
Note: Before using this information and the product it supports, read the information in “Notices” on page v.
## Contents

**Notices** ................................................................. .v
**Trademarks** .......................................................... vi

**Preface** ......................................................................... vii
The team that wrote this Redpaper ........................................ vii
Become a published author ................................................ vii
Comments welcome ....................................................... viii

**Chapter 1. Executive summary** ........................................ 1

**Chapter 2. Sizing tips and techniques** ............................... 3
2.1 Sizing IBM eServer xSeries for Domino ............................ 4
  2.1.1 xSeries servers for Lotus Domino ................................ 4
  2.1.2 BladeCenter servers for Lotus Domino ......................... 6
2.2 xSeries and BladeCenter family ........................................ 7
  2.2.1 xSeries ............................................................... 7
  2.2.2 BladeCenter .......................................................... 8
2.3 Standardizing your server hardware ................................. 10
2.4 Domino server roles ..................................................... 11
  2.4.1 Mail servers ........................................................ 11
  2.4.2 Database (or application) servers ............................... 12
  2.4.3 Hub servers ......................................................... 12
  2.4.4 Web servers ........................................................ 13
2.5 NotesBench reports ...................................................... 13
  2.5.1 Server.Load ........................................................ 14
2.6 Server requirements and recommendations ....................... 14
  2.6.1 Server memory .................................................... 15
  2.6.2 Server disk space ................................................ 16
  2.6.3 RAID and disk controllers ....................................... 18
  2.6.4 Processors .......................................................... 19
  2.6.5 Network adapter card ............................................. 19

**Chapter 3. Tuning your operating system on xSeries and BladeCenter for Domino** .............................. 21
3.1 xSeries and BladeCenter system requirements for Domino 6.5 .............................. 22
  3.1.1 Microsoft Windows ................................................ 22
  3.1.2 Linux .................................................................. 22
3.2 Performance basics ...................................................... 23
  3.2.1 Tuning Microsoft Windows for Domino 6.5 ................... 23
  3.2.2 Tuning Linux for Domino 6.5 ................................... 25

**Chapter 4. Tuning Domino performance on xSeries and BladeCenter** ........................................ 27
4.1 Domino tuning ............................................................ 28
4.2 Domino server performance .......................................... 28
  4.2.1 Optimizing Domino task performance ......................... 29
  4.2.2 Performance tuning the HTTP server stack .................... 32
  4.2.3 Network performance (compression) .......................... 38
  4.2.4 Improving mail performance .................................... 40
  4.2.5 Optimizing database performance .............................. 44
  4.2.6 Calendaring and scheduling resource usage .................. 53

© Copyright IBM Corp. 2004. All rights reserved.
4.2.7 Minimizing logging activity ........................................ 53
4.2.8 Improving Agent Manager performance .............................. 55
4.2.9 Managing server sessions .............................................. 56
4.2.10 Controlling user access ................................................. 58
4.2.11 Improving replication performance .................................... 59
4.2.12 Enabling transaction logging .......................................... 60
4.2.13 Using port encryption .................................................. 61
4.2.14 Lotus Domino partitioning ............................................. 62
4.2.15 Balancing workload in a Lotus Domino cluster ....................... 62
4.3 Maintaining Lotus Domino servers ...................................... 63
4.4 Planning for future growth ................................................ 64

Chapter 5. Monitoring Domino performance on xSeries and BladeCenter .......... 65
5.1 System monitoring tools .................................................. 66
5.2 IBM Tivoli Monitoring ..................................................... 67
   5.2.1 IBM Tivoli Analyzer for Lotus Domino ............................. 67
   5.2.2 IBM Tivoli Monitoring for Messaging and Collaboration ............. 69
5.3 Domino 6 Monitoring and Statistics tutorial .............................. 69

Appendix A. Summary: Common configuration problems that affect performance . 71
Appendix B. Protect your server from data loss ................................. 75
Uninterruptible power supply .................................................. 76
Disaster recovery software and virus protection ................................ 76

Appendix C. Domino 6 boundaries ............................................. 77

Related publications .......................................................... 79
Domino for eServer xSeries and BladeCenter Web sites ......................... 79
IBM Redbooks ..................................................................... 79
White papers about Domino for xSeries ...................................... 80
IBM eServer xSeries Web site .................................................. 81
IBM eServer BladeCenter Web site ............................................ 81
Experience reports ............................................................... 81
IBM Lotus Domino Web site .................................................. 81
Lotus Domino developerWorks ................................................ 81
Product installation information ................................................ 82
Getting the Most From Your Domino Directory: Revisited ...................... 82
How to get IBM Redbooks ..................................................... 82
Help from IBM ..................................................................... 82
Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law. INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:
This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM’s application programming interfaces.
Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

AIX®, BladeCenter™, Chipkill™, developerWorks®, Domino®, Domino Designer®, @server®, ibm.com®, iNotes™, Lotus®, Lotus Notes®, Notes®, POWER™, Predictive Failure Analysis®, Redbooks™, Redbooks (logo)™, ServeRAID™, Tivoli Enterprise Console®, Tivoli Enterprise™, Tivoli®, WebSphere®, X-Architecture™, xSeries®

The following terms are trademarks of other companies:

Intel, Intel Inside (logos), MMX, and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, and service names may be trademarks or service marks of others.
Preface

This IBM® Redpaper will guide and assist you in selecting the correct server configuration for your selected Domino® system, whether the system is for Web, application, or mail hosting. This Redpaper provides performance tuning techniques and best practices available for Lotus® Domino 6.5 running on IBM® Eserver® xSeries® and BladeCenter™ systems. We describe the Domino network design, operating system, and disk I/O subsystem, with hints and tips to help you investigate and solve performance bottlenecks on your server.

The team that wrote this Redpaper

This Redpaper was produced by a team of specialists from around the world working at the International Technical Support Organization, Raleigh Center.

Darren Belford is Site Services Manager for EOS Solutions, an IBM Business Partner based in Brisbane, Australia. He has 10 years of IT experience and is a Principal Certified Lotus Professional (PCLP) - Administration. He also holds a Microsoft® Certified Systems Engineer (MCSE) in Microsoft Windows® NT 4 and has a degree in Marketing and Business Management from Griffith University. His areas of expertise are Lotus Domino 6.5 migrations and infrastructure design and planning.

Darryl Miles is an IT Architect with IBM Australia. He has worked at IBM for 10 years and has supported various network operating systems, thin-client solutions, and electronic mail. Darryl is a Citrix Certified Administrator (CCA), Microsoft Certified Systems Engineer (MCSE), Certified Lotus Professional (CLP), Cisco Certified Network Associated (CCNA), Certified Novell Engineer (CNE), and an IBM Professional Server Engineer. He holds an honors degree in Electrical and Computer Systems Engineering from Monash University, Australia. Currently, his main focus is to provide thin-client solutions for customers in Asia Pacific.

Thanks to the following people for their contributions to this project:

Rufus P. Credle, Jeanne Tucker, Tamikia Barrow
International Technical Support Organization, Raleigh Center

Paul Branch, Steve Russell, Anssi Nummi
Redbook SG24-5313 team
IBM Raleigh

Susan Florio, James Grigsby, Razeyah Stephen, Carol Zimmett
IBM Westford

Gregg McKnight, Clement Moy
IBM Raleigh

Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners and/or customers.
Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our papers to be as helpful as possible. Send us your comments about this Redpaper or other Redbooks™ in one of the following ways:

- Use the online Contact us review redbook form found at:
  ibm.com/redbooks
- Send your comments in an Internet note to:
  redbook@us.ibm.com
- Mail your comments to:
  IBM Corporation, International Technical Support Organization
  Dept. HQ7B Building 662
  P.O. Box 12195
  Research Triangle Park, NC 27709-2195
IBM @server xSeries and BladeCenter are an ideal platform for Lotus Domino 6.5. Choosing the appropriate server for the correct Domino task can be a difficult undertaking. This Redpaper guides and assists you in selecting the correct server configuration for your selected Domino system, whether it is for Web, application, or mail hosting.

After you select the correct server, we assist you in ensuring that the maximum performance is gained from both the server operating system and Lotus Domino 6.5.

This Redpaper includes chapters about the following topics:

- An introduction to IBM @server xSeries and BladeCenter servers, including links to benchmarking and research performed on IBM hardware.
- Guidelines and recommendations that will assist in the planning and tuning of Microsoft Windows and Linux operating systems for Lotus Domino 6.5.
- Performance tuning techniques and best practices available for Lotus Domino 6.5 on xSeries and BladeCenter.
- Links to key information sources for more detail on Domino 6, xSeries platform and Domino 6 for xSeries.
- Guidelines and references to monitoring tools available in Domino 6.5.

This information has been collated from various sources to provide best practice guidelines and configurations with both hardware and software. With these guidelines and configurations, you can produce an efficient, stable, and reliable Lotus Domino 6.5 solution on both IBM @server xSeries and BladeCenter.
Sizing tips and techniques

This chapter provides an introduction to IBM @server xSeries and BladeCenter servers. It also details the different functions of Domino 6.5 servers. To assist in selecting the correct hardware for your solution, we provide you with some links to benchmarking and research performed on IBM hardware.

We also provide an analysis of key selection criteria to ensure that you choose the correct servers for the correct Domino 6.5 solution.
2.1 Sizing IBM eServer xSeries for Domino

Selecting the appropriate xSeries server to run Lotus Domino is not always as easy as selecting a file server for your network. Each Domino server can provide a variety of different functions such as electronic mail, database, intranet, or Internet Web server, or combinations of these.

By understanding the functions that your individual Domino servers will perform, you can better identify the hardware configuration that you require. The great news is that Domino Release 6.5 benefits from many performance improvements over its predecessor. Changes to database structure, thread processing, and buffer usage all contribute to an increase in the number of users supported over Release 5 on the same hardware.

The list of enhancements for Domino 6 and later includes:

- Rewriting and optimizing the Domino 6 HTTP stack for performance
- Introducing new Lotus Notes® memory management techniques to make memory allocation faster and to improve memory utilization
- Enhancing NSF database core code for performance
- Minimizing or eliminating semaphore contention in Domino code
- Optimizing the formula/compute engine code

This chapter outlines the xSeries server family of products and helps you confidently build a reliable foundation for your Domino installation. First, we provide a brief introduction to each server. We then look at the key selection criteria that need to be assessed to ensure an effective Domino solution.

We then discuss NotesBench reports and how we can relate these audited performance numbers to your real-world environment to define exactly which server you need. Some recommendations for disk space and memory and other server peripherals follow.

2.1.1 xSeries servers for Lotus Domino

xSeries servers provide IBM worldwide customers with a reliable foundation for their networked businesses by delivering outstanding power, scalability, control, and service. The xSeries family of products helps small, medium, and large companies run Domino in the configuration they need to solve their business problems.

This section outlines the benefits of xSeries servers. Although this information was accurate at the time of this writing, server hardware is updated frequently to use emerging technologies. For the most current compatibility information, refer to the IBM eServer xSeries server Web page at:


xSeries servers provide you with the following features:

- Latest Xeon processor technology
  xSeries servers featuring the Intel® Xeon processor provide outstanding availability and price and performance capabilities that help better manage and provision your IT environment.
  Up to 16 Intel Xeon Processors MP in rack-mountable xSeries 445.
Ultra fast throughput

IBM now offers Ultra320 SCSI hard disk drives (HDDs) with a data transfer rate up to 320 MBps, which is double the current data transfer rate of Ultra160 HDDs. IBM SCSI HDDs help you improve the transaction and cost performance of your xSeries server. No matter what your needs are, IBM provides the scalability in HDD offerings to complement your system.

X-Architecture™

X-Architecture technology is an evolving blueprint for xSeries servers that is drawn from the vast enterprise server heritage of IBM. xSeries engineers take the technologies that have already revolutionized larger IBM systems and bring them to the Intel-based platform. The result is that IBM has developed industry-standard servers designed to provide enterprise-inspired power, scalability, control, and service at an inexpensive price.

The strength of these technologies, and therefore of X-Architecture technology, lies in a research and development foundation that has no equal. For the past eight years, IBM has led the industry in the number of patents acquired. IBM has continuously landed number one benchmarks and received a multitude of product awards. Among those awards are titles won by xSeries servers.

Specific X-Architecture technology incorporated into the xSeries systems is:

- Active PCI (Peripheral Component Interconnect):
  - Hot Add: Adding a new PCI adapter to an empty slot while the system is running.
  - Hot Eject: Removing a PCI adapter from a slot.
- C2T Interconnect (a new cable connection design): A reduction in cable use.
- Chipkill™ memory: Corrects multiple bit errors using off-the-shelf, industry-standard ECC SDRAM memory.
- Predictive Failure Analysis® (PFA): Alerts you before a component actually fails.
- Light Path Diagnostics: In conjunction with the component Lighting Emitting Diodes (LEDs), the Light Path Diagnostics will help correctly identify a failed component.

For more information about X-Architecture, visit the following URL:


Rack mountable

Rack optimization for savings: All xSeries servers are rack mountable. Rack-optimized servers and Universal servers (with rack mounting kits) save valuable floor space and ease maintenance and operation.

Compatibility and performance

Dedicated IBM engineering teams conduct ongoing compatibility testing, certification testing, and performance measurement and optimization to help ensure that xSeries servers, network operating systems, and Lotus Domino run seamlessly together.

Manageability: IBM Director

IBM Director Version 4.1 is the newest release of the industry leading client/server workgroup manager. IBM Director tools provide customers with flexible capabilities to realize maximum system availability and lower IT costs. With IBM Director, IT administrators can view and track the hardware configuration of remote systems in detail and monitor the usage and performance of critical components, such as processors, disks, and memory.
In addition to making available a new version of IBM Director, enhanced extensions to IBM Director will also be available for IBM customers who want advanced capabilities. IBM Director Server Plus Pack consists of five additional tools for advanced server management to help optimize performance and maximize availability. These tools are tightly integrated with IBM Director for consistent management from a single console.

IBM Director features include:

- Self-managing, smart tools: Automated, proactive capabilities that reduce IT costs and maximize uptime.
- Support for non-IBM hardware: Innovative use of industry standards from CIM to SNMP enables heterogeneous hardware management, protecting your existing IT investment.
- Seamless integration: IBM Director protects your investments in other management packages by integrating with and complementing these with more extensive hardware manageability.
- Single-click management GUI: A convenient user interface delivers the ability to drag and drop tasks to specific systems or groups of systems.
- Integrated, centralized SQL database: An internal database makes system-related data available, even when the specific system is not directly available.
- Multiple operating system support: IBM Director smoothly handles a variety of popular operating systems.

For more information, see:


2.1.2 BladeCenter servers for Lotus Domino

BladeCenter and its offerings are positioned as the highest density servers of the IBM Eserver line. They represent a new approach to the deployment of application servers where two-way, SMP-capable Xeon processing, high-availability design, systems management, and easy setup features are combined in an extremely dense package.

This section outlines the benefits of BladeCenter servers. Although this information was accurate at the time of this writing, server hardware is updated frequently to use emerging technologies. For the most current compatibility information, refer to the BladeCenter servers Web page at:


For extreme flexibility and high scalability, the BladeCenter now supports 2-way POWER™ processor-based servers and 4-way Intel processor-based servers (Domino is not currently supported on POWER processor-based servers).

The benefits of BladeCenter servers include:

- BladeCenter performance and high scalability
- Manageability: IBM Director
- Space saving

Slim, hot-swappable blade servers fit in a single chassis like books in a bookshelf, and each is an independent server, with its own processors, memory, storage, network controllers, operating system, and applications. The blade server simply slides into a bay in the chassis and plugs in to a midplane or backplane, sharing power, fans, floppy drives, switches, and ports with other blade servers.
2.2 xSeries and BladeCenter family

The section provides further information about xSeries and BladeCenter servers.

2.2.1 xSeries

xSeries servers featuring the Intel Xeon processor provide outstanding availability and price/performance capabilities that help better manage and provision your IT environment.

Becoming an on demand business means finding new ways to grow, adapt, and respond in a dynamic business environment while managing costs. xSeries systems provide outstanding availability and systems management capabilities that help better manage and provision your computing resources for a responsive IT environment that can grow with your business. Just the capabilities you need in today’s on demand world.

Scalable Domino platforms can reduce the number of servers required to support an entire enterprise. The fewer the number of servers, the lower the complexity and cost of administration and management for a groupware solution and the operating system and procedures that support it.

---

Figure 2-1  xSeries servers
For building a Lotus Domino infrastructure, the following types of xSeries servers are available (Figure 2-1 on page 7):

- **Rack-optimized xSeries servers** help solve the data center space dilemma with outstanding performance and manageability in compact designs.

  Current models include:
  - xSeries 305: 1U rack-optimized chassis - Single Intel Pentium® 4 processor up to 3.06 GHz
  - xSeries 335: 1U rack-optimized chassis - Up to two Intel Xeon processors at up to 3.2 GHz
  - xSeries 343: NEBS Level 3 Compliant 2U - Two Intel Pentium III processors (1.26 GHz)
  - xSeries 345: 2U rack-optimized chassis - Up to two Intel Xeon processors up to 3.2 GHz
  - xSeries 365: 3U rack-dense chassis - Up to four Intel Xeon processors MP at up to 2.8 GHz
  - xSeries 382: 2U rack-optimized chassis - Two Intel Itanium 2 processors standard for compute-intensive 64-bit applications

- **Universal xSeries servers** deliver file, print, application, and mail serving solutions for small businesses or distributed departmental computing.

  Current models include:
  - xSeries 205: Tower with 4U rack capability with optional rack-mount kit - Intel Pentium 4 processor up to 3.06 GHz
  - xSeries 225: Tower with 4U rack capability with optional rack-mount kit - Intel Xeon processors at up to 3.06 GHz
  - xSeries 235: 5U tower with rack capability with optional rack-mount kit - Up to two Intel Xeon processors up to 3.2 GHz
  - xSeries 255: Versatile and simple-to-set-up 7U chassis; available as a rack or tower - Powered by up to four high-performance Intel Xeon Processors MP at up to 2.8 GHz

- **High-performance, scalable xSeries servers**, such as the new xSeries 445 with high-performance Intel Xeon Processors MP, feature outstanding levels of availability, scalability, and transaction performance. And the new x455, the next generation xSeries Intel Itanium 2 server, delivers pay-as-you-grow scalability and investment protection.

  Current Models include:
  - xSeries 445: 4U rack-dense form factor - Up to 16 Intel Xeon Processors MP (two chassis) at up to 2.8GHz; Up to four Intel Xeon processors (one chassis).
  - xSeries 450: Industry-leading 4U density saves data center floor space - Up to four Intel Itanium 2 processors at up to 1.0 GHz for compute-intensive 64-bit applications.
  - xSeries 455: Rack dense 4U form factor saves valuable data center floor space - Up to 16 Intel Itanium 2 processors at up to 1.5 GHz (up to four processors per chassis) for compute-intensive 64-bit applications

For more information about the entire xSeries product line, visit the following Web site:


### 2.2.2 BladeCenter

The BladeCenter and BladeCenter HS20 offerings are positioned as the highest density servers of the IBM @server line. They represent a new approach to the deployment of application servers where two-way, SMP-capable Xeon processing, high-availability design,
systems management, and easy setup features are combined in an extremely dense package.

![BladeCenter systems](image)

Figure 2-2 BladeCenter systems

The BladeCenter 8677-2XX (chassis shown in Figure 2-2) in just 7U of rack space contains up to 14 blade servers and four 1800-watt power supplies to provide the necessary I/O network switching, power, cooling, and control panel information. When compared to most Intel processor-based 1U high servers, the BladeCenter can support twice the amount of processing power in the same rack space.

The BladeCenter and HS20 blades can require less space and power resources because of their high-density design, reduced power requirements, and single environment systems management. This is an extremely important consideration for:

- Large enterprises
- Application service providers
- Scientific and technical computing businesses

The BladeCenter and HS20 blades are an excellent fit for Lotus Domino.

**BladeCenter HS20 performance and high scalability**

Highlights of the performance and scalability of the HS20 (shown in Figure 2-3 on page 10) include:

- One GB memory on 3.20 GHz system and 512 MB memory with 3.06 GHz system with Chipkill ECC support, supporting up to 8 GB maximum.
- Dual GB Ethernet connections with teaming and fail-over support.
- Integrated management processor, monitoring critical components on each blade.
- ATA-100 IDE controller, economical interface for up to two optional 40 GB IDE HDDs with IDE RAID 1 standard for operating system-independent mirroring.
- SCSI expansion connector supports optional storage unit containing an Ultra320 RAID 1 SCSI controller and backplane support for two hot-swap HDDs.

For more information about the entire BladeCenter product line, visit the following Web site:

2.3 Standardizing your server hardware

Using the information and tools available to you, you will choose the appropriate server for each location in your organization. For example, you might have sized all the Domino servers for a large organization using the suggested guidelines. You might have generated a list of sized servers, and this list might include several different models from the xSeries and BladeCenter range, such as x235, x440, x335, and HS20 blade.

However, many organizations standardize on only one or two models, and we recommend that you limit your choice of servers in this way for several reasons:
- Reduced training for administrative and technical support staff
- Fewer part numbers for on-site service replacement part inventory
- Easier ordering and installation for a customer who is implementing a large network in a short period of time
- Easier and quicker diagnosis of hardware and software problems by implementing a standard operating environment of particular models, network cards, BIOS levels, RAM, and disk space levels
- Easier and quicker maintenance because your technicians will have to be familiar with only a subset of the entire range

As an example, let us suppose that your estimates show that you need to install 15 Domino systems, and you chose the following xSeries servers:
- Four xSeries 445
- Eight xSeries 345
- Three xSeries 225
For the reasons given previously, you might decide that the minimum server platform will be the xSeries 345 server; the high-end server model will remain the xSeries 445. This reduces your selection to two server models.

You should also try to standardize the amount of RAM and disk space you have in your servers. This might not be possible in all cases, but try to reduce the variation in the systems you are purchasing because this has many long-term advantages for an enterprise Domino environment.

### 2.4 Domino server roles

When your Domino network grows to a very large size, you will find it easier to manage if certain Domino tasks are executed on dedicated servers. By defining roles for your servers, you determine what functions they perform and thus what tasks they need to execute. For example, if you specify that a Domino server will primarily support Domino Web Access (iNotes™) clients, your Domino server takes on a Web server role.

We describe a number of Domino server roles in the following sections, including some new roles you might not have seen before that are specific to Domino 6. Remember that these are not hard and fast specifications, just common server roles that many customers implement in their Domino environments. Defining the types of users each of your Domino servers will support will also make it easier to answer your capacity planning questions.

All Domino servers will benefit from having balanced performance across their major subsystems: CPU, disk, and network. However, some roles place heavier demand on the CPU and this is indicated in the descriptions that follow. In addition to any specific suggestions made here, you should also look at the performance tuning suggestions described in Chapter 4, “Tuning Domino performance on xSeries and BladeCenter” on page 27.

#### 2.4.1 Mail servers

Mail servers are used in the Domino network to store and route mail messages among users, servers, and applications. Dedicated one or more servers to these responsibilities gives you the following benefits:

- Easier administration of mailboxes and users. This allows improved auditing and simplifies installation procedures.
- Reduced load on the server by minimizing the number of databases that need to be replicated with other servers and by decreasing the amount of database indexing performed.
- Reduction in the amount of mail traffic (by careful organization of users per mail server). Mail traffic delivered to users on the same server will not be routed across the network. This can also allow for immediate mail delivery by eliminating the need to create mail delivery schedules.
- Establishment of a better scalable model for expansion as usage and storage needs increase by providing you with the ability to define more specific requirements, such as response time.
- Improved mail routing performance by adding multiple mail.box databases and tuning of mail router and local delivery threads.
- Improved I/O performance by distribution of mail files across multiple disk channels, controllers, or both.
2.4.2 Database (or application) servers

Database servers are used to store the data for Domino applications. Customized applications such as help desk databases or support tracking systems, for example, are typically implemented on dedicated servers. In sophisticated Domino-based applications, interaction between numerous databases is quite common. Application servers usually place heavier demands than usual on the CPU.

Distributing interdependent databases across multiple servers can place excessive loads on the servers, the network, or both, depending on the amount of interaction between applications. Dedicating your servers to application roles will provide the following benefits:

- Enables you to group applications by their usage. This will typically reduce replication needs and simplify administration of both the server and database security.
- Enables you to optimize server performance for application database usage without regard to mail usage.
- Enables you to define more specific requirements, such as response time, thereby establishing a better scalable model for expansion as usage and storage needs increase.
- Improves I/O performance by distribution of application databases across multiple disk controllers.

2.4.3 Hub servers

Hub servers are designed to be used as dedicated servers to route mail and replicate databases to other servers within the network. This typically results in hub servers being used only in larger Domino environments to minimize the performance impact on production servers. Domino hub servers are primarily implemented to facilitate communications between other servers, so users normally do not have access to these systems (or even know that they exist). Because of this, hub servers usually do not have large numbers of simultaneous connections. Therefore, you might assume that a hub server would not need to be based on a high-end server system to fulfill its role. However, this is typically not the case.

You must consider the fact that each hub server is required to support the storage of all common databases and be capable of replicating these databases to each additional server in a timely fashion. Each hub server must also be capable of routing mail and ensuring that messages are delivered within an acceptable time frame. To fulfill these requirements, each hub server should be designed with both powerful processing capacity and a high-performance disk subsystem.

We recommend that you use hub servers as the administration server for the Domino directory and applications. The Domino administration process uses this server to execute requests and updates to the directory and applications.

Providing database replication and mail routing to several other servers within the network places heavy loads on all server subsystems. Your target should be to design high-capacity scalable platforms. Areas you should consider include:

- The system should have a fast disk subsystem for replicating and archiving.
- Full-text indexing should not be allowed on these systems, because index updates degrade performance, and users will not be accessing these databases. Refer to “Domino database indexing: Controlling the UPDATE task” on page 31 for more details about how to disable full-text indexing on your server.
- Remove access to the server for all normal users so that they do not degrade performance.
- Improve mail routing performance by using multiple mail.box databases.
2.4.4 Web servers

Web servers provide data requested by Web browsers. This information can be normal Web pages consisting of graphics files and text files on the server stored in HTML format, or in a Domino environment, pages built from Notes databases. Where practical, all information available from your Web site should be placed on a single server to reduce stresses on both the system and the network, as discussed in 2.4.2, “Database (or application) servers” on page 12.

Web servers usually carry a lot of network traffic and can have a significant CPU load if complex Web applications are implemented. Dedicating servers to Web tasks will provide the following benefits:

- Simplifies security for both intranet use and external Internet connections. Only databases to be viewed by Web browsers should be placed on the server, especially if this server is to be browsed by Internet clients.
- Enables you to optimize server performance for Web usage without regard to mail usage.
- Enables you to define more specific requirements, such as response time, thereby establishing a better scalable model for expansion as usage and storage needs increase.

2.5 NotesBench reports

To help you to determine which servers you will need for your Domino network, we discuss a performance testing tool called NotesBench, developed by Lotus.

NotesBench enables hardware manufacturers to provide Lotus Domino customers directly with capacity and performance information about their server hardware. Using a standard benchmark allows you to compare one server with another, because the conditions under which the tests were carried out are defined as part of the test. NotesBench reports provide a great deal of information about the capacity of a Domino server.

At the time of writing, only a minimal number of Domino 6.5 NotesBench reports had been made available. However, several Domino 6 reports can be used at this time.

Table 2-1 and Table 2-2 on page 14 list some available reports.

Table 2-1  Tested NotesBench reports for xSeries 235

<table>
<thead>
<tr>
<th>System tested</th>
<th>xSeries 235 (2*Intel Xeon 2.8 GHz; 4 GB RAM)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domino version</td>
<td>Release 6.0</td>
</tr>
<tr>
<td>Workload</td>
<td>R6iNotes</td>
</tr>
<tr>
<td>OS version</td>
<td>Red Hat Linux 7.3</td>
</tr>
<tr>
<td>Users</td>
<td>3,000</td>
</tr>
</tbody>
</table>

For more information, see:

http://notesbench.org/r6report.nsf/ed670662098f24c68525687f006bf80e/704a8c1a5442ee8485256ce105df0aa?OpenDocument
Table 2-2  Tested NotesBench report for BladeCenter

<table>
<thead>
<tr>
<th>System tested</th>
<th>BladeCenter with one HS20 server module (2*Intel Xeon 3.06 GHz; 4 GB RAM)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domino version</td>
<td>Release 6.0</td>
</tr>
<tr>
<td>Workload</td>
<td>R6iNotes</td>
</tr>
<tr>
<td>OS version</td>
<td>Red Hat Linux 7.3</td>
</tr>
<tr>
<td>Users</td>
<td>4,000</td>
</tr>
</tbody>
</table>

For more information, see:

http://notesbench.org/r6report.nsf/ed670662098f24c68525687f006bf80e/d0d86751fb5f819c85256d710048f5?OpenDocument

The latest official NotesBench reports are available at the following sites:

http://www.notesbench.org

2.5.1 Server.Load

Lotus Server.Load is a tool that measures and characterizes various Domino server capacity and response metrics. Server.Load is installed on a Notes client. With Server.Load, you can select a workload and run it against a target server. The workloads (also called tests or scripts in the user guide) simulate the behavior of Domino workstation-to-server operations. Server.Load can be used by companies to stress test their servers before they are ready for production.

2.6 Server requirements and recommendations

To identify which xSeries system will satisfy your Domino server requirements, there are still some sizing issues that have to be quantified, in particular, the amount of memory and disk storage that you will need. We examine these choices next and also provide additional recommendations for your server to ensure high performance and smooth operation.

As you identify which subsystems are potential sources of bottlenecks, you will have a general idea of what initial actions to take regarding server requirements and performance optimization.

This section describes aspects of the key hardware subsystems and their optimal settings for the Domino 6.5 environment.

As previously discussed, Lotus Domino server can act as a mail, Web, application, and database server, handling mail routing and storage, as well as database and Web requests. The important subsystems for Lotus Domino server are:

- Server memory
- Server disk space
- Raid and disk controllers
- Processors
- Network interface
This section also provides an overview for performance tuning these subsystems. For more
detailed information about performance tuning xSeries hardware for Lotus Domino, refer to
Part I, “Server subsystems,” in the IBM Redbook *Tuning IBM xSeries Servers for
Performance*, SG24-5287, available at:


### 2.6.1 Server memory

The minimum amount of random access memory (RAM) you need in your Domino server
depends on two primary factors. Domino and the operating system need a base amount of
RAM to be able to execute on the hardware; this is the first component of the total RAM
required. The second component depends on how many Notes and Web clients have
sessions with your server and on what other tasks your server might have to perform. The
memory model has been redesigned in Domino 6.5, and this has meant a 30% reduction in
memory requirements per user.

Use the following formula to calculate the minimum amount of RAM you need in your Domino
6.5 server.

<table>
<thead>
<tr>
<th>Minimum memory requirements: 128 MB + (number of concurrent users/5) MB</th>
</tr>
</thead>
</table>

Remember that this RAM requirement includes only the operating system and Domino
running on your server. If you have any file and print services, backup and recovery, or
anti-virus software running on your server, you will need to configure additional RAM for your
server.

You should not install small increments of RAM in your server. If you have a requirement for
150 users on a Domino server, this formula tells you to install 158 MB (128 MB + 150/5 MB) of
RAM. However, you would not try to install exactly 158 MB of memory in your server. Choose
a convenient memory arrangement that uses the minimum number of dual in-line memory
modules (DIMMs) and provides equal to or more than your calculated minimum RAM amount.
In our example, this would give the total server memory of 256 MB from a single DIMM.

This algorithm is appropriate for mail and application servers and mail hubs. However, it is not
appropriate for replication hubs. Usually, replication hubs are heavily used but have few, if
any, sessions open to active users.

**Note**: Lotus Domino only addresses up to 2 GB of RAM. If you install extra memory, it will
only be used by the operating system and other additional tasks. This means that installing
more than 2.5 GB of RAM will not significantly improve server performance.

The IBM performance team has found that there is no guarantee that an existing server has
optimal memory utilization. For Microsoft Windows 2000 environments, you should
periodically monitor the memory allocation in the operating system's Task Manager, because
it clearly shows the total amount of memory installed and the average amount of available
memory.

Total memory minus available memory is the amount of memory the server is actually using,
sometimes called the “working set.” Because memory use is dynamic, it is best to monitor
memory utilization over an extended period of time to arrive at an accurate representation of
the memory working set. To determine the amount of memory needed to support double the
users, double the working set size, and then add 30% as a buffer for peak activity.
Servers should be configured so that average memory utilization does not exceed 70%. If this level is consistently exceeded, you run the risk that the server will expand storage onto disk (page memory onto disk) during periods of peak activity. Servers should never regularly page memory to disk, because performance will be very adversely affected.

**Memory guideline:** Paging I/O should occur only occasionally, such as when applications are initializing, never on a continuous basis. Average memory utilization should not exceed 70% of available memory.

Finally, we recommend that you always use error checking and correcting (ECC) RAM. All xSeries systems currently use this memory. This memory is significantly less prone to errors than normal parity memory.

### 2.6.2 Server disk space

The amount of disk space that your Domino server requires can vary greatly depending on what tasks the server needs to perform. We outline the Domino services that require disk space in the following sections, so you can easily determine how much your server requires.

New data compression techniques in Domino 6 reduce message sizes by 45% to 65%, saving valuable storage space and enabling faster backups.

#### Operating system and Domino

First determine how much disk space you need for your operating system and Domino software. The Lotus minimal requirement is 1 GB hard disk space (1.5 GB recommended) and 128 MB memory (192 MB or more recommended) for Domino running on Windows 2000. This is a good initial value for several reasons:

- It covers the minimum operating system requirements for Microsoft Windows 2000 Server.
- It gives you enough disk space in case you decide to install all of the Domino features, now or in the future.
- It provides sufficient disk space for future enhancements such as Domino Extended Search.

#### Operating system paging file

Windows 2000 Server uses a paging file stored on disk that allows the operating system to provide a virtual address space larger than the physical RAM size. Therefore, as you run more and more applications on your server, you will reach a point when the amount of physical RAM is exhausted. This will not cause your programs to fail. The operating system will simply page memory blocks between physical RAM and disk to meet the application’s needs. In Windows 2000, this file is called `pagefile.sys`. Because accessing a disk is slower than RAM, a server’s performance will dramatically decrease the more it has to page memory blocks between RAM and this paging file. You should make sure you have enough memory configured in your server (see 2.6.1, “Server memory” on page 15).

By default, Windows 2000 sets the `pagefile.sys` to the size of physical memory plus 12 MB, but making it larger often resolves virtual memory shortages.

We recommend that you make the initial size of the Windows 2000 paging file equal to the amount of memory you have in your server plus 10%. If you have a smaller paging file that expands and contracts in size, the operating system has to vary the size depending on the application load, further impacting performance. Having a large paging file has other advantages. It has been shown that a server will gain performance improvements if this file is
a contiguous set of disk blocks, because it reduces the chance of the paging file becoming fragmented.

Following this recommendation, our server with 256 MB of RAM (see 2.6.1, “Server memory” on page 15) will have a paging file of about 280 MB.

**Mail users**
You should add approximately 80-100 MB of disk space per mail user. This has been increased from our previous recommendation of 50 MB per user for R5.x versions of Domino. This is because of increasing Internet mail usage, where people are sending and receiving more messages with large attachments and rich text and HTML e-mail. In addition, most users, when they start using any e-mail system, have a tendency to keep copies of all their correspondence. Users’ mail file data will also increase as they begin to use features such as calendar and scheduling. Providing sufficient disk space for mail users initially gives you plenty of time to monitor the size of users’ mail databases and to make plans to add more if necessary.

**Application servers**
The types of application databases that can run on Domino can vary widely. Some databases, such as those for ordering systems, can grow quite slowly in size, while a discussion database is updated and added to by many users each day. We recommend that you initially provide 50 MB per standard (slow growing) Notes database and 100 MB per Notes discussion database.

**Mail hubs**
Mail hubs do not need as much disk space as other servers because they only route mail. However, if a link to a remote server is unavailable and users have sent many e-mails with large attachments, your mail hub can quickly run out of disk space. These mail items are temporarily stored in mail.box databases before they are delivered to another server or the recipient. You should initially provide 300-500 MB of additional disk space for your mail hub. If you have a very large number of users dependent on this mail hub, provide additional disk space as a precaution.

**Replication hubs**
These servers need large amounts of disk space to store complete replica copies of databases. Replica databases are copied from other servers and then to remote spoke servers. Determine how many application servers replicate to this hub. If all the databases are different on each application server, the total amount of disk space you will need would be the total of the disk space on each of the application servers added together. If there are databases common to multiple application servers, your total disk space requirement might be lower.

**Summary**
In summary, you can use the formulas shown in Table 2-3 on page 18 to calculate the minimum disk space you will need on your Domino server.
Table 2-3  Summary of minimum disk space requirements for Domino R6

<table>
<thead>
<tr>
<th>Item</th>
<th>Minimum disk space requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system + Domino</td>
<td>Windows 2000 (1 GB Minimum) / Domino (2 GB Minimum (1 GB free space))</td>
</tr>
<tr>
<td>Page file</td>
<td>Amount of RAM + 10%</td>
</tr>
<tr>
<td>If application or mail server</td>
<td>(Number of mail users + number of standard databases) x 80 + (number of discussion databases x 100)</td>
</tr>
<tr>
<td>If mail hub</td>
<td>100 MB for each mail.box</td>
</tr>
<tr>
<td>If replication hub</td>
<td>Add up the total amount of database space on application servers with which this hub replicates.</td>
</tr>
</tbody>
</table>

As an example, assume that we have a server that needs to support 100 mail users. The server is fitted with 1GB of RAM. There are only two discussion databases planned for this server, and the server acts as a mail hub to a remote Domino server.

The server would require the disk space shown in Table 2-4.

Table 2-4  Minimum disk space calculation example

<table>
<thead>
<tr>
<th>Item</th>
<th>Minimum disk space requirements (MB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system + Domino</td>
<td>3 GB hard disk space</td>
</tr>
<tr>
<td>Page file</td>
<td>256 MB + 256/10 = 281.6 MB</td>
</tr>
<tr>
<td>If application or mail server</td>
<td>(100 mail users) x 50 + (2 discussion databases x 100) = 5200 MB</td>
</tr>
<tr>
<td>If mail hub</td>
<td>3.5 GB of disk space is recommended as a minimum.</td>
</tr>
<tr>
<td>Total</td>
<td>Approx. 8.5 GB of disk space is recommended as a minimum (if application or mail server).</td>
</tr>
</tbody>
</table>

2.6.3 RAID and disk controllers

Several xSeries servers have redundant array of independent disks (RAID) technology as standard. This allows data to be striped across a number of hard disks that are seen by the operating system as one large disk drive. The three most popular levels of RAID used are RAID-0, RAID-1, and RAID-5. RAID-0 generally gives the highest disk performance, but there is no fault tolerance should one of the individual drives fail. RAID-1 provides excellent fault tolerance and is essentially disk mirroring. RAID-5 technology uses parity blocks so that, should one drive fail, the controller can rebuild the information on a replacement hard drive, while still providing valid data to the system. RAID-5 is more efficient in its use of disk space, but RAID-1 has 50% more throughput than RAID-5 when using an equivalent number of drives if the percentage of writes is high. The disk subsystem is the most overlooked performance bottleneck in the server.

We recommend the following for your Domino server:

- Implement RAID-1 on all of your servers if possible, or RAID-5 as your second choice. As your Domino servers become more critical to your business, you will want the best combination of disk fault tolerance and performance you can afford.
- Implement a second RAID-1 logical drive to store the Domino 6 transaction logs.
Only use hardware-controlled RAID arrays. Software-controlled RAID arrays place additional strain on your server in terms of CPU utilization. The CPU should be dedicated to providing Domino functions, not implementing a RAID subsystem.

If possible, install an additional hard disk for use as a hot spare. Hot spares can replace failed disks automatically, or be managed by the administrator.

Choose the fastest hard drives. High-performance disks provide fast data rates and low latency times. xSeries servers support very fast 15,000 RPM drives.

Do not mix SCSI Fast with SCSI Fast/Wide drives.

Do not mix drives that spin at different speeds.

Do not mix drives of different sizes in a RAID array.

Enable write-back cache on the RAID controller. This setting improves system performance by caching information before it is written back to disk. Some controllers do not have battery backups, and cached data will be lost in the event of a power failure. In this situation, installing an uninterruptible power supply (UPS) for your server is particularly important.

Set the RAID stripe size to 16 K.

Install the latest ServeRAID™ BIOS, firmware, and drivers, if applicable.

Use the fastest drives possible (for example, 15,000 RPM).

**2.6.4 Processors**

The CPU requirement per user has been reduced by as much as 23%, enabling servers to handle more users.

Many Domino functions are processor intensive. Tasks such as routing messages, indexing databases, searching databases, and dynamically creating HTML pages for Web clients all put heavy demands on the system's CPU. We, therefore, recommend that you purchase an xSeries server that is capable of symmetric multiprocessing (SMP). You might not need or be able to afford the extra CPU today, but you are providing room to grow as your use of Domino functions expands.

Domino is a highly scalable application that can take advantage of the benefits of SMP. It does so automatically; you do not need to tune or configure your hardware for Domino to take advantage of the additional CPUs. Domino takes advantage of multiple processors by allocating the replicator and router tasks to different CPUs, thereby spreading the load. Domino is a highly scalable application that can take advantage of the benefits of SMP.

**2.6.5 Network adapter card**

When analyzing network performance and potential bottlenecks, the following parameters should be investigated to maximize performance:

- Network adapters
- Ethernet adapter performance
- TCP/IP performance
- Operating system settings
- Network design

Network compression is an important performance feature offered in Domino 6.5. When you enable network compression, data is automatically compressed before it is sent over the network. This improves network performance, especially over slower line speeds. Network
compression can result in 34% to 52% less network traffic and enable faster message delivery.

Never assume that you do not have a LAN bottleneck by simply looking at LAN sustained throughput in bytes/sec. LAN adapter bottlenecks often occur at low LAN utilization but at high sustained packet rates. Observing packets/sec will often yield clues to these types of bottlenecks.

We recommend the following network adapter configuration for your server, especially if your Domino system is used by a large number of users:

- Use the onboard Ethernet adapter (10/100/1000 Mbps) in the server where possible.
- Use an IBM PCI Ethernet card with full-duplex capability.
- If you want to use another manufacturer’s network adapter, select one from a vendor who will provide you with excellent service and support and robust network drivers for the operating system you are using.
- For servers that will support a large number of users, consider using multiple network adapters.
- Connect your Ethernet adapters to a switch. This will provide extra bandwidth into the server.
Tuning your operating system on xSeries and BladeCenter for Domino

This chapter provides some guidelines and recommendations that will assist in the planning and tuning of Microsoft Windows and Linux operating systems for Lotus Domino 6.5.

We also provide links and access to additional resources to optimize the operating system settings in preparation for Domino 6.5.
3.1 xSeries and BladeCenter system requirements for Domino 6.5

The recommended requirements for Lotus Domino 6.5 should only be used as a guide. Capacity planning should be undertaken to ensure the servers are designed and built for the task assigned.

Note: Lotus Domino 6.5.1 was released on January 21, 2004. The information gathered for this chapter is based on this version.

3.1.1 Microsoft Windows

Lotus Domino 6.5.1 is prepared to run on the following Microsoft platforms:

- Microsoft Windows 2003 (Server and Advanced Server editions)
- Microsoft Windows 2000, Service Pack 3 (Server and Advanced Server editions)
- Microsoft Windows NT®, Version 4.0, Service Pack 6a (using an Intel processor)

Hardware requirements

The hardware requirements are:

- Processor supported: Intel Pentium
- Memory: 192 MB minimum, 256 MB or more recommended
- Disk space: 2 GB minimum (with a minimum of 1 GB free space)
- Disk swap space: 128 MB RAM minimum

A detailed explanation of these settings is available in Chapter 3 “Lotus Domino 6 prerequisites for IBM @server BladeCenter,” in the Redpaper Deploying Lotus Domino on IBM @server BladeCenter, REDP-3584, available at:


For more information about performance tuning xSeries hardware for Lotus Domino, refer to Tuning IBM @server xSeries Servers for Performance, SG24-5287, available at:


3.1.2 Linux

Lotus Domino 6.5.1 is prepared to run on the following Linux platforms:

- Red Hat Advanced Server, Version 2.1 (now supporting SMP)
- All United Linux/Powered by United Linux 1.0 SP2 for Linux on Intel distributions. That includes SUSE LINUX Enterprise Server (SLES) 8.0, Turbo Linux Enterprise Server, and Connectiva Linux Enterprise Server.

Hardware requirements

The hardware requirements are:

- Processor supported: Intel x86
- Memory: 128 MB minimum, 192 MB or more recommended
- Disk space: 1 GB minimum
- Disk swap space: Three times the physical RAM installed
3.2 Performance basics

Hardware and operating system tuning are key to delivering an effective Domino 6.5 server. We discuss several parameters that will enhance server performance.

Comprehensive details about these items are available through additional Redbooks and Redpapers discussed within this section.

3.2.1 Tuning Microsoft Windows for Domino 6.5

You can improve the performance of your Lotus Domino server by changing various operating system settings for Windows (the following settings relate to Windows 2000 Server). We recommend that you implement these changes immediately after you have installed your Lotus Domino server.

To use Windows 2000 as a Lotus Domino server platform, the operating system must be configured to service the unique demands of an application server. The following parameters are examples of performance tuning for Lotus Domino 6.5.

Application responsiveness

You can configure how Windows 2000 responds to foreground or background applications as a part of its multitasking capability.

In Windows 2000, click Start → Settings → Control Panel → System. Open the Advanced tab, and then click the Performance button. The panel shown in Figure 3-1 opens.

![Figure 3-1 Disabling boost of foreground applications in Windows 2000](image)

By doing this, the following Registry parameter will be altered:

HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\PriorityControl\Win32PrioritySeparation
You can control how a Windows 2000 server prioritizes its memory allocations and thread priorities for network services versus local applications and the size given to the system cache.

In Windows 2000, click **Start** → **Settings** → **Network and Dial-Up Connection**. To select a Local Area Connection, from the menu, choose **File** → **Properties**. Select **File and Print Sharing for Microsoft Networks**, and click the **Properties** button. The panel shown in Figure 3-2 opens. Then select **Maximize data throughput for network applications**.

By doing this, the following registry parameter is changed to zero:

HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\SessionManager\MemoryManagement\LargeSystemCache

And the following registry parameter is changed to 3:

HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\LanmanServer\Parameters\Size

**Note:** A message on the Lotus Domino server console recommends that you set the Win32PrioritySeparation value in the Registry to zero. This is valid only for Windows NT 4.0 and 3.51. The message, however, might also appear when you run Lotus Domino under Windows 2000.

We recommend that you always use the GUI controls to set the Win32PrioritySeparation value, instead of editing the Registry directly.

**Tip:** We recommend that you do not edit the registry manually using the registry editor. If system parameters can be set using the configuration panel provided, it is better to make use of those.
If you experience a high page fault rate and a high page I/O as a result of Lotus Domino server processes running on a Windows 2000 server configuration, check the file system cache setting.

The default setting specifies that the Windows 2000 system should work as a file server, rather than as an application server, because Windows 2000 will favor the system-cache working set rather than the processes working set.

If you are running a Lotus Domino server on a Windows 2000 system set up as a file server, the excessive page faults and paging activities can result from memory-intensive processes. For example, rebuilding the index for a large database might result in an undesirable level of paging, causing users to experience poor response times. To correct the problem, set the file system cache setting to **Maximize data throughput for network applications** and reboot the system.

**NTFS allocation unit and stripe size considerations**

Tests have shown that the optimal allocation unit size for Lotus Domino server is 16 KB. You should, therefore, format the Notes database partitions using an operating system file allocation unit size of 16 KB. We also recommend that you choose an adequate 16 KB RAID stripe unit size.

To format a drive using 16 KB allocation units, execute the following command:

```
format drive: /q /fs:ntfs /v:trlog /a:16K
```

In Windows 2000, the Disk Management console already supports an allocation unit size up to 64 KB. Figure 3-3 shows how to format a drive in Windows 2000 to achieve a 16 KB NTFS allocation unit size.

![Figure 3-3 Setting 16 KB NTFS allocation unit size in Windows 2000](image)

**Tip:** Always use the NTFS file system, not FAT. NTFS is more secure, provides better performance, and allows a larger partition size than FAT.

### 3.2.2 Tuning Linux for Domino 6.5

There are no recommended Linux-specific tuning settings. However, there is a limit on the open file handles that the Linux kernel can maintain. This poses a limit on the maximum number of concurrent users a Domino server running Linux can maintain. With the current maximum number of open file handles, you can expect to support between 200 and 300 real users on a Linux server.
The following settings are recommendations for setting up Linux and are not specific to Domino; however, they will provide advantages to running the application server.

Linux is a powerful operating system that can be molded to fit your needs. We discuss hard disk drive arrays, CPU, and memory monitoring to assist you in significantly increasing the scalability of Linux supporting Domino.

We recommend the following settings:

- IDE versus SCSI: We recommend that you use SCSI hard disks for your servers.
- Buses: Generally, your server will have two buses. You should split your heavily used cards, such as a primary network card and a RAID controller, so that they are installed in slots on different buses.
- Distributing I/O: If you are running a program that is disk I/O intensive, such as a Domino server, we recommend that you use separate, physical disks for the operating system, Domino data directory, and the Domino 6 transaction logs.
- RAID configurations: The main advantage of RAID is the redundancy. If one disk fails (for any reason), you will not lose your data. Another advantage is that you can stripe your data over multiple disks and gain speed for your applications.
- CPU utilization: We recommend that you monitor the CPU utilization regularly. If the average value is between 60% and 80%, you have reached the warning zone. If the average usage exceeds 80%, we recommend that you add a new processor or upgrade the old one.
- Memory usage: Linux uses memory efficiently. If you see the swap constantly in use, this indicates a lack of physical memory, and your server performance will suffer. Disk swap space should equal three times the physical RAM installed

A detailed explanation of these settings is available in Chapter 4 “performance, scalability, and troubleshooting” in Lotus Domino 6 for Linux, SG24-6835, available at: http://www.redbooks.ibm.com/redbooks/pdfs/sg246835.pdf

Swap space
When tuning Lotus Domino 6.5, be sure to refer to “Operating system paging file” on page 16.

Summary
Remember, the only sure way to know if one of your server resources is becoming a bottleneck is to monitor the system on an ongoing basis. Use the Windows Performance Monitor to collect data. Analyze it to learn how to correct the possible performance bottlenecks. Also use Domino’s own tools in the Statistics & Events database, as described in Chapter 5, “Monitoring Domino performance on xSeries and BladeCenter” on page 65.

Lotus Domino performance tuning on xSeries servers and Windows 2000 involves careful understanding of its features and capabilities. Thorough analysis of these functions is critical in integrating Lotus Domino to your system’s infrastructure.

There are tools built into Lotus Domino and Windows 2000 that enable you to perform maintenance and tuning procedures. These tools include the Administration Control Panel, Notes Log database, Statistics & Events database, the Lotus Notes console commands, and the Performance console. Performing regular server maintenance and tuning procedures helps improve server performance. Monitoring Lotus Domino server performance gives you an idea of future growth.
Tuning Domino performance on xSeries and BladeCenter

This chapter discusses the performance tuning techniques and best practices available for Lotus Domino 6.5 on xSeries and BladeCenter servers.

These settings and references to additional resources display the extensive support and reference material available for xSeries servers when using Domino.
4.1 Domino tuning

As discussed, Lotus Domino servers can act as a mail, Web, application, and database server. It might need to handle mail routing, storage, and database and Web requests. Parameters within Domino 6.5 can be adjusted to assist in the delivery of an effective solution.

It might also be necessary to modify server settings to manage customized development requirements (indexing, APIs, and so on).

This chapter provides Domino tuning best practices and will assist you in gaining greater server performance. We emphasize the possibilities a Domino administrator has to optimize the workload that a Lotus Domino 6.5 server can handle.

The best way to improve Domino performance is to:
- Plan ahead.
- Test the Domino application thoroughly on a test system and then on a pilot system.
- Start performance monitoring from day one, before the first user signs on.
- Constantly analyze your performance data and adjust your sizing based on actual facts.
- Roll it out gradually, a few users at a time, constantly monitoring its performance.
- Continue to analyze the performance data and report future trends and hardware needs.

It is ideal to use a centralized server to capture, collect, and report the statistics of a group of Domino servers. This is discussed further in Chapter 5, “Monitoring Domino performance on xSeries and BladeCenter” on page 65.

Tuning for the best performance can, however, start with good application design during the development phase.

For more information about application development performance for Domino, refer to Performance Considerations for Domino Applications, SG24-5602.


4.2 Domino server performance

Server overload affecting performance is possible if there was not sufficient planning for potential growth.

Many parameters can be tuned for optimal performance. As discussed earlier, Domino 6.5 servers can be used for numerous tasks (Web, application and mail server). Here, we describe the performance tuning best practices for these tasks. These tasks are:
- Optimizing Lotus Domino task performance (tasks, indexing)
- Performance tuning the HTTP stack
- Network performance (compression)
- Improving mail performance (routing)
- Optimizing database performance
- Calendaring and scheduling
- Improving the operational performance of the server (database activity, logging, agents, replication)
4.2.1 Optimizing Domino task performance

Lotus Domino installed straight out of the box is optimized for most configurations. However, there are some tuning parameters in specialized situations that can give your Lotus Domino server some added performance and also protect the server from overloading.

NOTES.INI

The Lotus Notes Domino server is controlled by the values stored in the NOTES.INI file. We recommend that you take a backup of your NOTES.INI file before you make any modifications in case your settings are not accepted by the system.

In order to tune various parameters of the Domino server, you need to manipulate statements in NOTES.INI configuration file. Before making changes to the NOTES.INI file, use the SHOW CONFIGURATION * server command at the console to check the existing settings. To edit the settings in the NOTES.INI file, use one of the following methods.

Using a Server Configuration document

The Server Configuration document is a form that exists in the Domino Directory. Use a Server Configuration document to specify settings for a single server, a group of servers, or all servers in a domain. You cannot specify all NOTES.INI settings in a Server Configuration document. The Set/Modify Parameter list in the Server Configuration document lists the default settings that you can specify. This is the recommended method of updating the server parameters.

Using the SET CONFIGURATION server command at the console

The SET CONFIGURATION server command writes the new setting to the NOTES.INI file. If you previously specified the setting in any Server Configuration document that affects this server, Lotus Domino writes the new setting to the Server Configuration document specific to the server being configured or creates a new document if necessary.

Editing the NOTES.INI file directly in a text editor

We do not recommend directly editing the NOTES.INI file because of the possibility of introducing errors into the file and impairing the operation of the Lotus Domino server. In most cases, you will have to restart the Lotus Domino server to put your changes into effect.

Tip: When you set variables using the Server Configuration document in the server's Domino Directory, you will get additional information about the particular variable's usage.

Configuring server tasks

Each task increases the server's load and can adversely affect server performance. Minimizing the number of server tasks that are run on the server, the frequency at which they run, and the time in which they run will enable you to increase the performance of the server.

The Lotus Domino server tasks to be automatically launched by the server are set in the NOTES.INI variables. For example:

- ServerTasks=Router,Replica,Update,Amgr,AdminP,CalConn,Event,Sched,Stats,HTTP,DIOP,IMAP,POP3,NNTP,DECS,maps,collect
- ServerTasksAt1=Catalog,Design
- ServerTasksAt2=UpdAll, Object Collect mailobj.nsf
- ServerTasksAt3=Object Info -Full
- ServerTasksAt5=Statlog

Each of these variables controls the schedules for automatic server and database maintenance tasks. The time is entered in a 24-hour format, where 0 is 12:00 a.m. and 23 is
11:00 p.m. In the example above, Catalog and Design tasks would initiate at 1:00 a.m., and the Statlog task would initiate at 5:00 a.m.

You can significantly improve performance by removing tasks that are not appropriate to the server if it is a specialized server. Consider the following suggestions to increase performance related to Lotus Domino server tasks.

- **These tasks should be turned off when they are not in use:**
  - **Scheduling:** Turn off this task if you are not using the server for scheduling and calendaring.
  - **AMgr:** Turn off this task if you are not using the server to run scheduled agents. Remember, this function is not required for WebQuery Agents.
  - **Collector, Reporter:** Turn off this task if you are not using the server to automatically track server statistics on a regular basis. You can collect them on demand.

- **Remove the Replicator (Replica) and Router tasks.**
  - Both of these tasks can be removed if they are not being used on the server, because each of these tasks take up a fair amount of server resources when loaded. For example, if you have only one Lotus Domino server in your organization that is used for both applications and mail routing, you might not need the Replicator task, because you do not have any other servers from which to replicate (and because clients will be replicating from the Lotus Domino server, not vice versa).
  - Another example might be that you have a hub replication server that never routes mail. In this case, you can remove the Router task. Be careful with this example because, although the server might not be used for routing mail between users, you might have some mail-enabled Lotus Domino database applications, in which case, you will need the Router task and should not remove it.

- **Carefully choose the times when server tasks are run.**
  - Daily server tasks should be run when other server tasks are not running and at times when few or no users are using the Lotus Domino server, such as before users begin using the server in the morning, during lunch time, or after normal working hours. This allows the maximum amount of server resources to be available for each server task that is currently executing and for user sessions. Examples of such server tasks are Design, Catalog, Statlog, and customized Notes API programs that need to be run only once a day.

**Task monitoring**

Domino runs a number of executables on the server. Each supports a different basic Domino task. The following list states their names and corresponding tasks:

- **ADMINP** The major Domino scheduler job running batch-like updates.
- **AMGR** Domino Agent Manager; takes care of predefined agents.
- **CALCONN** Connects the Domino calendar to other types of calendars.
- **EVENT** Domino Event Handler.
- **REPLICA** Domino database Replicator task.
- **REPORT** Domino Report Generator (adding data to STATREP.NSF).
- **ROUTER** Domino Mail Routing task.
- **SCHED** Domino Scheduler task.
- **SERVER** The main Domino Server task.
- **UPDATE** UPDATE task for Domino database indexes and views.
Domino database indexing: Controlling the UPDATE task

One of the most common, preventable causes of poor response time and performance of Domino servers is excessive and unnecessary activity of the Domino UPDATE task. This task is responsible for updating and rebuilding the indexes of Domino database views.

It is designed to run in the background and is intended to improve response time and performance by ensuring that, when a user opens a database view, the user does not have to wait for it to be indexed. Ironically, it often turns out to have the opposite effect.

The Indexer is composed of the following components:
- The UPDATE task
- The UPDALL task
- The Notes Indexing Facility (NIF) subsystem

Update task

The Update task is loaded at server startup by default and runs continually, checking its work queue for views and folders that require updating. When a view or folder change is recorded in the queue, the Update task waits approximately 15 minutes before updating all view indexes in the database so that the update can include any other database changes made during the 15-minute period. After updating the view indexes in a database, it then updates all the databases that have full-text search indexes set for immediate or hourly updates.

When the Update task encounters a corrupted view index or full-text index, it rebuilds the view index or full-text index in an attempt to correct the problem. This means that it deletes the view index or full-text index and rebuilds it.

To improve view-indexing performance, you can run multiple Update tasks if your server has adequate CPU power. Doing this can affect server performance and is recommended primarily for multiprocessor machines. On a server with multiple processors, enable a maximum of one Update task per processor.

You can modify the Update task using the Configuration Settings document.

Complete the following steps:
1. From the Domino Administrator, click Configuration tab.
2. Next to Use Directory on, select the server that stores the Domino Directory you want to modify.
3. Expand Servers → Configurations.
4. Do one of the following:
   - Click Edit Configuration to edit an existing Configuration Settings document.
   - Click Add Configuration to create a new Configuration Settings document.
5. Click the NOTES.INI Settings tab.
6. Click Set/Modify Parameters.
7. In the Item box, select **Updaters**. In the Value box, enter the number of Update tasks to run. Then click **OK**.

8. Click **Save and Close**.

9. Restart the server so that the setting takes effect.

The Indexer, as a whole, is responsible for keeping the active views and the full text indexes current within the databases. It processes any requests for changes to documents within a database so that the active view collections and the full text indexes display the most recent modifications and current information.

Review 4.2.5, “Optimizing database performance” on page 44 to assist in the management of the Update task by using some best practices for indexing and frequency of updates.

### 4.2.2 Performance tuning the HTTP server stack

The use of Domino 6.5 as a Web hosting solution on xSeries and BladeCenter has been enhanced, with developers rewriting and optimizing the Domino HTTP stack for improved performance. You can further enhance server performance and response times by modifying certain HTTP parameters.

In order to optimize performance of the Lotus Domino 6.5 Web server, it is ideal to gather a baseline for performance. Check and record the current server performance and response times first and then consider applying the tuning tips described in this section.

#### Internet Site documents

New to Domino 6, Internet Site documents are used to configure the Internet protocols supported by Domino servers. A separate Internet Site document is created for each protocol, Web (HTTP), IMAP, POP3, SMTP Inbound, LDAP, and IIOP, which is then used to provide protocol configuration information for a single server, or for multiple servers in a Domino organization. Specifically, you can create:

- Web Site documents. You create a Web Site document for each Web site hosted on the Domino server.
- LDAP Site documents. You create an LDAP Site document for LDAP to an organization in a directory.
- IMAP, POP3, and SMTP Site documents. You create an individual Internet Site document for each mail protocol for which you enter an IP address.
- IIOP Site documents. You create an IIOP Site document to enable the Domino IIOP (DIIOP) task on the server. This task allows Domino and the browser client to use the Domino Object Request Broker (ORB) server program.

Internet Site documents make it easier for administrators to configure and manage Internet protocols in their organizations. For example, prior to Domino 6, if you wanted to set up a Web site in your organization, it was necessary to configure each Domino server in the domain with Mapping documents, Web realms, and File Protection documents. If you had virtual servers and virtual hosts, you had to do the same thing for them. In Domino 6, you can configure a Web Site document so that all servers and hosts use it to get configuration information for a Web site, including mapping information, file protection information, and Web realm authentication information.

The Domino server is configured to use Internet Site documents if Load Internet configurations from the Server/Internet Sites Documents option is enabled on the Basics tab of the server document. If the option is not enabled, the server defaults to the Server
document settings to obtain configuration information for Internet protocols. We recommend that you enable this new function for ease of management.

Internet Site documents are created in the Internet Sites view, which is used to help manage Internet protocol configuration information by listing the configured Internet Site documents for each organization in the domain.

**Specifying network timeouts on the Web server**
Open, inactive sessions can prevent users from accessing the server. Specify time limits for activities between the Domino Web server and clients or CGI programs so that connections do not remain open if there is no network activity between them.

Network timeout options appear under the HTTP sub-tab of the Internet Protocols tab in the Timeouts section of the Server document, as shown in Figure 4-1.

![Figure 4-1 Network timeout tuning for Lotus 6.5 Domino Web server](image)

Table 4-1 on page 34 describes the fields and values you can use to tune network timeouts for the Lotus Domino Web server.
### Table 4-1 Setting the network timeout options for the Lotus Domino 6.5 Web server

<table>
<thead>
<tr>
<th>Field</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>HTTP persistent connections</strong></td>
<td>Specify whether you want to enable persistent HTTP connections on the Web server. These connections remain active under the following conditions:&lt;br&gt;▶ HTTP protocol is 1.1.&lt;br&gt;▶ The server application returns an HTTP response code less than 400. (If the server application returns an HTTP response code greater than or equal to 400, the connection will be closed by the server.)&lt;br&gt;▶ The HTTP request came through a proxy server.&lt;br&gt;▶ The client did not send a connection close header.&lt;br&gt;▶ The number of connections that the server can support is running low, or the number of connections queued for the thread processing the request is too large. If the connection is kept open, the following settings apply:&lt;br&gt;▶ The connection will be closed if the maximum number of requests per connection is exceeded.&lt;br&gt;▶ The connection will be closed if the persistent timeout is exceeded.&lt;br&gt;▶ The connection will be closed if no data is received by the server within the specified input timeout.&lt;br&gt;▶ The connection will be closed if a complete request is not received within the specified request timeout. <strong>Note:</strong> Persistent connections require more server overhead than connections that are limited by network activity.</td>
</tr>
<tr>
<td><strong>Maximum requests per persistent connection</strong></td>
<td>Specify the maximum number of HTTP requests that can be handled on one persistent connection. The default is 5.</td>
</tr>
<tr>
<td><strong>Persistent connection timeout</strong></td>
<td>Specify the length of time for which you want persistent connections to remain active. The default is 180 seconds.</td>
</tr>
<tr>
<td><strong>Request timeout</strong></td>
<td>Specify the amount of time for the server to wait to receive an entire request. The default is 60 seconds. If the server does not receive the entire request in the specified time interval, the server terminates the connection.</td>
</tr>
<tr>
<td><strong>Input timeout</strong></td>
<td>Enter the time, in seconds, that a client has to send a request after connecting to the server. The default is 15 seconds. If no request is sent in the specified time interval, the server terminates the connection. If only a partial request is sent, the input timer is reset to the specified time limit in anticipation of the rest of the data arriving.</td>
</tr>
<tr>
<td><strong>Output timeout</strong></td>
<td>Enter the maximum time, in seconds, that the server has to send output to a client. The default is 180 seconds.</td>
</tr>
<tr>
<td><strong>CGI timeout</strong></td>
<td>The maximum time, in seconds, that a CGI program started by the server has to finish. The default is 180 seconds.</td>
</tr>
</tbody>
</table>

### Specifying the number of threads used by the Web server

An HTTP request is processed by a thread. A thread, in turn, can handle a number of network connections. You can specify the number of threads the Web server can process. In general, the number of threads specified is an indication of the number of users who can access the server simultaneously.

If the number of active threads is reached, the Domino server queues new requests until another request finishes and threads become available. The more power your machine has, the higher the number of threads you should specify. If your machine spends too much time on overhead tasks, such as swapping memory, specify a lower number of threads.

Thread options appear under the HTTP sub-tab in the Internet Protocols tab in the Basics section of the Server document, as shown in Figure 4-2 on page 35.
Server: Domino651/ITSO

Basics  Security  Ports  Server Tasks  Internet Protocols  MTAs  Miscellaneous

HTTP  Domino Web Engine  DIOP  LDAP

Basics

Hostname(s):

Bind to host name: Disabled

DNS lookup: Disabled

DNS lookup cache: Enabled

DNS lookup cache size: 256

DNS lookup cache found 120 seconds timeout:

DNS lookup cache not found timeout: 240 seconds

Number active threads: 40

Web server threads options

Figure 4-2  Threads tuning for the Lotus Domino 6.5 Web server

Table 4-2 describes fields and values you can use to tune threads options for the Lotus Domino Web server.

Table 4-2  Setting threads options for Lotus Domino 6.5 Web server

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number active threads</td>
<td>The number of threads you want active on the server at the same time. The default is 40.</td>
</tr>
</tbody>
</table>

Restricting the amount of data users can send

The HTTP POST and PUT methods enable users to send data to the Domino server. The Server record field “Maximum size of request content” is new for Domino 6 and sets a limit on the amount of data that can be sent using either POST or PUT. This limit is enforced for all POST and PUT methods, whether the target is a database, CGI program, or Java™ servlet, and applies to all Web sites.

The Web Site document contains two additional settings that control POST and PUT methods that target a database (for example, filling in a form or uploading a file attachment). Formerly available in the Server document, for Domino 6, these settings been moved to the Web Site document so that you can specify different values for each Web site.

To restrict the amount of data that can be sent to a Domino database:

1. From the Domino Administrator, click the Configuration tab, expand the Web section, and click Internet Sites.
2. Choose the Web Site document you want to edit, and click Edit Document.
3. Click the Domino Web Engine tab. Under “POST Data,” complete the field shown in Table 4-3 on page 36.

Table 4-3 on page 36 describes fields and values you can use to manage the size of user’s uploads to the Lotus Domino Web server.
### Improving file download performance for Web clients

Web clients can download a file that is attached to a page or that is in a server directory that is mapped by a URL. If a client is using a product that supports byte-range serving (available in HTTP 1.1 and later), the client downloads the file in sections (ranges of bytes) and tracks the progress of each file download. If an interruption occurs, the client can resume the download from the point where it was interrupted. Without byte-range serving, users must repeat interrupted downloads from the beginning.

Domino is compatible with clients that support the HTTP 1.1 specification. The clients can be implemented in a variety of ways, for example, as browser plug-ins, applets, or stand-alone programs. Domino automatically uses byte-range serving if the Web client uses a product that supports this feature. No configuration is necessary.

However, attached files must be decompressed in order for clients that support byte-range serving to access them. When you attach a file, you must clear the Compress option. To verify that an existing attachment is decompressed, choose File → Document Properties, select the $FILE item, and verify that the Compression Type property is NONE, as shown in Figure 4-3.

### Managing Web agents

You can specify whether Web application agents, that is, agents triggered by browser clients, can run at the same time. These include application agents invoked by the WebQueryOpen and WebQuerySave form events and for agents invoked by the URL command “Penitent”. If you enable this option, the agents run asynchronously. Otherwise, the server runs one agent at a time.
You should set an execution time limit for Web application agents. The purpose of the time limit is to prevent Web agents from running indefinitely and using server resources. However, do not rely on this mechanism for the routine shutdown of agents. When the server shuts down an offending agent, resources that the agent was using (such as disk files) might be left open.

To run Web application agents:
1. Open the Server document you want to edit.
2. Choose Internet Protocols → Domino Web Engine. Under Web Agents, complete the fields as shown in Table 4-4.

<table>
<thead>
<tr>
<th>Field</th>
<th>Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run Web agents concurrently?</td>
<td>Choose one:</td>
</tr>
<tr>
<td></td>
<td>▶ Enabled: To allow more than one agent to run on the Web server at the same time (asynchronously).</td>
</tr>
<tr>
<td></td>
<td>▶ Disabled (default): To run only one agent at a time (serially).</td>
</tr>
<tr>
<td>Web agent timeout</td>
<td>The maximum number of seconds (elapsed clock time) for which a Web application agent is allowed to run. If you enter 0 for the value (default value), Web application agents can run indefinitely. Note: This setting has no effect on scheduled agents or other types of server or workstation agents.</td>
</tr>
</tbody>
</table>

### Domino Web Access (iNotes) performance

Domino 6 Web Access (iNotes) performance was a top priority for the Domino development team. The focus was placed on optimizing, streamlining, and removing bottlenecks in Domino Web Access and in the underlying Domino 6 code. The significant improvements made include:

- Rewriting and optimizing the Domino 6 HTTP stack for performance (this will benefit any Web-based application, not just Domino Web Access).
- Introducing new Notes memory management techniques to make memory allocation faster and to improve memory utilization.
- Enhancing NSF database core code for performance.
- Optimizing Domino Web Access code, especially for Calendar and Scheduling (C&S).
- The R6iNotes workload implements C&S.
- Minimizing or eliminating semaphore contention in Domino code.
- Optimizing the formula/compute engine code.

Results have shown that for Domino 6 on Windows 2000, the number of Domino Web Access active users increased by 40% with a CPU cost per user saving of 30%.

For further information about Domino 6 Web Access (iNotes) performance features, refer to the following article:

http://www-10.lotus.com/ldd/today.nsf/8a6d147cf55a7fd385256658007aacf1/fe7f745db1c68f2e85256c5d0050f07c?OpenDocument

### Compress HTTP response data

By default, Domino Web Access uses compression (GZIP format) to reduce network bandwidth consumption and provide better performance, particularly for users with slow
network connections. This option is set using the “Compress HTTP response data” setting on the Domino Web Access tab of the Configuration Settings document (Figure 4-4).

![Compress HTTP response data setting](image)

### 4.2.3 Network performance (compression)

Network compression is an important performance feature offered in Lotus Notes/Domino 6. When you enable network compression, data is automatically compressed before it is sent over the network. This improves network performance, especially over slower line speeds.

Notes/Domino 6 network compression offers a number of immediate benefits. For example, by reducing the amount of data being transmitted, you can improve the performance of your routing and replicating hub servers, especially if they are currently laboring under heavy workloads. In addition, you can enable it by default, so all your users can take advantage of this functionality without having to select it themselves. Because network compression is a standard, out-of-the-box feature, it does not require any additional code, which helps simplify administration and requires fewer CPU resources to run.

Of course, as with any new functionality, network compression raises some immediate questions:

- Just how much network bandwidth do you save with compression?
- How does network compression affect server performance?
- Do you need this new feature if you are already using 6.5 attachment compression?
- How will this work with encryption? Can encrypted data be compressed and decompressed?

Note the following statistics about the benefits of network compression:

- A 35-52% reduction in data transferred from the server to client
- A 26% reduction in data transferred from server to server

Network compression is performed by the Notes Remote Procedure Call (NRPC). NRPC compresses the data before it is transmitted over the network, using the LZ1 algorithm developed by Abraham Lempel and Jacob Ziv. For detailed information about the LZ1...
algorithm, see the IBM article, “A fast hardware data compression algorithm and some algorithmic extensions,” available at:


With network compression enabled on both the sending and receiving computers, the Notes client, Domino server, or both, will attempt to compress any data (for example, e-mail or replicated documents) before transmitting it over the network. Network compression works with mail routing, replication, or any other data sent through NRPC. This includes design elements in new or updated replica databases.

Network compression works for both Domino-to-Domino and Domino-to-Notes sessions. Both the sender and receiver must be running Notes/Domino 6 and have network compression enabled. Otherwise, the data will be sent uncompressed.

Data that is first encrypted cannot then be compressed. However, data that is first compressed can then be encrypted. With Notes/Domino 6, this is not an issue. If you use port-level encryption in Notes/Domino 6 combined with network compression, network data is automatically compressed first and then encrypted. When you do this, you receive the full benefit of both compression and encryption.

Network compression cannot further compress an already compressed attachment. This includes files compressed by Notes when attached (Huffman or LZ1 attachment compression) or in a compressed file format (such as a ZIP archive or JPEG). In other words, after a file attachment is compressed, it cannot be further compressed by network compression. This is an important point to remember when determining the results of network compression at your site. If you run a test in which you mail a large number of compressed attachments, the resulting gains produced by network compression might appear smaller compared to mailing documents with uncompressed attachments. (We further illustrate this fact in the test results that follow.)

In Domino 6.5, you can enable network compression using the Notes client, the Domino Administrator, or both.

From Notes:
1. Open Notes. Choose File → Preferences → User Preferences.
2. Click the Ports tab.
3. Select the port on which you want to enable network compression from the list.
4. Select the Compress network data check box. This enables network compression for the selected port on the local Notes client. (This does not affect the server.)

From the Domino Administrator:
1. In the Server pane, click the server on which you want to enable network compression.
2. Click the Server → Status tabs, and then click Server Tasks.
4. Select the port on which to turn on network compression. (Make sure Port enabled is selected for the port.)
5. Select Compress network data.
6. Click OK.
7. Restart the ports on the server so that network compression takes effect. This enables network compression for the selected ports on this server. (This does not affect the Notes client.)
4.2.4 Improving mail performance

Lotus Domino offers a variety of tuning options to adapt its mail subsystem to your needs. You might, for example, want to disable mail routing completely on hub replication servers or configure your server for the best performance as a mail routing hub.

Setting maximum mail transfer threads

The MailMaxThreads setting determines the maximum number of concurrent processes that the mail router can create to perform its mail transfers efficiently. The default setting is one thread per server port. Using more router threads allows mail to be routed more quickly, because there is more than one router thread looking in the mail.box database and distributing mail documents at the same time, instead of only one router routing the mail.box messages one by one. This is especially important on mail hub servers. However, additional threads can increase the demand for server processing time.

You specify this setting in the Server Configuration document in the Domino Directory (as shown in Figure 4-6 on page 41) Since Release 5 of Domino, the setting MailMaxThreads= [number] is no longer included in the NOTES.INI file.

Although it is often desirable to route mail as quickly as possible, it is also possible to have too many router tasks running. If you are using multiple router tasks, you should monitor the performance and use of each of the tasks to be sure that none of them remains idle most of the time, because extra router tasks can take up significant system resources.

We recommend that you also temporarily change the mail logging setting to show detailed thread information, so you can gain a better understanding of task usage. You specify this setting in the Server Configuration document in the Domino Directory, as shown in Figure 4-5.

![Configuration Settings window](Image)

**Figure 4-5 Configuration Settings window**
The available options are:

**Minimal**
Only logs errors and warnings.

**Normal**
Minimal, plus transfer output and delivery output.

**Informational**
Normal, plus information about transfer threads/processes, plus more detailed information about the transfers and deliveries.

**Verbose**
Information, plus more detailed information about transfer queues.

By setting this information, you will see the usage of various threads and whether adding more will be more effective for your server.

### Setting maximum mail threads for local delivery

The MailMaxDeliveryThreads setting determines the maximum number of threads the router can create to perform local mail delivery. The default number is 1. Increasing this value can improve message throughput for local deliveries. The ideal number usually falls in the range of 3 to 25, depending on the number of local deliveries on your Lotus Domino mail server.

### Setting maximum mail concurrent threads for a destination

If the MailMaxConcurrentXferThreads value is set to 1, mail between two servers is transferred sequentially. It can, however, create a serious bottleneck, heavily impacting Lotus Notes mail users. Sending any large mail will delay the transfer of all other mail routed to the same destination until the first mail transfer ends. We recommend that you monitor the mail traffic and balance the value between the network throughput and size structure of mail usually sent by users. See Figure 4-6 and Figure 4-7 on page 42.

**Note:** Since Release 5 of Domino, the settings for MailMaxThreads, MailMaxDeliveryThreads, and MailMaxConcurrentXferThreads are not included in the NOTES.INI file; they are represented by their fields in the Configuration Settings document, as shown in Figure 4-6 on page 41 and Figure 4-7 on page 42.

![Configuration Settings](image)
Disabling per-user message caching by the IMAP task

This setting disables per-user message caching by the IMAP task. It can improve the capacity of a server by reducing memory consumption. However, response time for some user operations might be slower. If this setting is omitted, IMAP per-user message caching will be enabled.

To disable per-user message caching by the IMAP task, set the following value to 1:

\[
\text{NoMsgCache} = [0 \text{ or } 1]
\]

Controlling type-ahead addressing

Type-ahead addressing displays names that match the letters a user types in the To, cc, and bcc fields in a mail message. The Lotus Domino server automatically completes the rest of the recipient's address for you. You can change or retype the address as needed. To save time, Domino enables type-ahead addressing by default.

To save bandwidth and improve server performance, you can disable type-ahead addressing. If you disable type-ahead addressing on a mail server, users can still use type-ahead to find addresses in their Personal Address Book or mobile Directory Catalog.

To enable or disable type-ahead addressing, complete the following steps:

1. Make sure you already have a Configuration document for the server or servers to be configured.
2. From the Domino Administrator, click the Configuration tab and expand the Messaging section.
3. Click Configurations.
4. Select the Configuration document for the mail server or servers for which you want to restrict mail, and click Edit Configuration.
5. Under the Basics tab, complete the Type-ahead field as shown in Figure 4-8 on page 43. Then, save the document.
Creating multiple mail.box databases

In Lotus Domino 6.5 server, you can continue to use a single mail.box database or improve performance significantly by creating multiple mail.box databases.

The mail router task uses a database called mail.box to store incoming and outgoing messages. Any process trying to write to the mail.box database needs exclusive access to the database. In other words, if another process is reading the mail.box database, the router write process must wait until the reading process relinquishes the database. As the mail load on the server increases, contention for access to the mail.box database creates a potential bottleneck that can limit performance. With multiple mail.box databases, Lotus Domino 6.5 server can use multiple simultaneous processes in accessing mail.box data, improving mail routing performance.

Because disk contention is rarely an issue for the mail.box database, you do not need to put the multiple mail.box databases on different disks. However, you should spread user mail files across multiple disks to ensure that all mail files and mail.box databases are not on the same disk.

The number of mail.box databases to be set depends on the number of mail users on the server. Based on the testing, Lotus recommends an appropriate number of mail.box databases for a given number of mail users, as shown in Table 4-5.

Table 4-5  Recommendations for number of mail.box databases

<table>
<thead>
<tr>
<th>Number of mail users</th>
<th>Number of mail.box databases</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-200</td>
<td>1</td>
</tr>
<tr>
<td>201-999</td>
<td>2</td>
</tr>
<tr>
<td>1000+</td>
<td>10</td>
</tr>
</tbody>
</table>

Adding even one extra mail.box database can result in a noticeable performance improvement in mail routing. The benefit of increasing the mail.box database count from two is more marginal, which might help to explain the non-intuitive figures in Table 4-5.
To create multiple mail.box databases, complete the following steps:

1. Make sure you already have a Configuration Settings document for the server or servers to be configured.
2. From the Domino Administrator, click the Configuration tab and expand the Messaging section.
3. Click Configurations.
4. Select the Configuration Settings document for the mail server or servers for which you want to restrict mail, and click Edit Configuration.
5. Click the Router/SMTP → Basics tab.
6. Under the Router/SMTP → Basics tab, complete the Number of mailboxes field as shown in Figure 4-9. Then, save the document.
7. Restart the server.

![Figure 4-9 Setting up multiple mailboxes on the Lotus Domino 6.5 server](image)

**Note:** You specify the number of mail.box databases in the Configuration Settings document. If you use the same Configuration document for multiple servers, Domino creates the number of mailboxes (mail.box databases) you set in that document on each server using that document.

### 4.2.5 Optimizing database performance

Database performance affects the overall Lotus Domino server performance in the most significant way because Lotus Domino uses databases for most of its activities.

**Using Lotus Domino 6.5 database format**

Databases that you create in Lotus Domino 6.5 or convert to the 6.5 format perform considerably better than databases created in previous releases: Database operations require less I/O and fewer CPU resources, view rebuilding and updating is quicker, and memory and disk space allocation is improved.
Because of these performance improvements, limiting the size of the databases to improve database performance is less important than in past releases. The maximum database size for Lotus Domino 6.5 format databases is 64 GB.

Note: You still have to take into account possible limitations of the underlying file system. For example, the ext2 file system, which is still the default for most major Linux distributions, can only support files up to 2 GB in size. Prepare for this item before you deploy Domino databases.

Keeping database size to a minimum
Despite improvements in database performance in Lotus Domino 6.5, you should still regularly monitor the size of databases and keep database size to a minimum. You can also set database size quotas to specify the maximum size a database can attain. Refer to the “Database size quotas” and “Setting database size quotas” articles in Lotus Domino 6.5 Administration Help for detailed descriptions of working with database size quotas.

Compacting databases
When documents and attachments are deleted from a database, Domino tries to reuse the unused space, rather than immediately reduce the file size. Sometimes, Domino will not be able to reuse the space or, because of fragmentation, cannot reuse the space effectively until you compact the database. Compacting databases helps you to keep them smaller and so you gain some performance points.

Note: Lotus Domino 6.5 server is much better at reusing space in 6.5 format databases, so compacting is needed less often than in past releases.

There are three styles of compacting used in Domino 6.5:

- In-place compacting with space recovery
  This style of compacting recovers unused space in a database but does not reduce the size of the database on disk. Databases retain the same database instance IDs (DBIIDs), so the relationship between the compacted databases and the transaction log remains intact. Users and servers can continue to access and edit databases during compacting. This style of compacting is useful for databases that you expect to stay the same size or to grow in size.
  When you run Compact without specifying options, Domino uses this style of compacting on all databases enabled for transaction logging. Domino also uses this style of compacting when you use the -b option (case sensitive) when compacting any database.
  Use this compacting method the most frequently, because it is the fastest method and causes the least system impact.

- In-place compacting with space recovery and reduction in file size
  This style of compacting reduces the file size of databases, as well as recovers unused space in databases. This style of compacting is somewhat slower than in-place compacting with space recovery only. This style of compacting assigns new DBIIDs to databases, so if you use it on logged databases, and you use a certified backup utility, perform full backups of the databases shortly after compacting is complete. This style of

Tip: If you upgraded to Release 6.5 from an earlier version, compacting a database will automatically upgrade the database format (on-disk structure, or ODS 43) to Release 6.5.

Note: You still have to take into account possible limitations of the underlying file system. For example, the ext2 file system, which is still the default for most major Linux distributions, can only support files up to 2 GB in size. Prepare for this item before you deploy Domino databases.
compacting allows users and servers to continue to access and edit databases during compacting.

When you run Compact without specifying options, Domino uses this style of compacting on databases that aren't enabled for transaction logging. Domino also uses this style of compacting when you use the -B option. To optimize disk space, we recommend that you run Compact using the -B option on all databases once a week or once a month.

- **Copy-style compacting**

  Copy-style compacting creates copies of databases and then deletes the original databases after compacting completes, so extra disk space is required to make the database copies. This style of compacting essentially creates a new database with a new database ID. If you use copy-style compacting on logged databases (using the -c option), compacting assigns new DBIIDs, so if you use a certified backup utility, you should perform full backups of databases shortly after compacting completes. When you use copy-style compacting, users and servers cannot edit databases during compacting, and they can only read databases if the -L option is used.

  Domino uses copy-style compacting by default when you use an option with Compact to enable a database property that requires a structural change to a database or when you run Compact on a database that has a structural change pending that was initiated from the Database Properties window. Enabling or disabling the database properties “Document table bitmap optimization” and “Don’t support specialized response hierarchy” require structural database changes.

---

**Defining the number of databases cached simultaneously**

If your server has sufficient memory, you can improve the performance of the server by increasing the number of databases that Lotus Domino can cache in memory at one time. To do so, use the NSF_DbCache_Maxentries statement in the NOTES.INI file. The default value is 25 or the NSF_Buffer_Pool_Size divided by 300 KB, whichever value is greater. The maximum number of databases that can be cached in memory is approximately 10,000. For short intervals, Lotus Domino will store up to 1.5 times the number entered for this variable.

Monitor the Database.DbCache.Hits statistic on your server. This indicates the number of times a database open request was satisfied by finding the database in cache. A high value indicates database cache is working effectively. If the ratio of Database.DbCache.Hits to InitialDbOpen is low, you might consider increasing NSF_DbCache_Maxentries.

To set the number of databases that a server can hold in its database cache at one time, set the NOTES.INI value as follows:

```
NSF_DbCache_Maxentries = [number]
```

In special circumstances, you might also want to disable the database caching. The database cache is enabled by default. To disable database caching, enter the following syntax on the Domino console:

```
Dbcache disable
```

The database cache keeps databases open. Use this command to disable the database cache when you need exclusive access to a file that might be in the database cache. For example, to run an application such as a virus checker or backup software, disable the cache. To re-enable the database cache, restart the server.

---

**Optimizing database index update**

Maximize disk space used for database view indexes and minimize the effort required to update view indexes by the Indexer server task. In general, the fewer view indexes that the Indexer server task must update, the fewer server resources are used by this task. You can...
use the NOTES.INI variable Default_Index_Lifetime_Days to minimize the amount of effort required by the Indexer task when updating view indexes. The Default_Index_Lifetime_Days variable controls how long a view index is kept in a database before it is deleted due to non-use:

Default_Index_Lifetime_Days = [number of days]

The default value for this variable is 45 days unless a database designer has specified a different lifetime by clicking Design → View Properties → Advanced Tab → Discard index to reach the dialog box shown in Figure 4-10.

![Figure 4-10 View window](image)

Setting this number to a lower value will reduce the number of view indexes on a server and therefore reduce the amount of time and effort that the Indexer server task must spend on updating view indexes.

**Full-text indexes**

Disable the updating of full-text indexes on a server if you do not have any full-text indexed databases on your server (and do not intend to have any). The NOTES.INI variable Update_No_Fulltext can be used to disable all full-text index updating on the server. You might want to use this variable to disable the updating of full-text indexes if, for example, you do not want users to create full-text indexes on their mail files on a mail server in order to save disk space on that server (the space that would be used for the full-text indexes) and to save the Indexer task the time and resources of updating these full-text indexes. This is a very good setting for mail and replication hub servers, which in most circumstances do not have any user connections. The full-text index variable is:

Update_No_Fulltext = [0 or 1]

Setting this value to 0 causes full-text indexes to be updated each time the Update task (a server command task) is executed, and setting the value to 1 disables all full-text indexing.
Tuning memory buffers

One of the most important memory pools in a Lotus Domino server is the NSF buffer pool. This memory pool is used for buffering I/O transfers between the NSF and NIF indexing functions and disk storage.

The memory pool size is automatically calculated by the server, except for partitioned servers. The maximum size depends on the limitations of the operating system and the amount of system memory available. The minimum size is 4 MB. On the server, the default maximum size is determined to be between 1/8 and 3/8 of the available physical memory, depending on the overall size of the physical memory.

The defaults are not automatically adjusted on partitioned servers, so it will usually be necessary to adjust the maximum values in each partition to a fraction of memory such that the memory used by all partitions adds up to approximately 1/4 to 3/8 of the memory. To increase or decrease the default size of the NSF buffer pool, the following value must be changed within the NOTES.INI file:

\[ \text{NSF\_Buffer\_Pool\_Size} = \text{[bytes]} \]

We recommend, however, that the NSF buffer size be allowed to take the default value. Changing this value from the default will stop the automatic calculation of this memory pool by the server, and your server's performance might actually worsen. Some Notes application vendors have set this parameter to 300-400 KB per active user.

On workstations, the default maximum setting of the NSF_Buffer_Pool_Size is 8 MB. The default value is usually sufficient, but if users get an error telling them to increase the value of this variable, the value should be increased a few megabytes at a time. You can use the Performance Monitor to find out if a larger value is causing too much swapping or paging.

**Note:** Use NSF_Buffer_Pool_Size_MB to avoid the 2 GB limitation that exists for NSF_Buffer_Pool_Size due to NOTES.INI variable limits.

Displaying images after documents

With large images or slow connections, loading images in order might slow the display of the document. To display documents of a database that contain images quickly, follow these steps:

1. Open the database.
2. Select File → Database → Properties.
3. Select the Basics tab.
4. Select the Display images after loading option, as shown in Figure 4-11 on page 49.
Lotus Notes users can then read the text while the images load. If you do not load images after text, Notes loads images in the order in which they appear in a document; if an image appears first, Notes loads it before displaying text.

**Note:** You can also set the Load images field to **On request** in the Advanced section of a Location document to display images only when users click them.

### Disabling the unread marks maintenance

By default, Lotus Domino 6.5 server maintains unread marks for each database. This takes system resources and can impact database performance. For many databases, such as mail files and discussion databases, unread marks are indispensable. But for many databases, this feature does not serve a useful purpose. Databases such as LOG.NSF, the Domino Directory, and the Help and Administration databases all belong to this category.

To disable the unread marks, follow these steps:

1. Open the database.
2. Select **File → Database → Properties** to open the Database Properties window.
3. Select the **Advanced** tab.
4. Select the **Don’t maintain unread marks** option, as shown in Figure 4-12 on page 50.
Optimizing document table bitmap

By default, during view updates and rebuilds, Domino searches each table for documents that appear in the view being updated. To update views more efficiently, do the following:

1. Open the database.
2. Select File → Database → Properties.
3. Select the Advanced tab.
4. Select the Optimize document table map option, as shown in Figure 4-12.

This property associates tables with the forms used by the documents the tables contain. Then, during a view update, Domino searches only the tables associated with the forms used by documents in the view being updated. This significantly improves the performance of view updates, especially updates of small views within large databases as, for example, the Connections view in the Domino Directory.

This property only works for views that use Form= as part of the selection criteria. There is a slight performance cost to maintaining the table/form association; however, when updating small views in large databases, the benefits offset the cost.

If you select or clear Optimize document table map, you must compact the database so that the setting takes effect. You can also run the compact server task with the -F or -f option to enable or disable this property and then compact.

Preventing overwriting of deleted data

By default, Lotus Domino 6.5 server overwrites data that has been deleted from a database to prevent users from accessing the deleted data. This feature improves security, but at the expense of more disk I/O, and therefore affects database performance.

You can consider disabling overwriting of deleted data if the following requirements are met:

- The data is already secured using other methods or security is not an issue.
- Deleted space in the database is constantly reallocated, such as for mail.box databases.
To prevent the overwriting of deleted data per a database, complete the following steps:

1. Open the database.
2. Select **File → Database → Properties**.
3. Select the **Advanced** tab.
4. Select the **Don’t overwrite free space** option, as shown in Figure 4-12 on page 50.

**Do not maintain Accessed document property**
The Document Properties box displays the Accessed property, which shows the date a document was last modified or read. The Maintain LastAccessed property selection controls whether the Accessed property is updated if the last document access was a read. Maintaining the Accessed property for reads causes disk I/O that would not otherwise occur.

By default, the database property Maintain LastAccessed is not selected, meaning that the Accessed (in this file) property is not updated when the last document access was a read, but only when the last access was a document modification.

You should consider selecting the Maintain LastAccessed property only in exceptional cases, for example, if you use the document archiving tool, available in the Database Properties window, to delete documents based on days of inactivity.

To prevent maintaining the Accessed document property, complete the following steps:

1. Open the database.
2. Select **File → Database → Properties**.
3. Select the **Advanced** tab.
4. Clear the **Maintain LastAccessed property** option if selected; see Figure 4-12 on page 50.

**Disabling specialized response hierarchy information**
By default, every document stores information that associates it with a parent document or a response document. Only two @ functions use this stored information. Maintaining this information has a significant, negative effect on database performance.

To improve database performance, disable the response hierarchy information in databases that do not use @AllChildren and @AllDescendants functions.

To disable the response hierarchy information, complete the following steps:

1. Open the database.
2. Select **File → Database → Properties**.
3. Select the **Advanced** tab.
4. Select the **Don’t support specialized response hierarchy** option, as shown in Figure 4-12 on page 50.

If you select or clear the Don’t support specialized response hierarchy property, you must compact the database so that the setting takes effect. You can also run the compact server task with the -h or -H option to enable or disable this property and then compact.

**Preventing headline monitoring**
Users can set up headline monitoring to automatically monitor databases for information that interests them. Monitoring a database this way affects performance, especially if many users do this.
To prevent users from monitoring a database, perform the following steps:

1. Open the database.
2. Select **File** → **Database** → **Properties**.
3. Select the **Advanced** tab.
4. Select the **Don't allow headline monitoring** option, as shown in Figure 4-12 on page 50.

Administrators can also use the Security section of a Server document in the Domino Directory to control headline monitoring at the server level.

**Limiting the number of entries in the $UpdatedBy field**

Every document includes an $UpdatedBy field that stores the names of users or servers who have edited the document. If the document has been heavily edited, the number of entries in the field can be large and both consume disk space and degrade the database performance. If there is no need to track update information, you can either limit the field size or disable the function totally by setting the value to 0.

To set the number of entries for these fields for each database, complete the following steps:

1. Open the database.
2. Select **File** → **Database** → **Properties** to display the Database Properties window.
3. Select the **Advanced** tab.
4. Enter the limit into the **Limit entries in the $UpdatedBy fields** box, as shown in Figure 4-12 on page 50.

**Limiting the number of entries in the $Revisions field**

The $Revisions field of a document stores the dates and times when changes were made to the document. This field is used to resolve, for example, replication conflicts. By default, the field stores up to 500 edit sessions. When the limit is reached, the oldest entry on the list is dropped.

You should consider limiting the entries in this field when the following characteristics apply:

- The database contains many documents.
- The database replicates often or has no replicas.
- Database documents are not subject to much editing.

The recommended value is 10 entries. If you set the limit to less than 10, you might see more replication or save conflicts. You can prevent the overwriting of deleted data in your databases.

To set the value for each database, complete the following steps:

1. Open the database.
2. Select **File** → **Database** → **Properties**.
3. Select the **Advanced** tab.
4. Type the limit into the **Limit entries in $Revisions fields** box, as shown in Figure 4-12 on page 50.
4.2.6 Calendaring and scheduling resource usage

Lotus Domino provides a great feature set to support calendaring and scheduling (C&S) for Lotus Notes users. You can expect that C&S tasks will increase resource usage to varying degrees in these basic ways:

- Workflow-oriented tasks, such as meetings, invitations, and responses create messages that are sent over the Domino messaging infrastructure. These messages tend to be short and are efficiently routed with minimal impact on the server.

- Freetime searches rely on the Notes RPC mechanism to get information from the clients’ home servers. Fundamentally, these searches are database queries to the busytime.nsf database, which are submitted through the client’s home server and possibly chained to other servers. The result set is small and is delivered with a minimum of network overhead. When requests need to be serviced by another server, the response is delivered efficiently without replication of data.

  Busytime is a very compact data structure that is purged regularly, thus minimizing data storage requirements. Typically, the busytime.nsf database can support hundreds of users with a few megabytes of data storage.

- The alarm system can add minimal overhead in conditions of extremely heavy usage. With alarms enabled, the alarm daemon process on the client will poll the home server for new alarms every 10 minutes, opening the user’s mail database to do so. If all calendaring users on a server enable this option, a significant number of additional RPC requests will be made against individual mail files. These requests are relatively insignificant compared to other mail activity on the server. However, administrators might notice additional transaction counts and log session activity because of this feature.

- Room and resource reservations are a final area for consideration. Rooms and resources are implemented as separate Domino database applications. Resource requests are processed by an agent, which determines whether the room or resource is available. If it is available, the agent creates a reservation in the database and sends an acceptance notice back to the client.

- Additional server processing bandwidth is required for specific calendaring processes: Calconn (Calendar Connector), which chains the freetime requests to other servers, and Sched (Schedule Manager), which updates the busytime.nsf database as users update their calendars. As described earlier, the freetime request is a chained RPC call to a remote server, so the Calconn overhead is minimal. Typically, it is not even measurable by monitoring tools.

- In addition to basic requirements for processing tasks and serving data previously listed, you can also expect to use additional hard disk space. Calendaring and scheduling adds documents to users’ mail files. The exact amount of overhead depends on the usage profile; the impact of a small text message is minimal, for example. However, an organization that uses detailed meeting agendas with large object attachments might eventually start pushing disk storage limits. Of course, this problem pertains to all mail-enabled applications. Good systems management practices can help administrators monitor the growth of application files.

Calendaring and scheduling has a minimal impact on network capacity and a small but measurable impact on server resources. In many cases, customers might find the total effect to be less than that of a traditional file-sharing system for calendaring.

4.2.7 Minimizing logging activity

Lotus Domino 6.5 server provides a variety of powerful logging options that gives an administrator the ability to collect statistics on a wide range of activities and use them for
security reasons, performance tuning, or deployment tasks. However, the intensive logging activity might have a significant negative impact on server performance and should be carefully considered.

**Notes log file optimization**

Keep the Notes log file (log.nsf) as small as possible and minimize the amount of database activity that is logged. Minimizing the activity of the Notes log will help increase server performance by minimizing the updates to the log file.

In general, you should log Lotus Domino server activities only when you are keeping regular statistics on certain activities, trying to diagnose a problem, or tracking the use of a specific resource. The following NOTES.INI variables all apply to logging Lotus Domino server activities and can be disabled if the results of their logging are not being used:

```
Log=logfilename, log_option, not_used, days, size
Log_AgentManager=value
LogDirCat=value
Log_Replication=value
Log_Session=value
Log_Tasks=value
Log_Update=value
Log_View_Events=value
PhoneLog=value
```

Although these settings do not have a UI equivalent and must be set in the NOTES.INI file, we recommend that you specify these setting in the NOTES.INI Settings tab of the Configuration Settings document in the Domino Directory.

For a complete set of Lotus Domino 6.5 server logging options, refer to Table 31 in *Netfinity and Domino R5.0 Integration Guide*, SG24-5313.

**Managing database activity recording in databases**

By default, Statlog reports database activity to all database User Activity dialog boxes when it runs. Even if a user disables User Activity reporting for a specific database, the next time Statlog runs, it enables recording in the dialog box again.

To prevent Statlog from automatically recording activity in User Activity dialog boxes, add `No_Force_Activity_Logging=1` to the NOTES.INI file. Then, you can enable activity recording per database, as needed. Because recording activity in the User Activity dialog box adds 64 KB to the size of each database, disabling automatic activity recording saves disk space on the server. The syntax of this variable is as follows:

```
No_Force_Activity_Logging = [0 or 1]
```

**Note:** If you use `No_Force_Activity_Logging`, Statlog still reports activity to the log file

Even if the server administrator uses the No_Force_Activity_Logging setting in the NOTES.INI file to disable automatic activity recording in databases, you can enable recording for a single database and use this logging option more selectively, as shown in Figure 4-13 on page 55.

To enable recording for a single database, complete the following steps:

1. Open the database, and select **File → Database → Properties**.
2. Select the **i** tab, and then click **User Detail**.
3. Select **Record activity** to enable activity recording.
4. Click **OK**.

![Figure 4-13  Selective enabling of database activity recording](image)

**4.2.8 Improving Agent Manager performance**

The Agent Manager controls when agents run on a server. Every time an agent runs, it uses server resources. To control when scheduled and event-triggered agents run, you specify settings in the Server document and in the NOTES.INI file. Customizing when agents run can conserve server resources, but it can also delay when agents run.

**Controlling when and how often the Agent Manager runs agents**

In general, the more frequently agents run, the sooner they perform their tasks. Running agents more frequently, however, can increase demand on server resources and adversely affect overall system performance.

When you create or modify an event-triggered agent, the Agent Manager schedules it to run immediately. This ensures that the agent can quickly process new documents. These NOTES.INI settings let you specify a time interval between subsequent runs of the agent. This can prevent repeated runs of the agent, for example, because of a rapid series of triggering events. Table 4-6 describes settings to control when and how often the Agent Manager executes agents.

<table>
<thead>
<tr>
<th>Statement = [minutes]</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMgr_DocUpdateEventDelay default = 5 minutes</td>
<td>Delay of execution that the Agent Manager includes in the schedule for a document update-triggered agent after a document update event. A longer interval results in the agent running less often, thus reducing demand for server time.</td>
</tr>
<tr>
<td>AMgr_NewMailEventDelay default = 1 minute</td>
<td>Delay of execution that the Agent Manager includes in the schedule for a new mail-triggered agent after a new mail message is delivered. A longer interval results in the agent running less often, thus reducing demand for server time.</td>
</tr>
<tr>
<td>AMgr_DocUpdateAgentMinInterval default = 30 minutes</td>
<td>The minimum elapsed time between executions of the same document update-triggered agent. Entering an interval can result in the agent running less frequently, reducing server demand.</td>
</tr>
<tr>
<td>AMgr_NewMailAgentMinInterval default = 0 (no interval between executions)(^a)</td>
<td>The minimum elapsed time between executions of the same new mail-triggered agent. Entering an interval can result in the agent running less frequently, reducing server demand.</td>
</tr>
</tbody>
</table>

\(^a\) Setting this and other Agent Manager variables to zero does not completely eliminate the delay; a built-in delay will always exist.
Controlling how quickly the Agent Manager queues agents

The Agent Manager periodically checks to see whether it has any new agents that it needs to schedule. These NOTES.INI settings control how quickly an agent gets into the schedule queue. Table 4-7 shows which settings control how quickly the Agent Manager queues agents.

Table 4-7  Controlling how quickly the Agent Manager queues agents

<table>
<thead>
<tr>
<th>Statement = [minutes]</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMgr_SchedulingInterval default = 1 minute</td>
<td>Delay between runnings of the Agent Manager’s scheduler</td>
</tr>
<tr>
<td>AMgr_UntriggeredMailInterval default = 60 minutes</td>
<td>Delay between runnings of the Agent Manager’s check for untriggered mail</td>
</tr>
</tbody>
</table>

Controlling how many concurrent agents are running

You can relieve a heavily loaded Agent Manager by allowing agents to run concurrently. To do this, modify the Max concurrent agents field in the Server Tasks/Agent Manager section of the Server document, as shown in Figure 4-14. Values greater than 1 allow more than one agent to run at the same time. Valid values are 1 through 10. Default values are 1 for daytime and 2 for nighttime.

4.2.9 Managing server sessions

You might want to set the maximum number of sessions that can run concurrently on your server. This is configured to prevent the server overloading because of a large number of users connecting to your server simultaneously. You can also define an inactivity period after which the server automatically terminates user connections. It will help you to regain resources they take up.
Maximum sessions

When a new user attempts to log on, if the current number of sessions is greater than the 
value of Server_MaxSessions (in the NOTES.INI file), the Lotus Domino server closes the 
least recently used session. In order for a session to be considered for closing, it must have 
been inactive for at least one minute. For example, if this parameter is set to 100, and the 
101st person tries to access the Lotus Domino server, the Lotus Domino server drops the 
least-used session from the server in favor of this new session.

Reducing maximum server sessions: Reducing the Server_MaxSessions value to a 
specific number will not prevent the server from allowing more than that number of 
concurrent active users on the server, but will drop the sessions soon after they become 
inactive. This frees up resources. Conversely, Domino will not close any session that has 
been idle for less than one minute regardless of the demand on the server.

Session details are found by using the following console command:

SHOW STAT NET

Sample output of this command is shown in Figure 4-15.

> show stat net
NET.TCPIP.BytesReceived = 13,512
NET.TCPIP.BytesSent = 663,232
NET.TCPIP.Sessions.Established.Incoming = 3
NET.TCPIP.Sessions.Established.Outgoing = 0
NET.TCPIP.Sessions.Limit = 65535
NET.TCPIP.Sessions.LimitMax = 65535
NET.TCPIP.Sessions.LimitMin = 10
NET.TCPIP.Sessions.Peak = 6
NET.TCPIP.Sessions.Recycled = 0
NET.TCPIP.Sessions.Recycling = 0

Figure 4-15 Sample output from the SHOW STAT NET command

Note: This server had a single network port named TCPIP. If your server has a different 
port name, this is shown after the word NET.

The key session details you want to investigate are:

- Sessions.Established.Incoming
- Sessions.Established.Outgoing
- Sessions.Peak

Note that these session values are per network port. So, if you have more than one network 
card in your server, you will need to add up all session values. If your server is overloaded, 
use these values to determine the value of the Server_MaxSessions value. The format of the 
Server_MaxSessions parameter in the NOTES.INI file is:

Server_MaxSessions = [number]
Session timeouts
You can specify the number of minutes of inactivity after which the server automatically terminates network and mobile connections. The default session timeout is four hours; the minimum recommended setting is 30-45 minutes. A lower setting might negatively affect server performance. The ideal setting depends on factors such as server load and the number of concurrent users on the server.

Inactive database server sessions do not take up many resources on a Lotus Domino server, but it is a good idea to minimize them anyway in order to regain the resources that they do take up. To do this, use the following NOTES.INI variable:

Server_Session_Timeout = [minutes]

Note: For mobile connections, XPC has its own internal timeout. If the XPC timeout value is shorter than the Server_Session_Timeout value, the XPC timeout takes precedence.

4.2.10 Controlling user access
To efficiently manage a Lotus Domino server, you might need to control the server access of your users. You can, for example, limit the maximum number of users accessing your server to protect the server against overloading or define Lotus Notes user access values that will override the client settings.

Limiting the maximum number of users
To protect your server from overloading, you can set the maximum number of users that are allowed to access a server. When you set this value and the maximum user limit is exceeded, Notes users are prompted with a message at their workstation when they try to access a database on the server, as shown in Figure 4-16.

To set the maximum number of users, set the following NOTES.INI value:

Server_MaxUsers = [number]

The default value is 0, which indicates unlimited access to the server by users.

Controlling minimum mail poll time
You can control the minimum allowable time in which Lotus Notes clients can poll for new mail. It is possible that your Lotus Domino server resources are being overtaxed by being constantly bombarded by requests for new mail from the Notes client machines if users have changed their default new mail notification check time from 15 minutes to a smaller number such as 2 or 5.

You can control the minimum frequency of these requests from the server by using the MinNewMailPoll NOTES.INI variable. This variable determines the minimum allowable checking time from clients regardless of the value specified on the client machines. No default is set during server setup. The syntax of this variable is as follows:

MinNewMailPoll = [minutes]
4.2.11 Improving replication performance

Streaming replication is a new Domino 6 feature enabled by default on servers and clients. To use streaming replication, both parties involved in the replication must be running Notes/Domino 6. Streaming replication is PULL only. This feature uses a single streamed RPC to read a number of documents and attachments. This is an improvement over the non-streaming method of requesting and acknowledging one database note at a time.

Streaming replication saves time and consumes less CPU and network activity to replicate the same number of documents. Network latency is reduced, especially on slower links. Documents are also replicated in ascending size order (smallest first). This allows the client to abort a long replication, but still receive some documents (for example, mail). Partial replication is also supported, allowing an aborted replication to be reinitiated and to continue where it left off. Views in folders are also updated incrementally while a replication is in progress (as opposed to the end of replication). This allows users to begin responding to initial documents (for example, mail messages) while larger documents continue downloading.

End-user benefits of streaming replication include:

- Shorter replication times.
- Ability to start viewing and working on documents before replication is completed.
- Replication of smallest documents first.
- If replication is cancelled, the target system still has the documents (especially smaller ones) that were replicated before the replication session ended.
- Incrementally updated folders.
- You no longer have to wait until the replication is over before seeing replicated documents in folders.

Advantages for IT departments include:

- Faster replication
- Partial replication
- Potentially less network traffic due to single streamed RPC and reduction of ACK (TCP/IP) responses

Here are some best practices to consider when managing system replication:

- Be careful about overscheduling the server. When the server becomes overloaded, calls back up, mail is not delivered, and users might receive poor service from the server.
- Check the log file log.nsf to check that the databases are replicating properly, mail is routing promptly, and the server is not overloaded.
- If necessary, modify the connection documents and make adjustments until the problem is solved.
- Use selective replication to replicate only documents from a certain person, about a certain topic, or subsets of databases.
- If replications are taking too long, change the schedule to replicate more often so that there are fewer updates per replication.
- Schedule replication at off-peak hours. If you are replicating internationally, consider the time zones for the source and destination servers and plan accordingly.
For the most dramatic improvement in performance, set up multiple replications so that a server, such as the hub, can replicate with multiple servers simultaneously. This should improve replication performance immediately.

Use replication options to shorten replication times. For example, if a hub server replicates to 50 servers, and you want to make sure an entire replication cycle occurs twice a day, limit the time the hub connects to each server. Be sure to check the log to see which databases replicate completely and which do not.

Set the replication priority to high, medium, or low to replicate databases of different priorities at different times. Set up replication groups based on replication priority.

Develop a policy that controls how database replicas get placed on servers. Creating unnecessary replicas consumes system resources.

Check the Statistics & Events database for events indicating server problems.

Setting up multiple replication tasks
You can improve server replication performance by running multiple replicator tasks simultaneously. By default, only one replicator task is executed. With a single replicator task running, if you want to replicate with multiple servers at the same time, the first replication must complete before the next one can start. Set the number of replicators by adding the following entry in the NOTES.INI file, where the value is the number of replicators:

Replicators = [number]

All other factors aside, we recommend that you set the number of replicator tasks equal to the number of spoke servers with which the hub replicates. However, you should not exceed 20 replicators to avoid putting too much load on the server. If the server you intended to replicate is not a hub server, the recommended number of replicators should equal the number of processors on the server.

Bear in mind that your network design might affect the optimal number of replicators for your implementation. If you are replicating over a single 256 Kbps frame relay connection to your spoke servers, for example, adding extra replicator tasks would not provide any performance improvement. This is because the bottleneck is not the replicator task but the network link itself. With a 155 Mbps ATM connection, you might obtain better utilization by changing the value.

4.2.12 Enabling transaction logging
Transaction logging is available in Lotus Domino 6.5 servers. With transaction logging, each transaction is transferred to its database not directly, but by posting sufficient information to complete the transaction to a high-speed access sequential log file. The transaction is finally posted to the database at some later time. Data in the log, but not yet posted to the database, can be recovered in the event of a server failure.

Tests have shown that enabling transactional logging gives a performance advantage of up to 20%, depending on the exact server configuration. The Lotus Domino transaction log is actually a set of sequential access files used to store the changes made to the databases. Sequential disk access to a dedicated physical disk is noticeably faster than random disk access.

Transaction logging disk optimization: For optimal performance, transaction logs should be placed on a separate physical disk device. If you place the logs on the same device as the databases, you lose the benefit of sequential access, and there is no performance improvement or very little.
Transaction logging also has system-level benefits. At the operating system level, all the file and disk operations are carried out by the kernel running in privilege mode. This requires a context switch to the kernel thread to handle the task of moving data to and from disk. Transaction logging improves the overall I/O throughput of Lotus Domino; therefore, it reduces kernel time and increases user time, which reduces overall processor time to do the same work.

To enable or disable transaction logging for all Lotus Domino 6.5 databases on the server, set the Transactional logging field to **Enabled** in the Transactional Logging tab of the Server document, as shown in Figure 4-17.

![Server: Domino651/ITSO](image)

You might also want to tune the Runtime/Restart performance value based on your needs. Runtime/Restart performance specifies the trade-off between transactional log runtime and restart recovery time. It is set through the “Runtime/Restart performance” field in the Transactional Logging tab of the Server document, as shown in Figure 4-17.

The available options are:

- Favor runtime. The system stores more database changes in memory and writes fewer changes to the transaction log. Having fewer writes to disk improves server runtime.
- Standard (default).
- Favor restart recovery time. The system stores fewer database changes in memory and writes more changes to the transaction log. Having more writes to the transaction log improves restart recovery time.

### 4.2.13 Using port encryption

You can encrypt network data on specific ports to prevent the network eavesdropping possible with a network protocol analyzer. Network encryption occurs at the network transfer layer of a selected protocol and is independent of other forms of encryption. Network data is encrypted only while it is in transit. After the data has been received and stored, network encryption is no longer in effect.
Network data encryption occurs if you enable network data encryption on either side of a network connection. For example, if you enable encryption on a TCP/IP port on a server, you do not need to enable encryption on the TCP/IP ports on workstations or servers that connect to the server.

Multiple high-speed encrypted connections to a server can affect server performance. Encrypting network data has little effect on client performance. In addition, encrypted network data cannot be compressed. Therefore, if you encrypt network data on a port that uses a data-compressing modem, you do not gain the throughput benefits from the modem.

In testing port encryption versus no port encryption, the relative processor utilization increases in the 5-10% range. Also, user response is slightly affected. However, if you are running the server in the less than 60% processor utilization range, the user's response time and performance degradation will not be significant.

Note: Network compression in Domino 6.5 will not yield benefits with bandwidth savings if the data is encrypted (such as SSL for Domino Web Access).

4.2.14 Lotus Domino partitioning

Using Domino server partitioning, you can run multiple instances of the Domino server on a single computer. By doing so, you reduce hardware expenses and minimize the number of computers to administer because, instead of purchasing multiple small computers to run Domino servers that might not take advantage of the resources available to them, you can purchase a single, more powerful computer and run multiple instances of the Domino server on that single machine.

On a Domino partitioned server, all partitions share the same Domino program directory, and thus share one set of Domino executable files. However, each partition has its own Domino data directory and NOTES.INI file; thus, each has its own copy of the Domino Directory and other administrative databases.

If one partition shuts down, the others continue to run. If a partition encounters a fatal error, the Domino fault recovery feature restarts only that partition, not the entire computer.

Use Lotus Domino partitioning technology to increase scaling of active user loads and to leverage more powerful configurations such as faster clock cycles, fibre-connected I/O subsystems, OS kernel to CPU binding, and multiple I/O controllers.

The Server_Max_Concurrent_trans NOTES.INI setting sets the limit for the number of concurrently scheduled transactions on a server. If you use this setting to set the maximum number of concurrent transactions on partitioned servers, Lotus recommends that the sum of the limits be 20 transactions or fewer. For example, if you are running four partitioned servers on a computer, you would set the limit for each partitioned server at five transactions. The syntax of the statement is as follows:

Server_Max_Concurrent_Trans = [number]

4.2.15 Balancing workload in a Lotus Domino cluster

If you are going to scale Domino to higher user loads, you might need to use Domino clustering. This will divide the load among the servers and you can take advantage of high-end hardware configurations.

Domino clusters provide you with scalability. It is easy to add servers to a cluster and also to remove them from a cluster as the workload changes. You can also balance the workload
among servers. This important feature lets the system distribute workload at the same time demand fluctuates or in the event of a server failure. Workload balancing lets the clustering technology distribute the workload dynamically, based on performance thresholds set in each server.

**Server availability threshold**
The server availability threshold specifies the lowest acceptable server availability index. By setting this percentage value for each server in a cluster, you determine how the workload is distributed among cluster members.

Approximately once each minute, Domino computes the server availability index and compares it to the server availability threshold that you set. If the availability index is less than the availability threshold, the server is marked as BUSY. When a server is marked as BUSY, requests to open databases are redirected to another server, if one is available. When the availability index becomes higher than the availability threshold again, the BUSY condition is withdrawn.

To set the server availability threshold, add the following line to the NOTES.INI file:

```
Server_Availability_Threshold = [number]
```

The higher the number you enter, the less workload the server can carry before going into the BUSY state. Entering the number 100 automatically puts the server into the BUSY state, regardless of its actual availability. Entering the number 0 disables workload balancing for that server. The default value is 0.

The server availability threshold is a key configuration setting for Domino cluster workload balancing. Setting the threshold too high can cause user requests to fail unnecessarily. Setting the threshold too low can result in poor performance for some users who might have received better service from another server. You should also consider how a server outage can affect server workload. For the fail-over capability, you might want to set the server availability threshold to allow some extra capacity to handle the fail-over workload.

**Network traffic considerations**
Cluster replication is constantly updating replicated databases over your network. These are not scheduled replication events; they are real-time updates to the databases. Cluster replication transactions are primarily small record updates, but you might be concerned about potential congestion on your network. In practice, many Domino cluster installations have found out that the network overhead is actually quite small. However, you do have the option of isolating cluster replication traffic by setting up a private LAN dedicated to intra-cluster communication between servers.

The additional expenditure required to do this is very small; all you need is an additional network adapter in each server, a hub, and wiring. A parameter in each server's NOTES.INI file, `Server_Cluster_Default_Port`, specifies which port is used for intra-cluster network traffic.

### 4.3 Maintaining Lotus Domino servers
Lotus Domino provides a wide range of tools for gathering information about a server and for diagnosing and fixing problems. Include the following maintenance tasks in your administration plan:

- Back up the server regularly. This is especially important for database recovery.
- Check for dead mail.
- Use the Administration Process to remove users’ names globally from access control lists (ACLs).
- If necessary, run the Database Fixup utility manually to fix corrupted databases and prevent server crashes.
- Check the log file for problems with replication, mail routing, scheduling, database integrity, and communication links.
- Monitor server statistics and events to track disk space, server load, memory and swap space, and to head off server trouble.
- Monitor database replication and access control list changes using event monitors.
- Periodically run the Compact utility (through program documents) to keep wasted database free space to a minimum.

### 4.4 Planning for future growth

The scalability, or capacity for expansion, of your system depends on how well you plan for growth. For example, if you initially deploy your system with just a few servers and decide on a peer-to-peer topology for server communication, you might need to rethink your topology as you add servers.

In addition, you might need to upgrade your network and computing infrastructure as you add servers and users. If you find yourself in this situation, consider deploying a Domino server cluster as a way to achieve scalability with the added advantages of server and database availability.
Monitoring Domino performance on xSeries and BladeCenter

This chapter provides guidelines and references to monitoring tools available in Domino 6.5. We also describe IBM Tivoli® products available to assist in managing your environment.

Analysis of the Domino 6.5 server can be achieved using the built-in system monitoring tools. These tools will also provide statistics, and if requested, alerts on operating system activity.
5.1 System monitoring tools

The Domino Administrator includes these system monitoring tools that you use to configure, view, and track the Domino system:

- **Statistics**: Domino gathers statistics that show the status of processes currently running on the system, for example, the statistic free space on drive C indicates the amount of free space available on drive C. You use these statistics along with the predetermined statistics thresholds to monitor both your Domino system and platform statistics.

- **Monitoring databases**: Store monitoring documents, information, and results. The Monitoring Configuration database stores the documents you use to set up monitoring. It also includes information about statistics, statistic thresholds, and event messages. The Monitoring Results database stores the gathered statistics reports and can be configured to store information about logged events. The log file stores the server's activity log documents.

- **Monitoring Configuration documents**: Define and configure what constitutes an event and how the event is handled. You can also customize the messages that appear on the console when an event occurs.

- **Server tasks**: Collect and record information about the Domino system. The Event Monitor task determines if an Event Handler has been configured for the event, and if so, routes the event to the specified person, database, or server management program for processing. The Statistic collector task gathers Domino server statistics and creates statistics reports in the Monitoring Results database or to another database you can specify.

- **Domino server monitor**: Provides a visual representation of the status of the servers you are monitoring.

Every occurrence that happens on the Domino system is an event. Events signal both that the system is working smoothly, processing data, and performing tasks and that the system is malfunctioning, perhaps by not processing data or performing required tasks.

Domino generates events continuously. Therefore, to monitor the Domino System efficiently, you must decide which events you want to know about. For example, the following event occurs every time a file replicates with this server; consequently, you might want to know about the event only if it fails:

```
The Event "Replicating files with Domino651/ITSO"
```

When you configure an event, you determine the three critical pieces of information:

- What type the event is
- What the severity is
- How you want it handled

The Event Handler is shown in Figure 5-1 on page 67.
You configure your events using Event Generator and Event Handler documents. Event Generators describe the condition that must be met for an event to be generated; Event Handlers describe what happens when the event occurs.

After deciding which events you want to know about, decide what will happen when the event occurs. You have several choices. You can log the event to the log file; you can mail a notification of the event to a file or an administrator; or you can mail the event to another application for further processing.

You create an Event Handler document to specify to log the event to a specified destination and simultaneously receive notification of the event's occurrence and run a program for additional processing. You can also prevent the event from being logged or handled at all. However, if you want to know about an event, you must have an Event Handler document. Otherwise, the event is not recorded. There is no default way of handling an event. So if you do not create event handlers, events are not logged or stored anywhere (except for server or add-in task events, which are stored in the log). After an event is passed to the Event Monitor task, it can invoke one or more configured Event Handlers.

For further information about these monitoring facilities in Domino 6.5, refer to the Lotus Domino Administration 6.5.1 Help database.

5.2 IBM Tivoli Monitoring

This section describes Tivoli tools for Domino.

5.2.1 IBM Tivoli Analyzer for Lotus Domino

Tivoli Analyzer for Lotus Domino provides a performance analysis and management solution that runs seamlessly within the IBM Lotus Domino 6 Administrator, helping to give IBM customers the ability to maximize the performance of their Lotus Domino server. Tivoli Analyzer for Lotus Domino demonstrates the tighter integration within the major software solutions of IBM, and provides customers with a built-in, easy-to-use solution that speeds deployment and increases the return on their IT investments.
Tivoli Analyzer for Lotus Domino is the first product to take advantage of new Lotus Domino 6 statistics and activity measurements, and one of the only products that runs seamlessly inside the Domino Administrator. Lotus Domino 6 servers generate comprehensive, detailed statistics and measurements of the server’s activity. This product offers a groundbreaking approach to Domino server management and uniquely analyzes these new statistics and activity measurements, turning raw data into clear and meaningful knowledge that administrators will use to take actions to improve the overall health of Domino servers, predict sizing requirements for the Domino environment, and predict capacity bottlenecks before they become critical.

IBM Tivoli Analyzer for Lotus Domino 6 provides:

- Powerful predictive analysis and intelligent server health check functionality that expand the capabilities of the Lotus Domino Administrator by evaluating hundreds of data points and operating system statistics. Through the constant assessment of the health of Lotus Domino servers, short-term and long-term performance can be improved, and the ability to meet service level agreements can be increased.
- Advanced resource planning tools necessary to predict growth within the Lotus Domino environment can reduce the amount of hands-on effort required and ultimately decrease the total cost of ownership.
- Workload balancing tools that match a server’s workload to the server hardware and operating system.
- Change management tools automate the workflow required to approve and then implement the workload balancing tool’s recommended plan.

For more information, visit the following Web site:

Domino Server Health Monitoring

Domino Server Health Monitoring is an add-on feature of Domino 6. It is provided in the IBM Tivoli Analyzer for Domino product, and is a tool that monitors and analyzes the status of Domino servers on a periodic basis. The health monitor integrates fully with the Domino 6 administration client.

After it is installed, and after setting a few options in the administration client, you will see a thermometer next to the server on the administration client monitoring tab. This thermometer will be green when the server is performing well, yellow if service is starting to degrade, and red if a major problem has occurred.

Server Health Monitoring is able to monitor all operating system statistics (AIX®, Microsoft Windows 2000 and Windows NT, OS/400, and Sun Solaris) and is smart enough to know how to evaluate the health of the servers using predictive algorithms and trend analysis. It has a collection of all the performance tuning options and settings for all the operating systems built-in. It is no longer necessary to read through articles or search the Web looking for performance tuning information or correct Domino settings. Server Health Monitoring is also smart enough to provide possible causes for the problems, enabling administrators and IT managers to proactively monitor their servers. They will be able to discover problems long before they have a critical impact on the user population.

Server Health Monitoring performs analysis over multiple monitoring intervals. This means that the Server Health Monitoring checks on these different areas, such as the operating systems statistics, Server Monitoring set, and Domino statistics, on a regular basis. Multiple monitoring intervals are reviewed before any assessments are made; this prevents making decisions based on spike conditions, such as server startup, or when various maintenance
activities, such as compact, are running. Instead, it looks for a sustained or repeated pattern over multiple intervals.

Server Health Monitoring is provided in the IBM Tivoli Analyzer for Domino product. Although this Tivoli product does not require any other Tivoli products as prerequisites, it can be used in conjunction with other Tivoli Monitoring products. Also, after Server Health Monitoring is configured on the administrator’s workstation, no configuration changes are necessary to the Domino 6 servers. It should be noted that administrators can still monitor their server performance using Domino statistics and events. This will log various performance statistics and can be set to enable warnings when statistics reach a critical value.

5.2.2 IBM Tivoli Monitoring for Messaging and Collaboration

IBM Tivoli Monitoring for Messaging and Collaboration ensures the availability and optimal performance of Lotus Domino server environments. Tivoli Monitoring for Messaging and Collaboration provides stable, secure, and proactive monitoring and management tools that leverage common Tivoli technology to provide rapid time to value and greater ease of use.

Tivoli Monitoring for Messaging and Collaboration monitors the status of Domino servers, identifies server and system problems in real time, notifies administrators, and takes automated actions to resolve Domino server problems. The product also collects monitoring data to help you analyze performance and trends and helps you address problems before they affect end users. Using Tivoli Monitoring for Messaging and Collaboration can help you maximize the business impact of your messaging and collaboration systems.

Integration with IBM Tivoli Enterprise™ Data Warehouse provides enterprise-wide access to data for historical and trending analysis, reporting, and graphing. Tivoli Monitoring for Messaging and Collaboration provides seamless integration with other Tivoli Performance and Availability solutions, including IBM Tivoli Enterprise Console® and IBM Tivoli Business Systems Manager, to provide a true end-to-end solution.

The key features of Tivoli Monitoring for Messaging and Collaboration include:

- Auto discovery of Domino servers.
- Best practices Resource Models that are deployed during installation and configuration.
- The Resource Models focus on key areas: Domino server availability, database management, mail routing, replication, server processes, and server health.
- Enhanced probes to measure and record application and server performance from an end-user point of view.

For Domino administrators, this includes a very intuitive and highly functional GUI that provides access to all the operational information, troubleshooting tools, monitoring, and event configuration interface required to manage a Domino environment. This user interface is provided in addition to the IBM Tivoli Monitoring Health Console, which is available for all IBM Tivoli Monitoring products. For more information, see:


5.3 Domino 6 Monitoring and Statistics tutorial

Lotus developerWorks provides numerous tutorials to assist with Domino functions. An excellent tutorial is available for Monitoring and Statistics for Domino 6 (and can also be used for Lotus Domino 6.5).
The following excerpt is from the tutorial Web site that focuses on Domino 6 Monitoring and Statistics:

To keep your systems healthy and ready for use, you need to know when problems first appear and how to gather information about capacity, workload, and system/user activities. This statistical information is invaluable in resource planning and finding and fixing bottlenecks. This tutorial shows how to set up and use monitoring features in Domino 6. The tutorial begins by explaining how monitoring and statistics work in Domino 6. It then shows how to use event-driven monitoring features for effective Domino server management. You will also learn how to gather and create graphical representations of statistics from both Domino events and operating system events. Finally, you will see how to use the Domino SNMP Agent for monitoring Domino.

This tutorial is for administrators who want to use the Domino 6 monitoring and statistics infrastructure. This infrastructure features flexible event monitoring that gives administrators real-time control over Domino domains. It can also generate numerous statistics and reports that can help you to recognize and remove bottlenecks in your Domino domain.

The tutorial is available at:

Summary: Common configuration problems that affect performance

This appendix provides direction when encountering problems with Domino performance. We describe common issues and causes of performance issues.

Lotus Domino server performance is not determined purely by the power of your hardware. Other factors can have a significant effect on the overall system performance. Making sure your network design, replication schedules, and database distribution are not causing inefficiencies are just three examples.

In the following list, we tried to put items in order of potential improvement, highest first, but that might not be the case for your specific environment. Also, clustering can provide significant improvement, but appears toward the end of the list because it is not usually implemented for performance alone.

Common configuration issues include:

- **Views (4.2.5, “Optimizing database performance” on page 44).**
  
  Does it take a long time to display a view? If so, you can increase the view buffer on the server. This will increase performance, because it will not force the server to read the view from the disk. You have to be careful because the maximum size you can increase to is about one-fourth of the physical memory.

- **Accessing databases (4.2.5, “Optimizing database performance” on page 44).**
  
  You can increase the maximum number of database open/close handles in the cache. The maximum size you can increase to is 1872. You can also decrease the size of the databases. The larger databases require more I/O and use more memory. This will use up other resources. Because some databases have to be large, you can increase the I/O subsystem or add more memory to the server.
- Unnecessary work (4.2.1, “Optimizing Domino task performance” on page 29).
  Many server tasks are started by default. You might not need every task to run your
  Domino environment, because some tasks such as Scheduling could take up about 20% of
  your server’s CPU.
- Increase the indexer tasks.
  Multiple processors can handle more than one indexer task. It is recommended that you
  set the number of indexers to the number of CPUs minus one. For example, if you have a
  four processor machine or environment, set the indexer to three. The is done by modifying
  an entry in the NOTES.INI file. See the next bullet.
- Increase the NOTES.INI settings.
  As you would make an entry change for the previous bullet, do so for the parameters
  UPDATE_SUPPRESSION_TIME and UPDATE_SUPPRESSION_LIMIT. This setting
determines the minimum amount of time before the indexer runs again. Increasing this
value will mean that the indexer will run less frequently.
- Increase the Mail box number (4.2.4, “Improving mail performance” on page 40).
  Increasing this will improve mail routing performance, but it does require additional
memory. The default value is 1. This change is completed in the Server Configuration
- Enable transaction logging (4.2.12, “Enabling transaction logging” on page 60).
  The easiest way to improve disk performance is to enable transaction logging. Always
place the log files on a separate disk device, preferably running RAID-1.
  Lotus recommends archive logging.
- Verify your memory configuration (2.6.1, “Server memory” on page 15).
  After the disk subsystem, memory is a potential bottleneck. Verify that you have installed
enough memory to support the number of users on your server. If required, add more
memory.
  The suggested minimum is 0.2 MB per active user. You might want to reserve more
memory for more advanced users performing more complicated tasks, perhaps 1 MB for
each concurrent user. Also reserve memory for any additional server task, such as
directory assistance or indexing.
- Distribute your databases over multiple physical disk devices.
  You can greatly improve disk utilization by spreading the disk workload across multiple
physical disk devices or RAID arrays. Configure several disk devices and create directory
links pointing to these devices using the Domino Administrator tool. Spread databases
and user mail files roughly evenly across the disks.
- Use RAID-1 disk subsystems (2.6.3, “RAID and disk controllers” on page 18).
  For performance coupled with fault tolerance, RAID-1 is superior to both RAID-0 and
RAID-5. Configure one RAID-1 array for the operating system, paging file, and Domino
program file, another array for transaction logging, and one or more arrays for Domino
data.
  Configure more disks per RAID array to get better disk read/write data throughput. Use a
stripe size of 16 KB on IBM ServeRAID-3H and 3L adapters. This is the optimal stripe unit
size for Domino. Span disk devices across multiple SCSI channels (when large numbers
of disks are being used).
Appendix A. Summary: Common configuration problems that affect performance

- Improve your network design (2.6.5, “Network adapter card” on page 19).
  If the network is highly used, split your network into smaller segments. Connect your critical servers directly to a network switch.
  Upgrade your network adapters from 10 Mbps to 100 Mbps or faster adapters. If even better network performance is required, consider ATM.

- Change your network protocol to TCP/IP.
  Implement your network using the TCP/IP protocol, and configure WINS servers and DNS for more growth potential. If you want to implement Domino clustering, you have to use TCP/IP.

- Improve your Domino design (2.4, “Domino server roles” on page 11).
  The Domino network and system design is at the heart of every system. Consider at least the following:
  - Plan replication schedules carefully. Consider hub and spoke design if there are several Domino servers in your network.
  - Dedicate Domino servers to roles: one server for mail, one for Remote Access Service (RAS), another for databases.

- Implement Domino clustering (4.2.15, “Balancing workload in a Lotus Domino cluster” on page 62)
  Connect your servers into a cluster to achieve better availability, scalability, and workload balancing.

- Upgrade to faster CPUs (2.6.4, “Processors” on page 19).
  Add more CPUs to improve raw processing power. Domino is a highly scalable application that can take advantage of the benefits of SMP. It does so automatically; you do not need to tune or configure your hardware for Domino to take advantage of the additional CPUs.
Protect your server from data loss

You should consider the following suggestions to help protect your servers from data loss when in operation.
Uninterruptible power supply

We recommend that you protect your server from data loss by connecting it to the main electricity supply through an uninterruptible power supply (UPS). A UPS keeps your server running for a short period in the event of a power failure by drawing power from a backup battery. When the battery gets too low (assuming power is not restored), the UPS instructs the server to shut down. This allows adequate time for your server to save data from the SCSI controller and operating system caches to disk before power is lost completely. Database corruption is much less likely if a properly configured UPS is part of your server system.

You should make sure that the UPS you choose has compatible software that will allow commands to be executed on your server in the event of a power failure. These commands can stop the Domino server and also perform a backup of important databases, such as the Domino Directory, before the server is shut down.

Disaster recovery software and virus protection

Hardly any significant computing environment today operates without some sort of protection against disasters. It is important that you purchase a high-quality backup and recovery system that allows you to easily and quickly restore your system in the event of a complete failure. Many companies only install software that allows them to recover files from tape. Unfortunately, it can take quite some time to install your operating system again and restore your Domino server from tape. Therefore, purchase a backup and recovery system that allows you to restore the operating system and applications completely by a simple boot disk process. This will enable you to get your Domino server operational as quickly as possible.

Ensure that your backup system can use the new Domino 6.5 APIs that allow incremental backup and restoration of Notes databases while online. You will need to enable the Transaction Log facility and read the instructions that come with your backup system to enable these features.

Domino is the center of your electronic mail and business processes. Users might e-mail others with software that is infected with a computer virus. We recommend that you purchase anti-virus software that can protect all of your systems against these attacks. Because some anti-virus software will check only files in the computer's file system, consider buying software that can scan attachments in Notes documents as well.
Domino 6 boundaries

Table C-1 summarizes the known maximum limits of various Lotus Notes and Domino 6 features as noted in the Notes Domino documentation.

<table>
<thead>
<tr>
<th>Item</th>
<th>Maximum limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database size</td>
<td>The maximum OS file size limit (up to 64GB).</td>
</tr>
<tr>
<td>Text field size</td>
<td>32 KB (storage); 32 KB displayed in a view’s column.</td>
</tr>
<tr>
<td>Rich text field size</td>
<td>Limited only by available disk space up to 1G.</td>
</tr>
<tr>
<td>Response levels in a hierarchical view; number of</td>
<td>31 levels; 300,000 documents.</td>
</tr>
<tr>
<td>documents per level</td>
<td></td>
</tr>
<tr>
<td>Characters in names</td>
<td>Database title: 96 bytes.</td>
</tr>
<tr>
<td></td>
<td>File names: On Windows and UNIX® platforms, minimum of 255 or OS limits, or both; on local Macintosh workstation, 31.</td>
</tr>
<tr>
<td></td>
<td>Field names: 32.</td>
</tr>
<tr>
<td></td>
<td>View names: 64.</td>
</tr>
<tr>
<td></td>
<td>Form names: 32.</td>
</tr>
<tr>
<td></td>
<td>Agent names: 32.</td>
</tr>
<tr>
<td>Fields in a database</td>
<td>~3000 (limited to ~64 K total length for all field names). You can enable the database property “Allow more fields in database” to get up to 22,893 uniquely-named fields in the database.</td>
</tr>
<tr>
<td>Columns in a table</td>
<td>64.</td>
</tr>
<tr>
<td>Rows in a table</td>
<td>255.</td>
</tr>
<tr>
<td>Views in a database</td>
<td>No limit; however, as the number of views increases, the length of time to display other views also increases.</td>
</tr>
<tr>
<td>Forms in a database</td>
<td>Limited only by database size.</td>
</tr>
<tr>
<td>Columns in a view</td>
<td>289 ten-character columns; dependent on # or characters per column.</td>
</tr>
<tr>
<td>Documents imported into a view</td>
<td>Documents totaling at least 350 K.</td>
</tr>
<tr>
<td>Item</td>
<td>Maximum limit</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>-------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Cascading views in a database</td>
<td>200.</td>
</tr>
<tr>
<td>Margin size (in inches)</td>
<td>46.</td>
</tr>
<tr>
<td>Page cropping size (in inches)</td>
<td>250.</td>
</tr>
<tr>
<td>Documents in a view</td>
<td>Maximum of 130 MB for a view index.</td>
</tr>
<tr>
<td>Documents that can be exported to tabular text</td>
<td>Limited only by available disk space.</td>
</tr>
<tr>
<td>Entries in an access control list (ACL)</td>
<td>~950 names (ACL size is limited to 32767 bytes).</td>
</tr>
<tr>
<td>Roles in an access control list</td>
<td>75 roles.</td>
</tr>
<tr>
<td>ID password length</td>
<td>63 characters.</td>
</tr>
<tr>
<td>Authorized users on a multiple password ID</td>
<td>8 users.</td>
</tr>
<tr>
<td>Outline entries in an outline</td>
<td>~21,000 entries.</td>
</tr>
</tbody>
</table>
Related publications

This section provides several key information sources for Domino 6 for xSeries and BladeCenter. Customers are encouraged to explore these resources for the most recent product-level and technical how-to information.

Domino for eServer xSeries and BladeCenter Web sites

The primary resource is the Domino for xSeries and BladeCenter Web site, available at:


This site provides the most current product release information, as well as additional information. Domino 6.5 for Windows and Linux on xSeries was announced in September, 2003.

Another important Web site is the Domino for xSeries road map to publications site, available at:


The information provided in this reference is intended for the product installation and maintenance staff. It provides you with links to the latest Domino for xSeries publications, Redbooks, and Web sites.

The xSeries benchmarks Web site provides information for the successful deployment of Domino for xSeries (NotesBench section).

NotesBench provides an objective method for evaluating the performance of different platforms running the Lotus Domino server. NotesBench generates a transactions-per-minute (tpm) metric, called a NotesMark for each test, along with a value for the maximum capacity (number of users) supported, and the average response time. Both performance metrics have an associated price/performance metric based on the cost of the hardware and software used to run the benchmark. The price/performance metric is not based on total cost of ownership (TCO), which includes warranty and maintenance coverage.


IBM Redbooks

Redbooks typically provide positioning and value guidance, installation and implementation experiences, typical solution scenarios, step-by-step how-to guidelines, and sample code. They are available in one or more of the following formats: hardcopy, softcopy on the IBM Redbooks Web site, and in the IBM CD-ROM collections.

Refer to the following list for Domino 6 for xSeries Redbooks that were recently published. This list also includes some Redbooks originally published for R5. To access these and cross-platform Domino 6 documents, visit the Redbooks Web site at:

http://www.redbooks.ibm.com/

Recent Domino 6 for xSeries Redbooks and Redpapers include:

- Tuning IBM eServer xSeries Servers for Performance, SG24-5287
White papers about Domino for xSeries

For white papers about Domino for xSeries, go to the following Web site:


Select Literature to read what IBM @server xSeries experts have to say on key topics such as:

- ZDNET Make the Case Series. Why x? Making the case for IBM @server xSeries Servers
  This paper explores the opportunities and benefits that can be realized in the use of xSeries systems for enterprise-based needs
- Systems Management for IBM @server xSeries servers
  This white paper describes how IBM is an industry leader in this area because it provides systems management features that give the administrator the tools needed to effectively plan for, deploy, manage, support, and retire networked clients and servers.

For xSeries and Domino benchmarking white papers, see:


The Third Party report--Use of Lotus Notes and Domino 6 saves money paper, a report from Ferris Research and IBM about the savings you can expect with IBM Lotus Notes and Domino 6, is available at:

http://www.ibm.com/software/lotus/info/literature/whitepapers.jsp

- IBM @server xSeries 440 Planning and Installation Guide, SG24-6196
- Server Consolidation with the IBM @server xSeries 440 and VMware ESX Server, SG24-6852
- How to install Red Hat Linux on the xSeries 440, TIPS0042
- IBM @server BladeCenter Systems Management, REDP-3582
- Deploying Lotus Domino on IBM @server BladeCenter, REDP-3584
- Domino Web Access 6.5 on Linux, SG24-7060
- A Roadmap for Deploying Domino in the Organization, SG24-5617
- Upgrading to Notes & Domino 6 diagram, TIPS0068
- Upgrading to Lotus Notes and Domino 6, SG24-6889
- Domino Designer 6: A Developer's Handbook, SG24-6854
- Lotus Domino 6 for Linux, SG24-6835
- Lotus Domino 6 spam Survival Guide for IBM @server, SG24-6930
- IBM WebSphere V5.0 Security WebSphere Handbook Series, SG24-6573
- Lotus Security Handbook, SG24-7017
- New features of Lotus Domino 6.0.1: Single Copy Template, REDP-3681
- Upgrading to Domino 6: Performance Benefits, REDP-3634
IBM eServer xSeries Web site

For the latest information about the xSeries family of products, see the following Web site:


IBM eServer BladeCenter Web site

For the latest information about the BladeCenter family of products, see the following Web site:

http://www.ibm.com/servers/eserver/bladecenter/

Experience reports

You can view real-world customer success stories at the success stories for IBM software for xSeries and BladeCenter Web site:


Search the keyword Domino for recent customer experiences.

IBM Lotus Domino Web site

See the IBM Lotus Domino Web site at:

http://www.lotus.com/products/r5web.nsf/webhome/nr5serverhp-new

Lotus Domino developerWorks

You can register at the Lotus developerWorks® Web site (formerly known as Notes.net) to receive updates on areas of interest, available at:

http://www.ibm.com/developerworks/lotus

You will find quick access to information about the following topics:

- FastPath to Lotus by product name
- Downloads and CDs
- Learning resources
- Support
- Community

Note: The forums on the Lotus developerWorks Web site are an invaluable resource for troubleshooting. All Lotus products are provided with a forum, updated constantly by the Domino community.
Product installation information

The installation guide for each release contains the most up-to-date information for that release. This is shipped on the full release CD in both NSF and PDF format. It can also be downloaded from the Lotus developerWorks at:

http://www.ibm.com/developerworks/lotus

Select Technical library → Product Documentation → by Product → Domino. Select the appropriate release.

Getting the Most From Your Domino Directory: Revisited

The IBM Redbook *Getting the Most From Your Domino Directory*, SG24-5986, contains the most complete information for beginning, intermediate, and skilled Domino administrators for managing your Domino Directory. It includes information about understanding the Domino directory services, planning directory services, and setting up, configuring, administering, and using Domino directory services.

How to get IBM Redbooks

You can search for, view, or download Redbooks, Redpapers, Hints and Tips, draft publications and Additional materials, as well as order hardcopy Redbooks or CD-ROMs, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services
Domino for IBM @server xSeries and BladeCenter Sizing and Performance Tuning

This IBM Redpaper will guide and assist you in selecting the correct server configuration for your selected Domino system, whether the system is for Web, application, or mail hosting. This Redpaper provides performance tuning techniques and best practices available for Lotus Domino 6.5 running on the IBM @server xSeries and BladeCenter systems. We describe the Domino network design, operating system, and disk I/O subsystem, with hints and tips to help you investigate and solve performance bottlenecks on your server.