Lotus Domino Domain Monitoring

Introduction to the powerful new Domino 7 features

Probes, corrective actions, and collection hierarchies

Examples of monitoring scenarios with tips and techniques

Philip Monson
Thomas Gumz
Frank Nostrame
Leah Busque

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

Notices ........................................................................................................................................ v
Trademarks .................................................................................................................................. viii

Preface ........................................................................................................................................... ix
The team that wrote this Redpaper ............................................................................................... ix
Become a published author ........................................................................................................... x
Comments welcome ................................................................................................................... x

Chapter 1. Overview of Lotus Domino domain monitoring ......................................................... 1
1.1 What is Domino domain monitoring? ....................................................................................... 2
1.2 Domino domain monitoring components ................................................................................ 3
1.3 Other Lotus Domino monitoring features ................................................................................ 4
  1.3.1 Event generators and handlers ....................................................................................... 4
  1.3.2 Statistics collection .......................................................................................................... 5
  1.3.3 Domino server monitor ................................................................................................... 5
  1.3.4 Activity logging ............................................................................................................... 6
  1.3.5 Other monitoring tools included with Lotus Domino ..................................................... 6
  1.3.6 Custom monitoring ......................................................................................................... 6
  1.3.7 Third-party products ....................................................................................................... 7
1.4 Why Domino domain monitoring? .......................................................................................... 7
1.5 Version and platform compatibility ........................................................................................ 8
  1.5.1 Domino version compatibility ........................................................................................ 8
  1.5.2 Platform compatibility ................................................................................................... 8
1.6 Event generator and Simple Event reporting into DDM.nsf ..................................................... 8
  1.6.1 Event generators and DDM.nsf. .................................................................................... 8
  1.6.2 Simple Events and DDM.nsf. ....................................................................................... 8
  1.6.3 Monitoring compatibility between Domino 7 and pre-Domino 7 servers ....................... 9
1.7 Total cost of ownership (TCO) reduction with Domino domain monitoring ......................... 9
1.8 IBM monitoring strategy ......................................................................................................... 10
1.9 Goals of this paper ................................................................................................................ 10

Chapter 2. Domino domain monitoring: components and “out-of-the-box” configuration ........ 11
2.1 Component overview .............................................................................................................. 12
2.2 Probes ..................................................................................................................................... 12
  2.2.1 Probe configuration documents ..................................................................................... 12
  2.2.2 Probe types and architecture ......................................................................................... 14
2.3 Collection hierarchies .............................................................................................................. 15
  2.3.1 Overview of collection hierarchies. ............................................................................... 15
  2.3.2 Architecture behind server collection hierarchies ......................................................... 16
2.4 Filters ....................................................................................................................................... 17
2.5 User interfaces ....................................................................................................................... 18
  2.5.1 Monitoring Configuration (Events4.nsf). ................................................................... 18
  2.5.2 Domino domain monitor database (DDM.nsf). .............................................................. 20
2.6 Events ..................................................................................................................................... 22
  2.6.1 Event types ................................................................................................................... 22
  2.6.2 Event statuses .............................................................................................................. 22
  2.6.3 Event severities ............................................................................................................ 23
  2.6.4 Event document example ............................................................................................... 24
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.7 Customize DDM.nsf from Event documents</td>
<td>25</td>
</tr>
<tr>
<td>2.8 Out-of-the-box configuration</td>
<td>28</td>
</tr>
<tr>
<td>2.8.1 Domino 7 upgrade and Domino domain monitoring</td>
<td>28</td>
</tr>
<tr>
<td>2.8.2 Default probes</td>
<td>29</td>
</tr>
<tr>
<td>2.8.3 No collection hierarchies created by default</td>
<td>29</td>
</tr>
<tr>
<td>2.8.4 One default filter</td>
<td>30</td>
</tr>
<tr>
<td>2.8.5 “Out-of-the-box” user interfaces</td>
<td>30</td>
</tr>
<tr>
<td>2.8.6 Events created by default</td>
<td>30</td>
</tr>
<tr>
<td>2.9 Management of DDM.nsf</td>
<td>33</td>
</tr>
<tr>
<td>2.9.1 ACL settings for DDM.nsf access and usage</td>
<td>34</td>
</tr>
<tr>
<td>2.9.2 Replication settings to control the size of DDM.nsf.</td>
<td>34</td>
</tr>
<tr>
<td>2.9.3 Domino domain monitoring filters</td>
<td>35</td>
</tr>
<tr>
<td>2.9.4 Managing Events</td>
<td>36</td>
</tr>
<tr>
<td>2.9.5 ST Awareness</td>
<td>39</td>
</tr>
<tr>
<td>Chapter 3. A Domino domain monitoring scenario</td>
<td>43</td>
</tr>
<tr>
<td>3.1 Overview of the Domino server environment</td>
<td>44</td>
</tr>
<tr>
<td>3.1.1 The Redbooks lab Domino server topology</td>
<td>44</td>
</tr>
<tr>
<td>3.1.2 Planning the server collection hierarchy</td>
<td>44</td>
</tr>
<tr>
<td>3.1.3 Configuring the server collection hierarchy</td>
<td>45</td>
</tr>
<tr>
<td>3.1.4 Verifying the server collection hierarchy</td>
<td>47</td>
</tr>
<tr>
<td>3.2 Monitoring the external mail server</td>
<td>47</td>
</tr>
<tr>
<td>3.2.1 Monitoring TCP ports on pre-Domino 7 servers</td>
<td>48</td>
</tr>
<tr>
<td>3.2.2 Verifying the TCP port checks</td>
<td>49</td>
</tr>
<tr>
<td>3.3 Upgrading the server to Domino 7</td>
<td>54</td>
</tr>
<tr>
<td>3.3.1 Updating the server collection hierarchy</td>
<td>55</td>
</tr>
<tr>
<td>3.3.2 Creating and configuring Domino domain monitoring Messaging Probes</td>
<td>56</td>
</tr>
<tr>
<td>3.3.3 Verifying operational status of the Domino domain monitoring probes</td>
<td>58</td>
</tr>
<tr>
<td>3.4 Fixing Out of Office agents after a mail template upgrade</td>
<td>60</td>
</tr>
<tr>
<td>3.5 Custom corrective action code</td>
<td>63</td>
</tr>
<tr>
<td>3.5.1 Events with default corrective actions</td>
<td>63</td>
</tr>
<tr>
<td>3.5.2 Writing your own custom corrective actions</td>
<td>64</td>
</tr>
<tr>
<td>3.5.3 Keyword substitution in corrective actions</td>
<td>66</td>
</tr>
<tr>
<td>3.5.4 Corrective action code to fix a corrupt database</td>
<td>67</td>
</tr>
<tr>
<td>3.5.5 Testing our new custom corrective action code</td>
<td>69</td>
</tr>
<tr>
<td>3.6 Enabling your existing applications for Event reporting</td>
<td>73</td>
</tr>
<tr>
<td>3.7 Summary</td>
<td>76</td>
</tr>
<tr>
<td>Chapter 4. More Domino domain monitoring scenarios and tips</td>
<td>77</td>
</tr>
<tr>
<td>4.1 Additional monitoring scenarios</td>
<td>78</td>
</tr>
<tr>
<td>4.1.1 Security monitoring</td>
<td>78</td>
</tr>
<tr>
<td>4.1.2 Directory monitoring</td>
<td>81</td>
</tr>
<tr>
<td>4.1.3 Web Configuration monitoring</td>
<td>86</td>
</tr>
<tr>
<td>4.1.4 Replication monitoring</td>
<td>88</td>
</tr>
<tr>
<td>4.2 Domino domain monitoring console commands and debugging information</td>
<td>91</td>
</tr>
<tr>
<td>4.2.1 Debug notes.ini setting for Domino domain monitoring</td>
<td>92</td>
</tr>
<tr>
<td>4.3 Forwarding OS Events to Tivoli Enterprise Console (TEC)</td>
<td>92</td>
</tr>
<tr>
<td>Appendix A. Domino domain monitoring probes</td>
<td>95</td>
</tr>
<tr>
<td>Probes by feature area</td>
<td>96</td>
</tr>
<tr>
<td>Application probes</td>
<td>96</td>
</tr>
<tr>
<td>Database probes</td>
<td>96</td>
</tr>
<tr>
<td>Directory probes</td>
<td>97</td>
</tr>
<tr>
<td>Messaging probes</td>
<td>98</td>
</tr>
</tbody>
</table>
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®
- Candle®
- Domino Designer®
- Domino®
- IBM®
- Lotus Notes®
- Lotus®
- Notes®
- OMEGAMON®
- Redbooks™
- Redbooks (logo)™
- Sametime®
- Tivoli Enterprise™
- Tivoli Enterprise Console®
- Tivoli®
- WebSphere®
- z/OS®

The following terms are trademarks of other companies:

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

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

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

The purpose of this IBM® Redpaper is to show system administrators and IT architects how to use and leverage the capabilities of the new Lotus® Domino® domain monitoring feature of IBM Lotus Domino Version 7. The Lotus Domino domain monitoring feature provides customizable monitoring capabilities and fast recognition and reporting of critical server issues by delivering a single, feature-oriented view through which administrators can see the status of multiple servers across a domain.

By reading this Redpaper, you will gain intimate knowledge of Domino domain monitoring and see examples of its true power, such as autonomic corrective actions and more.

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, Cambridge Center.

Philip Monson is a Project Leader at the ITSO Lotus Center in Cambridge MA. Phil has been with Lotus / IBM for 15 years, joining the company when the early versions of Notes were rolled out for internal use. He has served in management, technical, and consulting roles in the IT, Sales, and Development organizations.

Thomas Gumz is an Advisory Software Engineer and Product Designer in the IBM Lotus Software Group, working on IBM Lotus Domino Server Administration and IBM WebSphere® Portal Server software. He is the principal product designer of the Domino Administration clients, and also designed and built the new Domino domain monitor database in Domino 7. He is based in NYC, NY.

Frank Nostrame is an IT Specialist in IBM Global Services. Frank specializes in Domino infrastructure, deployment, and migration projects on the IBM account. As a project and technical lead, he has extensive experience in monitoring, automation, and early deployment of Lotus Domino products. Frank holds a CS/MIS degree from Boston College. His other professional areas of interest include SOA/Web Services and open source Web development.

Leah Busque is a Staff Software Engineer for Lotus Development in Cambridge, MA. While working on Domino Server development for over four years, Leah has been a part of the Messaging and Administration teams focusing on the Mail Server, Domino Archiving, and Server Administration. She holds two degrees from Sweet Briar College in Virginia, one in Mathematics and the other in Computer Science. She worked extensively on the development of Domino domain monitoring for the Domino 7.0 release. Her professional areas of interest include Server Administration Development, Autonomic Computing, and NP Completeness Complexity Theory.

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

Harry Peebles, Senior Software Engineer. IBM Software Group, Lotus
Julie Kadashevich, Senior Software Engineer. IBM Software Group, Lotus
Mark Skurla, Advisory Software Engineer, Technical Lead. IBM Software Group, Lotus
Scott O'Keefe, Domino Administration Template Developer. IBM Software Group, Lotus
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 customers.

Your efforts will help increase product acceptance and client 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 e-mail to:
  redbook@us.ibm.com

► Mail your comments to:
  IBM Corporation, International Technical Support Organization
  Dept. HYJ Mail Station P099
  2455 South Road
  Poughkeepsie, New York 12601-5400
Overview of Lotus Domino domain monitoring

This IBM Redpaper provides an overview and example scenario of Lotus Domino domain monitoring, the new server monitoring feature available with Lotus Domino 7. This Redpaper will focus on one end-to-end example scenario that will highlight many of the features of Domino domain monitoring, as well as provide tips and techniques. In the final chapter, we provide additional example scenarios that will highlight other key features of Domino domain monitoring.

In this chapter, we introduce Domino domain monitoring and its components. We describe some of the tools available prior to Domino 7 in order to show where Domino domain monitoring fits into your monitoring infrastructure. Finally, we will discuss compatibility and the business value gains associated with Domino domain monitoring.

This chapter contains the following topics:

- Introduction to Domino domain monitoring
- Other Domino monitoring features included with Lotus Domino 7
- Why Domino domain monitoring?
- Version and platform compatibility
- Business value of Domino domain monitoring
1.1 What is Domino domain monitoring?

Domino domain monitoring, a new feature of Domino 7, provides a single feature-oriented view through which administrators can see the status of multiple servers and then use the information provided to quickly resolve problems.

Domino domain monitoring automates problem determination and analysis in order to help reduce the costs and time associated with monitoring and troubleshooting your Domino environment. With Domino domain monitoring, administrators can quickly locate and resolve issues before they cause more serious problems or outages.

The key features of Domino domain monitoring include:

- Top-down, feature-oriented view of the domain status
- Highly-configurable probes categorized by feature areas
- Probable cause and possible solution determination
- Automation of corrective actions
- Default settings for easy out-of-the-box setup
- Domino domain monitoring data aggregation with collection hierarchies

The Monitoring Configuration (Events4.nsf) database is used for all Domino domain monitoring configuration. The new Domino domain monitoring probes generate Event report documents that get consolidated and reported into the new Domino Domain Monitor database (DDM.nsf).

Domino 7 is required to enable Domino domain monitoring probes, but there are Best Practices probes that work with pre-Domino 7 servers. Simple Events and Event generator triggered Events also get reported to DDM.nsf.

Domino domain monitoring is enabled with Domino 7.0 “out of the box,” and comes with a default set of probes. The default configuration will be discussed in detail in 2.8, “Out-of-the-box configuration” on page 28.

Domino domain monitoring is an end-to-end monitoring solution that is highly customizable and extensible. The administrative interface (DDM.nsf) is intuitive and contains built-in collaboration features, including Event assignment and Sametime® awareness (see Figure 1-1 on page 3). The new Domino domain monitoring probes can be customized to meet the needs of your organization.

Important: Simple Event is the new term used in Domino 7 for an Event that is not associated with or does not contain target information. We discuss Simple Events in more detail in 1.6, “Event generator and Simple Event reporting into DDM.nsf” on page 8.
1.2 Domino domain monitoring components

Domino domain monitoring can be broken down into architectural components. These components will be explained in more detail in Chapter 2, “Domino domain monitoring: components and “out-of-the-box” configuration” on page 11, but you should have a basic understanding of them before we dive into the nuts and bolts of Domino domain monitoring. A high-level description of the five components is provided in Table 1-1 on page 4.
1.3 Other Lotus Domino monitoring features

This section describes some of the other monitoring tools and utilities that are available as part of the Lotus Domino product, as well as custom monitoring applications and third-party products. Our intention is to summarize the monitoring capabilities of Domino domain monitoring while keeping in mind that these tools should be used in conjunction with the new Domino domain monitoring feature of Domino 7 as necessary.

In some cases, there may be a need to utilize another monitoring feature to supplement the reporting provided with Domino domain monitoring. For example, as an administrator or IT architect, you might use the statistics collection feature to get more granular platform statistics. Similarly, if you need a dashboard view that shows current server or task availability from your Notes client location, you might use the Domino server monitor.

To summarize, Domino domain monitoring is intended to be used as the primary monitoring administration interface and in conjunction with other Domino features, custom applications, or third-party products, as necessary.

1.3.1 Event generators and handlers

Event generators and Event handlers are used to monitor and report Domino Events, statistics, and to probe a server for access or connectivity reports. Every Event occurrence on a Domino system can be monitored via an Event generator and reported via a corresponding Event handler. An Event generator can produce an alert on the occurrence of one of the following:

- Any Event that matches a criteria. The criteria are type, severity, and message text.
- A built-in or add-in task Event, which allows you to specify an Event message, which has a corresponding Event code, type, and severity.

### Table 1-1 Domino domain monitoring components

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Probes</td>
<td>Probes do the majority of the work and can be configured to monitor Events in multiple feature areas (Replication, Security, and so on).</td>
</tr>
<tr>
<td>Collection Hierarchy (Optional)</td>
<td>One or more collection hierarchies can be created to roll up the Domino domain monitoring results (Event) information into one or more collection servers.</td>
</tr>
<tr>
<td>Filter (Optional)</td>
<td>Filters are utilized to increase or decrease the amount of Events tracked via DDM.nsf.</td>
</tr>
<tr>
<td>User Interfaces</td>
<td>The Events4.nsf (Monitoring Configuration) database and DDM.nsf database are the two user interfaces used with Domino domain monitoring. Events4.nsf is used to configure Domino domain monitoring probes, collection hierarchies and filters, and DDM.nsf stores probe results (Events) by severity, date, and type.</td>
</tr>
<tr>
<td>Events</td>
<td>Events are generated by enabled probes, enabled Event generators and non-configurable embedded probes.</td>
</tr>
</tbody>
</table>
A custom Event generator, which can generate Events in the following feature areas: Database, Domino Server, TCP Server, Mail Routing, Statistic, and Task Status.

There are several notification methods available to report Events, including Mail, Log to a database (usually statrep.nsf), and Log to NT Event Viewer.

**Note:** With Domino 7, there are three new notification methods: Run an agent, Send Java™ Console Controller Command, and Send a console command to the server.

Both Event generator and Event handler documents are stored in the Monitoring Configuration database (Events4.nsf).

**Important:** In addition to Event handlers, *all* Event generators report Events into the DDM.nsf database, as detailed in 1.6, “Event generator and Simple Event reporting into DDM.nsf” on page 8.

### 1.3.2 Statistics collection

Domino server statistics are collected by configuring a Server Statistic Collection document in the Monitoring Configuration database and running the Collect task on the server. One server can collect from multiple servers or each server in the domain can collect and report statistics locally. The Monitoring Results database (statrep.nsf) is the default database to store Statistics Reports.

A statistics report is created for each of the following feature areas:

- Calendaring and Scheduling
- Clusters
- Communications
- Mail and Database
- Network
- Platform
- System
- Web server and Retriever

### 1.3.3 Domino server monitor

The Domino server monitor feature provides a visual representation of the status of servers and server tasks and displays real-time statistics. You can view servers across multiple domains or a subset of servers and save server groups to Monitoring profiles for future use.

The Domino server monitor provides shortcuts to administrative tasks, including:

- Opening a different Domino Administrator tab from the Domino server monitor
- Displaying the differences in current and previous statistic values
- Viewing additional information about a server or server task
- Creating an Event handler for a server that is down
As part of the Domino server monitoring feature, you can locally collect statistics on the servers that you are monitoring and graphically display the statistics by creating statistics charts. Both real-time and historical charts can be created and saved using profiles. You can also create statistic profiles so that you can routinely chart a specified set of statistics.

### 1.3.4 Activity logging

Activity logging is used to collect information about activity on your Domino servers and replaced the Billing task used on pre-Domino 6 versions. This activity information can be used for resource planning, performance monitoring, and end-user billing purposes.

Activity logging is enabled and configured in a server configuration document under the Activity Logging tab. The activity information, which is stored in log.nsf, can optionally be stored in the Activity Trends (activity.nsf) database. The information is presented in trended and historical form and is categorized by Server, Database, User, and Connections.

In addition to logging the activity data to the Activity Trends database, you can also manually pull and view the data using advanced search functionality. The Notes Log Analysis database (loga4.nsf) is the default location to store log analysis data.

### 1.3.5 Other monitoring tools included with Lotus Domino

- **Server console:** View and interact with a custom console that uses your predefined colors and text attributes to illustrate the status of a process.
- **Log.nsf:** A default server log that includes multiple views and advanced search capabilities via the Domino Administrator client.
- **SNMP agent traps:** Enable third-party products to manage aspects of the Domino server. (SNMP Trap is an Event handler option.)

**Attention:** The IBM Redbook *Upgrading to Lotus Notes and Domino 6*, SG24-6889 contains additional information about pre-Domino 7 monitoring. This Redbook is available at the following URL:


### 1.3.6 Custom monitoring

In addition to the Domino monitoring product features, there are many options available to develop custom tools for Domino monitoring. Some examples are:

- Shell/batch scripts to monitor ASCII Domino console log output
- C/C++ API Toolkit to access Domino database components
- Java/COM/OLE/LotusScript to access the Notes and Domino environment

There are also many possibilities when it comes to choosing a tool or enterprise wide application to monitor your platform level statistics, availability, and performance:

- Platform specific statistic utilities (topas, cpumon, and so on)
- Custom written data collection software, including back-end storage and retrieval
- User interface development to back-end data
1.3.7 Third-party products

While the scope of this Redpaper does not include any direct mention of Domino monitoring software vendors, there are many products in the market that provide various types of Domino monitoring, from feature specific to comprehensive, domain-wide monitoring.

A recommended monitoring suite of IBM products will be covered in 1.8, “IBM monitoring strategy” on page 10.

1.4 Why Domino domain monitoring?

The pre-Domino 7 monitoring features discussed so far in this chapter are comprehensive and fully integrated into the Domino product. However, administrators still felt they needed additional functionality in order to resolve some of the following issues:

- The cost of monitoring and troubleshooting servers.
- There can be problems reported on many servers and in many places.
- Understanding and analyzing all Events and errors is impossible.
- Domino has great tools for single Event problems, but:
  - No correlation capability for related problems.
  - Root cause analysis of complex data is onerous.
  - No cross-server notification and reporting mechanism.
- No corrective action context in many cases.

Domino domain monitoring was developed to address these specific issues, allowing administrators to quickly resolve problems across their entire Domino environment. Domino domain monitoring utilizes a new administration paradigm to detect, analyze, present, and help resolve the most important and costly Domino problems. Problems in the areas of messaging, directory, replication, security, applications, agents, Web server, and more are filtered and prioritized to direct administrators to the most serious issues in the domain. Where problem details are important, administrators have drill-down capabilities and the ability to look at historical information.

Domino domain monitoring automates problem analysis with default and customizable probable causes, possible solutions, and corrective actions for Events. Other tools require that you manually diagnose server problems and correlate Event information; Domino domain monitoring does this for you automatically. Domino domain monitoring is also proactive since it has embedded probes that are built into the server and add-in task code. No other monitoring could provide this type of real-time monitoring because access to this level of the code is not feasible. Other tools are reactive because they can only respond to Events or server conditions from the outside looking in.

This Redpaper will discuss the autonomic capabilities of Domino domain monitoring in detail. We will explain and demonstrate the power of corrective actions and show how you can customize the probable cause and possible solution for Events. Domino domain monitoring utilizes Tivoli® Autonomic Management Engine (TAME) for the operating system probes, which allows the OS probes to report to Tivoli Enterprise™ Console (TEC), which we will discuss in 4.3, “Forwarding OS Events to Tivoli Enterprise Console (TEC)” on page 92.

Chapter 3, “A Domino domain monitoring scenario” on page 43 will explain how Notes application developers can create Events within their code that can be probed and reported into Domino domain monitoring. This is a little known feature that allows for the integration of
application specific Events into Domino domain monitoring, which ensures higher levels of application performance and availability.

This Redpaper will help you full optimize Domino domain monitoring.

1.5 Version and platform compatibility

The subsequent sections explain the Lotus Notes and Domino version and operating system platform compatibility with Domino domain monitoring.

1.5.1 Domino version compatibility

Domino 7 is required to run the new Domino domain monitoring probes because the code associated with these probes is built into the new Domino 7 server and add-in task code.

Two exceptions to this Domino 7 code requirement are the Best Practices probes for security and Web. Pre-Domino 7 servers cannot run these probes, but they can be probed by a Domino 7 server running Domino domain monitoring. The reason is because these best practice probes check configuration settings in the Domino directory (names.nsf) on the probing server (running Domino 7). The probes do not actually run any code on the server being probed.

1.5.2 Platform compatibility

Domino domain monitoring is available on all Domino 7 supported platforms.

There are a few differences in configuration and operating system probe availability, as listed in Appendix A, “Domino domain monitoring probes” on page 95.

1.6 Event generator and Simple Event reporting into DDM.nsf

This section explains how, in addition to the new Domino domain monitoring probes, the pre-Domino Event generators and Simple Events both report into the DDM.nsf. Domino domain monitoring will correlate all Events in DDM.nsf.

1.6.1 Event generators and DDM.nsf

Event generators can be reported with Event handlers, but they are also reported into DDM.nsf by default. This inclusion of Event generator triggered Events is a bonus for administrators, who no longer have to search through statrep.nsf to piece together the status or history of a server issue.

1.6.2 Simple Events and DDM.nsf

Simple Event is the new term used in Domino 7 for an Event that is not associated with or does not contain target information. Generally, most Events logged to the Domino server console are Simple Events. These Simple Events are also reported into the DDM.nsf database. We discuss Simple Events in more detail in 2.6, “Events” on page 22.
1.6.3  Monitoring compatibility between Domino 7 and pre-Domino 7 servers

The following bullets summarize the monitoring compatibility between Domino 7 and pre-Domino 7:

- A Domino 7 server can monitor any pre-Domino 7 server version using pre-Domino 7 methods, which generate Simple Events in the DDM.nsf database.
- A Domino 7 server can monitor any Domino 7 server version using pre-Domino 7 methods, which generate Enhanced Events in the DDM.nsf database.
- A Domino 7 server can monitor Domino 7 servers using the new Domino domain monitoring probes, which generate Enhanced Events in the DDM.nsf database.
- A Domino 7 server can monitor pre-Domino 7 servers only using the best practices Domino domain monitoring probes, which generate Enhanced Events in the DDM.nsf database.

**Note:** “Pre-Domino 7 methods” refer to Event generators and Simple Events.

Enhanced Event is the new term used in Domino 7 for an Event that contains at least one field with specific target information, which can be servers, databases, agents, or a user-specified target. Enhanced Events will be discussed in detail in 2.6.1, “Event types” on page 22.

1.7  Total cost of ownership (TCO) reduction with Domino domain monitoring

Domino domain monitoring gives organizations the opportunity to reduce TCO by providing a means to monitor and determine the health of an entire domain at a single UI location and quickly resolve problems.

Domino domain monitoring is included with Domino 7, and enables monitoring at a granular level that no other feature or product can provide.

Here are some of the high-level benefits of Domino domain monitoring that reduce TCO:

- One top-down, feature-oriented interface for monitoring and problem resolution
- Distills/correlates a huge amount of otherwise indigestible data using Event correlation
- Interface allows systematic approach to resolving and organizing server issues
- Configurability and flexibility accommodates diverse needs

The ability to easily link to and modify Event documents with customized probable cause, possible solutions, and corrective actions allows for unlimited possibilities. Customization of the new Domino domain monitoring probes is accessible and straightforward, so that either your Domino administrator or development team can easily create custom probes.

Domino domain monitoring is a key part of the Domino 7 value proposition, preventing server downtime and providing optimal performance.
1.8 IBM monitoring strategy

This section provides an overview of the IBM monitoring strategy for Domino, which needs to be considered at two levels: enterprise-wide monitoring tools and Domino monitoring tools.

**Enterprise-wide monitoring**
For customers deploying an enterprise-wide monitoring environment, the IBM Tivoli Monitoring (ITM) product family is recommended. Tivoli's strategic direction is for ITM and the former Candle® OMEGAMON® products to become more tightly integrated in a future release of ITM.

**Domino monitoring**
For monitoring of Lotus Domino servers, IBM recommends that Domino administrators also use a dedicated Domino monitoring tool. Only dedicated Domino tools can provide monitoring at the correct level of granularity to track all Domino server task activity. Several Domino monitoring tools are available from IBM, including Domino domain monitoring, IBM Tivoli Intelliiwatch (formerly a Candle product) and IBM Tivoli Monitoring for Messaging and Collaboration (ITMMC). Each product has its own particular strengths and uses:

- Domino domain monitoring offers advantages in terms of cost (free with Domino), low server processing overhead, and deeply integrated Domino monitoring functionality.
- Intelliiwatch advantages include support for Domino 6 servers and Performance Manager to customize Domino native statistics.
- ITMMC is tightly integrated with ITM, but only offers basic Domino server monitoring capabilities compared to either Domino domain monitoring or Intelliiwatch. ITMMC is recommended for ITM customers who want oversight of Domino servers, but it should be complemented by other Domino specific monitoring tools, such as Domino domain monitoring or Intelliiwatch.

1.9 Goals of this paper

The goals of this Redpaper are three-fold:

- The first goal is to provide a thorough overview of the Domino domain monitoring feature. We will discuss the components of Domino domain monitoring in detail and provide tips that will assist you with the out-of-the-box configuration of the feature in order to get you up and running quickly.
- The second goal is to provide an example scenario to demonstrate the server monitoring and autonomic corrective action capabilities of Domino domain monitoring. We utilize multiple probes, filters, and a collection hierarchy to provide in-depth analysis of our example Redbooks Domino lab environment. We expose the full potential of Domino domain monitoring in this example scenario.
- Finally, we explore more of the Domino domain monitoring feature area probes to get you thinking about how to satisfy your server monitoring requirements. We provide additional monitoring scenarios and some tips and techniques to accomplish this final goal.
Domino domain monitoring: components and “out-of-the-box” configuration

In this chapter, we will explore the components of Domino domain monitoring in detail in order to prepare you for the example scenario in Chapter 3, “A Domino domain monitoring scenario” on page 43.

The following topics will be discussed in this chapter:

► Components of Domino domain monitoring
  – Probes
  – Collection hierarchies
  – Filters
  – User interfaces
  – Events
► Out-of-the-box configuration
► Management of DDM.nsf
2.1 Component overview

This section describes the components that make up Domino domain monitoring. As described in 1.2, “Domino domain monitoring components” on page 3, there are five components that comprise Domino domain monitoring:

- Probes
- Collection hierarchies
- Filters
- User interfaces
- Events

We will take a closer look at each component in this chapter.

Note: Collection hierarchies and filters are optional, but as described in 2.8, “Out-of-the-box configuration” on page 28, there is a default filter to allow reporting of all Events with Fatal or Failure severity and exclude Simple Events with any other severity levels.

2.2 Probes

Probes are the backbone of Domino domain monitoring and require Domino 7 because they are embedded into the new server and addin task code.

Important: Probes run locally on each Domino server and they do not remotely monitor other servers. When probes are triggered, they generate Events in the local DDM.nsf database. Collection hierarchies are used to aggregate or “roll up” DDM.nsf data across multiple servers in a domain. Collection hierarchies are discussed in 2.3, “Collection hierarchies” on page 15.

2.2.1 Probe configuration documents

Probes are set and stored in individual Probe configuration documents, which are stored in the Monitoring Configuration database (events4.nsf). Probe configuration documents contain Basics, Target, and Specifics sections in the Basics tab. Depending on the probe type, there may also be a Schedule tab in the document.

Basics tab
The Basics tab contains the Basics, Target, and Specifics sections, as displayed in Figure 2-1 on page 13. Depending on the probe type, there may also be a Schedule tab in the document where you can specify how often the probe should run.
Chapter 2. Domino domain monitoring: components and “out-of-the-box” configuration

Figure 2-1 Database Scheduled Checks probe document to check for 30% or more unused space

**Basics section**
The Basics section contains the following required information:

- Probe Type
- Probe Subtype
- Description: User defined description of the probe
- Probe description: General description of the probe that is built into the template (not editable)

**Target section**
The Target section contains the information related to where the probe will run and optionally on what servers, databases and processes. The first field, ‘Server to run probe,’ will appear on all Probe configuration documents. The rest will appear depending on the type of probe:

- Server to run probe
- Servers to probe
- Databases to probe
- Process to probe

One feature of note is the Special Target Servers option for the field “Which servers should run this probe?” Domino domain monitoring dynamically figures out what servers run certain tasks. Using this target option makes sense for certain probes. For example, if you are configuring a Messaging Router Process State probe, you may only want to enable this on mail servers. If so, you can simply select Special Target Servers and specify Mail Servers in the drop-down list box.
The Special Target Servers option is also available for Filter documents. Detailed information about how servers get categorized into these Special Target Servers groups can be found in “Special Target Server groups” on page 57.

**Specifics section**
The Specifics section contains various configuration settings specific to the probe function, for example, thresholds for Events it generates. In addition, most probes contain a Severity setting, where you can specify Normal, Warning (low), Warning (high), Failure, or Fatal. Some probes have no additional configuration settings available in the Specifics section.

**Note:** Usually you will find a Specifics section on the Basic tab, but if the Specifics section is unusually long, the Specifics configuration information will be moved to its own tab (for example, Security Best Practices probe).

**Schedule tab**
The Schedule tab is where you define the schedule for a probe. The Lotus Domino 7 Administrator Help documentation provides the full details of how you can schedule probes and can be located at the following URL:

http://www-10.lotus.com/ldd/notesua.nsf/ddaf2e7f76d2cfbf8525674b00508d2b/de2da7b609c6cb038525703b005e2f5970openDocument

The top level schedule setting is “How often should this probe run?”, which can be set to run multiple times per day (daily, weekly, or monthly).

**Note:** Some probes run on a schedule, but Domino domain monitoring does not allow the schedule to be modified. In this case, the probe description will note this and no Schedule tab is available.

One configuration setting to note is “How should missed probes be handled?” This setting is available when scheduling a probe to run weekly or monthly and can be set to one of the following:

- Ignore missed probe
- Run missed probe on startup
- Run missed probe at next time range

**Note:** The last option applies to probes run weekly or monthly. For example, if a probe scheduled to run weekly at 4:30 AM on Tuesday does not run because the server is down from 4:30 AM - 5:30 AM, the probe will run on the next day, Wednesday, at 4:30 AM.

### 2.2.2 Probe types and architecture

This section explains how probes are categorized into nine different probe types, also known as feature areas. In the second section, “Probe architecture” on page 15 we will explain the different ways in which probes are scheduled and triggered so that you have a better understanding of the Domino domain monitoring architecture.

**Probe types (feature areas)**
All probes are categorized by a probe type. For each probe type, there are one or more probe subtypes. The probe subtypes further categorize the function of the probe. For example, messaging is a probe type and one of its associated probe subtypes is router process state. This combination of probe type and probe subtype creates a mail router status probe.
The probe types and their associated subtypes are listed with descriptions in Appendix A, “Domino domain monitoring probes” on page 95.

Note: There are some probes that should only have one probe configuration document, like operating system probes. Most other probes can have multiple probe configuration documents, as discussed in 4.1.2, “Directory monitoring” on page 81.

**Probe architecture**
When Domino domain monitoring probes were designed, they were categorized into three different styles: Scheduled, Embedded, and Listening probes. These categories are described in depth below and should give you a better feel for the architecture behind Domino domain monitoring. It is not imperative that you remember these different categories to successfully use Domino domain monitoring, but we thought it would be interesting to give you a glimpse into the probe architecture.

**Scheduled**
Scheduled probes run according to a customizable or predefined schedule, as discussed in “Schedule tab” on page 14.

**Embedded**
Embedded probes “instrument” the feature area code. They have been integrated directly into the code so that they can report on problems in real time.

**Listening**
Listening probes listen for certain errors to occur. When one of these errors does occur, the probe wakes up and starts the work of processing the Event and reporting it into DDM.nsf.

---

### 2.3 Collection hierarchies

This section will focus on the architecture behind a server collection hierarchy. We cover the steps required to create a collection hierarchy in 3.1.3, “Configuring the server collection hierarchy” on page 45.

#### 2.3.1 Overview of collection hierarchies

The Lotus Domino 7 Administrator Help documentation contains very thorough documentation about server collection hierarchies. High level points include the ability to:

- Create a server collection hierarchy or multiple collection hierarchies to “roll up” or aggregate Domino domain monitoring Event data.
- Set up the collection hierarchy to represent an existing Domino server topology (for example, hub and spoke) or set it up to ensure an administrator only receives information that is pertinent to the servers he or she maintains.
- Data aggregation is done using selective replication from the collection server. The collection server pulls Event data when it is notified about the change in an Event state and when the collection interval time (five minutes, not modifiable) has been exceeded.

**Important:** A collection hierarchy does all of the work of aggregating Domino domain monitoring data. Domino domain monitoring automatically creates the selective replication formulas needed to collect the data. You do not need to set up any connection documents to replicate Domino domain monitoring data.
2.3.2 Architecture behind server collection hierarchies

Collection hierarchies allow for a great deal of flexibility. Multiple hierarchies can exist within one domain and any given server can be the collection server and a monitored (node) server in one or more hierarchies. A collection hierarchy uses selective replication behind the scenes. Domino domain monitoring does all of the work associated with consolidating your data so that you do not have to worry about it.

**Tip:** You can even monitor the “roll up” of Domino domain monitoring with Domino domain monitoring. If the Domino domain monitoring data misses its scheduled pull of data, you will find an Error retrieving DDM data from server Mail1/Caribbean: Remote system no longer responding Event in DDM.nsf on the collection server.

We found that a simple collection hierarchy (hub and spoke) worked best in our lab environment. In larger production environments, though, you might want to set up multiple collection hierarchies based on server types (mail, Web, and so on) or by application. We also recommend configuring a backup collection server for fault tolerance. If your primary collection server fails, you will continue to collect data on the backup server without interruption.

As depicted in Figure 2-2 on page 17, Domino domain monitoring Event data from the mail servers is aggregated on MailHub and data from the application and Web servers is aggregated on AppHub. HighHub aggregates from both MailHub and AppHub and therefore contains Domino domain monitoring Event data for the entire domain. With this particular setup, mail and application administrators would utilize Domino domain monitoring on MailHub and AppHub, respectively. Administrators with domain-wide monitoring responsibilities would utilize Domino domain monitoring on HighHub.

In addition to these three collection hierarchies, there is another collection hierarchy configured to monitor Application1, located at the bottom of Figure 2-2 on page 17. Application1 runs on server App1 and the Application1 user mail files reside on Mail2. In this scenario, there is one administrator assigned to monitor Application1, and he or she would utilize Domino domain monitoring on App1.
2.4 Filters

Filters control the volume of Events that appear in DDM.nsf. Filters are used to specify what type of Events should be included in the Domino domain monitor database, not what Events should be excluded.

The configuration of filter documents is very straightforward. If you select Log All Event Types, you have the option to select the severities that you want to get filtered through. If you select Log Selected Event Types, you have the option to select the Event types and the severity level for each type that you want to get filtered through, as seen in Figure 2-3 on page 18. A filter can be applied to all servers in the domain, Special Target Servers, or servers that you select.
2.5 User interfaces

The following sections describe the user interfaces for Domino domain monitoring: the Monitoring Configuration database (Events4.nsf) and the Domino Domain Monitor database (DDM.nsf).

2.5.1 Monitoring Configuration (Events4.nsf)

All Domino domain monitoring configuration is done in the Monitoring Configuration database (Events4.nsf). This configuration database is also used to store Event generators and handlers, statistic collection documents, and other non-Domino domain monitoring configuration related documents.

Note: The Monitoring Configuration database (Events4.nsf) is named Statistics & Events on pre-Domino 6.0 servers.
To create Domino domain monitoring configuration documents, you can use the Domino Administrator client. The Domino domain monitoring configuration views can be accessed under the Configuration tab and Monitoring Configuration section, as seen in Figure 2-4.

![Figure 2-4 Monitoring Configuration views from the Domino Administrator client](image)

**Attention**: As of the writing of this Redpaper, the Server Collection Hierarchy view is not available from the Domino Administrator client.

Alternatively, you can directly open the Monitoring Configuration database on the appropriate server with your Lotus Notes® client by selecting **File → Database → Open**, as shown in Figure 2-5 on page 20.
As seen in the left navigation section in Figure 2-5, the Probe, Filter, and Collection Hierarchy configuration views are under the DDM Configuration section. The views appear slightly different in the Notes client than the Domino Administrator client. The DDM probe views are by author, probe server, and type (feature area). Open Monitoring is not a view but a link to the Domino domain monitor database (DDM.nsf).

### 2.5.2 Domino domain monitor database (DDM.nsf)

The result of a triggered probe is an Event that is stored in an Event document in the Domino domain monitor (DDM.nsf) database, commonly referred to as DDM.nsf. DDM.nsf can be opened in the Notes client, or the Domino Administrator client by clicking the **Open Domain Monitoring** button, which is under the Analyze section in the Server/Analysis tab.

The Domino domain monitor database is the user interface for Domino domain monitoring and contains many features. This section will only focus on the Event views, which are highlighted with boxes in Figure 2-6 on page 21. The rest of the user interface components will be covered in 2.9, “Management of DDM.nsf” on page 33, where we discuss how to best use the Domino domain monitor database for management of server issues within a matrixed Domino administration team.
Chapter 2. Domino domain monitoring: components and “out-of-the-box” configuration

Figure 2-6  DDM.nsf database view by Type

We will quickly review the views available in the Domino domain monitor database. The box labeled A in Figure 2-6 contains several views that all display Events based on different criteria, with the exception of Open Configuration. The tabs in box B allow you to change the view based on the status of the Event documents. The button labeled C toggles between all view documents and unread view documents.

The first view in box A, My Events, displays all open and closed Events that have been assigned to you. When this view is selected, the tabs in the blue box are replaced with a My Events tab.

The rest of the views are self-explanatory; however, there are a few features to note. The middle tab in the blue box, Recent Events, displays all open and closed Event documents that have been created or modified in the last seven days. The All Events tab in the blue box displays all Event documents, including permanently closed Event report documents. Event document statuses will be discussed in 2.6, “Events” on page 22.

Similar to the Monitoring Configuration database, the last option in the box A is not a view, but a link to the Monitoring Configuration database (Events4.nsf). The Show Administrators tab on the bottom-right of Figure 2-6 is technically a view, but we will cover this feature in 2.9, “Management of DDM.nsf” on page 33.

Important: For performance reasons, the DDM.nsf views do not automatically get updated when opened. The views are updated every 15 minutes, so they are relatively up-to-date and should open quickly. You can issue <F9> to refresh a view manually.
2.6 Events

Domino domain monitoring events have different types, statuses, and severities, as detailed in the following sections.

2.6.1 Event types

Domino domain monitoring Events are categorized into two types: Simple and Enhanced.

**Simple**

Simple Events is the new term used in Domino 7 for Events that are not associated with or do not contain target information. Generally, most Events logged to the Domino server console are Simple Events.

**Tip:** The list of Simple Events is available in the Advanced/Event Messages by Text or Advanced/Event Messages by Task views in the Monitoring Configuration database.

**Enhanced**

Enhanced Events must contain at least one field with specific target information, which can be servers, databases, agents, or a user-specified target. Target information defines the context of the Event.

Enhanced Events are generated in one of the following ways:

1. A Domino domain monitoring probe configuration document. These are the new Domino domain monitoring probe documents, which are stored in the DDM Configuration/DDM Probes views in the Monitoring Configuration database.
2. A Domino Event generator, as described in 1.6, “Event generator and Simple Event reporting into DDM.nsf” on page 8. Event generators are “old” pre-Domino 7 functionality, which are stored in the Event Generators views in the Monitoring Configuration database.
3. Any other Event with specific target information appearing in the DDM Event report.

**Important:** To clarify point number three, there are some Events that have no associated configuration documents. These enhanced Events are triggered by probes that are embedded into the Domino code and are not configurable.

2.6.2 Event statuses

Domino domain monitoring Events can have one of three different statuses: Open, Closed, and Permanently Closed

**Open**

All new Events are created in the Open state. If closed, Events can also be reopened by either a Domino administrator or by Domino domain monitoring if the same Event is triggered again with a different text message or severity.

**Important:** An Event report document for a particular Event will remain closed if the Event is reported with the same exact text message and severity. Otherwise, Domino domain monitoring will reopen the Event with the new Event text or severity.
Closed
Events can be closed by either an administrator, via the Domino domain monitor database interface, or via a clearing Event. A clearing Event, which must have a severity value of Normal, will close an Event report document if a resolution of the problem is detected. An example closed Event is “Server task IMAP Server on Mail2/Caribbean is running.” Not all Events have associated clearing Events.

Permanently Closed
Events that are permanently closed will be reported in the Domino domain monitor database, but will not be reopened. Event occurrence counts and prior Event history continues to be tracked for Permanently Closed Events.

Permanently Closed Events will only appear in the All Events view and you can reopen or close them at any time. You would set a Permanently Closed Event to Closed if you have manually resolved the issue and still want Domino domain monitoring to report it as Open if the issue occurs again in the future.

Note: You may see Events that are similar to Events you have set to Permanently Closed still being generated in DDM.nsf. The reason is because the entire text of an Event, including any user or database name, is taken into account when the Domino domain monitoring code compares Events.

2.6.3 Event severities

There are five different severity levels in Domino domain monitoring: Fatal, Failure, Warning (High), Warning (Low), and Normal. Each severity is listed below starting with the most severe and ending with Normal.

1. Fatal: The highest severity level reserved for the most critical errors.
2. Failure: The second highest severity level. Events categorized in this way are still considered to be major problems.
3. Warning (High): Used for many configuration type issues or moderate errors that occur. These are not as critical as a Fatal or Failure Event.
4. Warning (Low): Used for many configuration type issues, non-critical errors, or informational type Events.
5. Normal: Used mostly when clearing problems out of the Event queue. Events categorized at this severity level could have previously been at higher severity levels and are now resolved. Domino domain monitoring detects that fixes have been made and will change the Event to a normal severity level. Older, informational Events from Events4.nsf can also have a severity level of Normal.

Note: Normal severity Events will never be in an Open State. They can only be seen in a view that includes Closed or Permanently Close Events. Event statuses were discussed in detail in 2.6.2, “Event statuses” on page 22.
2.6.4 Event document example

This section provides an example Event document. While this Redpaper will not cover all of the possible information contained within an Event document, this is a representative example of typical Event documents.

The following document components are highlighted by numbers in Figure 2-7:

1. Event document status (Open, Closed, or Permanently Close) and team member assigned to address the Event.
2. Server and domain that generated the Event and Event class (Simple or Enhanced).
3. Target information (Server, Database, Agent, User). Simple Events do not have any target information.
4. Corrective action solution box (appears after clicking the Choose solution button). If there is only one possible solution, the solution will be performed immediately, and there will not be a pop-up box presented with only one option.
5. Event change history (updated manually by users and automatically by Domino domain monitoring).

Figure 2-7 Example Event document generated with a built-in probe

The sections that are not highlighted with boxes in Figure 2-7 contain more specific information related to the Event. We will explain how to customize this information, as well as how to create or modify corrective actions, in 2.7, “Customize DDM.nsf from Event documents” on page 25.

Some Event documents will contain two related errors: the root cause and a secondary error. An example of this type of Event is shown in Figure 2-8 on page 25.
Chapter 2. Domino domain monitoring: components and “out-of-the-box” configuration

2.7 Customize DDM.nsf from Event documents

In this section, we will discuss how to customize DDM.nsf with Server and Addin Task Event documents (not to be confused with Event report documents in DDM.nsf), which is where you can add or modify probable cause, possible solution, and corrective actions for Events.

The Server and Addin Task Event documents are located in the Advanced/Event Messages by Text view in the Monitoring Configuration database, and can be accessed from any Event document in the DDM.nsf database by clicking the document link located in the Severity and Type field, as shown in Figure 2-7 on page 24.
Each Server and Addin Task Event document contains three tabs.

**Basics**
The first tab, as shown in Figure 2-9, contains Event type, subtype, and severity information, as well as other information related to the Event. We recommend you do *not* modify the fields on this tab because Domino domain monitoring relies heavily on certain default values for its operation, especially the Original text and Value fields. You might want to increase or decrease the severity level of an Event depending on how important the Event is in your Domino environment. Any other updates on this tab should be performed with caution.

![Figure 2-9 Basics tab on the Server Addin Task Event document](image)

**Probable Cause/Possible Solution**
The middle tab, as shown in Figure 2-10 on page 27, is where you can add or modify probable cause and possible solution information. Again, the defaults should be used when available.

---

*Note:* There are default probable cause, possible solution, and corrective actions for *some* Events, not *all* of them. The documents with defaults are covered in 3.5.1, “Events with default corrective actions” on page 63.
Corrective Actions

The final tab, as shown in Figure 2-11, contains two input fields for corrective action scripts. The first input field is used for formula language scripts and the second is used for LotusScript.
2.8 Out-of-the-box configuration

The goal of this section is to help you get up and running with the “out of the box” Domino domain monitoring configuration. We will focus on the Domino domain monitoring components that are enabled or disabled by default in the Monitoring Configuration database. After covering what is configured by default, we will explain why Event documents might appear in the DDM.nsf database once you first open it for use.

As of Domino 7.0, Domino domain monitoring ships with the default settings listed in Table 2-1.

Table 2-1  Components with their default settings

<table>
<thead>
<tr>
<th>Components</th>
<th>Default settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Probes</td>
<td>47 probes in all nine feature areas are configured. Note: All 47 probes are disabled by default.</td>
</tr>
<tr>
<td>Collection Hierarchies (Optional)</td>
<td>No collection hierarchy is created by default.</td>
</tr>
<tr>
<td>Filters (Optional)</td>
<td>One filter is enabled to filter only Simple Events with a Fatal and Failure severity.</td>
</tr>
<tr>
<td>User interfaces</td>
<td>Events4.nsf and DDM.nsf are created by default.</td>
</tr>
<tr>
<td>Events</td>
<td>Events might start to get generated. This will be discussed in detail in 2.8.6, “Events created by default” on page 30</td>
</tr>
</tbody>
</table>

2.8.1 Domino 7 upgrade and Domino domain monitoring

As mentioned in Chapter 1, “Overview of Lotus Domino domain monitoring” on page 1, Domino domain monitoring is included (free) with Domino 7. There are no additional installation steps needed to enable the feature. Domino domain monitoring is automatically running once you bring your server(s) up for the first time on Domino 7.

Note: Domino domain monitoring requires the Event task to be running to work properly. The Event task loads automatically during server startup.

Before upgrading to Domino 7, there are some optional configuration steps to ensure Domino domain monitoring runs correctly across your Domino environment(s):

- If you are upgrading from a Domino 7 beta release to Domino 7 (non-beta), you should delete all of the beta copies of DDM.nsf in your domain before you upgrade. After the upgrade, the DDM.nsf will get recreated automatically, but you will have to manually recreate the Probe configuration documents, or enable any default probes that you were using. Recreating the Domino domain monitor database and probes will eliminate any
possible design and Event mismatch issues between the Domino 7 beta release versus
the Domino 7 (non-beta) version.

- The second step is to ensure that the design of the Domino 7 Events4.nsf databases do
not have their design replaced with pre-Domino 7 templates. You might already have this
design control for system databases set up via ACL settings or connection documents, but
now is a good time to ensure your new Events4.nsf design, with the DDM.nsf
Configuration views, will not get overwritten after the upgrade.

### 2.8.2 Default probes

There are 47 default probes in the Monitoring Configuration database, and they are all
disabled. The probes are disabled so that the DDM.nsf database does not get flooded with
Events once you upgrade to Domino 7. There is one default probe created for every probe
type and associated subtype, with a few extras for Application Code and Messaging default
probes.

**Note:** There are a select number of probes that cannot be immediately enabled from their
default configuration because they require additional field values that are specific to the
probe or environment. For example, there are Messaging probes that require a Mail
Recipient to be specified, and there is no way for Domino domain monitoring to know what
mail file to select.

The default probes are there to quickly get you started with Domino domain monitoring. By no
means do you have to, or should you, enable all of the default probes immediately. The best
way to get familiar with Domino domain monitoring is to enable a select few of the default
probes that are applicable to your server(s).

For example, you might want to start off by monitoring your hub(s) for replication of
names.nsf, admin4.nsf, and other important system databases with the Scheduled
Replication Checks or the Replication Errors probes. Once you are comfortable with how
these probes work, you can enable probes in the rest of the feature areas that you need to
monitor. Details about enabling default probes and creating new probes are available in 3.3.2,
“Creating and configuring Domino domain monitoring Messaging Probes” on page 56.

**Note:** Default probes are authored by the Lotus Notes Template Development ID.

### 2.8.3 No collection hierarchies created by default

There is no collection hierarchy created by default because the requirements for data
aggregation are unique to different Domino environments. Collection hierarchy topologies
should be planned by a Domino administrator or architect before being implemented.

Collection hierarchies are a very important component in Domino domain monitoring. At least
one collection hierarchy should be created to get you started with data aggregation. As you
get more familiar with Domino domain monitoring and get more administrators using the
feature, you will want to create additional collection hierarchies for more specialized data
“rollup” across different servers within your Domino domain.
2.8.4 One default filter

There is one filter created by default, which allows the reporting of all Simple Events with Fatal or Failure severity on all servers. This default filter is created for the same reason the new Domino domain monitoring probes are disabled: to reduce the volume of Events created in the DDM.nsf database “out-of-the-box.”

You might find the default filter is sufficient, but you can increase or decrease the volume of Simple Events (or Simple and Enhanced) as needed by modifying the default filter or creating additional filters. As described in 2.4, “Filters” on page 17, you can create filters for specific server groups and only allow certain Event types to display in DDM.nsf.

Attention: Filters give you the option of allowing Simple Events of severity Normal to get reported in the DDM.nsf database. To prevent your DDM.nsf database from getting flooded with informational only Events, we suggest you do not filter Simple Events of severity Normal.

2.8.5 “Out-of-the-box” user interfaces

The Monitoring Configuration and DDM.nsf database are both available once you bring your server(s) up for the first time on Domino 7.

Note: By default, DDM.nsf has the “Enforce a consistent Access Control List across all replicas” option checked. We recommend that you keep this setting enabled, because you will lose the action button functionality in local replicas of DDM.nsf if you disable it.

2.8.6 Events created by default

This section explains what Events you might find in DDM.nsf before you enable any of the new Domino domain monitoring probes. We present example Events and explain what triggered the creation of the Event.

Simple Events of Fatal or Failure severity example

Simple Events of Fatal or Failure severity might appear in the DDM.nsf database given the default filter documented in 2.8.4, “One default filter” on page 30. An example of a Simple Event is displayed in Figure 2-12 on page 31.
As described in 2.7, “Customize DDM.nsf from Event documents” on page 25, clicking the document link in the Severity and Type field will bring you to the Server and Addin Task Event message document, as shown in Figure 2-13.

Enhanced Events from Domino Event generators example
As discussed in 1.6, “Event generator and Simple Event reporting into DDM.nsf” on page 8, Event generators are reported into the DDM.nsf database by default. An example of a Enhanced Event triggered by a Domino Event generator is displayed in Figure 2-14 on page 32.
Figure 2-14  Example Enhanced Event document created by an Event generator

Clicking the Triggered by: document link will bring you to the Event generator document, as shown in Figure 2-15.

Figure 2-15  Task status Event Generator associated with the enhanced Event shown in Figure 2-14
Enhanced Events from built-in probes example
Since built-in probes are automatically running, you will probably find some of these Events when you first get started using the DDM.nsf database. An example of a Enhanced Event document triggered by a built-in probe is displayed in Figure 2-16.

Figure 2-16 Example Event document generated by a built-in probe

Note: Simple Events and built-in probes are reported into DDM.nsf automatically, which is why there is no Triggered by field in their Event documents.

2.9 Management of DDM.nsf
An important aspect of Domino domain monitoring is the integrated collaboration features, which can be fully utilized by a small Domino administration team or a highly matrixed enterprise with multiple support teams. This section provides tips and best practices on how to manage your DDM.nsf database.

You have set up Domino domain monitoring, configured probes, and seen Events pour into the DDM.nsf database. Now what? How do you keep the problems and information you are reviewing at a manageable level, and how can you best utilize DDM.nsf on a daily basis? This section will discuss the different collaboration features and help you to manage your DDM.nsf database in the most efficient way.

We will cover issues related to user access to certain Domino domain monitoring features, maintaining a reasonable DDM.nsf database size, Sametime awareness, and other important topics that address the manageability of the Domino domain monitor database.
2.9.1 ACL settings for DDM.nsf access and usage

ACL settings for DDM.nsf define what buttons and functionality you will see when you open the database. For example, with Author access or lower, you are only allowed to open, modify, and close your own Event documents. With Editor access or higher, you are able to add comments to any documents, not just your own. Table 2-2 lists the minimum access and roles in the ACL that are required to perform certain actions in DDM.nsf.

<table>
<thead>
<tr>
<th>Action</th>
<th>Minimum access and required roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read Event documents.</td>
<td>Read access</td>
</tr>
<tr>
<td>Add comments to your own Events.</td>
<td>Author access</td>
</tr>
<tr>
<td>Add comments to another administrator's Events.</td>
<td>Editor access</td>
</tr>
<tr>
<td>Assign or reassign your own Events.</td>
<td>Author access and the [Assign Events] role</td>
</tr>
<tr>
<td>Assign or reassign Events regardless of owner.</td>
<td>Editor access and the [Assign Events] role</td>
</tr>
<tr>
<td>Change state of your own Events.</td>
<td>Author access and the [Change State] role</td>
</tr>
<tr>
<td>Change state of Event regardless of owner.</td>
<td>Editor access and the [Change State] role</td>
</tr>
</tbody>
</table>

**Note:** The [Assign Events] and [Change State] ACL roles are specific to DDM.nsf.

Using these ACL settings is the best way to control the access and use of DDM.nsf. Many users might be able to see the Events being reported, but the actual management of the database should be granted to a select few. The administrators and architects who oversee the entire domain are the best ones to manage DDM.nsf at a high-level, delegating Events to the appropriate administrators as necessary.

2.9.2 Replication settings to control the size of DDM.nsf

One recommended setting to help control the size of DDM.nsf is found as a space saver option for the database. This option will purge documents that have not been modified in X number of days. The default value for X in this case is 90 days, and it is recommended to keep this setting at the default for now. As you become more familiar with DDM.nsf in your domain, you can tweak this setting to the number of days that makes the most sense for you.

You will find this setting on the File menu of DDM.nsf under **Replication → Settings** and by clicking the **Space Savers** tab, as shown in Figure 2-17 on page 35.
2.9.3 Domino domain monitoring filters

Filters were discussed briefly in 2.4, “Filters” on page 17, but we will take a more in-depth look at them here. As described previously, there is one filter included out-of-the-box when you first upgrade to Domino 7.0. The out-of-the-box filter allows Simple Events of severity Fatal or Failure to be reported to DDM.nsf. All other Simple Events will not be reported to DDM.nsf with this default filter enabled. You will probably find at first that this default Filter is sufficient for you. As you continue to use DDM.nsf you will learn what types of issues you are finding in your domain, you may consider using the full functionality provided by the Filter feature. You can filter Events down to a very granular level, and you can have multiple filters running on many different machines at the same time.

You may be wondering why you would want to use a complex filtering scheme? Well, filters can be tweaked further to provide more control over what is reported in DDM.nsf. An example could be a mail server running Domino domain monitoring where you only want to see the most severe messaging problems reported. You can use a filter to run on the Special Target Group ‘Mail Servers’ and only report on Messaging Events of type Fatal and Failure. With the high volume of Events coming in from all over the domain, setting up a filter such as this one would allow you to focus on your most important problems pertinent to your mail servers. The same scenario could be applied to an application server where you may only want to see Events of type Application reported to ddm.nsf.

In general, do not be afraid to create different Domino domain monitoring filters and see how they help you focus on the most important problems quickly on a per server basis.

**Tip:** As shown in Figure 2-17, a selection formula is configured in DDM.nsf. This is the selection formula we discussed earlier in 2.3, “Collection hierarchies” on page 15. We recommend you do not modify this selection formula. If you modify this replication formula, you risk breaking the collection hierarchy data aggregation.
2.9.4 Managing Events

Managing events is an important component of your overall management of Domino domain monitoring. The following sections provide details on the key aspects of an event.

An Event life cycle
Another important concept of Domino domain monitoring is understanding how the Event life cycle works. Once you have a solid understanding of this aspect you will be able to move problems through DDM.nsf in a quick and efficient manner. The interesting thing about an Event's life cycle is that an Event is never really “dead.” An Event can always come back to life by being reopened again.

There are four major steps to an Event's life cycle:

1. Open
2. Assigned
3. Closed
4. Reopened

The Event is opened
This occurs when Domino domain monitoring finds a problem and raises either a Simple or Enhanced Event. The Event types are discussed in detail in 2.6.1, “Event types” on page 22.

The Event is assigned
Event assignment is an important feature in DDM.nsf. It allows the volume of problems and Events reported to DDM.nsf to be divided up among different users so that many people can work together to run a solid, problem free domain. Assigning Events is easy and can be done from two different view levels.

1. On the main DDM.nsf view, there is a large "Assign..." button on the top of the window pane. Simply place your focus on a specific Event or place a checkmark next to multiple Events, and click the Assign button. A pop-up box appears and gives you the option to select who you would like to assign the Event(s) to. With the first option, you can select administrators from a list from the Domino directory or you can assign the Event to yourself by choosing the second radio button Myself. It is a good idea to add comments in the assignment dialog, as shown in Figure 2-18 on page 37, these will be included in the Event history at the bottom of every Event.

2. The second way to assign an Event is after you have already opened it. The "Assign..." button will now appear in the top left hand corner of the open Event, and you would follow the same directions to complete the assignment as described above.
Figure 2-18 Assigning an Event with comments

Tip: Filling in the comments box as you are assigning Events is a great way to keep a detailed history of a specific problem. The comment box allows you to write a great deal of information, so feel free to get as detailed as you need to. Also, if the same Event has been closed and then reopened, you will still have these comments in the history to refer back to and help you to remember how you reached a solution the last time the problem occurred.

For the smaller IT departments, think of using the assignment feature as another way to focus on the most important problems. For large enterprises with multiple administrators, the assignment feature will become invaluable to you for keeping tabs on problems and following the progress of solutions.

When you open DDM.nsf, you will see the My Events view at the very top of the left hand pane. By clicking My Events, you will see a new view that encompasses your queue. It is first sorted by Open Events in your queue, and then by the severity of these Events. By working out of this queue, you will be able to focus on Events of top priority and work much more efficiently than just solely working in larger views in DDM.nsf.

For large enterprises, using the assignment feature will help you move through issues quickly. As new Events come into DDM.nsf, perhaps it will be a good idea to assign them based on feature area. For Events of type Messaging, quickly move the Event into the mail administrator's Event queue so that he or she is able to apply their expertise to finding the right solution.

Important: When an Event goes through any state change initiated by a user (opened, closed, or reopened), the Event is automatically assigned to this user.

The Event is closed
Events can be closed by either an administrator or via clearing Events, which occur after a previously reported problem has been fixed.

Closing versus Permanently Closing Events was briefly discussed in 2.6, “Events” on page 22. In summary, an Event that has been closed can be reopened by an administrator or by Domino domain monitoring if the problem occurs again. Permanently Closed Events will never be Reopened by Domino domain monitoring and can only be "brought back to life" by an administrator.
Why would you want to Permanently Close an Event versus just closing an Event? If you do not want to see a specific Event reported to DDM.nsf any longer, you can Permanently Close the Event. The next time this problem occurs, Domino domain monitoring will realize that you have Permanently Closed the Event and will not reopen it in the DDM.nsf database. Note that when Domino domain monitoring makes the decision of whether to reopen the Event or not, the exact text of the Permanently Closed Event must match what Domino domain monitoring is trying to report at that time. So database names, server names, and user names, which in some Events are embedded into the Event Text itself, all must match.

In Figure 2-19, the user name "External Admin" is embedded in the Event text. If this Event is now Permanently Closed, any other attempt to access the database by External Admin will not be reopened as a problem in ddm.nsf. However, an attempt to access database by “Internal Admin” will still be reported to ddm.nsf. Because of the different user name embedded into the Event text, the two Events are considered separate and different Events. But say you want to get rid of all of these "ATTEMPT TO ACCESS DATABASE... " Events all together. How do you do that? Well, using the Permanently Close feature is not the way. What you can do is reduce the severity of the "ATTEMPT TO ACCESS DATABASE... " error in Events4.nsf, the configuration database. By default, the Event severity for this error is set to Warning (High), as shown in Figure 2-19. If you change this Event’s severity to Normal, it will never be reported as an Open Event. But you could also change the severity to Warning (Low) and then use Domino domain monitoring Filters to stop the reporting of Warning (Low) Events, or more specifically, Warning (Low) Security Events. There are many options here, and many ways to see exactly what you want while removing all the things that you do not want to see.

Figure 2-19 ATTEMPT TO ACCESS DB Event message

Testing your solution

You have found a solution to an open Event, closed the Event, and now maybe you would like to test your solution. You can rerun probes manually with console commands in order to immediately test whether a problem reported in DDM.nsf is actually fixed or not. If the Event you closed remains closed after your manual test, then you can be confident the issue is resolved.

This test can be done with any enabled, scheduled Domino domain monitoring probe in Events4.nsf. To run a probe on demand, simply go to the server console and type the command `tell Event dumpprobes`, as shown in Example 2-1. This will give you a list of all enabled, scheduled probes on the current server by ID number.

Example 2-1 Console output from `tell Event dumpprobes` command

```
> tell Event dumpprobes
> ID      Probe Message Queue Name
-----   ------------------------
1   ReplicationEvents-Events-1
32630   Database-32630
38178   ReplicationCheck-Events-3
34138   Messaging-34138
7990   Messaging-7990
7998   Messaging-7998
7994   Messaging-7994
```
You can then choose which probe you want to run by ID number and type the console command `tell Event run probe xxxxx`, where `xxxxx` is the ID number. The following example command will run the Database probe with ID #32630:

```
> tell Event runprobe 32630
```

**Important:** ID number is the decimal format of the Probe Configuration Document's Note ID. If there are multiple probes in one feature area, like Messaging in Example 2-1, you will need to discover which ID number to use in the `runprobe` command. To do this, go to the Probe Configuration Document in Events4.nsf of the probe that you would like to run. Look at the document properties, and find the Note ID on the Document IDs Tab. Use the calculator function to translate this hexadecimal number into decimal format.

You may not see the probe being run from the server console unless you have special debug variables turned on (see 4.2, “Domino domain monitoring console commands and debugging information” on page 91), but rest assured that the probe is actually running in the background.

To summarize, after you have resolved an enhanced Event reported to DDM.nsf, you can test your solution by running a probe on demand and seeing if the Event remains closed in ddm.nsf. Closing and Permanently Closing Events remains an important part of an Event's life cycle and should be used as part of the management process of DDM.nsf.

**Note:** Not all Events have an associated “clearing” Event. If you are testing an Event that does not have a “clearing” Event, instead of checking to see if the your Event report document automatically closes in DDM.nsf, make sure that the occurrence count and time does not get updates. Once you confirm that, you can close the Event report manually.

### The Event can be reopened

As previously discussed, Events can be reopened either by DDM.nsf or manually by an administrator. If DDM.nsf determines that a Closed Event has occurred again, it will automatically be reopened and the occurrence count will be incremented in the Event document (see Figure 2-20).

![Figure 2-20 Occurrence count message generated by DDM.nsf](image)

At any time an Administrator can also Reopen any Event document if the problem remains unresolved, or if it is closed manually by mistake.

#### 2.9.5 ST Awareness

Another collaboration feature built into ddm.nsf is the Sametime awareness feature. To see this feature you will need a Sametime server specified in the Location document from which you are viewing ddm.nsf. Once you have a Sametime server specified you will notice the text, “Show Administrators” or “Hide Administrators”, in the bottom left hand pane of DDM.nsf.

This view contains the names from two fields in the server document - the Full Access Administrators field and the Administrators field. Any groups or nested groups listed in these two fields will be expanded down to the name level. As of Domino 7.0, server names can also be listed in the Sametime awareness list. If you are logged onto Sametime, you will be able to see the administrator's Sametime status.
As you can see in Figure 2-21, the collection server replica of DDM.nsf, in this case Hub, will display a drop down box where you can select either the collection server where you are accessing DDM.nsf or other servers in the collection hierarchy where Hub is the collection server. For example, Hub is a collection server for Mail1, so from the Hub replica of DDM.nsf you can select Mail2 from this drop down list, as shown in the first box in Figure 2-22.

Selecting Mail2 will display the administrators that are defined for the Mail2 server, as shown in the second box in Figure 2-22. At the click of a button you will able to start a Sametime chat with one or more of the administrators for any given server that reports into DDM.nsf. Perhaps a critical problem has come into ddm.nsf and you need to quickly get in contact with the right administrator. Simply use this built in Sametime feature to ask questions about the problem and collaborate within Domino domain monitoring in real-time.

Note: If changes are made to the server document fields Full Access Administrators or Administrators, these changes will not be reflected in the Sametime Awareness list in ddm.nsf until the Event task has been reloaded on the server from which you are viewing ddm.nsf.

Another example of Sametime integration is depicted in Figure 2-23 on page 41. Using a new Domino 7 feature that dynamically builds a contact list group based on the active documents User fields, administrators can access end users or other administrators directly from their Sametime client.
Figure 2-23  Collaboration with dynamic Sametime contact list creation
A Domino domain monitoring scenario

In Chapter 2, “Domino domain monitoring: components and “out-of-the-box” configuration” on page 11, we discussed the general architecture, functionality, and components of Domino domain monitoring. In this chapter, we will provide a scenario that shows how Domino domain monitoring can be leveraged to resolve real world issues in a mail-centric example environment, including:

- Proactively monitor IMAP, POP3, and SMTP
- Detect and automatically fix a corrupt database using custom corrective action
- Proactively monitor Out of Office agents after mail template upgrades
- Extend existing application agents to raise Events
3.1 Overview of the Domino server environment

This section describes the Domino server environment we use throughout the chapter. Our Redbooks lab Domino domain is not intended to simulate a “real-world” production environment, but serves as a small and easy to understand example installation.

3.1.1 The Redbooks lab Domino server topology

Our server topology (see Figure 3-1) represents a somewhat mail-centric environment. The private intranet behind the firewall contains two clustered Domino 7.0 mail servers (Mail1 and Mail2) and one Domino 7.0 hub server (Hub), which acts as the administration server of the Domino directory. We also installed a Sametime 7.0 server on the hub system to provide instant messaging and awareness capabilities. Outside of the firewall, we have a Domino 6.54 server (External) to send and receive external e-mails via SMTP and allow mail retrieval via IMAP or POP3 for selected mail files.

Note: A Sametime server is not required for Domino domain monitoring. However, DDM.nsf is fully Sametime-enabled, and we wanted to take advantage of this in our example environment.

![Figure 3-1 The Redpaper team's Domino server topology](image)

3.1.2 Planning the server collection hierarchy

Domino domain monitoring probes run on Domino 7 and higher servers and report Events to the local DDM.nsf. Rather than visit every Domino Domain Monitor database on every server to review the probe results, we created a Domino domain monitoring server collection hierarchy to have the DDM.nsf data from the Mail1 and Mail2 servers aggregated into DDM.nsf on our hub server (see Figure 3-2 on page 45).
For more details on collection hierarchies, see 2.3, “Collection hierarchies” on page 15 and visit the online Lotus Domino Administrator 7 Help documentation at the following URL:

http://www-12.lotus.com/ldd/doc/domino_notes/7.0/help7_admin.nsf

**Tip:** For a large server environment, we recommend configuring backup collection servers for fault tolerance. If one collection server fails, you will continue to aggregate probe results on the backup collection server without interruption.

### 3.1.3 Configuring the server collection hierarchy

Domino domain monitoring server collection hierarchies are configured in the ‘Monitoring Configuration’ (Events4.nsf) database by doing the following steps:

1. Open the Monitoring Configuration database (Events4.nsf).
2. Make sure the DDM Configuration view section is expanded.
3. Select **Server Collection Hierarchy**.
4. Click the **New Server Collection Hierarchy** action button.

For our example Domino environment, we will select **One server will collect from all servers in the domain** (see Figure 3-3 on page 46).
**Important:** Make sure the design of the Domino directory has been upgraded to Domino 7. If you have a pre-Domino 7 names.nsf design, the server collection hierarchy code will allow you to pick pre-Domino 7 servers as collection or subordinate servers, which is not supported, since pre-Domino 7 servers cannot run Domino domain monitoring.

![New Server Collection Hierarchy](image)

Click **OK** to automatically create the new server collection hierarchy (represented as a new outline) with the hub server collecting from all available Domino 7 servers (Mail1 and Mail2, in our case) (see Figure 3-4).

![Monitoring Configuration](image)

**Figure 3-3** New Server Collection Hierarchy

**Figure 3-4** The automatically created server collection hierarchy
The server will automatically detect and recognize any newly created or modified server collection hierarchies. There is no need to restart the server or the Event task.

**Note:** The time it takes a collection hierarchy to recognize changes depends on where in the hierarchy the change is made. Changes to the immediate children of a given server will be detected within a few minutes. The further down the hierarchy a change is made, the longer it will take the selective replication to update all the servers in the hierarchy.

### 3.1.4 Verifying the server collection hierarchy

To verify that the Hub server now aggregates the probe results from the Domino domain monitor databases on Mail1 and Mail2 we open the DDM.nsf on Hub and select the **By Server** view. As expected, data from all three servers is aggregated, as shown in Figure 3-5.

![DDM](image)

**Figure 3-5** Hub/Caribbean now contains rolled-up data from the collected Mail servers

### 3.2 Monitoring the external mail server

To make sure our users can retrieve their mail messages remotely when on the road or working from home, we need to ensure that the POP3, IMAP, and HTTP (for Domino Web Access) server tasks are running and functional. We also want to make sure that the SMTP service is available for proper mail delivery.

In our current Domino Topology, the External/Caribbean server, which acts as our external SMTP, POP3, and IMAP server, is still running Domino R6.54.

First, we are going to describe how to set up and monitor the tasks mention above on our External server, a pre-Domino 7 server where native Domino domain monitoring probes are not available.

Then we are going to upgrade our R6.54 server to Domino 7 and make use of the more powerful and more comprehensive Domino domain monitoring Messaging probes.
3.2.1 Monitoring TCP ports on pre-Domino 7 servers

Since our External mail server is running Domino R6.54, we created a “non-DDM” TCP Server Event Generator probe in the Monitoring Configuration database (Events4.nsf) (see Figure 3-6). The probe document enables the Hub server to probe External, which uses the I Spy task. The resulting Events are raised as Simple Events and get reported in DDM.nsf. So, even though we are not using the new Domino domain monitoring probes, the Events generated from this pre-Domino Event generator are stored and managed in the DDM.nsf database.

**Important:** We needed to change the Event severity raised by this Event generator from the default “Warning High” value to at least “Failure” to make this Event appear in the Hub servers DDM.nsf database. The reason is because the default Domino domain monitoring filter excludes all Simple Events that are not of severity Failure or Fatal. More information about the default filter can be found in 2.4, “Filters” on page 17.

**Tip:** We used the fully qualified DNS host name (external.cam.itso.ibm.com) as the server name to be probed rather than the Notes abbreviated name (External/Caribbean). Using the fully qualified name allows us to monitor any Domino server, even if it is not in the same Domino domain as the probing server. This allows you to set up a single Domino 7 server outside of your production Domino 6.x domain to start using Domino domain monitoring for managing the reported Events in DDM.nsf.

The I Spy server add-in task performs the actual port probes configured by this Event generator. If the I Spy task is not already loaded on server Hub/Caribbean (remember, we probe the External/Caribbean server remotely via the Hub server), we can load it manually, as shown in Example 3-1 on page 49.
Example 3-1  Loading ISpy server addin task manually

> load runjava ISpy
09/14/2005 04:42:00 PM  JVM: Java Virtual Machine initialized.
09/14/2005 04:42:00 PM  RunJava: Started lotus/notes/addins/ispy/ISpy Java task.
09/14/2005 04:42:06 PM  ISpy: Initialization complete.
>

Note: Make sure you type ISpy case sensitive, otherwise the server will report an error. This is because the word ‘ISpy’ in this particular context is actually referring to the name of a Java class name, and such references are always case sensitive. You are actually loading the runjava server task (which is case insensitive), which in turn starts an instance of the ISpy Java class.

3.2.2 Verifying the TCP port checks

To visually verify that the TCP port checks are actually performed, we checked the server console of server External/Caribbean and saw that Hub/Caribbean indeed started to probe the server:

09/14/2005 04:35:21 PM  SMTP Server: hub.cam.itso.ibm.com (9.33.85.101) connected
09/14/2005 04:35:21 PM  SMTP Server: hub.cam.itso.ibm.com (9.33.85.101) disconnected. 0 message[s] received

Unfortunately, only the SMTP server task logged the incoming connection requests to the Domino server console. The POP3 and IMAP server tasks do not do this by default, although there is a notes.ini variable that can be used to simulate the default SMTP logging of incoming connection requests. Setting the notes.ini variable LOG_CONNECTIONS=1 will log all incoming connections to the Domino server console, regardless of the Domino server addin task involved.

Important: After setting LOG_CONNECTIONS=1, we had to restart the POP3, IMAP, SMTP, and HTTP server tasks for them to pick up the newly set configuration variable.

Now we can clearly see the probes are working from the Domino server console, as shown in Example 3-2. The Hub server (originating from IP address 9.33.85.101) opened port 143 (IMAP), 110 (POP3), 80 (HTTP), and 25 (SMTP) on the External server (on IP address 9.33.85.112).

Example 3-2  Console output resulting from setting LOG_CONNECTIONS=1 in notes.ini

<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>09/15/2005</td>
<td>02:42:12</td>
<td>Opened TCP/IP connection from 9.33.85.101,1675 to 9.33.85.112,143</td>
</tr>
<tr>
<td>09/15/2005</td>
<td>02:42:12</td>
<td>Opened TCP/IP connection from 9.33.85.101,1673 to 9.33.85.112,110</td>
</tr>
<tr>
<td>09/15/2005</td>
<td>02:42:12</td>
<td>Opened TCP/IP connection from 9.33.85.101,1672 to 9.33.85.112,80</td>
</tr>
<tr>
<td>09/15/2005</td>
<td>02:42:12</td>
<td>Closed TCP/IP connection from 9.33.85.101,1675 to 9.33.85.112,143</td>
</tr>
<tr>
<td>09/15/2005</td>
<td>02:42:12</td>
<td>Closed TCP/IP connection from 9.33.85.101,1673 to 9.33.85.112,110</td>
</tr>
<tr>
<td>09/15/2005</td>
<td>02:42:12</td>
<td>SMTP Server: hub.cam.itso.ibm.com (9.33.85.101) disconnected. 0 message[s] received</td>
</tr>
<tr>
<td>09/15/2005</td>
<td>02:42:12</td>
<td>Closed TCP/IP connection from 9.33.85.101,1672 to 9.33.85.112,80</td>
</tr>
</tbody>
</table>

We can enable even more output on the probing Hub/Caribbean server to see ISpy’s activity by setting DEBUG_ISPY=2 in the notes.ini of server Hub/Caribbean.
Attention: Debug settings can produce enormous amounts of additional output and can adversely affect server performance. It is recommended to only temporarily enable debug output for testing purposes or if instructed by IBM technical support personnel.

The last six lines of Example 3-3 show that ISpy is performing the port probes as configured.

Example 3-3  Console output resulting from setting DEBUG_ISPY=2

<table>
<thead>
<tr>
<th>&gt;ID</th>
<th>Probe Message</th>
<th>Queue Name</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>No probes loaded</td>
<td></td>
</tr>
<tr>
<td>09/15/2005 03:23:55 PM</td>
<td>RunJava JVM: Thu Sep 15 15:23:55 EDT 2005 ISpy: Starting IP probes...</td>
<td></td>
</tr>
<tr>
<td>09/15/2005 03:23:55 PM</td>
<td>RunJava JVM: Result IMAP response time = 31</td>
<td></td>
</tr>
<tr>
<td>09/15/2005 03:23:55 PM</td>
<td>RunJava JVM: Result SMTP response time = 31</td>
<td></td>
</tr>
<tr>
<td>09/15/2005 03:23:55 PM</td>
<td>RunJava JVM: Result HTTP response time = 188</td>
<td></td>
</tr>
<tr>
<td>09/15/2005 03:23:55 PM</td>
<td>RunJava JVM: Result POP3 response time = 266</td>
<td></td>
</tr>
</tbody>
</table>

Since the POP3, IMAP, HTTP, and SMTP server tasks are all running on the External/Caribbean server, no Event was raised and therefore no related output gets created in the DDM.nsf database on Hub.

To force the probe to trigger an Event, we are going to unload the IMAP server task on server External/Caribbean. The probing server should detect this, raise an Event of type Simple and severity Failure (as configured in our TCP Server Event Generator document in Events4.nsf), and create an Event Report document in the probing server's DDM.nsf database.

After waiting for about 15 minutes (the probe schedule interval, as shown in Figure 3-6 on page 48), we see on the probing Domino servers console that the IMAP probe timed out trying to probe the remote IMAP port (reported as -1) (see Example 3-4).

Example 3-4  IMAP probe timeout reported as -1 on the probing Domino server's console

<table>
<thead>
<tr>
<th>&gt;ID</th>
<th>Probe Message</th>
<th>Queue Name</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>No probes loaded</td>
<td></td>
</tr>
<tr>
<td>09/15/2005 03:53:15 PM</td>
<td>RunJava JVM: ISpy: Reloading DDM probe config if needed</td>
<td></td>
</tr>
<tr>
<td>09/15/2005 03:53:57 PM</td>
<td>RunJava JVM: Thu Sep 15 15:53:57 EDT 2005 ISpy: Starting IP probes...</td>
<td></td>
</tr>
<tr>
<td>09/15/2005 03:53:58 PM</td>
<td>RunJava JVM: Result POP3 response time = 16</td>
<td></td>
</tr>
<tr>
<td>09/15/2005 03:53:58 PM</td>
<td>RunJava JVM: Result SMTP response time = 16</td>
<td></td>
</tr>
<tr>
<td>09/15/2005 03:53:58 PM</td>
<td>RunJava JVM: Result HTTP response time = 31</td>
<td></td>
</tr>
<tr>
<td>09/15/2005 03:53:58 PM</td>
<td>RunJava JVM: Result IMAP response time = -1</td>
<td></td>
</tr>
</tbody>
</table>
Per our TCP Server Probe, this raised an Event of severity Failure, which is now reported in server Hub/Caribbean's DDM.nsf database as a new Event document, as shown in Figure 3-7.

![Figure 3-7 The IMAP Event report in the DDM.nsf view](image)

As shown in Figure 3-8 on page 52, the newly created Event is not assigned to anyone yet. Also, since this Event is a Simple Event, it does not contain additional information or even corrective actions. The Event report document is classified as Simple because it did not get created by a new Domino domain monitoring probe, but with a TCP Server Event Generator probe from a pre-Domino 7 server.
We will assign this Event to the Network Admin, who will address the issue and document his or her work within the Event document itself. The Network Admin closes the issue after reloading the IMAP task on External/Caribbean. All assignments, comments, and status changes to the Event document are recorded in the Event Change History section, which is shown at the bottom of Figure 3-9.

**Note:** As of Domino 7.0, I Spy probes always end up in DDM.nsf as Events of type Server, even though they show up in other databases such as STATREP.NSF as type Web or Mail.
We then wanted to see what happens when the same situation occurs again, or if any of the other server tasks probed by the TCP Server Event generator probe are unavailable, so we shut down the POP3 server task. As expected, the ISpy task timed out probing for the (now unavailable) POP3 port, as seen in Example 3-5.

**Example 3-5  ISpy task timing out**

<table>
<thead>
<tr>
<th>Date/Time</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>09/16/2005 01:03:48 PM</td>
<td>RunJava JVM: Fri Sep 16 13:03:48 EDT 2005 ISpy: Starting IP probes...</td>
</tr>
<tr>
<td>09/16/2005 01:03:49 PM</td>
<td>RunJava JVM: Result IMAP response time = 32</td>
</tr>
<tr>
<td>09/16/2005 01:03:49 PM</td>
<td>RunJava JVM: Result HTTP response time = 15</td>
</tr>
<tr>
<td>09/16/2005 01:03:49 PM</td>
<td>RunJava JVM: Result SMTP response time = 22</td>
</tr>
<tr>
<td>09/16/2005 01:03:49 PM</td>
<td>RunJava JVM: Result POP3 response time = -1</td>
</tr>
<tr>
<td>09/16/2005 01:03:49 PM</td>
<td>RunJava JVM: Fri Sep 16 13:03:49 EDT 2005 ISpy: ...IP probes complete.</td>
</tr>
</tbody>
</table>

Instead of creating a new Event report document in DDM.nsf, the server updated the existing Event report document by:

- Changing the Event report status from Closed back to Open.
- Making the POP3 task error the Most Recent Event.
- Moving the prior IMAP task error (Figure 3-9 on page 52) to the Prior Events tab.

Figure 3-10 shows the updated report.

**Figure 3-10  Updated existing Event report**
3.3 Upgrading the server to Domino 7

As we have seen in 3.2, “Monitoring the external mail server” on page 47, servers running pre-Domino 7 code can only report into DDM.nsf with pre-Domino 7 methods.

To take full advantage of Domino domain monitoring, we are going to upgrade our External/Caribbean mail server to Domino 7. Not only does this allow us to use many of the new features available in Domino 7, such as new support for SMTP DNS white lists, and so on, but it also allows us to use the following Domino domain monitoring Messaging probes:

- Mail DSN Probe
- Mail Flow Statistic Check Probe
- Mail Reflector Probe
- Message Retrieval Process State Probe
- Message Retrieval TCP Port Health Probe
- NRPC Routing Status Probe
- Router Process State Probe
- SMTP Process State Probe
- SMTP TCP Port Health Probe
- Transfer Queue Check Probe

These Domino domain monitoring probes offer several advantages over the pre-Domino 7 Event Generator probes, including:

- Easier configuration with Special Target Servers (see “Special Target Server groups” on page 57)
- Generation of Enhanced Events (see 2.6, “Events” on page 22)
- Availability of corrective actions to fix the root cause of Events

After the server has been upgraded to Domino 7, the design of Events4.nsf gets automatically upgraded to the latest design (Events4.ntf) on server startup.

**Note:** Since our External/Caribbean server was a member of a domain with other Domino 7 servers, the design of most of its system databases, such as names.nsf and Events4.nsf, were already up to date.

The server then runs a special check on the Events4.nsf database to detect replication conflicts or duplicate Event documents in Events4.nsf. This check is performed because any conflict or duplicate documents could cause problems with Domino domain monitoring. If any conflicts or duplicates are found, an Event report document is created in DDM.nsf, which by default provides a console command that can be used to resolve the problem. The console output for this check is shown in Example 3-6.

**Example 3-6   Events4.nsf upgrade checks**

09/16/2005 03:54:41 PM  From the ‘($Messages)’ view in ‘Events4.nsf’, 2 duplicate notes have been deleted and 0 replication conflict notes have been deleted.
09/16/2005 03:54:41 PM  Detection and removal of duplicate configuration documents in Events4.nsf is complete. 2 documents were deleted.
09/16/2005 03:55:59 PM  Opened TCP/IP connection from 9.33.85.112,1941 to 9.33.85.112,1352
09/16/2005 04:08:47 PM  Detection and removal of duplicate configuration documents in Events4.nsf is initiated ...
Additionally, there is a check performed to detect documents with duplicate PUIDs. Every probe document in DDM.nsf has an associated PUID (probe unique ID) and duplicate PUIDs could cause problems with Domino domain monitoring. If duplicate PUIDs are found, an Event report document with the text *X duplicate PUIDs found* is generated and the console command to resolve the duplicates is provided.

**Note:** As of Domino 7.01, these checks are performed shortly after server start-up *and* on a nightly basis.

### 3.3.1 Updating the server collection hierarchy

Now that the External/Caribbean server is upgraded to Domino 7, we want to include it in our server collection hierarchy.

We opened EVENTS4.NSF, selected the **Server Collection Hierarchy** entry, and then clicked the topmost Hub/Caribbean outline entry, which allowed us to add a new server to the existing collection hierarchy (see Figure 3-11 and Figure 3-12 on page 56).

**Figure 3-11**  Adding a new server to our existing server collection hierarchy
3.3.2 Creating and configuring Domino domain monitoring Messaging Probes

We opened the Monitoring Configuration database (Events4.nsf) and selected the DDM Configuration\DDM Probes\By Type view and enabled the following messaging probes, as shown in Figure 3-13 on page 57:

- Default Messaging/IMAP Message Retrieval Process State Probe
- Default Messaging/POP3 Message Retrieval Process State Probe
- Default Messaging/IMAP Message Retrieval TCP Port Health Probe
- Default Messaging/POP3 Message Retrieval TCP Port Health Probe
- Default Messaging/SMTP Process State Probe
- Default Messaging/SMTP TCP Port Health Probe

Note: The TCP Event Generator Probes previously configured in 3.2.1, “Monitoring TCP ports on pre-Domino 7 servers” on page 48 will continue to work against Domino 7 servers. However, for the purpose of this Redpaper, we have disabled them in favor of the new Domino domain monitoring Messaging Probes.
Figure 3-13  Domino domain monitoring messaging probes to enable

These default probes can simply be enabled from the view level without any additional configuration.

**Special Target Server groups**

The messaging probes run on Special Target Server groups by default, namely all SMTP, IMAP, and POP3 servers. This way, if we later add a new server to the domain, or change a server function from a mail server to an application server, we do not need to modify the probes to account for the change. The servers will automatically determine whether they still belong to any of these special target server groups and if not will automatically stop performing the probes.

**Note:** Each server determines at runtime whether it is part of a special target server group. For example, if a server has the IMAP task listed in the `SERVERTASKS=` line in its `NOTES.INI` file and either the IMAP or IMAP SSL port is enabled, this server gets classified as an IMAP server.

**Important:** You can also create new Domino domain monitoring probes by clicking the **New DDM Probe** view action button instead of just modifying the existing probe configurations.

These newly configured probes have more functionality and create “richer” Enhanced Events (see 2.6, “Events” on page 22). Not only do the probes now check on the protocol level via the various Port Health Probes, but they also check the actual server processes via the various Process State Probes. These probes create Enhanced Events and offer Probable cause/possible solution information as well as corrective actions.
3.3.3 Verifying operational status of the Domino domain monitoring probes

Unlike the TCP Server Event generators configured in 3.2.1, “Monitoring TCP ports on pre-Domino 7 servers” on page 48, which raise Events only if an error-condition actually occurs, certain Domino domain monitoring probes will automatically clear themselves when the problem is fixed. If this happens, the Event document will be updated to a Normal status. In other words, some Event documents will clear themselves if during a subsequent probe run the problem fixes itself. This functionality is not available for every Event document reported into DDM.nsf.

![DDM Image]

**Figure 3-14  Event document reports with Normal status**

As shown in Figure 3-14, the Event task created Event documents in DDM.NSF with the statuses set to Closed and the Event severities set to Normal, indicating that both TCP port checks and server process checks for the IMAP, POP3, and SMTP server tasks are operating normally after being reported as down.

**Attention:** In order to see closed Events, make sure you select the **Recent Events** or **All Events** tab in DDM.NSF.

After manually shutting down the IMAP task again, the respective Event report documents changed their status from Closed to Open and their severity from normal to Warning - High and appear now under the Open Events tab, as shown in Figure 3-15 on page 59.
Opening an Event report document (Figure 3-16 on page 60) shows the following Event information:

- The “Triggered by” entry provides a document link back to the DDM probe configuration document (in Events4.nsf). This allows us to quickly open the probe and modify it.
- The “Reported by” entry tells us the server task name reporting/raising the Event.
- The “Severity and Type” entry provides a doclink back to the Monitoring Configuration (Events4.nsf) document that contains the details about this Event. This allows us to quickly adjust the severity level or edit corrective action code.
- The “Probable Cause” and “Possible Solution” information provides detailed descriptions of the problem and suggested solutions on how to fix them.

**Note:** The prior normal and closed state of the Event has been moved to the Prior Events tab, making way for the updated Event status and severity on the Most Recent Event tab, as shown in Figure 3-17 on page 60.
### 3.4 Fixing Out of Office agents after a mail template upgrade

One implication of upgrading our mail server to a newer version could have been that the mail template included a newer version of the Out Of Office agent. The design task, that by default runs every night at 1 AM, will replace all instances of this agent in all mail files on the server. This design task run will reset the signature of the Out Of Office agents back to the standard Lotus Notes Template Development/Lotus Notes signature, removing any existing “Run on behalf of” user name and disable the agent.
The consequences of the Out Of Office being disabled are that currently enabled Out Of Office agents at the time of the server upgrade will stop sending out notifications.

Domino domain monitoring provides an Application Code probe that does not require any configuration on your part, is always running, and identifies several error conditions:

- Agent security errors
- Disabled agents for the design update task
- Agents that terminate abnormally when the maximum time to run is exceeded
- Full-text search operations performed on databases that are not full-text indexed

To show how to resolve this issue with Domino domain monitoring, we enabled the Out of Office agent in two mail files (mail\ccarter.nsf and mail\athomas.nsf) on the External/Caribbean server.

We then changed and upgraded the mail file design from the R6 template to the Domino 7 template by manually loading the design task, as shown in Example 3-7.

Example 3-7  Server console output showing the design task disabling previously enabled Out Of Office agents

```
> load design
09/22/2005 05:42:31 PM Database Designer started
09/22/2005 05:42:53 PM Updating 'OutOfOffice' into database 'Art Thomas' from template 'Mail (R7)'
09/22/2005 05:42:53 PM Designer: Agent 'OutOfOffice' in 'Art Thomas' disabled during Design Update from template 'Mail (R7)'. Agent signer 'External Admin/Caribbean'.
09/22/2005 05:42:53 PM Updating 'OutOfOffice' into database 'Chip Carter' from template 'Mail (R7)'
09/22/2005 05:42:53 PM Designer: Agent 'OutOfOffice' in 'Chip Carter' disabled during Design Update from template 'Mail (R7)'. Agent signer 'External Admin/Caribbean'.
09/22/2005 05:42:57 PM Database Designer shutdown
```

As expected, the application probe caught this condition and reported it as new Event status reports in DDM.NSF (Figure 3-18 on page 62 and Figure 3-19 on page 62).
Lotus Domino Domain Monitoring

Figure 3-18 DDM Application Probe results detecting the disabled Out of Office agents

Figure 3-19 Event report providing corrective actions

Clicking the **Choose Solution** button brings up the dialog box shown in Figure 3-20 on page 63.
3.5 Custom corrective action code

With Domino 7, it is also possible to write your own corrective action code (either via the LotusScript or @Formula language). This is a very useful feature that allows you to run more specialized scripts tailored to your own environment, or to add completely new corrective action code to Events that currently do not have any code yet.

**Tip:** Event documents are located in the “Advanced: Event Messages by Text” view in Events4.nsf. You can also click the doclink provided in the “Severity and type” line in any Event report document in DDM.nsf to open the associated Event document in Events4.nsf.

### 3.5.1 Events with default corrective actions

To see a list of Events that have associated corrective action code, we created a new view in Events4.nsf with this view selection formula:

```sql
SELECT ( @Trim(CorrectiveFormula) != "" | @Trim(CorrectiveScript) != "" )
```

Then we created three columns to show us the Event, and the type of code available (see Table 3-1).

**Table 3-1 Columns for custom view to list Events with corrective actions**

<table>
<thead>
<tr>
<th>Column name</th>
<th>Column formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event</td>
<td>OriginalText</td>
</tr>
<tr>
<td>Has Formula</td>
<td>@If(@Trim(CorrectiveFormula) != &quot;&quot;; &quot;yes&quot;; &quot;&quot;)</td>
</tr>
<tr>
<td>Has Script</td>
<td>@If(@Trim(CorrectiveScript) != &quot;&quot;; &quot;yes&quot;; &quot;&quot;)</td>
</tr>
</tbody>
</table>
Important: Not all Events in the Monitoring Configuration database (Advanced/Event Message by Text) have corrective action code available by default. In Domino 7, it is mostly the new Events that have specifically been created for Domino domain monitoring that have corrective actions by default. We found a total of 93 Event documents in Domino 7.0 that contain corrective action code.

To see the existing corrective action code, simply open any Event document from this custom view and select the Corrective Action(s) tab (see Figure 3-21).

![Server and Addin Task Event]

3.5.2 Writing your own custom corrective actions

In this section, we are going to write custom corrective action code for an existing Event that currently does not have a corrective action out of the box. The Event we have chosen is one that can occur in any environment: a Database is corrupt -- Cannot allocate space error, as shown in Figure 3-22 on page 65.
Figure 3-22 The database corruption Event in EVENTS4.NSF

When this Event is reported in DDM.NSF, it offers Probable Cause/Possible Solution text, but no automatic corrective action (Figure 3-23).

Figure 3-23 The database corruption Event as reported in DDM.NSF without corrective actions
Custom corrective action code can be provided as either @Formula code or LotusScript code and is stored as text inside the Event message document in EVENTS4.NSF (see Figure 3-24).

![Server and Addin Task Event](image)

**Figure 3-24 Fields to store corrective action code**

**Attention:** All default corrective actions are either implemented as @formula code or as LotusScript code. If you were to provide corrective action code in both languages, the LotusScript implementation will be chosen over the @formula implementation at runtime in DDM.NSF.

We are going to implement our corrective action code in LotusScript. Our solution includes sending a remote console command to the server (to fix a corrupt database), and only LotusScript offers this functionality.

**Restriction:** In Domino 7.0, it is not possible to write corrective action code in Java.

### 3.5.3 Keyword substitution in corrective actions

The key to writing good corrective actions is to utilize *keyword substitution*. Keyword substitution allows you to utilize any of the fields in an Event document, which provide the context to the actual problem being reported.

Keyword substitution allows you to use *any* of the field values from the Event report document from which your script is executed. This allows your code at runtime to refer to the current database name, user name, server name, domain name, and so on, stored in the Event report document.

To see the list of available fields you can refer to, highlight a DDM Event report document in any of the views in DDM.NSF and select **File → Document Properties...**, and then click the second tab in the dialog box (see Figure 3-25 on page 67).
As shown in Figure 3-25, the database file name is stored as a value in a field named TargetDatabase. Likewise, the server name is available in the field TargetServer.

Using keyword substitution, we can refer to the values in these fields by enclosing the field names with % signs in our code, such as %TargetDatabase% and %TargetServer% (see Example 3-8).

Example 3-8 Corrective action source code using keywords

```notes
Dim s As New NotesSession
Dim db As NotesDatabase
Set db = s.GetDatabase("%TargetServer%", "%TargetDatabase%", False)
```

Domino domain monitoring will dynamically substitute these placeholder field names with the actual values of these field names at runtime when you open the Event report document in DDM.NSF (see Example 3-9).

Example 3-9 The same corrective action code at runtime with expanded keywords

```notes
Dim s As New NotesSession
Dim db As NotesDatabase
Set db = s.GetDatabase("CN=External/O=Caribbean", "mail\gumz.nsf", False)
```

3.5.4 Corrective action code to fix a corrupt database

Our custom corrective action code for the database corruption Event (shown in Figure 3-23 on page 65) will:

- Check if the current user of the DDM.NSF database executing our corrective action code has enough administrative access to the remote server to send server console commands.
- Send a remote server console command to perform a full fix of the corrupt database.

The LotusScript code implementing this is shown in Example 3-10 on page 68.
Example 3-10  LotusScript code to check access and send a remote console command to fix a database

Dim session As New NotesSession
Dim sUser As String
Dim sCommand As String
Dim sServer As String
Dim vAccess As Variant

'check if the current DDM user has remote console access to the target server
sServer = "%TargetServer%"
sUser = session.UserName
vAccess = Evaluate("@ServerAccess([RemoteAccess]; "| + sUser + "; "| + sServer + ")")

If Cbool(vAccess(0)) = True Then
' send remote console command to perform a full fixup the target database
sCommand = "load fixup %TargetDatabase% -f -j"
Call session.SendConsoleCommand(sServer, sCommand)
Msgbox "Sent " + sCommand + " to " + sServer
Else
' not enough access
Msgbox "You are not authorized to send remote console commands.", MB_OK + MB_ICONSTOP
End If

Note: DDM.nsf executes corrective action code inline via the Execute() function for LotusScript code, and the @Eval() function for @formula code. All limitations and restrictions associated with Execute() and @Eval() apply to Domino domain monitoring corrective action code.

See the Lotus Domino Designer 7 Help database for more details:
http://www-12.lotus.com/ldd/doc/dominotoes/7.0/help7_designer.nsf

Next, we pasted our LotusScript code into the Corrective LScript text field on the Corrective Action(s) tab of the Event text document in EVENTS4.NSF and saved it (see Figure 3-26).
3.5.5 Testing our new custom corrective action code

Now that we have written and saved our corrective action code to the Event document, we are ready to test it to see that it works as designed.

In Domino 7, we can manually raise any Event at will by sending a TELL command to the EVENT server task. By manually raising Events, we do not have to wait until the error (in our example, the database corruption error) occurs again in a production environment.

To raise or fire an Event, we need to know its associated Event status code.

One way to obtain the Event status code for a given error or message is to find the Event in the Advanced\Event Messages by Text view in EVENTS4.NSF and open it. The Event status code is found on the Basics tab in the Value: field (see Figure 3-27).

![Figure 3-27 Finding the Event status code value](image)

In our specific case, the value is 0x0238. Now that we have the Event status code, we can tell the Event task to fire that particular Event for us via the following server console command:

tell Event fire 0x0238

The server console then shows the manually fired Event:

09/28/2005 05:38:52 PM  Database is corrupt -- Cannot allocate space
>

More importantly, a new Event report document has been created in DDM.nsf for this Event (see Figure 3-28 on page 70).

\[\text{Note: The Check Syntax button is only available for Corrective formula code. To perform a similar check with LotusScript, we wrote, compiled, and tested our LotusScript code as part of an agent or button in a separate database.}\]
Lotus Domino Domain Monitoring

One important thing to note is that when Domino domain monitoring expands your custom corrective action code at runtime, it checks to ensure all of the %fieldname% keywords exist. This check is performed when the Event report document is opened. If a referenced field cannot be found or is empty, keyword expansion will be aborted and the Choose Solution... button will be hidden for this particular Event. Hiding the Choose Solution... button prevents the user from running a script that will potentially cause runtime errors due to empty field values.

You can set `debug_ddm=1` in the notes.ini file of your Lotus Notes client, restart the client and open the Event report document in DDM.NSF again. With the debug option enabled, the bottom of the document will provide additional debug information, as seen in Figure 3-29 on page 71.
The tabs 1 to 10 each represent the respective corrective action formula or script code for up to 10 Events (the first Event being the Most Recent Event, and the second to tenth Events being the Prior Events).

![Debug information showing the expanded corrective action code and any related errors](image)

As indicated in Figure 3-29, the expansion of our corrective action has been aborted due to two referenced fields not being present in the current Event document.

To populate the missing fields, we created a simple formula agent in DDM.nsf that populates these fields (see Example 3-11).

**Example 3-11  Simple formula agent to populate fields**

```plaintext
@SetField("TargetServer": @Prompt([OkCancelEdit]; ""; "TargetServer:""); "")
@SetField("TargetDatabase": @Prompt([OkCancelEdit]; ""; "TargetDatabase:""); "")
```

After running the agent on the Event report document and setting the TargetServer field value to External/Caribbean and the TargetDatabase field value to mail@gumz.nsf, the Choose Solution... button finally shows up after re-opening the Event report document (see Figure 3-30 on page 72).
We are finally ready to test our corrective action code within the context of an Event report document in Domino domain monitoring.

**Important:** You must restart the Event task after making changes to Events4.nsf; otherwise, your custom corrective actions will not become immediately active.

As expected, after clicking the **Choose Solution...** button, our corrective action LotusScript code ran and sent a remote console command to the server (see Figure 3-31 on page 73 and Example 3-12 on page 73).
Chapter 3. A Domino domain monitoring scenario

3.6 Enabling your existing applications for Event reporting

In this section, we will build a Simple custom Event to show how you can easily enable your existing Lotus Notes or Domino applications with a few lines of additional LotusScript or Java code to raise and report Simple Events containing your own Event text, an Event type, and Event severity. These Events are genuine Domino Events and can be handled with Event handlers, appear in statrep.nsf, and to a certain extent in DDM.NSF.

To raise Events from within applications, you need to call the LogEvent() method of the NotesLog back-end class. The queue name must be EventDispatcher.

For more information about the LogEvent() method, visit the Lotus Domino Designer 7 Help at the following URL:

http://www-12.lotus.com/ldd/doc/domino_notes/7.0/help7_designer.nsf
The code in Example 3-13 will raise a custom Mail Event of severity Fatal (see Figure 3-32 and Figure 3-33 on page 75).

**Example 3-13  LotusScript code to raise our custom Event**

```
Dim session as NotesSession
Dim log as NotesLog
set session = New NotesSession
set log = session.CreateLog(""")
call log.LogEvent("A custom Event!", "EventDispatcher", EV_MAIL, SEV_FATAL)
```

**Figure 3-32  Our custom Event reported in STATREP.NSF**
You cannot specify your own Event status code. All Events generated via the LogEvent() method share the same Event status code 0x1AC6, which is defined as the standard Application Event in Events4.nsf (you can find the Event document in EVENTS4.NSF in the view Advanced/Event Messages by Text). Also, the LogEvent() method only works when the script code is running on the server. Also, it is not feasible to add custom corrective action code to the generic Application Event, since this Event is used for all Events raised by the LogEvent() method, and the custom corrective action code has no enhanced Event data, such as TargetServer, TargetDatabase, and so on, available, as it is a Simple Event.

Our example custom Event also shows up in DDM.NSF (see Figure 3-34 on page 76).
Figure 3-34   Our custom Event reported in DDM.NSF

**Restriction:** As of Domino 7.0, the Event gets categorized as a Server Event, regardless of the original Event type raised by our code. This issue is scheduled to be corrected in Domino 7.0.1.

### 3.7 Summary

This chapter provided a scenario of how we set up and configured our Redbooks lab Domino environment to tackle some common issues that happen in production environments. While we used a somewhat mail centric environment, all the items discussed can be applied to any type of Domino infrastructure.

In Chapter 4, "More Domino domain monitoring scenarios and tips" on page 77, you will read about some more brief scenarios that will help you realize the potential uses of Domino domain monitoring.
Chapter 3, “A Domino domain monitoring scenario” on page 43 took you through a complete sample scenario to solve some common issues. This chapter provides additional monitoring scenarios that utilize different probe types, including Replication, Security, Directory, and Web. We expose these additional probes to get you thinking about how to utilize Domino domain monitoring to its fullest.

We also provide some tips to help bring out all of the features of Domino domain monitoring. The final section lists console commands and debug information that can be used for testing Domino domain monitoring probes.
4.1 Additional monitoring scenarios

The scenarios covered in this section will focus on feature areas that are crucial to most, if not all, Domino infrastructures. We will cover the Replication, Security, Directory, and Web feature areas.

We chose these probes because they apply to most Domino domains, and we will be able to show you some unique ways in which to use them.

4.1.1 Security monitoring

The Security Probes included in the configuration database cover a wide range of security problems. Security probes validate database ACLs, report on the security of agents in a database, and analyze the security configuration of your domain. In this mini-scenario, we will focus on the security probes that analyze the configurations in your domain. Emphasis is placed on three basic documents in your Domino directory: Server Documents, Configuration Documents, and Person Documents. We will show you how to leverage the Security Best Practices, Review, and Configuration probes to do the grueling work of assessing the security configurations in your domain.

Perhaps there is a third-party vendor running a security audit against your company. You are the Domino Administrator, IT Architect, or IT Manager; or maybe you work for a small company and your responsibilities include all three of these job areas. Getting ready for a security audit can be a daunting process. Manually checking multiple security fields for each machine in your domain could take days, and your time is better spent on other responsibilities. To have an automated process do these security checks for you just makes sense, and since these probes check hundreds of settings at a time, this automated process tends to be much more thorough than manual verifying all of these settings are correct.

To start off, you can set up a Security Review Probe. This probe will generate a report of important security fields across as many servers as you specify. The configuration of this probe is simple. Similar to most other probe documents, the Basics tab allows you to fill in a probe description, which server will run the probe, and which servers will actually be probed (see Figure 4-1).

![Figure 4-1 Security probe basics](image-url)
On the Specifics Tab (see Figure 4-2), you will find all the different security areas that can be probed. If the audit you are preparing for is Web specific, for example, you can decide to only have Domino domain monitoring report on Web specific security fields. If the audit is more general, you can decide to probe all security areas for your Domino servers.

Once you run the Security Review Probe, a detailed security report for each requested server will be generated. Print this report for easy analysis against your auditor’s recommendations. If you need to tweak any security settings, use the Choose Solution button to easily access the configuration settings you need to change.

Now imagine that you have made the auditor’s requested changes to one of your servers. You now have a guideline server that has the security settings you desire and you would like to measure the other servers in your domain against this one. You can use the Security Configuration Probe to compare your guideline server to all the other servers in your domain.

The configuration for the Security Configuration Probe is exactly the same as the Security Review Probe, with one exception: on the Specifics Tab, you are able to specify a guideline server (see Figure 4-3 on page 80).
Lotus Domino Domain Monitoring

Figure 4-3   Security Configuration probe: guideline server

This probe will now do an analysis of all of the same configurations available to the Security Review Probe, but it will also take the analysis one step further by reporting on discrepancies between the guideline server and the target servers you specify. This could prove to be an invaluable tool as you are getting ready for your security audit. You can focus on configuring one server to the exact security specifications provided by the auditors, and then use this probe to see how far off the rest of your domain is. Figure 4-4 shows an example of an Event report generated by the Security Configuration Probe.

Figure 4-4   Security Configuration probe details

You will notice that links are provided back to each probed server document, and the analysis is broken down into sections: Administration Section, Programmability Restrictions Section, and so on. You may find that fields that hold name lists, such as the “Run Unrestricted methods and operations” field in this example, will many times not match between server documents. When this probe is run and the comparison is done between the guideline server document and the target server document, a thorough analysis of name list fields will also be done and missing names or additional names will be brought to your attention under the bolded headings “Additions in Probe Server list” or “Missing from Probe Server list”, as seen in Figure 4-4. Just like the Security Review Probe, a Choose Solution button is provided in the Event document so you can easily make changes to the security settings that are reported as discrepancies.

Another useful probe that relates to security auditing is the Security Best Practices Probe. It is one of the more popular probes and is really easy to set up and use. This probe will run its own audit in your domain, making comparisons against recommended security values, what we call the “best practices” security values. The configuration of this probe has all the same
options as the Security Review Probe and all the same specifics to check for as the Security Configuration Probe. You can run your own audit in your domain and allow Domino domain monitoring to be the auditor. If you have security configurations that are not set up according to best practices, they will be brought to your attention with Events in the DDM.nsf database. These Events will contain recommendations on how to fix these settings. An additional probing option of the Security Best Practices Probe is that you can check security settings for the Person Documents in your domain. Fields such as “Check Password” or “Password Change Interval” will be probed and the recommended settings of these fields will be shown to you. Domino domain monitoring can also alert you if any person document has a user.id file attached to it, since this not a best practice.

**Note:** severities were discussed in detail in 2.6.3, “Event severities” on page 23. You will see that the severity of a Security Best Practices Probe and the Security Configuration Probe is calculated on the fly while the probe is run. Depending on how much of the configuration does not match, a percentage security score is calculated. Security scores of 0 - 50% mean 50% or less of the configuration did not match, and this generates a severity of Warning (Low). Scores of 51% and above means over half of the configuration did not match and this will generate a Warning (High) Event.

By default, this probe is a scheduled probe, but it has the unique capability to run in real time. You will notice an option on the Schedule Tab of the probe configuration document under the heading Real Time (see Figure 4-5). By enabling this Real Time schedule option, you can catch Security holes being implemented in your domain as they occur. The way this works is if any security setting in a Server, Configuration, or Person document is changed, the Best Practices probe will automatically be kicked off and will do an analysis of the security of your domain immediately. This real time probing can be used to find and fix problem security settings right when they occur, so that vulnerabilities can be brought to your attention right when they are implemented, and not days later when your security has already been breached. This real time option is also available for the Security Configuration Probe.

![Figure 4-5  Security Best Practices probe: Real Time](image)

### 4.1.2 Directory monitoring

In order to effectively monitor your Domino servers for availability and performance, we recommend utilizing the Domino directory probes, which include four probes to monitor LDAP as well. The Domino directory is the heart of the Domino server. Many basic server functions require a Name Lookup to be performed on Domino directories, including Notes client user access, mail processing, authentication issues within Notes applications, as well as many others.

This section provides an example of how to configure and customize the Name Lookup Search Response probe. We conclude the section with a discussion on how the LDAP Process State can be used to monitor your LDAP servers. There are many Directory probes available, but we will focus on these high-visibility probes. You can enable additional probes to
generate more detailed information related to your Domino directories, including Directory
catalog and Directory Indexer process information. The full list of Directory probes is located
in Appendix A, “Domino domain monitoring probes” on page 95.

We configured the Name Lookup Search Response to be extremely sensitive so that we
could generate Events for demonstration purposes. As shown in Figure 4-6, we have enabled
the probe to alert you with a Failure severity Event if the Name Lookup search exceeds 1
milliseconds. While we have the lowest possible setting, we recommend you start with the
default Name Lookup response time thresholds.

Figure 4-6 Directory Name Lookup Search Response probe document

The corresponding Event generated in DDM.nsf is pictured in Figure 4-7.

Figure 4-7 Event generated from the Directory Name Lookup Search Response probe

This is a very simple example, so we will take it a step further. Your Domino directory has
exceeded the Name Lookup search threshold and the probe provides a default Possible
solution of “Provide additional directories to better balance requests.” Well, first of all, this
could be a one time occurrence. As an administrator or architect who knows the history of
the domain issues, you have to decide whether to pursue this particular issue further or close
the issue. Also, adding additional directories might not be feasible, since that might complicate
your environment unnecessarily and require additional maintenance.
We will close the Event, but not without customizing the Name Lookup Search Response probe first. The first goal is to reduce the sensitivity of the probe because a 1 millisecond Name Lookup response time on the Domino directory is not realistic. The second goal is to provide more probable causes and possible solutions that are applicable to our environment.

We click the **Triggered by** link, shown in Figure 4-7 on page 82, to access the Name Lookup Search Response probe. In the probe document, we simply update the Failure threshold value from 1 to 5 milliseconds and save the document. Next, while back in the Event document (Figure 4-7 on page 82), we click the **Severity and type** link to access the Server and Addin Task Event for the Event document we want to modify, “The Domino Directory search response (<number> ms) on <name> has reached its threshold (<number> ms).”

As mentioned in 2.7, “Customize DDM.nsf from Event documents” on page 25, be careful when making modifications to these documents. In general, you will leave all fields on the Basics tab with their default values. We are adding some additional Probable causes and Possible solutions, as shown in Figure 4-8.

Our updates to the probe and Server and Addin Task Event document are complete. We will now test the probe to see, first, if our Name Lookup slow response problem is still occurring with the new threshold and, second, if our updates to the Server and Addin Task Event document show up in the Domino domain monitoring Event document.

To test our updates, we follow the steps documented in “Testing your solution” on page 38. First, we issue the command in Example 4-1 on page 84 on Mail2, the server that has the problem.
Example 4-1 Mail2 tell Event dumpprobes console output

> tell Event dumpprobes
>
<table>
<thead>
<tr>
<th>ID</th>
<th>Probe Message Queue Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>-----</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>1</td>
<td>ReplicationEvents-Events-1</td>
</tr>
<tr>
<td>38166</td>
<td>Database-38166</td>
</tr>
<tr>
<td>38170</td>
<td>Database-38170</td>
</tr>
<tr>
<td>30346</td>
<td>Database-30346</td>
</tr>
<tr>
<td>30394</td>
<td>Messaging-30394</td>
</tr>
<tr>
<td>30282</td>
<td>Messaging-30282</td>
</tr>
<tr>
<td>30290</td>
<td>Messaging-30290</td>
</tr>
<tr>
<td>30286</td>
<td>Messaging-30286</td>
</tr>
<tr>
<td>30382</td>
<td>Directory-30382</td>
</tr>
</tbody>
</table>

We then run the 30382 Directory probe manually by issuing `tell Event runprobe 30382`. We will not see any console output stating the probe is running, so we need to verify the status of the probe run in DDM.nsf. The problem is still occurring, but the response time is getting better. As shown in Figure 4-9, the response time is at 5 milliseconds, so the new threshold is being hit exactly at 5 seconds.

![Figure 4-9](image.png)

Note that the Probable cause and Possible solution have been updated per the customization in the Server and Addin Task Event document.

After troubleshooting the problem, we have determined that the Name Lookup response delays have been caused by the indexing of names.nsf. To remedy the problem, we have adjusted the replication schedule in the appropriate connection document so that the Name
Lookup search delays do not occur during prime shift. This information is fully documented by the Mail2 Admin in the Event Change History section at the bottom of Figure 4-9 on page 84.

Now that we have resolved that issue, we will focus on monitoring the LDAP process. In this mini-scenario, we have an application on our External server that uses LDAP to authenticate users. User authentication is a critical component to any application. The authentication prompt is the first thing users see when hitting your application, so first, we want to ensure that the LDAP process is running, and second, ensure that the LDAP process is responding within its predefined threshold. To achieve these goals, we will use the LDAP Process State and LDAP Search Response probes, respectively.

The LDAP Process State probe configuration is very straightforward, as seen in Figure 4-10.

![Figure 4-10 Directory LDAP Process State probe configuration](image)

We will throw a little twist into this example. In the Caribbean domain we perform maintenance on the Domino servers from 12:00 AM to 7:00 AM. This is maintenance that is done every week, and the URL for the Web site even displays a message stating that the site is down for maintenance. We do not want to monitor LDAP during this time frame.

To schedule around this maintenance, we create two probes with different schedules, as seen in Figure 4-11. The first probe runs 24x7 Monday through Saturday, and the second runs 7:00 AM - 12:00 AM.

![Figure 4-11 Directory LDAP Process State probe schedules](image)
Now that we have the LDAP process monitored, we can also set up a probe to monitor the LDAP search response. Luckily, the configuration for this probe, Directory LDAP Search Response Probe, is identical to the Name Lookup Search Response, which we used above in our example that monitored the primary Domino directory (names.nsf).

One recommendation we have with the Directory LDAP Search Response probe is to set a low response time threshold value in order to get a base line of what the LDAP directory response time is during normal operation of your server, and specifically when customers can quickly authenticate with their applications on the server. When baselining, make a note of what the search response time is in milliseconds, and run the baseline for at least one full day with a normal user load hitting your applications. Once you have a valid baseline, you can set a threshold that is higher than the “normal” search response, and therefore optimize the probe for your server. This will also give you the information you need to correlate user complaints with LDAP response times in the future, modifying the probe as necessary. We discuss using probes for obtaining a baseline of statistics in Appendix A, “Domino domain monitoring probes” on page 95.

To summarize, we started out with a very Simple Event. As administrators or architects, we know some issues come and go and might not have a negative effect on service delivery. We have the ability to simply close these Events knowing that Domino domain monitoring will automatically open the Event if it occurs again, at which point we can reassess the Event severity.

As seen in this example, there are also ways to easily customize the probe and Event information so when the probe gets triggered again it will provide more meaningful information geared specifically to your Domino domain. This information, including the Event Change History information, enables administrators to immediately get up-to-speed on the current and historical issues, and take the appropriate steps to resolve them. We also customized the LDAP probe schedule to work around a weekly server maintenance window, and discussed using probes to get a baseline of statistics to better optimize your monitoring using Domino domain monitoring.

### 4.1.3 Web Configuration monitoring

Perhaps you are the Webmaster for your organization and you have configured your Domino server as a Web server and are facing complaints from users about Web server performance. The Web Best Practices Probe was created to validate the Web settings on your Domino server ensuring you have your Web server optimized for performance.

Much like the Security Best Practices probe, when there is a mismatch of recommended web settings, Event reports are generated. Severities are calculated during the probe run based on the percentage of mismatched settings.

As seen in Figure 4-12 on page 87, you can choose to only check your Web Server Performance Settings, and focus on how you can optimize these settings for your environment.
Chapter 4. More Domino domain monitoring scenarios and tips

Figure 4-12  Web Best Practices probe

A report generated by the Web Best Practices Probe is very detailed. Many different settings across three different documents in your domino domain will be probed: Internet Site Documents, Web Configuration Documents, and Server Documents. Figure 4-13 shows some of the details generated by a Web Best Practices Probe for a Server Document.

Table 1: Details of Web Best Practices probe

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Field Value</th>
<th>Recommendation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current Site configuration</td>
<td>No</td>
<td>Internet Site documents are recommended for Web Server configuration as of Domino 6 and higher.</td>
</tr>
<tr>
<td>Basic Web Server Configuration</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Who can - Access Server</td>
<td>All users can access this server</td>
<td>Currently, all users can access this server, you may only want to grant access to certain people.</td>
</tr>
<tr>
<td>Who can - NotAccess Server</td>
<td>EMPTY</td>
<td>Change this value to a specific name or group of names. Leaving this value open is a security risk.</td>
</tr>
<tr>
<td>Session authentication</td>
<td>Disabled</td>
<td>Enabling this field allows for more granularly defining configuration.</td>
</tr>
<tr>
<td>Web Server Performance</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maximum cached designs</td>
<td>125</td>
<td>This is the default value. If this value is tweaked, it might influence server performance.</td>
</tr>
</tbody>
</table>

Figure 4-13  Details of Web Best Practices probe

Figure 4-14 on page 88 shows another example of details, this time for a Web Configuration Document.
You will notice in the figures above that there is a section broken down for Web Server Performance, so you can quickly look at your performance settings and make decisions on how to tweak them. In the recommendation column for some of these settings, you will be alerted that you are running with default values. Default values are alright, but this probe is designed to make you aware that for some settings, you should decide how you can optimize them for your own environment. There will be differences based on machine speed, size, and other factors, so it is difficult for Domino domain monitoring to make general recommendations for each of these settings. The report generated into DDM.nsf simply brings some of these settings to your attention and then allows you to easily optimize them via the Choose Solution button.

The second flavor of Web Probes is the Web Configuration Probe, and it is similar in design to the Security Configuration Probe. Imagine that you have users complaining about the performance of one specific Web server, but you also have another Web server that is running smoothly. The Web Configuration Probe allows you to specify this good Web server as the guideline server for your domain. All other Web servers can be measured against this guideline server and discrepancies will be reported to DDM.nsf. This is handy especially if you have the same type of hardware running each of these Web servers. You will be able to easily assess which settings should be optimized.

In summary, the Web Best Practices and Configuration Probes are useful tools to help set up a new server in your domain, or to troubleshoot a problematic Web server. They were designed to save you time and to do some of the more grueling work and analysis on your behalf.

### 4.1.4 Replication monitoring

Monitoring replication in your Domino environment is important. Whether it is sales opportunities in a Notes database, notes in a mail file or Domino directory updates, the replication of data is crucial to the functioning of a Domino server. The Replication Errors and Scheduled Replication Checks probes are ideal to ensure replication occurs error free and on time. Our replication scenario will monitor replication (push only) of the names.nsf, admin4.nsf, and Events4.nsf from our Hub server to our spoke servers in our Domino lab domain, Caribbean. We can monitor replication of all of these databases, from Hub to the spoke servers with Domino domain monitoring Replication probes.

To ensure replication occurs on schedule, we created a Scheduled Replication Checks probe, as shown in Figure 4-15 on page 89. The probe ensures replication occurs between the Administration Server of the Domino Directory, which not surprisingly is our Hub server, to our spoke servers within eight hours. As seen in the bottom right of Figure 4-15 on page 89, we scheduled the probe to run daily, seven days a week at 5:00 AM. The reason we check for replication within the last eight hours at 5:00 AM is because we have a connection document to push replication of the databases from the hub to the spoke servers from 9:00 PM - 5:00
AM. With this probe, we will be alerted if the databases did not replicate during their normal scheduled times.

Figure 4-15  Scheduled Replication Checks probe document with Schedule tab

Now that we ensure replication occurs on time, what about replication errors? We will use a Replication Errors probe, as shown in Figure 4-16. The probe monitors any push replication errors from our Administration Server to our spoke servers. We also specify the three databases we want to monitor, or alternatively we could have specified all databases with an asterisk (*) and then list the databases you do not want to monitor.

Figure 4-16  Replication Errors probe document
These probes provide a complete solution to your replication monitoring needs. You can modify the severity levels of the probes to prioritize replication delays or errors. The ability to specify either push, pull, or both gives you the flexibility to monitor the flow of replication that you care about, whether it is for system databases or replication of high priority/availability end-user applications.

Now that we have probes set up, we will look at some example Event documents that were generated from Domino domain monitoring Replication probes.

**Attention:** The example Replication Events were not necessarily created by the exact probes described above. The point is to show some of the detailed diagnostic information available within the Domino domain monitoring Events.

In Figure 4-17, we have a delay in the pull of replication from Hub to Mail2 for Events4.nsf.

![Figure 4-17 Detailed Event information from a Scheduled Replication Checks probe](image)

**Note:** The Replication Diagnosis Report, shown above on the Details Tab, is available because we checked “Attempt to diagnose problems” in the Scheduled Replication Checks probe, as seen in Figure 4-15 on page 89, and is highly recommended. This report provides possible root cause information to assist with problem determination. This feature is optional for Scheduled Replication Checks probes, and is built into the Replication Errors probes.

In Figure 4-18, we have a replication failure because replication was temporarily disabled on the Mail2 replica of names.nsf.

![Figure 4-18 Detailed Event information from a Replication Errors probe](image)
## 4.2 Domino domain monitoring console commands and debugging information

There are some interesting commands you can run on your Domino domain monitoring Server’s console. You will find these console commands extremely helpful when you are troubleshooting problems or would like to test your solution. Some of these console commands have been mentioned in the scenarios above, but Table 4-1 consolidates them, including a quick description of what they do.

<table>
<thead>
<tr>
<th>Console command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>tell Event fire 0x????</code></td>
<td>Generates an Event in Domino domain monitoring with Value 0x????.</td>
</tr>
<tr>
<td><code>tell Event fire 0x???? 0x????</code></td>
<td>Generates two Events to Domino domain monitoring and connects them in the form 0x????:0x????, so that the first Event is the Root Cause, and the last Event is secondary.</td>
</tr>
<tr>
<td><code>tell Event dumpprobes</code></td>
<td>Will show a list of enabled DDM Probes on the Server Console by numerical ID and type.</td>
</tr>
<tr>
<td><code>tell Event runprobe ?????</code></td>
<td>Designed to be used with the <code>dumpprobes</code> command, this will run a probe with ID ?????? immediately.</td>
</tr>
</tbody>
</table>

**Note:** The Value for an Event can be found in Events4.nsf. If you look at the Event Messages by Text View, under the Advanced Views, you will find all the Events that can be fired off into DDM.nsf. By opening any one of these Events, you will see the Value on the Basic Tab, and you can substitute this hexadecimal value for the 0x???? string in the above table.

Example 4-2 displays another interesting console command that shows the DDM Probe Schedule for your server. The format of this command is `sh sch -ddm`. When executed on the server console, it will show a table of enabled probes and when the next run time for each scheduled probe is.

```bash
> sh sch -ddm
Scheduled Type Next schedule
Default Messaging/Router Proc Event Monitor 09/30/2005 12:31:03 PM
Default Messaging/SMTP Proces Event Monitor 09/30/2005 12:30:40 PM
Default Messaging/POP3 Messag Event Monitor 09/30/2005 12:30:40 PM
Default Messaging/IMAP Messag Event Monitor 09/30/2005 12:30:40 PM
Monitor the replication of na Event Monitor 10/01/2005 05:00:00 AM
Default Database/Scheduled Da Event Monitor 10/01/2005 04:30:00 AM
Default Web/Best Practices Pr Event Monitor 10/01/2005 02:00:00 AM
Default Security/Best Practic Event Monitor 09/30/2005 12:30:40 PM
Default Security/Configuratio Event Monitor 10/01/2005 04:00:00 AM
Default Messaging/SMTP TCP Po ISpy 09/30/2005 12:29:51 PM
Default Messaging/POP3 Messag ISpy 09/30/2005 12:29:51 PM
Default Messaging/IMAP Messag ISpy 09/30/2005 12:29:51 PM
```

This is very useful in giving you a birds eye view of the Domino domain monitoring schedule in your domain and may help in the planning of future DDM Scheduled Probes.
4.2.1 Debug notes.ini setting for Domino domain monitoring

The notes.ini setting DEBUG_EVENT=1 can be used to test Domino domain monitoring probes. This is a global debug variable for the Event task and includes information about probes schedules and information that can be used to test your Domino domain monitoring probes.

Remember, enabling additional debug logging can adversely affect server performance, so we do not recommend that you use this variable at all times. We recommend using DEBUG_EVENT=1 with another Notes.ini setting, DEBUG_OUTFILE = “filename”. DEBUG_OUTFILE enables console logging to ASCII (text) files that will contain debug information as well. More information about the DEBUG_OUTFILE notes.ini parameter can be found in the Lotus Domino Administrator 7 Help file.

4.3 Forwarding OS Events to Tivoli Enterprise Console (TEC)

Domino 7 is capable of reporting Domino domain monitoring operating system Events to Tivoli Enterprise Console® (TEC). This capability is possible because the new OS probes were implemented using the Tivoli Autonomic Management Engine (TAME). As of Domino 7.0, all other Domino domain monitoring probes do not use TAME, which is why other Events do not get forwarded to TEC. For more information about autonomic computing initiatives at IBM, visit the following URL:


Custom resource models can be developed to run under the Domino RMEval server task, although this is unsupported as of this writing.

Note: Resource models contain specific metrics, Events, thresholds, and parameters, which are used to determine the health of your resources along with specifications for corrective actions in the Event of failures.

The Basics tab in the Configuration Settings document in the Domino directory contains the Tivoli Enterprise Console settings, as seen at the bottom of Figure 4-19 on page 93.
The steps to configure the TEC settings in Figure 4-19 and install the Domino domain monitoring OS probe definitions on a TEC server are located in the Lotus Domino Administrator 7 help file.
Domino domain monitoring probes

This appendix lists all probes available with Domino 7.0 by probe type. Additional information is provided for the operating system probes, including the platforms supported for each probe subtype.
Probes by feature area

For the full details of each probe type, go to the Lotus Domino Administrator 7 Help at the following URL:

http://www-12.lotus.com/ldd/doc/domino_notes/7.0/help7_admin.nsf/Main?OpenFrameSet

Application probes

Table A-1 lists the application probes and their descriptions.

Table A-1  Application probes list with descriptions

<table>
<thead>
<tr>
<th>Application code probe name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application code -- Agents behind schedule</td>
<td>Identifies which agents have fallen behind schedule by the greatest amount of time for the current day. The minimum configuration for this probe is five minutes, which corresponds with the polling interval. This probe applies to agents run by the Agent Manager.</td>
</tr>
<tr>
<td>Application code -- Agents ranked by CPU usage</td>
<td>Generates a list of the 100 agents that use the greatest amounts of CPU for the current day. This probe has a relatively high overhead. When the probe is enabled, all agents are monitored. This probe applies to agents run by the Agent Manager and Web agents.</td>
</tr>
<tr>
<td>Application code -- Agents ranked by memory usage</td>
<td>Tracks memory use of LotusScript and Java agents. This probe applies to agents run by the Agent Manager and Web agents.</td>
</tr>
<tr>
<td>Application code -- Long running agents</td>
<td>Identifies potential run-away agents and generates a ranked list of agents that have run the longest. This probe applies to agents run by the Agent Manager and Web agents. The minimum configuration for Web agents is one minute; the minimum configuration for scheduled agents run by the Agent Manager is five minutes.</td>
</tr>
</tbody>
</table>

Attention: Information on how best to utilize the Application code probes to monitoring agent performance can be found at the following URL:


Database probes

Table A-2 lists the database probes and their descriptions.

Table A-2  Database probes list with descriptions

<table>
<thead>
<tr>
<th>Database probe name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database -- Compact</td>
<td>Reports errors that occur when a database is compacted.</td>
</tr>
<tr>
<td>Database -- Design</td>
<td>Reports errors that occur when the design of a database is updated.</td>
</tr>
</tbody>
</table>
Directory probes

Table A-3 lists the directory probes and their descriptions.

<table>
<thead>
<tr>
<th>Directory Probe name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database -- Error monitoring</td>
<td>Monitors key APIs in NSF and NIF, which are used to perform database operations and generates Events for errors that occur.</td>
</tr>
<tr>
<td>Database -- Scheduled Checks</td>
<td>Reports if the named database cannot be opened. The probe can also check for unused space in the database and for database inactivity.</td>
</tr>
<tr>
<td>Directory -- Directory Availability</td>
<td>Monitors the availability of all directories being served by the server and reports any errors. The probe also reports on namesake update issues and local LDAP server issues.</td>
</tr>
<tr>
<td>Directory -- Directory Catalog Aggregation Schedule</td>
<td>Based on the Directory Catalog task schedule, this probe subtype monitors missed aggregations and aggregations in progress longer than expected.</td>
</tr>
<tr>
<td>Directory -- Directory Catalog Creation</td>
<td>Monitors the Directory Catalog task process of creating directory catalogs by aggregating source directories into destination directory catalogs.</td>
</tr>
<tr>
<td>Directory -- Directory Indexer Process State</td>
<td>Monitors whether the Directory Indexer task is running.</td>
</tr>
<tr>
<td>Directory -- LDAP Process State</td>
<td>Monitors whether the LDAP process is running.</td>
</tr>
<tr>
<td>Directory -- LDAP Search Response</td>
<td>Monitors the server's average search response time for LDAP searches.</td>
</tr>
<tr>
<td>Directory -- LDAP TCP Port Health</td>
<td>Monitors whether LDAP is listening on its ports and responding promptly.</td>
</tr>
<tr>
<td>Directory -- LDAP View Update Algorithm</td>
<td>Monitors the LDAP view-update algorithm that is being used by the server.</td>
</tr>
<tr>
<td>Directory -- NAMELookup Search Response</td>
<td>Monitors the average search response time of all NAMELookups performed on the server.</td>
</tr>
<tr>
<td>Directory -- Secondary LDAP Search Response</td>
<td>Monitors the average search response time of searches of secondary LDAP servers that are performed on the server.</td>
</tr>
</tbody>
</table>
### Messaging probes

Table A-4 lists the messaging probes and their descriptions.

**Table A-4  Messaging probes list with descriptions**

<table>
<thead>
<tr>
<th>Messaging probe name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Messaging -- SMTP TCP Port Health</td>
<td>Verifies that SMTP services are successfully processing protocol requests.</td>
</tr>
<tr>
<td>Messaging -- Message retrieval TCP Port Health</td>
<td>Verifies that the IMAP and POP3 services are successfully processing protocol requests.</td>
</tr>
<tr>
<td>Messaging -- Route Process State</td>
<td>Monitors the status of the router.</td>
</tr>
<tr>
<td>Messaging -- SMTP Process State</td>
<td>Monitors the status of the SMTP process.</td>
</tr>
<tr>
<td>Messaging -- Message retrieval process state</td>
<td>Verifies the health of the IMAP and POP3 tasks that are configured to run on the specified servers.</td>
</tr>
<tr>
<td>Messaging -- NRPC Routing Status</td>
<td>Using NRPC, the probe sends a message to verify that mail routing is working.</td>
</tr>
<tr>
<td>Messaging -- Transfer Queue Check</td>
<td>Monitors mail flow to individual destinations.</td>
</tr>
<tr>
<td>Messaging -- Mail Flow Statistic check</td>
<td>Monitors the amount of mail on a Domino server and checks whether the number of mail messages in the MAIL.BOX file exceeds the number of mail messages that the router can process.</td>
</tr>
<tr>
<td>Messaging -- Mail Reflector</td>
<td>Tests mail routing to Domino and non-Domino mail systems.</td>
</tr>
<tr>
<td>Messaging -- Mail DSN</td>
<td>Uses the Delivery Status Notification (DSN) method to test the mail flow to an SMTP mail system. The destination domain must support the .dsn extension.</td>
</tr>
</tbody>
</table>

### Operating system probes

Table A-5 lists the operating system probes and their descriptions.

**Table A-5  Operating system probe list with descriptions**

<table>
<thead>
<tr>
<th>Operating system probe name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System -- CPU</td>
<td>Monitors CPU usage.</td>
</tr>
<tr>
<td>Operating System -- Disk</td>
<td>Monitors activity on all disks.</td>
</tr>
<tr>
<td>Operating System -- Memory</td>
<td>Monitors the amount of physical memory being used.</td>
</tr>
<tr>
<td>Operating System -- Network</td>
<td>Monitors the amount of network traffic on the server.</td>
</tr>
</tbody>
</table>
Appendix A. Domino domain monitoring probes

OS Probes by platform
There are a limited number of probe subtypes that are not available on certain platforms. Table A-6 outlines what probe subtypes are available on what platforms.

Table A-6   Operating system probes Subtype availability for each platform

<table>
<thead>
<tr>
<th></th>
<th>Windows®</th>
<th>Solaris™</th>
<th>AIX®</th>
<th>Linux/ zLinux</th>
<th>OS400</th>
<th>z/OS®</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Disk</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Not Available</td>
</tr>
<tr>
<td>Memory</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Not Available</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Network</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Not Available</td>
</tr>
</tbody>
</table>

Due to differences in the underlying architecture of the platforms listed in Table A-5 on page 98, there are also differences in the available configuration settings and the default threshold values for the probe subtypes. The Domino domain monitoring development team set default values after baseline testing to verify consistency of monitoring results across different platforms.

Tip: In order to get a baseline of the Operating Statistics for your Domino server(s), you can set extremely low thresholds for the OS probes. With these low thresholds, you can gauge what your servers statistics look like under normal operating conditions. With this information, you can set thresholds that will accurately tell you when your servers are exceeding their normal levels.

Replication probes
Table A-7 lists the replication probes and their descriptions.

Table A-7   Replication probes list with descriptions

<table>
<thead>
<tr>
<th>Replication probe name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Replication -- Errors</td>
<td>Monitors replication and reports replication errors to DDM.nsf.</td>
</tr>
<tr>
<td>Replication -- Replication Check</td>
<td>Monitors whether a specified database replicates within a specified time interval.</td>
</tr>
</tbody>
</table>
Security probes

Table A-8 lists the security probes and their descriptions.

<table>
<thead>
<tr>
<th>Security probe name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Security -- Best Practices</td>
<td>Compares a set of baseline security configuration settings to the same settings in a domain. This probe is a &quot;Best Practices&quot; security audit of the domain.</td>
</tr>
<tr>
<td></td>
<td>Note: To create your own Best Practices probe, modify the security configuration settings on the Specifics tab.</td>
</tr>
<tr>
<td>Security -- Database ACL</td>
<td>Monitors the access control privileges that groups and individuals have in specified databases on the server running the probe. You designate the acceptable access levels on the Specifics tab.</td>
</tr>
<tr>
<td>Security -- Database Review</td>
<td>Reviews the security properties for a specified database and generates a report on the probe findings.</td>
</tr>
<tr>
<td>Security -- Review</td>
<td>Generates a report on the security settings specified in the Specifics tab of the Probe document. You have the option of selecting the Directory Profile Note and the Security settings in my configuration document options if you want the settings in those documents reviewed by the probe.</td>
</tr>
</tbody>
</table>

Server probes

Table A-9 lists the server probes and their descriptions.

<table>
<thead>
<tr>
<th>Server Probe name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server -- Administration</td>
<td>Reports any administration process requests that fail to run when the request is scheduled to do so.</td>
</tr>
</tbody>
</table>
Web probes

Table A-10 lists the Web probes and their descriptions.

Table A-10  Web probes list with description

<table>
<thead>
<tr>
<th>Web probe name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web -- Best Practices</td>
<td>Reviews Web server configuration values on specified servers against a set of predefined values.</td>
</tr>
<tr>
<td>Web -- Configuration</td>
<td>Performs a comparison of Web server configuration values on specified servers against the same values for a known good server or guideline server.</td>
</tr>
</tbody>
</table>
Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this Redpaper.

IBM Redbooks

For information about ordering these publications, see “How to get IBM Redbooks” on page 103. Note that some of the documents referenced here may be available in softcopy only.

- *Upgrading to Lotus Notes and Domino 6*, SG24-6889

Online resources

These Web sites and URLs are also relevant as further information sources:

- IBM developerWorks : Autonomic Computing Toolkit overview
  

  
  [http://www-10.lotus.com/ldd/notesua.nsf/ddaf2e7f76d2cfbf8525674b00508d2b/de2da7b609c6cb038525703b005e2f5970openDocument](http://www-10.lotus.com/ldd/notesua.nsf/ddaf2e7f76d2cfbf8525674b00508d2b/de2da7b609c6cb038525703b005e2f5970openDocument)

- IBM Lotus software: Lotus Documentation: Lotus Domino Designer 7 (English)
  

- Lotus Domino 7 Administrator Help
  

- Lotus Domino Designer 7 Help
  

  

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](http://ibm.com/redbooks)
Help from IBM

IBM Support and downloads
ibm.com/support

IBM Global Services
ibm.com/services
Lotus Domino Domain Monitoring

Introduction to the powerful new Domino 7 features

Probes, corrective actions, and collection hierarchies

Examples of monitoring scenarios with tips and techniques

The purpose of this IBM Redpaper is to show system administrators and IT architects how to use and leverage the capabilities of the new Lotus® Domino® domain monitoring feature of IBM® Lotus Domino Version 7. The Lotus Domino domain monitoring feature provides customizable monitoring capabilities and fast recognition and reporting of critical server issues by delivering a single, feature-oriented view through which administrators can see the status of multiple servers across a domain.

By reading this Redpaper, you will gain intimate knowledge of Domino domain monitoring and see examples of its true power, such as autonomic corrective actions and more.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information: ibm.com/redbooks