

macromedia®

Using ClusterCATS



Trademarks

Afterburner, AppletAce, Attain, Attain Enterprise Learning System, Attain Essentials, Attain Objects for Dreamweaver, Authorware, Authorware Attain, Authorware Interactive Studio, Authorware Star, Authorware Synergy, Backstage, Backstage Designer, Backstage Desktop Studio, Backstage Enterprise Studio, Backstage Internet Studio, ColdFusion, Design in Motion, Director, Director Multimedia Studio, Doc Around the Clock, Dreamweaver, Dreamweaver Attain, Drumbeat, Drumbeat 2000, Extreme 3D, Fireworks, Flash, Fontographer, FreeHand, FreeHand Graphics Studio, Generator, Generator Developer's Studio, Generator Dynamic Graphics Server, JRun, Knowledge Objects, Knowledge Stream, Knowledge Track, Lingo, Live Effects, Macromedia, Macromedia M Logo & Design, Macromedia Flash, Macromedia Xres, Macromind, Macromind Action, MAGIC, Mediamaker, Object Authoring, Power Applets, Priority Access, Roundtrip HTML, Scriptlets, SoundEdit, ShockRave, Shockmachine, Shockwave, Shockwave Remote, Shockwave Internet Studio, Showcase, Tools to Power Your Ideas, Universal Media, Virtuoso, Web Design 101, Whirlwind and Xtra are trademarks of Macromedia, Inc. and may be registered in the United States or in other jurisdictions including internationally. Other product names, logos, designs, titles, words or phrases mentioned within this publication may be trademarks, servicemarks, or tradenames of Macromedia, Inc. or other entities and may be registered in certain jurisdictions including internationally.

This product includes code licensed from RSA Data Security.

This guide contains links to third-party websites that are not under the control of Macromedia, and Macromedia is not responsible for the content on any linked site. If you access a third-party website mentioned in this guide, then you do so at your own risk. Macromedia provides these links only as a convenience, and the inclusion of the link does not imply that Macromedia endorses or accepts any responsibility for the content on those third-party sites.

Apple Disclaimer

APPLE COMPUTER, INC. MAKES NO WARRANTIES, EITHER EXPRESS OR IMPLIED, REGARDING THE ENCLOSED COMPUTER SOFTWARE PACKAGE, ITS MERCHANTABILITY OR ITS FITNESS FOR ANY PARTICULAR PURPOSE. THE EXCLUSION OF IMPLIED WARRANTIES IS NOT PERMITTED BY SOME STATES. THE ABOVE EXCLUSION MAY NOT APPLY TO YOU. THIS WARRANTY PROVIDES YOU WITH SPECIFIC LEGAL RIGHTS. THERE MAY BE OTHER RIGHTS THAT YOU MAY HAVE WHICH VARY FROM STATE TO STATE.

Copyright © 1999–2002 Macromedia, Inc. All rights reserved. This manual may not be copied, photocopied, reproduced, translated, or converted to any electronic or machine-readable form in whole or in part without prior written approval of Macromedia, Inc.
Part Number ZCL2M100

Acknowledgments

Project Management: Stephen M. Gilson

Writing: Stephen M. Gilson

Editing: Linda Adler and Noreen Maher

First Edition: May 2002

Macromedia, Inc.
600 Townsend St.
San Francisco, CA 94103

CONTENTS

ABOUT THIS BOOK VII

Developer resources	viii
About Macromedia documentation	ix
Viewing online documentation.	ix
Contacting Macromedia	x

CHAPTER 1 Before You Begin 1

ClusterCATS overview.	2
ClusterCATS capabilities	2
Detailed overview.	3
ClusterCATS product configurations	5
ClusterCATS components	6
System requirements	7
ClusterCATS Server system requirements.	7
ClusterCATS Explorer and Web Explorer system requirements	8

CHAPTER 2 Scalability and Availability Overview. 9

What is scalability?	10
Performance.	10
Load management	12
Successful scalability implementations	13
Designing and coding scalable applications.	13
Avoiding common bottlenecks	16
DNS effects on website performance and availability	17
Load testing your web applications.	20
What is website availability?	23
Availability and reliability.	23
Common failures	24
Website availability scenario	25
Failover considerations	25
Creating scalable and highly available sites	28
What is clustering?	28
Hardware-based clustering solutions.	29
Software-based clustering solutions.	30
Combining hardware and software clustering solutions	32

CHAPTER 3 Installing ClusterCATS	33
Before you install	34
Upgrading from a previous version of ClusterCATS	34
Configuring DNS servers	34
Configuring server failover	38
Using ClusterCATS dynamic IP addressing	38
Configuring firewalls	38
Analyzing web server content	39
Considering domain controllers (Windows NT only)	40
Installing ClusterCATS	41
Installing ClusterCATS on Windows	41
Installing ClusterCATS on UNIX	42
After you install	45
 CHAPTER 4 Configuring Clusters	 47
Introduction to ClusterCATS Administration	48
ClusterCATS Server	48
ClusterCATS Explorer (Windows only)	48
ClusterCATS Web Explorer (UNIX only)	49
ClusterCATS Server Administrator	52
btadmin	53
Creating clusters	54
Creating clusters in Windows	54
Creating clusters in UNIX	60
Removing clusters	62
Adding cluster members	63
Adding cluster members in Windows	63
Adding cluster members in UNIX	64
Removing cluster members	65
Removing cluster members in Windows	65
Removing cluster members in UNIX	65
Server load thresholds	66
Configuring load thresholds in Windows	66
Configuring load thresholds on UNIX	69
Session-aware load balancing	72
Enabling session-aware load balancing on Windows	72
Enabling session-aware load balancing on UNIX	72
Persistent session failover in JRun	74
Session swapping overview	74
Configuring JRun for session swapping	74
Configuring ClusterCATS for session swapping	74
Using shared files for session swapping	75
Using JDBC for session swapping	76
Using ColdFusion probes	77
Configuring ColdFusion probes in Windows	77
Configuring ColdFusion probes in UNIX	81
Using JRun probes	84
Configuring JRun probes in Windows	84
Configuring JRun probes in UNIX	88

Load-balancing devices	92
Using Cisco LocalDirector	92
Using third-party load-balancing devices	95
Administrator alarm notifications	98
Configuring administrator alarm notifications on Windows	98
Configuring administrator alarm notifications on UNIX	99
Administrator e-mail options	100
Configuring administration e-mail options on Windows	100
Configuring administration e-mail options on UNIX	101
Administering security	103
Configuring authentication on Windows	103
Configuring authentication on UNIX	106
CHAPTER 5 Maintaining Cluster Members	109
Understanding ClusterCATS server modes	110
Changing active/passive settings	111
Changing active/passive settings in Windows	111
Changing active/passive settings in UNIX	112
Changing restricted/unrestricted settings	113
Restricting/unrestricting servers in Windows	113
Restricting/unrestricting servers in UNIX	114
Using maintenance mode (Windows only)	115
Updating a cluster member (Windows only)	118
Resetting cluster members	120
Resetting cluster members on Windows	120
Resetting cluster members on UNIX	120
CHAPTER 6 ClusterCATS Utilities	121
Using btadmin	122
Using btadmin on Windows	122
Using btadmin on UNIX	122
Using bt-start-server and bt-stop-server (UNIX only)	125
Using btcfgchk	126
Syntax	126
Sample output	126
btcfgchk DNS errors	126
Using hostinfo	129
Syntax	129
Sample output	129
Using sniff	130
Syntax	130
Sample output	130

CHAPTER 7 Optimizing ClusterCATS131

- ClusterCATS dynamic IP addressing (Windows only)132
 - Understanding static and dynamic IP address configurations132
 - Benefits of ClusterCATS dynamic IP addressing133
 - Setting up maintenance IP addresses.133
 - Enabling ClusterCATS dynamic IP addressing.135
- Using server failover.137
 - Static versus ClusterCATS dynamic IP addressing137
 - Windows domain controllers137
- Configuring load-balancing metrics138
 - Overview of metrics138
 - Load types139
 - Output variables.....139
 - Troubleshooting the load-balancing metrics.....140

INDEX 141

ABOUT THIS BOOK

Using ClusterCATS describes how to use ClusterCATS, the clustering technology that provides load-balancing and failover services to assure high availability for your web servers.

Contents

- [Developer resources](#) viii
- [About Macromedia documentation](#) ix
- [Contacting Macromedia](#) x

Developer resources

Macromedia, Inc. is committed to setting the standard for customer support in developer education, documentation, technical support, and professional services. The Macromedia website is designed to give you quick access to the entire range of online resources. The following table shows the locations of these resources.

Resource	Description	URL
Macromedia website	General information about Macromedia products and services	http://www.macromedia.com
Information on ColdFusion	Detailed product information on ColdFusion and related topics	http://www.macromedia.com/coldfusion
Macromedia ColdFusion Support Center	Professional support programs that Macromedia offers	http://www.macromedia.com/support/coldfusion
ColdFusion Online Forums	Access to experienced ColdFusion developers through participation in the Online Forums, where you can post messages and read replies on many subjects relating to ColdFusion	http://webforums.macromedia.com/coldfusion/
Information on JRun	Detailed product information on JRun and related topics.	http://www.macromedia.com/products/jrun/
JRun Support Center	Professional support programs that Macromedia offers.	JRun Support Center http://www.macromedia.com/support/jrun
JRun Online Forums	Access to experienced JRun developers through participation in the Macromedia Online Forums, where you can post messages and read replies on many subjects relating to JRun.	http://webforums.macromedia.com/jrun
Installation Support	Support for installation-related issues for all Macromedia products	http://www.macromedia.com/support/email/isupport
Training	Information about classes, on-site training, and online courses offered by Macromedia	http://www.macromedia.com/support/training
ColdFusion Developer Resources	All the resources that you need to stay on the cutting edge of ColdFusion development, including online discussion groups, Knowledge Base, technical papers, and more	http://www.macromedia.com/desdev/developer/
ColdFusion Reference Desk	Development tips, articles, documentation, and white papers	http://www.macromedia.com/v1/developer/TechnologyReference/index.cfm

Resource	Description	URL
JRun Developer Resources	All of the resources that you need to stay on the cutting edge of JRun development, including online discussion groups, Component Exchange, Resource Library, technical papers, and more.	http://www.macromedia.com/desdev/developer/
Macromedia Alliance	Connection with the growing network of solution providers, application developers, resellers, and hosting services creating solutions with ColdFusion	http://www.macromedia.com/partners/

About Macromedia documentation

Macromedia documentation is designed to provide support for the complete spectrum of participants. The print and online versions are organized to let you quickly locate the information that you need. The Macromedia online documentation is provided in HTML and Adobe Acrobat formats.

Viewing online documentation

All Macromedia documentation is available online in HTML and Adobe Acrobat Portable Document Format (PDF) files.

The PDF files are included on the product CDs and are installed in the docs directory, although they are an optional part of the installation.

Contacting Macromedia

Corporate
headquarters

Macromedia, Inc.
600 Townsend Street
San Francisco, CA 94103
Tel: 415.252.2000
Fax: 415.626.0554
Web: <http://www.macromedia.com>

Technical support

Macromedia offers a range of telephone and web-based support options. Go to <http://www.macromedia.com/support> for a complete description of technical support services.

Sales

Toll Free: 888.939.2545
Tel: 617.219.2100
Fax: 617.219.2101
E-mail: sales@macromedia.com
Web: <http://www.macromedia.com/store>

CHAPTER 1

Before You Begin

ClusterCATS is a web server clustering technology that provides load-balancing and failover services that assure high availability for your web servers. ClusterCATS lets you cluster distributed servers into a single, high-performance, highly available environment of web server resources.

A cluster consists of two or more web servers located on a LAN or across a WAN. Web servers included in a cluster operate as a single entity to provide rapid and reliable access to resources on those web servers. A cluster can help your website avoid the consequence of busy and failed servers — slow networks. With ClusterCATS you can avoid bandwidth, latency, and congestion problems.

Contents

- [ClusterCATS overview..... 2](#)
- [ClusterCATS components 6](#)
- [System requirements..... 7](#)

ClusterCATS overview

The ClusterCATS technology provides robust features for website availability, load balancing, and failing-over servers.

A website is no longer just a web server. Most websites have moved beyond static HTML pages on a web server. To generate dynamic content or process transactions, a website now includes multiple resources — web servers, files, applications, databases, and other software processes on multiple servers in one or more locations. The move to a more advanced site, consisting of multiple resources, often from multiple vendors, introduces a major problem — overall site availability and performance. More resources, especially software resources, and more links between them exponentially increase the probability of failure. Creating a fast, reliable website becomes substantially more challenging.

Macromedia created ClusterCATS, a complete website resource management solution, to enable service level agreements offering 24x7 availability and optimal response time for e-commerce, customer self-service, sales automation, customer support, and other critical business functions. With ClusterCATS, you can build and manage advanced websites, consisting of multiple resources spanning multiple servers, in one or more locations.

ClusterCATS builds and manages clusters. A cluster is a group of website resources, including web servers, files, applications, databases, and even the network, that act in unison, providing reliable and rapid user access. These resources can be clustered in a single building, distributed in a local area network (LAN), or distributed in a wide-area network (WAN) in multiple locations across the world. A cluster intelligently detects and transparently shields users from the following critical problems:

- Failed and busy servers
- Failed and busy applications and databases
- Slow networks caused by congestion, latency, and bandwidth problems

ClusterCATS capabilities

ClusterCATS delivers critical capabilities required by advanced websites today. These capabilities produce important benefits in the areas of website performance, availability, manageability, and scalability.

User response time is accelerated with ClusterCATS application and server load management.

ClusterCATS consists of server and client components. The ClusterCATS Server runs as a Windows service and ISAPI filter, NSAPI plug-in, or Apache module. ClusterCATS Explorer is the client-based management application used to build and manage clusters. Operating in conjunction with an administrative agent on each ClusterCATS Server, ClusterCATS Explorer provides all the required tools for centrally managing one or more clusters from any location.

You can also configure ClusterCATS software to enhance simple web server load-balancing products, such as Cisco's special-purpose LocalDirector hardware device.

The following table introduces the ClusterCATS capabilities:

Feature	Description
Application and server load management	Allows administrators to configure server load thresholds to provide optimum user response time in JRun/ColdFusion applications. ClusterCATS Server load management protects users from overloaded servers.
Server failover	Provides seamless failover of a web server because of a hardware, software, or network connection to another member in the cluster. ClusterCATS shields users from unplanned or planned server failures.
Session state management	Allows session state to be maintained across your website using a unique method that eliminates the source IP address server overload problems caused by proxy users. ClusterCATS application state management ensures users are not redirected away from a server while maintaining state.
Application monitor	You can configure ClusterCATS to monitor the JRun/ColdFusion server or a JRun/ColdFusion application, and restart the server or application if a failure occurs.
Distributed operations	Exploits a distributed operations model, eliminating traffic bottlenecks and maximizing performance and availability. All servers share knowledge of application or web server performance, and server availability. Each server can respond directly to a request or redirect a request to a faster server.
Centralized management	Provides a central console to manage and configure all web servers in your cluster. ClusterCATS Explorer provides both high-level and detailed views of the status of one or more websites and all the resources within a website.

Detailed overview

Application and server load management

ClusterCATS improves user response time by managing application load and web server load across multiple servers.

You establish load management policy through two administrator-defined response time thresholds. You configure these for each server. One threshold sets the level at which load management is activated. If this level of activity is reached, ClusterCATS gradually redirects a percentage of new server requests to the least-loaded server.

The other threshold defines the peak, or maximum, load level. This is defined as the load level that should not be reached on that server. If this threshold is attained, an alarm is sent and requests is redirected.

Session state management and failover

For some applications, it is important that a user session is completed on one server. ClusterCATS offers a session state management option that ensures that the same web server services requests from a user. When enabled, this option sends the user to the best-performing server. The user session then remains on that server until completion.

ClusterCATS defines a new session for the following:

- A user comes from a different domain
- A user enters a new URL
- A user employs a bookmark

This approach has distinct advantages over other methods, such as using a source IP address to define a user session. The ClusterCATS definition of a session is particularly beneficial if many visitors come from a large proxy server (for example, America Online). In that scenario, web servers could easily become overloaded.

Should user state be lost completely due to a resource failure, ClusterCATS provides graceful state failover. This capability automatically displays an administrator-defined URL for a custom HTML page or JRun page upon resource failure. This page can be designed to apologize for the failure and, if replicated resources are available, direct the user to restart the application at the beginning via a specific URL.

Distributed operations

ClusterCATS uses a distributed operations model, eliminating traffic bottlenecks and maximizing performance. While other hardware and software load-balancing solutions force all user requests and, typically, all responses through a single special-purpose network device or server, each ClusterCATS Server can receive a request, respond to a request, manage traffic load, and support failover. Unlike hardware load-balancing solutions, ClusterCATS performance is not throttled by network media limitations and ClusterCATS is network media independent. Consequently, performance scales linearly as servers and resources are added.

Centralized management

ClusterCATS Explorer, operating in conjunction with an administrative agent on each ClusterCATS Server, provides all the required tools for building and managing a website from any location, whether it be an operations center, hotel, or home.

ClusterCATS Explorer features a familiar Windows Explorer-like user interface and provides both detailed and high-level status views of one or more websites and all resources within a website.

ClusterCATS Explorer views include:

- Simplifies user interface for the configuration tasks of building a website, including adding and removing resources, setting load thresholds, selecting alarms, designating administrators, configuring replication, and state management capabilities
- Real-time graphs of the actual application or HTTP server load and load thresholds

ClusterCATS product configurations

ClusterCATS includes a comprehensive core set of features and offers several add-on options for extending its capabilities. All ClusterCATS configurations include:

- Macromedia Enterprise Server (ColdFusion and JRun) load manager
- Configurable load thresholds
- Real-time load monitor
- Session state management (server level)
- HTTP server monitor and auto-restart
- Real-time web server availability monitor
- Web server failover option
- Web server restriction
- Macromedia Enterprise Server monitor and auto-restart
- Macromedia Enterprise Server application monitor and auto-restart
- Administrator authentication
- Alarms
- Daily reports

ClusterCATS components

ClusterCATS consists of these primary components:

- **Server** Resides on each computer in a cluster. It communicates with the web server and other ClusterCATS Servers. For more information, see [“ClusterCATS Server” on page 48](#).
- **Server Administrator** (Windows only) or btadmin Lets you perform server-specific administration tasks through a graphical interface. For UNIX-based administration, use the scriptable btadmin utility, which is also available for Windows users. For more information, see [“ClusterCATS Server Administrator” on page 52](#) and [“Using btadmin” on page 122](#).
- **ClusterCATS Explorer and Web Explorer** Graphical utilities for creating and managing clusters in Windows and UNIX environments, respectively. For more information, see [“ClusterCATS Explorer \(Windows only\)” on page 48](#) and [“ClusterCATS Web Explorer \(UNIX only\)” on page 49](#).

The following table shows which components ClusterCATS installs on each platform:

Windows Installation	UNIX Installation
ClusterCATS Server	ClusterCATS Server
btadmin and ClusterCATS Server Administrator	btadmin (Note: You can administer a UNIX cluster with the ClusterCATS Server Administrator from a Windows computer outside the cluster.)
ClusterCATS Explorer	ClusterCATS Web Explorer (Note: You can access this from a Windows or UNIX computer.)

You must run the installation program on each server that will be part of your cluster and on the Windows computer (NT, 2000, .NET Server, 98, or 95) from which you will use ClusterCATS Explorer to administer the cluster. Even if your clusters run on Solaris or Linux platforms, obtain a Windows computer for running ClusterCATS Explorer. If you cannot, use the ClusterCATS Web Explorer in conjunction with the included server utilities to administer your clusters.

System requirements

This section describes the platforms on which the ClusterCATS components run and their minimum system requirements.

ClusterCATS Server system requirements

You must install the ClusterCATS Server component on each server in your cluster. Ensure that your server meets the minimum system requirements for your platform.

Windows system requirements for ClusterCATS Server

- Intel Pentium 200 Mhz or greater CPU
- 100 MB of free disk space
- 128 MB of RAM
- Windows NT (with SP 4 or greater), Windows 2000, or Windows .NET Server
- Internet Information Server or greater; Netscape Enterprise Server v3.5.1 or greater
- Administrative privileges on each server
- A unique IP address assigned to each web server
- Correct DNS entries and configurations (see [“Configuring DNS servers” on page 34](#))

Note: ClusterCATS Server does not run on Windows 98 or Windows 95.

Sun Solaris system requirements for ClusterCATS Server

- Sun SPARC workstation
- 100 MB of free disk space
- 128 MB of RAM (more recommended)
- Solaris operating system v2.5.1 or greater with Patch 103582-18 or higher
- Netscape Enterprise Server v3.5.1 or greater or Apache Web Server v1.3.6 or greater
- Administrative root privileges on each server
- A unique IP address assigned to each web server
- Correct DNS entries and configurations (see [“Configuring DNS servers” on page 34](#))

Linux system requirements for ClusterCATS Server

- Intel Pentium 200 Mhz or greater
- 100 MB of free disk space
- 128 MB of RAM (more recommended)
- Red Hat operating system v6.0 or greater
- Apache Web Server v1.3.6 or greater
- Administrative root privileges on each server
- A unique IP address assigned to each web server
- Correct DNS entries and configurations (see [“Configuring DNS servers” on page 34](#))

ClusterCATS Explorer and Web Explorer system requirements

You can install the ClusterCATS Explorer or Web Explorer component on a computer outside the cluster, so you can administer the cluster from a central location. Ensure the computer on which you install one of these components meets the minimum system requirements.

System requirements for the Windows-based Explorer

The Windows-based ClusterCATS Explorer runs from a Windows computer (NT, 2000, .NET Server, 98, or 95), regardless of the platform on which you install ClusterCATS Server. Its system requirements are as follows:

- Intel Pentium 200 Mhz or greater CPU
- 100 MB of free disk space
- 64 MB of RAM (128 MB recommended)
- Windows NT Service Pack 5 or greater (if running Windows NT)
- Administrative privileges

System requirements for the ClusterCATS Web Explorer

Use the ClusterCATS Web Explorer if you have a UNIX-only environment. Install the ClusterCATS Web Explorer program on a UNIX server that meets the following requirements:

- Sun SPARC workstation
- 75 MB of free disk space
- 128 MB of RAM
- Solaris operating system v2.51 or greater with Patch 103582-18 or higher
- Netscape Enterprise Server v3.5.1 or greater or Apache Web Server v1.3.6 or greater
- Microsoft Internet Explorer 4.0 or greater or Netscape Navigator 3.0 or greater

CHAPTER 2

Scalability and Availability Overview

This chapter describes the concepts involved in achieving scalable and highly available web applications.

Contents

- [What is scalability? 10](#)
- [Successful scalability implementations 13](#)
- [What is website availability?..... 23](#)
- [Creating scalable and highly available sites..... 28](#)

What is scalability?

As an administrator, you probably hear about the importance of having web servers that scale well. But what exactly is scalability? Simply, scalability is a web server's ability to maintain a site's availability, reliability, and performance as the amount of simultaneous web traffic, or load, hitting the web server increases.

The major issues that affect website scalability include:

- [“Performance” on page 10](#)
- [“Load management” on page 12](#)

Performance

Performance refers to how efficiently a site responds to browser requests according to defined benchmarks. You can design, tune, and measure application performance.

Performance can also be affected by many complex factors, including application design and construction, database connectivity, network capacity and bandwidth, back office services (such as mail, proxy, and security services), and hardware server resources.

Web application architects and developers must design and code an application with performance in mind. When the application is built, administrators can tune performance by setting specific flags and options on the database, the operating system, and often the application itself to achieve peak performance. Following the construction and tuning efforts, quality assurance testers should test and measure an application's performance prior to deployment to establish acceptable quality benchmarks. If these efforts are performed well, you can better diagnose whether the website is operating within established operating parameters, when reviewing the statistics generated by web server monitoring and logging programs.

Depending on the size and complexity of your web application, it may be able to handle from ten to thousands of concurrent users. The number of concurrent connections to your web server(s) ultimately has a direct impact on your site's performance. Therefore, your performance objectives must include two dimensions:

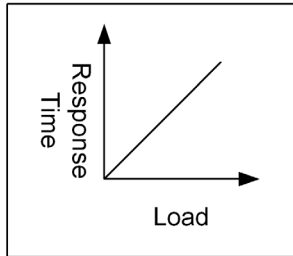
- Speed of a single user's transaction
- Amount of performance degradation related to the increasing number of concurrent users on your web servers

Thus, you must establish response benchmarks for your site and then achieve the highest number of concurrent users connected to your site at the response rates. By doing so, you will be able to determine a rough number of concurrent users for each web server and then scale your website by adding additional servers.

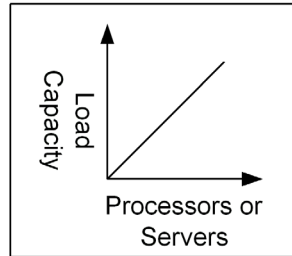
When your site runs on multiple web servers, you must monitor and manage the traffic and load across the group of servers. To learn how to do these tasks, see [“Hardware planning” on page 26](#) and [“Creating scalable and highly available sites” on page 28](#).

Linear scalability

Perfect scalability — excluding cache initializations — is linear. Linear scalability, relative to load, means that with fixed resources, performance decreases at a constant rate relative to load increases. Linear scalability, relative to resources, means that with a constant load, performance improves at a constant rate relative to additional resources.



Linear Scalability Relative
to Load



Linear Scalability Relative
to System Resources
(e.g., hardware)

Caching and resource management overhead affect an application server's ability to approach linear scalability. Caching allows processing and resources to be reused, alleviating the need to reprocess pages or reallocate resources. Disregarding other influences, efficient caching can result in superior linear application server scalability.

Resource management becomes more complicated as the quantity of resources increases. The extra overhead for resource management, including resource reuse mechanisms, reduces the ability of application servers to scale linearly relative to constraining resources. For example, when a processor is added to a single processor server, the operating system incurs extra overhead in synchronizing threads and resources across processors to provide symmetric multiprocessing. Part of the additional processing power that the second processor provides is used by the operating system to manage the additional processor, and is not available to help scale the application servers.

It is important to note that application servers can scale relative to resources only when the resource changes affect the constraining resources. For example, adding processor resources to an application server that is constrained by network bandwidth would provide, at best, minor performance improvements. When discussing linear scalability relative to server resources, you should assume that it is relative to the constraining server resources.

Understanding linear scalability in relation to your site's performance is important because it affects not only your application design and construction, but also indirectly related concerns, such as capital equipment budgets.

Load management

Load management refers to the method by which simultaneous user requests are distributed and balanced among multiple servers (Web, JRun, ColdFusion, DBMS, file, and search servers). Effectively balancing load across your servers ensures that they do not become overloaded and eventually unavailable.

There are several different methods that you can use to achieve load management:

- Hardware-based solutions
- Software-based solutions, including round-robin Internet DNS or third-party clustering packages
- Hardware and software combinations

Each option has distinct merits.

Most load-balancing solutions today manage traffic based on IP packet flow. This approach effectively handles non-application-centric sites. However, to effectively manage web application traffic, you must implement a mechanism that monitors and balances load based on specific web application load. ClusterCATS ensures that the JRun or ColdFusion server, the web server, and other servers on which your applications depend remain highly available.

For more information on using hardware and software for load balancing, see [“Creating scalable and highly available sites” on page 28](#).

Successful scalability implementations

Achieving scalable web servers is not a trivial task. There are various solutions from which to pick, setup and configuration tasks to understand and perform, and many delicate dependencies between related but heterogeneous technologies. This section describes some of the major issues affecting successful scalability implementations:

- [“Designing and coding scalable applications” on page 13](#)
- [“Avoiding common bottlenecks” on page 16](#)
- [“DNS effects on website performance and availability” on page 17](#)
- [“Load testing your web applications” on page 20](#)

Designing and coding scalable applications

Application architects must create designs that are inherently flexible by relying on open standards that don't restrict the application's construction and implementation to vendor-specific interfaces and tools. Similarly, web developers that construct the designed application must be aware that they can significantly impact the application's scalability in the way in which they write their code, build their SQL queries, invoke thread management, access databases, and partition the application.

This section discusses the following topics to consider when designing and building a web application:

- [“Application session and state management” on page 13](#)
- [“Database locking and concurrency issues” on page 14](#)
- [“Application partitioning” on page 15](#)

Application session and state management

As you create web applications, you will probably create specific variables that you intend to carry across multiple interactions between a user's browser and a site's web server(s). Using **client** variables that are stored in a shared state repository, or **session** variables that are stored in memory of a specific server, are popular approaches for accomplishing this task. The latter approach, however, introduces a significant challenge for a website that is supported by multiple servers. When a user has begun a session and variables are stored on a specific server, the user must return to that server for the life of the session to maintain correct state information.

An example that illustrates this concept is an e-commerce application that uses shopping carts. With this type of application, as a customer accumulates items in a cart, there must be a mechanism to ensure that the user can see the items as they are added. One approach is to store these items in session variables on a specific web server. However, if you use this approach, there must also be a way to ensure that the user always returns to the same server for the life of the session. ClusterCATS automatically handles this challenge for you.

Another approach to solving this problem is to store client variables in a back-end common state repository. This approach enables all web servers in a cluster to access variables in a common, shared back-end data store, such as a database. However, this approach can potentially affect your site's performance.

Web developers must think through the user scenarios in which application session and state are affected, and engineer appropriate mechanisms to handle them. The most common ways to handle session data are:

- Client-side options consisting of cookies, hidden fields, a get list, or URL parameters
- Server-side session variables

Note: Storing session data on the server requires that a simple identifier is stored on the client, such as a cookie.

- An open state repository consisting of a common back-end database or other shared storage device

Whatever mechanism your architects and engineers use, they must anticipate the scenarios in which maintaining an application's state is vital to a good user experience. See [“Session-aware load balancing” on page 72](#).

Database locking and concurrency issues

Dynamic web applications that allow users to modify a database must ensure appropriate database **concurrency handling**. This term refers to how an application manages concurrent user requests when accessing the same database records. If an application does not impose a database-locking mechanism on multiple requests to update a record, data integrity can be compromised in the database — two users could make simultaneous modifications to a record, but only the second change would take effect.

For example, consider a Human Resources web application on a company intranet. The HR Generalist adds two new employee records to the HR database by filling out a web form, because two new employees have been hired. The Generalist enters most of the vital information into the records, but doesn't yet have the new employees' phone extensions or HMO selections, so leaves those fields blank. Later in the day, the HR Generalist's manager, the HR Director, obtains this information from both new hires and decides to enter it in the database. However, one of the new employees, after speaking with her husband, decides to change her HMO selection from the basic selection to the PPO choice. The employee calls the HR Generalist to tell him of the change, and the Generalist says he will take care of it immediately. Without talking to the HR Director, the HR Generalist adds the information into the employee records at the same time that the HR Director is attempting to update the information.

In this scenario, if the application uses an appropriate database concurrency validation mechanism, the HR Director receives a message indicating that she could not access the employee record because it was in use, thereby alerting her that someone in her department was trying to change the record. However, if the application did not use such a validation mechanism, the HR Director would overwrite the new data that the Generalist had just entered, resulting in data integrity problems. This example illustrates the importance of your dynamic web applications handling database concurrency issues well.

Application partitioning

The way an application is partitioned and deployed dramatically affects its ability to scale. A key development objective must be to ensure that each partition scales independently of the others, thereby eliminating application bottlenecks.

Application partitioning refers to the logical and physical deployment of an application's three core types of logic, or services — presentation, business, and data access. If you are familiar with the concept of tiered client/server application development, you already understand the rationale for developing applications in this way. The following short review highlights this methodology's benefits.

An application, regardless of whether it is a web application or a more traditional client/server application, has three main categories of logic, or services:

- **Presentation services** — a user interface, by which users interact with the application's features. In a traditional client/server application, this logic resides on a client computer, typically as a proprietary executable file. In a web paradigm, there is no specific proprietary client software required, other than a browser. Emerging web technologies can help you leverage powerful client-side processing available through a browser. These technologies include Enterprise JavaBeans (EJB), scriptlets, JavaScript, applets, and Dynamic HTML. Well-planned use of these technologies can reduce unnecessary trips to the server, thereby minimizing performance degradation.
- **Business services** — the custom business logic and rules that an application uses to perform calculations and application-specific functions. An example of a business service is an algorithm that automatically calculates shipping and handling charges for an order, based on the total cost of the order. In JRun, this logic is contained within scriptlets and EJBs. In ColdFusion, this logic is contained in ColdFusion pages. Depending on the nature of the business and how often the business rules change, business logic can be partitioned to reside on its own server for easier access that expedites frequent logic modifications, or it can reside in stored procedures on the database server.
- **Data services** — the interaction between the application and the database in which the application stores and manipulates data. The way application manages data services is directly tied to the application's performance capability. In short, accessing a database can be costly and can cause significant performance degradation, depending on a variety of factors. For example, the types of database drivers used for connections, the construction of SQL queries, the manner in which database connections are pooled and maintained, and whether stored procedures are implemented for frequent database access, all directly impact the application's performance.

The way that architects and web developers decide to partition and deploy these core application services significantly affects the application's ability to scale. Although your development efforts may no longer be burdened with developing, distributing, customizing, and updating proprietary client software for your applications, the ubiquitous graphical user interface (GUI) — the web browser — presents new interface issues and challenges. For example, you must ensure that your application's presentation remains performance-friendly. It should minimize the number and size of graphic elements that must be downloaded to the client. Also, because some browsers cannot

cleanly display all technologies, such as cascading style sheets (CSS), Java applets, and frames, you must carefully evaluate their use in your applications.

Bear in mind these presentation guidelines, to aid your applications' performance and user experience, and be sure to plan and test for the lowest common denominator that all browsers can accommodate.

Often, partitioning business services to a separate business logic application server from the primary application server, if necessary, can yield better application organization and easier maintenance. You can maximize your application's data services by carefully constructing them and by ensuring that a separate database server (in this case, a separate computer) is used to increase processor capacity for any database transactions.

These are several of the most important topics you and the developers creating your web applications should consider early on. In doing so, you ensure that your web applications are designed and coded with scalability in mind.

Avoiding common bottlenecks

In addition to application design and construction considerations, you must plan to avoid common bottlenecks that can negatively affect a web application's performance.

Following are typical bottlenecks that can affect an application's ability to perform and scale well:

- Poorly written application logic — inefficient programming is probably the most common reason applications perform poorly. Instituting industry best practices, such as coding standards, design reviews, and code walkthroughs, can significantly help to alleviate this problem.
- Processor capacity — even a well-architected and programmed web application can perform poorly if the web server's CPU is unable to provide sufficient processing power. Ensure that heavy-load, mission-critical applications reside on hardware that can effectively do the job.
- Memory — insufficient random access memory (RAM) limits the amount of application data that can be cached. Ensure that the amount of memory installed on the application server computer is commensurate with the needs of the web application.
- Server congestion — server congestion refers to all types of servers, not just the web server. Your application, proxy, search and index, and back-office servers can periodically experience high volume that indirectly degrades the performance of your web application. When planning the physical design of the system, investigate carefully the network topology that will be implemented to ensure that existing servers are sufficient. If they are not, you may need to add new servers to the topology to ensure uninterrupted service and performance expectations.
- Firewalls — some dynamic applications that must restrict anonymous access because they present or share confidential information must pass through a corporate firewall, which can slow down requests and responses. Ensure that the correct ports are open on the firewall to ensure valid security authentication and to enable appropriate client/server communications. (You may be able to open additional secure ports to accommodate increased traffic.)

- Network connectivity and bandwidth — consider the type of network your application will run on (LAN/WAN/Internet) and how much traffic it typically receives. If traffic is consistently heavy, you may need to add additional nodes, routers, switches, or hubs to the network to handle the increased traffic.
- Databases — database access, while vitally important to your application's capabilities and feature set, can be costly in terms of performance and scalability if it is not engineered efficiently. When creating data sources for accessing your database, use a native database driver rather than an ODBC driver, if possible, because it will provide faster access. Similarly, try to reduce the number of individual SQL queries that must be repetitiously constructed and submitted, by placing common database queries in stored procedures that reside on the database server. Tune your databases and queries for maximum efficiency.

DNS effects on website performance and availability

Improper Domain Name System (DNS) setup and configuration on web servers is one of the most common problems administrators encounter. This section addresses the following topics:

- [“What is DNS?” on page 17](#)
- [“DNS effects on site performance and availability” on page 17](#)
- [“DNS core elements” on page 18](#)

What is DNS?

DNS is a set of protocols and services on a TCP/IP network that allows network users to use hierarchical natural language names, rather than computer IP addresses, when searching for computer hosts (servers) on a network. DNS is used extensively on the Internet and on private enterprise networks, including LANs and WANs.

The primary capability of DNS is its ability to map host names to IP addresses, and vice versa. For example, suppose the web server at Macromedia has an IP address of 157.55.100.1. Most people would connect to this server by entering the domain name (www.macromedia.com), not the less-friendly IP address. Besides being easier to remember, the name is more reliable, because the numeric address could change for a variety of reasons, but the name can always be reserved.

DNS effects on site performance and availability

Internet DNS is a powerful and successful mechanism that has enabled huge numbers of individuals and organizations to create easily locatable websites on the Internet. However, DNS by itself may not allow your website to perform and scale as it should, thus causing it to become unavailable and unreliable. Whether you use DNS by itself to load balance inbound traffic depends largely on the site's purpose and the amount of concurrent activity you expect on it. For instance, a low-volume, static site that provides only textual HTML information can probably be accommodated by round-robin DNS. However, a high-volume, dynamic, e-commerce site that you anticipate doing lots of volume won't perform or scale well if it is only supported by round-robin DNS.

To understand why, let's look at the e-commerce example. Even if you have planned ahead and set up multiple servers to support this high-volume site, if you rely only on DNS, it can only perform two tasks:

- Translate natural language names to server IP address mappings so that users can find the site
- Distribute load among servers in a rote, sequential distribution manner, if you have enabled round-robin distribution for multiserver load balancing

However, if a spike in user activity causes servers to overload or fail, round-robin DNS keeps distributing requests among all servers, even if some are not operating.

In short, Internet DNS is limited in its capabilities, and its round-robin distribution mechanism does not include intelligence for monitoring, managing, and reacting to overloaded or failed servers. Consequently, DNS *by itself* is not a sound load-balancing or failover solution for your business-critical sites. ClusterCATS compensates for DNS limitations and lets you create highly available, reliable, scalable web applications.

DNS core elements

The following are core DNS elements that you must be able to configure if your web applications are to work well with DNS:

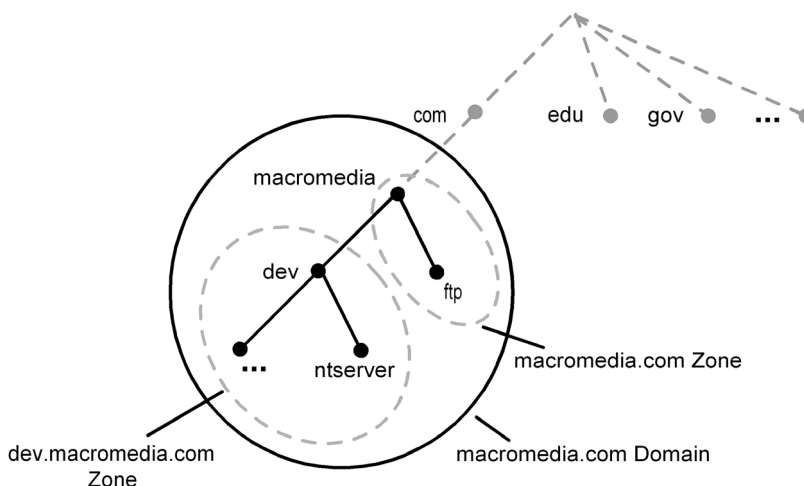
- [“Zones and domains” on page 18](#)
- [“DNS record types, server aliases, and round-robin distribution” on page 19](#)

Zones and domains

A Domain Name System is composed of a distributed database of names. The names in the DNS database establish a logical tree structure called the **domain name space**. On the Internet, the root of the DNS database is managed by the Internet Network Information Center (InterNIC). The top-level domains were originally assigned organizationally and by country. Two-letter and three-letter abbreviations are used for countries. Some abbreviations are reserved for use by organizations — for example, .com, .gov, and .edu for business, government, and educational organizations, respectively.

A **domain** is a node on a network and all the nodes below it (subdomains) that are contained within the DNS database tree structure. Domains and subdomains can be grouped into zones to allow distributed administration of the name space. More specifically, a **zone** is a portion of the DNS name space whose database records exist and are managed in one physical file. One DNS server may be configured to manage one or multiple zone files. Each zone is anchored at a specific domain node. You use zones for breaking up domains across multiple segments to distribute the management of the domain to multiple groups, and to replicate data more efficiently.

The following figure shows these concepts:



DNS servers store information about the domain name space and are referred to as **name servers**. Name servers typically have one or more zones for which they are responsible. The name server has authority for those zones and is aware of all the other DNS name servers that are in the same domain.

DNS record types, server aliases, and round-robin distribution

There are three DNS record types that you must define and configure for each web server in order for ClusterCATS load-balancing and failover technology to work correctly. These records must be defined and configured on your local and primary DNS servers.

- A Record — contains a host-name-to-IP-address mapping, where the natural language name is the primary name representing the IP address.
- PTR Record — contains the IP-address-to-host-name mapping. This is the reverse lookup of the A record, in which, given the IP address, the natural language host name for the IP address is displayed.
- CNAME Record — short for canonical record. This record contains an alias name that maps to the primary host name of a web server. For example, you can assign a server named `www1.yourcompany.com` an alias of `www.yourcompany.com`, so that users never see `www1.yourcompany.com`, in the event of a server redirection.

To see how all of these records work together, let's look at a simple example. There are two web servers, named `www1.yourcompany.com` and `www2.yourcompany.com`. You don't want users to see the primary host names (A records) for these servers in their browser; you want them to see only their assigned aliases (CNAME records), when being redirected.

The DNS entries would look like the following:

; Entries for forward-resolution: A-records		
www1.yourcompany.com	IN A	192.168.0.1
www2.yourcompany.com	IN A	192.168.0.2
; Entries for reverse-resolution: PTR-records		
192.168.0.1	PTR	www1.yourcompany.com
192.168.0.2	PTR	www2.yourcompany.com
; Round Robin entries		
www.yourcompany.com	IN A	192.168.0.1
www.yourcompany.com	IN A	192.168.0.2

To ensure that your site lookups and translations occur as intended, you must provide correct entries in your DNS records, as shown. Also, to enable round-robin DNS functionality, you must create round-robin entries as shown.

On the Windows platform, you make DNS entries using the Domain Name Service Manager utility.

On UNIX platforms, you make DNS entries in the name.db file, which is read by the DNS server's Berkeley Internet Name Daemon (BIND).

Load testing your web applications

Load testing is the process of defining acceptable benchmarks for your web application's performance, and then simulating load and measuring resulting response times and throughput against the benchmarks. You perform load testing to measure the application's ability to scale.

This section discusses the following topics:

- [“Reasons to perform load testing” on page 20](#)
- [“How to load test your web applications” on page 21](#)
- [“Load-testing considerations” on page 22](#)

Reasons to perform load testing

Load testing is important to your website's success because it lets you test its capacities before you deploy it, so you can find and fix problems before they are exposed to your users. Determining your site's purpose, and the amount of traffic you anticipate, may affect how you load test it.

Managers of small sites, who don't expect heavy concurrent loads, might be able to organize actual users to simultaneously access the site to perform load testing. However, this is difficult to accomplish well, because it introduces many human variables. In fact, for larger business-critical systems that expect heavy concurrent load, this type of testing is not feasible and does not provide satisfactory or realistic results.

A better approach to load testing is to use load simulation software. There are some excellent software load-testing tools on the market that let you *simulate* heavy loads

hitting your web server. By using the software in conjunction with your defined benchmarks and formal test plans, you can confidently determine whether your web application is ready for deployment.

Another reason to load test is to verify your failover capabilities. Failover ensures that if a primary server within a cluster of servers stops functioning, subsequent user requests are directed to another server within the cluster. Failover is addressed in more depth in [“What is website availability?” on page 23](#). Using load-testing software, you can essentially force a server redirection by designating a computer as “unavailable” or by shutting it down.

Note: ClusterCATS uses the HTTP protocol to redirect packets of data from a failed server to an available server. Therefore, it is important to verify that your load-testing tool can handle HTTP redirections properly before you initiate load testing.

How to load test your web applications

Before you can load test, you must purchase a load-testing software tool and learn how to use it.

There is a variety of good load-testing software tools on the market, including Segue’s SilkPerformer, Mercury Interactive’s LoadRunner, and RSW’s e-LOAD. Each of these packages provides substantial Web-enabled software-testing solutions that help you effectively simulate and test load.

After you purchase, install, and learn to use load-testing software, you determine benchmarks that you want to—or *must*—achieve for your website, to ensure a good user experience. Following that, you formalize your testing strategy by designing and developing written test plans against which you execute your tests.

When the test plans are written and approved, you run the tests. After you do so, you capture and analyze the load-testing results and report the statistics to the development team. From there, you’ll need to reach consensus about the most serious problems you discovered, the necessary changes to make, and the best way to implement the fixes. After the changes are made and a new build of the application is available, you rerun the tests to look for performance improvements. Again, you analyze the testing results, and continue this cycle until the site is operating within the established parameters that you’ve set. When your team agrees that the site scales well and is operating at peak performance under heavy stress, you’re ready to deploy the application into a production environment.

Load-testing considerations

Before starting your load testing, consider the following:

- Define benchmarks early — ensure that you understand your website's performance and scalability requirements before you start running tests against it. Otherwise, you won't know what you're testing for and the statistics you capture won't have significance. Also, remember that the benchmarks you define should be customized for the current application; don't simply reuse benchmarks from an earlier site on which you may have worked. Each web application is distinct in terms of its design, construction, back-office integration, and user experience requirements.
- Ensure that the test environment mirrors the production environment— create a test environment that is as close as possible to the production environment in which the website will be hosted. If you don't simulate a similar network and bandwidth scenario, or use the same types of servers, or ensure that the same versions of software (operating system, service packs, web server, and third-party tools) reside on the test and production servers, you can't anticipate problems nor determine why they occur. The number of possibilities would be too large.
- Minimize distributed environment load testing — load testing in a distributed environment can be problematic if the network on which you perform load tests becomes congested, resulting in poor response times. Also, if everyone else in the organization uses the network for their everyday activities, such as e-mail, source control, and file management, an increased load on the network will probably cause significant network degradation for them, and accompanying frustration.

In such a scenario, it might be more effective to physically sit at the server on which the application resides and perform the tests locally, rather than bring the entire LAN or WAN to a slow crawl. Also, by testing locally, you can better rule out the network as the source of the scalability problems. Alternatively, you might be able to configure a separate subnet on the LAN or WAN that is distinct from the subnet on which everybody else in your environment uses network services.

You should have a good overview of what scalability implies, the core elements that compose it, some of the issues that affect successful implementations, and the tasks that must be performed to verify that your web applications are able to achieve satisfactory scalability.

The next section describes website availability and reliability concepts and considerations.

What is website availability?

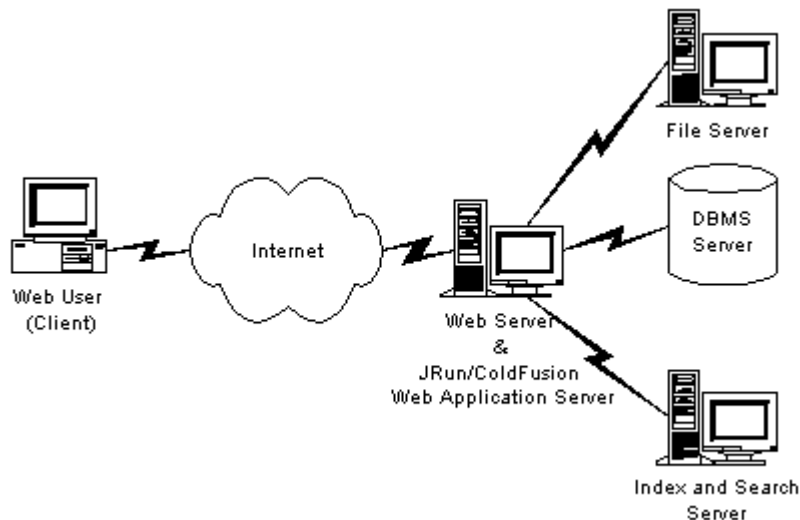
It is critical to design, develop, test, and deploy web applications so they can scale well under heavy and ever-increasing load. However, in spite of the best-laid plans and preparations, servers can fail for seemingly unknown reasons, causing your site to become unavailable. If and when a server fails or becomes overloaded, you want to ensure that the failure won't adversely affect your business by preventing your customers from accessing and using your web application. If it does, you risk jeopardizing your bottom line with lost sales and disgruntled customers who will look to your competitors for goods and services.

This section defines and describes website availability and failover:

- [“Availability and reliability” on page 23](#)
- [“Common failures” on page 24](#)
- [“Website availability scenario” on page 25](#)
- [“Failover considerations” on page 25](#)

Availability and reliability

In simple terms, availability and reliability mean that you can access a website by entering the site's URL in your browser, and all of its features work as intended. Thus, availability and reliability refer to the uptime of a website, which is often directly related to the uptime of the web server and other dependent servers, such as a database server, an application server, or a file server. For a site to be considered available, all of the servers that provide its functionality must work.



For JRun and ColdFusion web applications, it is particularly important that the servers remain as highly available and responsive as the web server and other dependent servers. JRun and ColdFusion process requests sent to them from the web server. Upon successfully processing the application logic, JRun and ColdFusion return the results to the web server, which in turn returns an HTML response to the browser.

Availability and reliability are concerned with keeping the relevant servers that provide services to your web application available at all times. However, if a server on which your site depends becomes unavailable, you must have a sound redundancy scheme to make certain that your site remains available. As your organization moves into an e-business paradigm, you must plan, design, and implement load-balancing and failover strategies that guarantee that your servers will remain operational.

If servers employ a good strategy for load balancing and failover, they provide high availability and reliability to their users. In fact, Internet Service Providers (ISPs) that host commercial websites and offer 24x7 technical support as a competitive service differentiator typically specify in written service-level agreements a percentage of time that they guarantee a website will be available. If the ISP has a sound scalability and failover strategy in place, this figure is usually in the range of 99% or better.

Common failures

Following are typical types of failures that can negatively impact your web application's availability and reliability:

- **Hardware failures** — while less common than software failures, hardware failures do occur, and can include crashed hard drives, blown processors, and corrupted network cards. Diagnosing and fixing these issues can be a lengthy endeavor because of time spent getting parts and performing labor. If your web application is mission-critical, you should ensure a sound hardware redundancy strategy to avoid costly downtime. A sound strategy includes a minimum of two, but preferably three, web servers.
- **Software failures** — the software failures that most affect a web application involve the web server's operating system, the web server software itself, or the web application software. If the operating system crashes or becomes corrupt, the web server cannot function properly (or perhaps at all), compromising your web application's availability, reliability, and performance. Similarly, if the web server software crashes or acts erratically, it will probably cause the web server to stop running. Preparing for software failures is difficult, but if you have mirrored secondary hardware systems in place to account for failures, you'll minimize your web application's downtime.
- **Server failures** — other servers on which your web application depends can also fail, causing downtime or diminished capabilities on your site. For example, for distributed applications, a proxy server might go down, causing requests for your web application's services to go unanswered. Or the database server might crash, making it impossible for users to submit or retrieve information from your database. Or a mail server might go down, making it impossible for your users to successfully send mail to you. Ensure that your organization's IT architecture includes network monitoring and notification software that can quickly report on the general health of your network and alert you about any failed servers.

Website availability scenario

Imagine that you have just built a robust, interactive e-commerce website on which you plan to sell the most sought-after books and music in the world. You have used Java scriptlets to build the application, so of course you've taken advantage of its many built-in features, including secure database access, multithreading, and integrated session management.

Upon finishing the development work and quality assurance testing, you deploy the website onto one production web server that is hosted within your IT department. The IT department informs you that it can use its existing Internet connection to make your site live, avoiding the additional hosting support cost of using an outside vendor.

The site goes live, and it's an instant success. Orders start pouring in the very first day, and huge numbers of people log on to browse and buy. Everything seems perfect. Then, on the second day of business, the load hitting the site is so high, the web server's performance slows to a crawl, eventually causing the server to become unavailable. Suddenly, your tech support lines are ringing off the hook with complaints that users cannot access your site, causing you to lose significant business.

Although the application provided many useful features and capabilities, customers could not access them, because the site's performance degraded to the point that the site became unavailable. Because the site was deployed on only one server, the incoming traffic could not be load balanced. Also, without redundant servers in place, the site could not intelligently load balance increasing traffic nor redirect traffic to other available servers (no failover).

This simple scenario illustrates how critical adequate scalability, performance, and failover planning are to any successful web development effort. Servers can become overloaded or fail at any time, so ensure that your design, development, testing, and deployment strategies are sound, promote good communication between necessary departments, and include adequate disaster recovery capabilities.

Failover considerations

The ability to failover unavailable servers to redundant servers is a cornerstone of any mission-critical application, one that ensures an application's continuous and reliable operation. Such disaster planning and recovery can be broken down into these topics:

- [“Hardware planning” on page 26](#)
- [“Systems monitoring” on page 26](#)
- [“Corrective actions” on page 26](#)

Review the following considerations to ensure that you have a sound failover strategy in place — one that guarantees your website's availability.

Hardware planning

As indicated in the availability example above, you must acquire all necessary hardware and configure it before you deploy an application. All websites have different requirements, feature sets, purposes, audiences, and budgets, and therefore different needs. However, if your site is a business-critical system that affects your company's bottom line, you *must* ensure an appropriate redundancy strategy by having two or more redundant systems in place. In fact, Macromedia recommends that you use a minimum of three servers to support a critical website, so you can take one server offline to perform update and maintenance tasks while maintaining at least two servers in production at all times. This scheme provides administrative flexibility and protects your site from hardware or software failures.

The two predominant redundancy models used today are:

- Primary/backup servers — an example of this model would be an important web application that receives relatively little traffic, such as an intranet. Typically, this redundancy model uses an expensive, high-capacity server for the primary server, and an inexpensive, lower-quality server for the backup server in case the primary server fails.
- Parallel servers — this is a classic load-balancing/redundancy mode, and is used most often for business-critical applications. Unlike the primary backup scheme, the multiple servers in a parallel scheme are considered peers and are grouped as a single entity to support one or more applications.

You can use identical cloned hardware in your server clusters, or you can mix hardware sizes and models. Cloned, higher-capacity, higher-end hardware might have greater up-front hardware costs, but help minimize long-term administration costs. Conversely, mixing hardware models and capacities might be less expensive in the short term, but could add administrative costs later on.

If you plan to use a parallel model, using many middle-range servers, rather than fewer high-end ones, or many inexpensive ones, is recommended. Servers that provide adequate capacity and are moderately priced can generally accommodate your needs as well as expensive ones, but at a fraction of the cost.

Systems monitoring

Ensure that your network and the mission-critical sites that reside on its servers are supported by systems-monitoring software. This type of software actively and continuously monitors an application's availability and service levels. These monitoring programs must be able to not only detect problems, but also route alerts to administrators for immediate notification of problems.

Corrective actions

The third major failover consideration is the corrective actions that must occur if a failure causes a server to become unavailable. Generally, if a server goes down and causes your site to become unavailable, some level of human interaction is usually required to effectively diagnose and correct the problem.

However, before the analysis and repair can occur, the administrator must be notified. Whatever failover system you put in place, it should include an automated notification system that can route alerts through your telecommunications infrastructure (e-mail, pagers, real time Web-based alerts, and so on) to the appropriate administrator for prompt attention.

Besides notifying the administrator that a problem has occurred, you also want your failover solution to automatically redirect traffic intended for the unavailable server to other available servers until the unavailable server is fixed. This crucial corrective action is what keeps your website up and available to your users even if one of the servers supporting it is experiencing problems.

Creating scalable and highly available sites

When you understand the issues of scalability and availability, the next step is to learn the techniques you can use to achieve scalable and highly available websites.

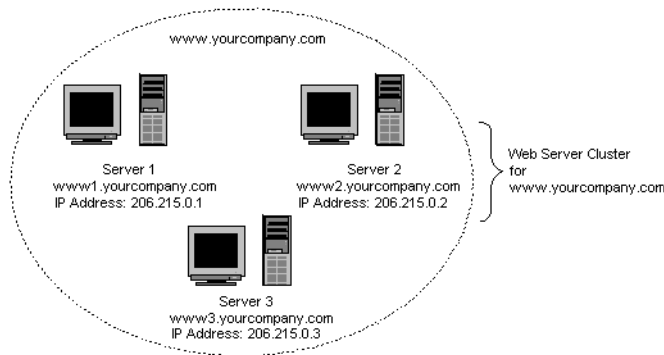
This section describes the following topics:

- “What is clustering?” on page 28
- “Hardware-based clustering solutions” on page 29
- “Software-based clustering solutions” on page 30
- “Combining hardware and software clustering solutions” on page 32

What is clustering?

Clustering is a technique in which two or more web servers supporting one or more domains (such as `www.yourcompany.com`) are grouped as a **cluster** of servers, to collectively accommodate increases in load and provide system redundancy.

The following figure shows an example of a server cluster for a website:



Clustering for scalability works by distributing load among servers in the cluster (load balancing) using an unintelligent-but-regular distribution sequence (round-robin DNS and routers) or a predefined threshold or algorithm (specialized clustering software) that you specify and can adjust for each server in the cluster.

Clustering for failover relies on redundant servers to ensure that business-critical applications remain available if one of the servers in a cluster fails. Intelligent software-based failover solutions can detect when a server has failed and automatically redirect new incoming HTTP requests to available cluster members. Some hardware-based failover devices that have less built-in intelligence require an administrator's intervention when a failure is detected.

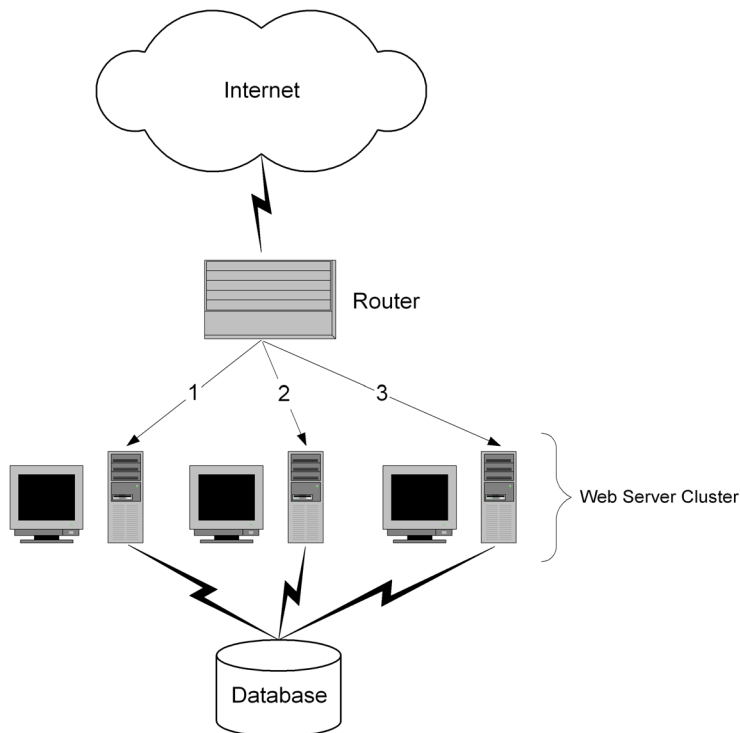
Clustering can be accomplished using software-based solutions, such as round-robin DNS alone or together with a third-party package; a hardware-based solution, such as a packet router; or a combination of the two.

Hardware-based clustering solutions

A common and reliable hardware-based clustering solution is a packet router. One of the most popular routers is Cisco Systems' LocalDirector. A router, in front of a cluster of web servers, directs incoming HTTP requests to available web servers in the cluster. A router works by assessing the rate and volume of IP packet flow to and from web servers, and selecting the best server to accommodate the traffic. This process is fast and efficient. The router and clustered web servers comprise a **virtual server**.

Routers are considered semi-intelligent devices because they can detect a server failure and redirect requests to other servers. If a web server fails or stops responding, the router stops sending packets to the unresponsive server. Routers are not considered fully intelligent because, while they can redirect requests upon discovering a failure, they do not let you configure redirection thresholds for individual servers. They also do not support application-aware load balancing.

The following figure shows a router distributing requests in round-robin fashion to the available servers in a web server cluster:



Advantages of hardware-based solutions

A hardware-based clustering solution, such as a router, is an attractive solution for the following reasons:

- It uses proven technology
- It has relatively low complexity

- There are no recurrent licensing fees
- It is semi-intelligent; routers can load balance in a round-robin fashion, detect failures, redirect traffic, and remove failed servers from a cluster.

Note: Load-balancing devices offer different features and capabilities.

Considerations

Carefully evaluate the following issues against a router's attributes:

- Expense — hardware devices can be expensive relative to some software solutions, even without yearly licensing fees.
- Single point of failure — if a problem develops on the load-balancing device *itself* and *it* fails, your load-balancing and failover strategies do not work. Although some load-balancing devices come with secondary systems for this reason, this additional equipment often inflates the overall price of a hardware solution.
- Lack of application-awareness — the device cannot be tuned for particular types of web applications (static vs. dynamic sites) or for the development tools used to build them (scriptlets vs. JSP vs. CGI vs. ASP and so on). Consequently, a router cannot measure the performance of a web application server.
- Limited intelligence — the device does not let you configure individual load and redirection thresholds for each server in a cluster, so it cannot effectively manage load to prevent failures.

Software-based clustering solutions

There are several kinds of software-based clustering solutions on the market. As with hardware-based clustering solutions, there are strengths and weaknesses associated with each. These software solutions include:

- Round-robin DNS — a very popular choice because of its relative simplicity and low implementation cost, but does not include intelligence for load balancing or failover.
- Primary/backup clustering — cloned systems provide redundancy for one another. This type of clustering does not provide parallel server load balancing.
- Smart clustering — combines the advantages of round-robin DNS and backup clustering to provide simplicity with intelligence and redundancy.

ClusterCATS lets you easily create, optimize, and maintain “smart” clusters to support your web applications. ClusterCATS runs on Windows, Solaris, and Linux platforms and works with leading mission-critical web servers, including Microsoft IIS, Netscape Enterprise Server, and Apache. It is easily administered from remote locations and provides robust features, including:

- Configuring load and redirection thresholds by server
- Optimizing the load-balancing scheme with application-aware and session-aware load balancing
- Automatically detecting failures
- Automatically redirecting traffic to available servers
- Automatically notifying administrators of problems

Advantages

The following benefits make a software-based clustering solution attractive:

- Relatively low expense — compared to the cost of hardware devices, such as routers or switches, software-based clustering solutions are relatively inexpensive. In fact, you can cheaply implement Internet DNS on UNIX and Windows platforms for initial load-balancing needs, and augment with third-party clustering software.
- Flexibility — some clustering software can augment existing hardware devices, providing a more robust load-balancing and failover solution. By integrating hardware with software, you diminish, if not eliminate, losses on capital expenditures that your organization has already made. For more information, see [“Combining hardware and software clustering solutions” on page 32](#) and [“Load-balancing devices” on page 92](#).
- Intelligence — some software solutions provide a level of intelligence that enables preventive load-balancing measures that actually minimize the chance of servers becoming unavailable. If a server does become overloaded or actually fails, some software can automatically detect the problem and reroute HTTP requests to available servers in the cluster.
- No single point of failure — by distributing the load-balancing and failover capabilities among multiple servers in a cluster or multiple clusters, as opposed to relying on only a single device, no individual server failure can disable your application.

Considerations

Consider the following issues when evaluating software-based solutions for your environment:

- Differences among feature sets — software-based clustering solutions differ in their of capabilities and features. For instance, some lack automatic failure detection, notification, or IP address assumption, and others have significantly delayed detection. Some let you configure load thresholds to enable preventive measures, and some don't. Determine your scalability and failover needs in advance and pick your solution accordingly.
- Platform constraints — determine whether the software solution is available on your platform and operates with your preferred web server. When reviewing data sheets and other marketing collateral from vendors, ensure that the robust features you want are available on the platform you need.
- Level of complexity — some software-based clustering solutions are relatively simple. Others introduce more complexity because of the features offered, the amount of initial configuration and subsequent administration, or the amount of integration that must occur between other systems and devices.

Combining hardware and software clustering solutions

Instead of having to choose either a hardware solution or a software solution, you can combine both types of clustering choices. Combining hardware and software solutions certainly provides the greatest scalability and availability capabilities for a site. A combined solution is an attractive option if your organization has already invested in one, but is looking for more comprehensive coverage. Having the flexibility to integrate hardware with software means that your organization won't necessarily have to absorb a capital loss on a previous technology investment if you decide to purchase additional clustering technology.

However, as already discussed, all hardware or software solutions are not equal. Many have different features and capabilities, and not all hardware and software integrate well together. Investigate thoroughly when purchasing technology to augment your current solution.

For a visual representation of hardware and software clustering solutions working together, see [“Hardware-based clustering solutions”](#) on page 29.

CHAPTER 3

Installing ClusterCATS

Before installing ClusterCATS, you must make many important decisions about the architecture of your website. Use the first section in this chapter to guide you through the decision-making process. When you have installed ClusterCATS, read the last section in this chapter for important information on how to make your site secure and reliable.

Contents

- [Before you install.....](#) 34
- [Installing ClusterCATS.....](#) 41
- [After you install.....](#) 45

Before you install

Before installing ClusterCATS and creating server clusters, you must perform the following pre-installation tasks:

- [“Upgrading from a previous version of ClusterCATS” on page 34](#)
- [“Configuring DNS servers” on page 34](#)
- [“Configuring server failover” on page 38](#)
- [“Using ClusterCATS dynamic IP addressing” on page 38](#)
- [“Configuring firewalls” on page 38](#)
- [“Analyzing web server content” on page 39](#)
- [“Considering domain controllers \(Windows NT only\)” on page 40](#)

Upgrading from a previous version of ClusterCATS

To update the ClusterCATS application while preserving your configuration settings, re-install ClusterCATS using the instructions in this chapter. The ClusterCATS installation detects that a configuration is available for use and prompts you by asking whether you want to use the configuration in the new installation. You can use the ClusterCATS Server setup options, or keep the existing cluster configurations.

Configuring DNS servers

ClusterCATS software requires that both the forward lookup (host name-to-address translation) and reverse lookup (address-to-host name translation) be registered with your DNS server. For evaluation purposes, you can use host files, but this is not recommended in a production environment.

Note: ClusterCATS does not support Dynamic Host Configuration Protocol (DHCP). A unique IP address must be assigned to each web server.

This section addresses the following topics:

- [“Understanding DNS servers” on page 34](#)
- [“Configuring your primary DNS server” on page 36](#)

Understanding DNS servers

When you enter a URL into a web browser, the browser is able to locate the website you want to visit because of the name-to-IP address translation that the Internet Domain Name System (DNS) performs. This section reviews two important components of the DNS infrastructure that enable this capability — primary DNS servers and local DNS servers.

Primary DNS servers

The primary DNS server provides the final mapping of a website name to the computer on which a website resides. The primary DNS server can be located anywhere on the Internet, but most reside either in the same physical location as the web servers or at the ISP that provides the connection between the web servers and the Internet.

The primary DNS server contains tables of forward and reverse name translations. For example, forward translation entries (A records) look like this:

www1.company.com	192.168.0.1
www2.company.com	192.168.0.2

Reverse translation entries (PTR records) are opposite, and look like this:

192.168.0.1	www1.company.com
192.168.0.2	www2.company.com

Configure your websites with forward *and* reverse DNS entries on your primary DNS server. If you are not responsible for maintaining your primary DNS server, tell your DNS administrator to add forward and reverse entries for your explicit web server names (www1.company.com, www2.company.com, and so on).

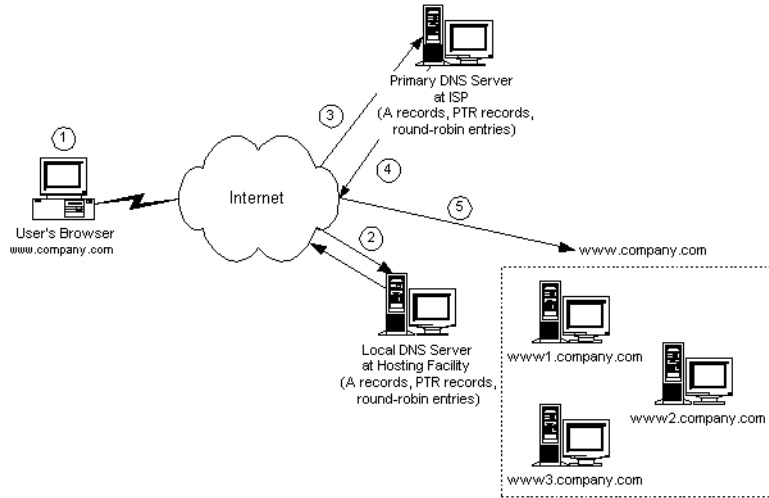
Note: ClusterCATS does not operate correctly unless both forward and reverse translations are configured for each explicit web server.

Local DNS servers

A local DNS server usually resides at a web hosting facility. The local DNS server stores its own local table of name translations for the websites that the browser visits. If a user enters a URL in a browser that the browser has already visited, it retrieves the host name-to-IP address translation from the local DNS server's table. However, if a user enters a URL for a site that the browser on that computer has never visited, the local DNS server must access the primary DNS server on the Internet to resolve the name-to-IP mapping before the browser can send a request to the appropriate web server. To summarize, primary and local DNS servers work together to resolve name-to-IP address mappings in the following way:

- 1 A user enters a website URL in a browser.
- 2 The browser checks the local DNS server for the name-to-IP address mapping. This server typically resides at the facility where the web servers are hosted.
- 3 If the local DNS server does not have the mapping, it goes to the Internet and locates the primary DNS server to look up the name-to-IP address mapping.
If round-robin DNS is in use, the primary DNS server determines which server in the cluster is next in line to receive the request.
- 4 The primary DNS server sends the translation to the local DNS server, which in turn sends it to the user's browser.
- 5 The browser sends an HTTP request to the correct web server hosting the site.

The following diagram shows this process:



Configuring your primary DNS server

You must configure DNS so the forward *and* reverse lookup translation entries are entered and registered correctly with your primary DNS server. To do this, you must define DNS A and PTR DNS records for the web servers on your primary DNS server.

Besides standard name translations, your primary DNS server can distribute HTTP requests sequentially across clustered servers, using a technique called round-robin DNS. This service lets DNS return a list of multiple servers to the browser that requests a name translation.

Round-robin DNS and ClusterCATS work well together. You should not rely on just round-robin DNS for distributing load for your business-critical sites, because DNS functionality is limited. In short, DNS is a good load *distribution* technique, but it cannot *manage* load because it cannot react to increases in server traffic. It also cannot detect server failures, nor redirect requests among available servers. ClusterCATS compensates for these limitations.

Macromedia recommends that you use round-robin DNS or a hardware load-balancing device to distribute requests initially to the web servers in your cluster. After the initial distribution, ClusterCATS load management and failover features automatically take over and ensure that your web applications remain up and running.

Using ClusterCATS with round-robin DNS

For high-volume sites, you should use round robin DNS to initially distribute requests to the web servers in your cluster. The load management component of ClusterCATS enhances round-robin DNS by eliminating its two major limitations:

- Server failure — round-robin DNS cannot detect server failure. Should a server in a cluster fail, another server on the subnet immediately and transparently assumes the IP address of the failed server.
- Server overload — round-robin DNS cannot detect server overloads. ClusterCATS lets you configure load thresholds for each server. Should actual server load exceed the load threshold, ClusterCATS transparently redirects users to another web server, using an HTTP redirect. When redirected, user requests and responses flow to and from that server directly, minimizing response time throughout the user session.

You must ensure that round-robin DNS entries are configured correctly on your primary DNS server so ClusterCATS operates effectively with round-robin DNS. For example, for a single-location server cluster consisting of four servers, you must configure round-robin DNS across all four servers for the domain name, and individual IP addresses for each explicit server name.

For example, the DNS table forward entries on your primary DNS server would be similar to these:

Host Name	IP Address
www.company.com	193.168.0.1
www.company.com	193.168.0.2
www.company.com	193.168.0.3
www.company.com	193.168.0.4
www1.company.com	193.168.0.1
www2.company.com	193.168.0.2
www3.company.com	193.168.0.3
www4.company.com	193.168.0.4

The DNS table reverse entries on your primary DNS server would be similar to these:

IP Address	Host Name
193.168.0.1	www1.company.com
193.168.0.2	www2.company.com
193.168.0.3	www3.company.com
193.168.0.4	www4.company.com

Note: When using round-robin DNS, do not define a reverse mapping (PTR record) for the site name (www.company.com); the cluster will not operate properly if you do. Define only forward mappings (A records) for www.company.com. Define A records and PTR records for all explicit servers (www1, www2,...) in the cluster. This configuration ensures that requests cycle through the servers sequentially, or “round-robin.”

Round-robin DNS distributes the initial domain-level requests across all four servers. Thereafter, ClusterCATS distributes load to avoid failed or overloaded servers.

Configuring server failover

ClusterCATS protects clusters from server hardware and software failures. When a server is no longer sending or receiving packets from the network, its IP address (and, therefore, its HTTP requests) are assumed by another cluster member, which picks up HTTP traffic originally addressed to the failed server. Server failover services are provided per subnet.

Server failover is an option to select during the installation process. If you do not do so, you must reinstall ClusterCATS and select that option. On Windows systems, preparing your site for ClusterCATS Server failover can require uninstalling your web server software. For more information, see [“Using server failover” on page 137](#).

Using ClusterCATS dynamic IP addressing

You can set up your website so ClusterCATS dynamically assigns IP addresses to your web servers. This addressing scheme includes a static maintenance address for each server that lets you and ClusterCATS contact the server at any time, even during a web server failure.

The setup for ClusterCATS dynamic IP addressing varies depending on your cluster’s operating system:

- Windows — If your IP address for the local system is the same as the IP of your web server, setting up your site for ClusterCATS dynamic IP addressing can involve reinstalling your web server software and resetting your TCP/IP settings. Consider this carefully before installing ClusterCATS. For more information, see [“ClusterCATS dynamic IP addressing \(Windows only\)” on page 132](#).
- UNIX — It is not necessary to configure a UNIX system for dynamic IP addressing because it is set up by default if you select that option during installation.

Configuring firewalls

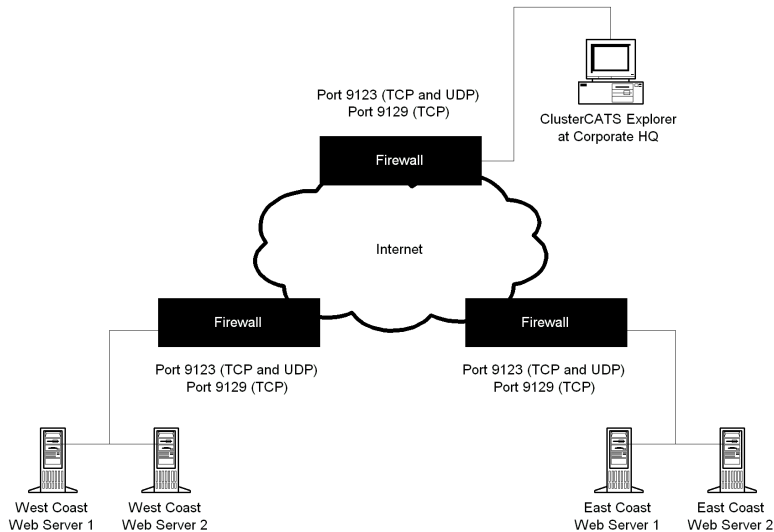
Many corporate environments today rely on firewalls to securely control access to proprietary knowledge that resides on public Internet sites, internal intranet sites, or private extranet sites. You can configure ClusterCATS to work seamlessly across one or more firewalls.

A common technique is to use NAT as a security precaution on your firewall. This configuration segregates internal and external resources and facilitates extra control and monitoring of web traffic. For more information, see Macromedia Knowledge Base Article 15339.

If multiple, distributed server clusters support a domain, you must open appropriate ports on each firewall to ensure that the server clusters’ load-balancing and failover features work. For example, if you cluster multiple, distributed web servers that have a firewall between them, you must open ports 9123 and 9129 on the firewall that separates them to enable server-to-server communications. Also, if you will manage your cluster

from behind another firewall, you must open both ports so the ClusterCATS Explorer can communicate with the cluster.

The following diagram shows this scenario:



This scenario involves Company ABC, which has East Coast and West Coast server groups connected to the Internet, protected by several firewalls. The ClusterCATS Explorer resides at the corporate headquarters behind a firewall with a direct connection to the Internet.

You must open and configure the appropriate communication ports on your firewalls to allow server-to-server communication in a distributed setting and server-to-client communication.

Note: You must open both ports on all affected firewalls.

These ports include the following:

- Port 9123 (for TCP and UDP access) — opening port 9123 on a firewall allows multiple, distributed server clusters residing in different locations to communicate with one another across firewalls
- Port 9129 (for TCP and UDP access) — opening port 9129 on a firewall allows ClusterCATS Explorer to communicate with multiple, distributed server clusters across firewalls

Analyzing web server content

All web servers, virtual servers, or websites in a cluster must have identical content.

You should specify the same default document for each web server in the cluster. For example, set the default document to `default.jsp`.

Considering domain controllers (Windows NT only)

If you use Windows NT Domain server authentication, each web server in a cluster must participate as a member NT server in a domain. Do not set a server in your cluster as the primary domain controller (PDC). ClusterCATS Server failover will interfere with the function of the PDC. An NT server can be a backup domain controller, but this is not the recommended configuration.

Installing ClusterCATS

ClusterCATS is a separate installation package from the JRun server installation program. You must install the ClusterCATS load-balancing and high-availability software on each servers in your cluster. You can run the installation program to install ClusterCATS on a separate, nonclustered computer from which you will administer your clusters with ClusterCATS Explorer (Windows).

Before installing ClusterCATS on a platform, review the pre-installation steps described in [“Before you install” on page 34](#).

This section describes the following:

- [“Installing ClusterCATS on Windows” on page 41](#)
- [“Installing ClusterCATS on UNIX” on page 42](#)

Installing ClusterCATS on Windows

ClusterCATS supports the Microsoft Internet Information Server (IIS) 4.0 and Netscape/iPlanet Enterprise Server on Windows NT 4.0 (with Service Pack 4 or later) and IIS 5.0 on Windows 2000 and Windows .NET Server.

To install ClusterCATS on Windows servers:

- 1 Run the ClusterCATS `setup.exe` file from Windows Explorer or the Run dialog box.
- 2 Accept the default installation directory, or click Browse to select a different directory and click Next. In this manual, the installation directory is referred to as `<CC_install_directory>`.
- 3 Use the following table to determine which components to install. For more information, see [“ClusterCATS components” on page 6](#).

Component	Reason to Select Component
ClusterCATS Explorer	To use this computer to administer other ClusterCATS Servers in your clusters. This computer can also be a member of the cluster with the ClusterCATS Server component.
ClusterCATS Server	If you are adding this computer as a member of a cluster. This component includes the Server Administrator.
Documentation	To make ClusterCATS documentation accessible from this computer.

If more than one of the supported web servers is running, the Select Web Server dialog box appears.

- 4 Select the web server to which you want ClusterCATS to bind. ClusterCATS does not support clustering different types of web servers on one computer.
- 5 Select a method of load management.

Note: You must select JRun (JSP) Performance to have ClusterCATS manage the JRun server load.

The following table describes your options:

Method	Reason to Select Option
HyperText Transport Protocol (HTTP)	To have ClusterCATS load manage just your web server's HTTP requests.
JRun (JSP)	To have ClusterCATS load manage your JRun server's JSP and servlet requests.
ColdFusion (CFM)	To have ClusterCATS load manage your ColdFusion server's CFML page requests. You must have ColdFusion Server installed for this option to take effect.

- 6 In the Server Fail-Over dialog box, to enable this server to assume the IP address of a failed server, click Yes. To prevent this computer from assuming a failed server's IP address, click No.

Note: If you use Cisco LocalDirector with ClusterCATS, select No for server failover. If ClusterCATS performs dynamic IP addressing, the LocalDirector will not be able to recover the failed-over IP address.

For more information, see [“Using server failover” on page 137](#).

- 7 You are prompted to open ClusterCATS Explorer right away. By doing so, you can add the new server to a cluster or create a cluster. If you are going to use this server as the administration computer, you are not required to add it to a cluster.

Installing ClusterCATS on UNIX

To install ClusterCATS on UNIX, you must have the following information:

- The names of the web server on which you will install ClusterCATS
- root access to the systems on which you will install ClusterCATS
- The names, types, and install directories of the web servers you will be clustering
- The directory where JRun is installed

Note: The procedures in this section assume you have installed JRun.

ClusterCATS supports the Apache Web Server and the Netscape Enterprise Server on Solaris platforms, and the Apache Web Server on Linux.

For more information about installation options, enter ? at any prompt during the installation procedure. Default selections appear in brackets [default].

To install ClusterCATS on UNIX:

- 1 Enter the following command as root:

```
./cluster_install
```

The ClusterCATS installation welcome message appears.

Do you wish to continue with this installation? [yes]?

- 2 Enter Yes to continue with the installation or No to exit the installation.

The license agreement confirmation message appears.

Do you agree to the license terms? [no]?

- 3 Review the `license.txt` file that is supplied with ClusterCATS. If you agree with the licensing terms, enter Yes at the prompt. If you do not agree with the licensing terms, enter No. Entering No terminates the installation procedure.
The installation directory prompt appears.
Enter install directory for ClusterCATS Server: [/opt]:
- 4 Enter the base directory where ClusterCATS will be installed, or press Enter to accept the default. The installation creates a `./btcats` subdirectory under the base directory.
Note: The base directory (such as /opt) must exist prior to the installation.
The installation continues with the Configure Web Server Specific Information section.
If you are installing ClusterCATS on Linux, skip to the next step.
If you are installing ClusterCATS on Solaris, you are prompted to enter the type of web server to bind ClusterCATS to:
Enter Web Server type (Netscape, Apache, or <cr> to continue):
- 5 On Solaris, enter your web server type. On Linux, continue with the Apache instructions in this step.
Apache: You are prompted to enter Apache's installation directory and then the location of the Apache config file:
Enter Apache installation directory: [/usr/local/apache]:

Enter location of Apache config file httpd.conf: [/usr/local/apache/conf]:
Netscape: You are prompted to enter the location of Netscape's root directory:
Enter Netscape Enterprise Server Root: [/usr/netscape/suitespot/https-server]:
- 6 Enter the configuration file location. ClusterCATS prompts you to optimize load balancing for this server:
Optimize load balancing for either HTTP or JRun on https-<yourserver> [HTTP]:
ClusterCATS prompts you for the JRun installation directory:
Enter install directory for JRun []:
- 7 Enter the directory. ClusterCATS prompts you to turn on failure monitoring:
Monitor Web Server for Failures? [yes]?
- 8 Enter Yes to have monitoring turned on for this server. Enter No to disable monitoring.
ClusterCATS optionally supports monitoring and restarting the web server on server failure. Failures include the server not running or not responding to HTTP requests.
ClusterCATS prompts you to enable server failover on this server:
Enable ClusterCATS Server instance for Failover? [yes]?
- 9 Enter Yes to enable this server to assume the IP addresses for a failed cluster member, and to pick up all the HTTP traffic originally addressed to the failed server. Enter No to skip failover support for this server.
For more information, see [“Using server failover” on page 137](#).

If you are configuring ClusterCATS with Netscape and selected Yes, you are prompted to decide which servers in the cluster this server will provide failover support for:

Cluster Mates to provide Failover for: all, subset, none [all]:

- 10 Netscape only: Enter all to provide failover support to all members of this server's cluster. Enter subset to explicitly define the cluster members for which this server will provide failover support. Enter none to disable server failover.

If you entered subset, you are prompted to enter a list of the ClusterCATS Server Members that are allowed to fail over to this server. For more information, see the online help.

- 11 Restart your web server for the changes to take effect.

After you install

When you have successfully installed ClusterCATS on all members of the cluster and any administrative computers, you are ready to create your first cluster.

If you administer ClusterCATS from a Windows computer, you can use the Cluster Setup Wizard described in [“Creating clusters with the Cluster Setup Wizard” on page 54](#), or manually create the cluster using the procedure described in [“Creating clusters” on page 54](#).

If you administer ClusterCATS from a UNIX computer, see [“Creating clusters in UNIX” on page 60](#).

Regardless of the method you use to create your first cluster, you should familiarize yourself with the procedures in the following table when implementing your web applications in a clustered environment:

Option	Description
Load thresholds for servers	Two response time thresholds configured for each server. The first defines maximum or busy load; the second activates load management. If the load for the server exceeds the busy threshold, no new sessions can start on that server. If another server in the cluster has the capacity to handle additional users, requests are redirected to it. The load management activation threshold is referred to as the gradual redirection threshold and is designed to prevent the server from reaching the peak threshold. For more information, see “Server load thresholds” on page 66 .
Email addresses for alarm recipients	ClusterCATS generates alarm notifications for several events, including HTTP server failures, low disk space, server busy, and web server failover. You provide e-mail addresses of all administrators for ClusterCATS to notify for each generated alarm notification. For more information, see “Administrator alarm notifications” on page 98 .
Session-aware load balancing	If your web applications use session variables that store information in web server memory, you should enable session-aware load balancing. This feature prevents users who have established a session from being redirected to another server as a result of load balancing. For more information, see “Session-aware load balancing” on page 72 .
Administering with the ClusterCATS Web Explorer	If you use a UNIX computer to administer your cluster with ClusterCATS Web Explorer, you must configure your web server to host the Web Explorer pages. For more information, see “Configuring the communications port on your web server” on page 50 .
Administrative authentication	Password protect administrative access to your cluster members using domain accounts (NT only) or local accounts on each system (UNIX and NT). Administrative users must also be members of the group <code>sys</code> , or a special <code>BT_<clustername></code> group. For more information, see “Administering security” on page 103 .

CHAPTER 4

Configuring Clusters

When you have configured your website and installed ClusterCATS, use the procedures in this chapter to create and configure clusters.

Contents

• Introduction to ClusterCATS Administration.....	48
• Creating clusters	54
• Removing clusters.....	62
• Adding cluster members	63
• Removing cluster members.....	65
• Server load thresholds	66
• Session-aware load balancing	72
• Persistent session failover in JRun	74
• Using ColdFusion probes	77
• Using JRun probes.....	84
• Load-balancing devices	92
• Administrator alarm notifications	98
• Administrator e-mail options	100
• Administering security	103

Introduction to ClusterCATS Administration

ClusterCATS consists of these components:

- ClusterCATS Server
- ClusterCATS Explorer and ClusterCATS Web Explorer
- ClusterCATS Server Administrator and `btadmin`

The following sections describe these components.

All the components are installed on a computer when you run the ClusterCATS installation program.

You must run the installation program on each server that will be part of your cluster, and on the Windows computer from which you will use ClusterCATS Explorer to administer the cluster. Even if your clusters run on Solaris or Linux platforms, you can use a Windows computer to run the ClusterCATS Explorer (recommended). You can also use the Web-based Explorer in conjunction with included server utilities to administer your clusters.

Note: Read the description of each component that is relevant to your installation in the sections that follow. These sections contain important configuration information.

ClusterCATS Server

The ClusterCATS Server is the heart of the clustering and load balancing of ClusterCATS. It must be installed on each server in your cluster. The server monitors the status of all other web servers in a cluster and tracks application and transaction resource availability. ClusterCATS Server runs on Windows, Sun Solaris, and Linux platforms. To administer the ClusterCATS Server, use the ClusterCATS Server Administrator (Windows) or the `btadmin` utility (UNIX).

Each ClusterCATS Server component performs the following functions:

- Intelligently manages HTTP load across web servers
- Proactively manages JRun or ColdFusion server load
- Provides failover support for every server in your cluster
- Proactively monitors JRun and or ColdFusion servers and applications

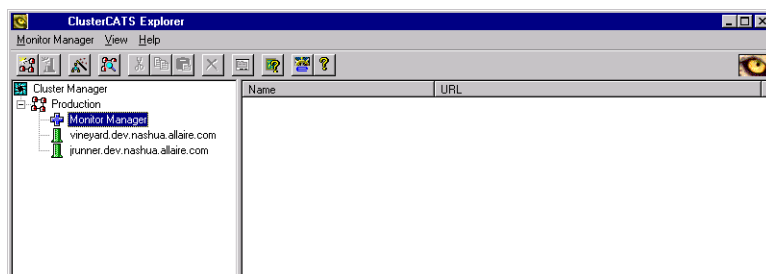
ClusterCATS Explorer (Windows only)

ClusterCATS Explorer is a Windows-based administration utility that you use to create and manage clusters from one computer. Using a Windows Explorer-like graphical interface, you perform management tasks such as:

- Creating and removing clusters
- Adding and removing servers from a cluster
- Configuring load-balancing and high availability features
- Enabling administrator authentication privileges
- Configuring e-mail-based alarm notifications
- Monitoring clusters

Note: You can run the ClusterCATS Explorer from any server in the cluster, or you can run it remotely. This flexibility gives administrators in different geographic locations the ability to administer distributed clusters. You can also use ClusterCATS Explorer to administer UNIX clusters from a single Windows computer. You can view multiple clusters from a single Explorer.

The ClusterCATS Explorer presents a view of your cluster that is much like the view that the Windows Explorer presents of the files and directories that reside on a PC, as shown below.



The ClusterCATS Explorer interface includes four distinct areas:

- Menu bar — menu access to all ClusterCATS functionality
- Toolbar — shortcuts to the most frequently used ClusterCATS functions
- Left pane — views of cluster objects.
- Right pane — the folder and files for the object currently selected in the left pane

Each object in a ClusterCATS cluster configuration — clusters, servers, monitors, and probes — is represented by a unique icon. You can manipulate the icons in much the same manner as you expand and collapse directory trees in the Windows Explorer. For a list of which icons represent which objects in the ClusterCATS Explorer, click the Icon Legend button.

ClusterCATS Web Explorer (UNIX only)

You use the ClusterCATS Web Explorer (`btweb`) for administering UNIX-only clusters. It is a graphical, cross-platform, Web-based utility used to create, configure, and administer ClusterCATS clusters.

Note: ClusterCATS only installs ClusterCATS Web Explorer on UNIX servers, but you can access it from any computer with an Internet browser.

The Web Explorer, like its Windows counterpart, is quite robust and lets you configure and administer clusters easily. However, it does not contain the identical functionality provided by the Windows-based ClusterCATS Explorer. The Web Explorer does not let you do the following:

- Install ClusterCATS Web Explorer on an Windows server; it runs only from UNIX servers
- Create and administer Windows servers that have security enabled
- Set or modify load thresholds with a *graphical* display

- Monitor the load hitting the server via a *graphical* display; the server's load statistics are only displayed textually on the Cluster Member List and Server Properties pages
- Integrate ClusterCATS with Cisco LocalDirector

If you require any of these capabilities, you should obtain a Windows computer and use the Windows-based ClusterCATS Explorer for your cluster administration.

Configuring the communications port on your web server

Before you can open and use the ClusterCATS Web Explorer, you must ensure that a communications port is configured to listen for HTTP requests on the Netscape or Apache web server for which you installed ClusterCATS. You can access the ClusterCATS Web Explorer only through the defined communications port on your web server, which you configure using your web server's administration utilities.

Note: For availability and security reasons, be sure to allow access to the ClusterCATS Web Explorer only from a separate IP-based virtual host server on a port other than 80 and password protect access to it.

Netscape considerations

By default, Netscape Enterprise Server assigns your web server a random, six-digit communication port number. You can either use this assigned number or change it to something easier to remember, like port 81.

If you are not familiar with configuring your web server's communications ports, see the Netscape Enterprise Server Administrator online help for instructions.

Apache considerations

Make the following changes to the Apache Web Server's `httpd.conf` file to enable the ClusterCATS Web Explorer (btweb). Replace the IP address (192.168.96.71) and port (2222) specified in the example below with values appropriate for your system and enable authentication for the virtual directory.

```
### BTWeb Administration

Listen 192.168.96.71:2222
<VirtualHost 192.168.96.71:2222>
    ServerAdmin root@localhost
    DocumentRoot /usr/lib/btcats/btweb
    DirectoryIndex default.htm
    ServerName btweb
    ErrorLog logs/btweb_error_log
    CustomLog logs/btweb_access_log combined
    ### BTWeb stuff ###
    AddHandler cgi-script .exe
    <Directory "/usr/lib/btcats/btweb/">
        Options FollowSymLinks
        Options ExecCGI
        AllowOverride None
        Order allow,deny
        Allow from all
        AuthName "btcats admin tools"
        AuthType Basic
```

```

        AuthUserFile /usr/local/apache/conf/users
        require user admin
    </Directory>
</VirtualHost>

```

When you have configured your server, restart Apache. To access the Web Explorer, point your browser to the IP address you entered as the `VirtualHost`.

For information on using the `htpasswd` utility to create and manage your authentication file list, see the Apache documentation.

Opening the Web Explorer

The ClusterCATS Web Explorer can be used from a computer that runs either Netscape Navigator or Microsoft Internet Explorer versions 4.0 or greater.

To open the Web Explorer:

- 1 Open a web browser.
- 2 Enter the following URL in the browser's address field:

For Netscape Enterprise Server v3.x:

`http://<server-name>:<admin-port>/admin-serv/btweb/default.html`

For Netscape Enterprise Server v4.0x:

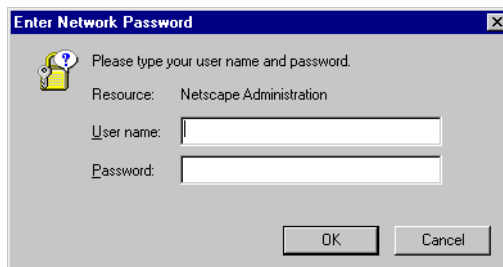
`http://<server-name>:<admin-port>/https-admserv/btweb/default.html`

For Apache:

`http://<virtual_host>:<admin-port>/default.html`

`servername` or `virtual_host` is the name of the web server on which you installed ClusterCATS, and `<admin-port>` is the communication port number on which the web server or virtual host is configured to listen for HTTP requests.

The Enter Network Password dialog box appears:



- 3 Enter your user name and password in the appropriate fields and click OK.

Note: The default user name and password is `admin`.

The ClusterCATS Web Explorer opens.

ClusterCATS Server Administrator

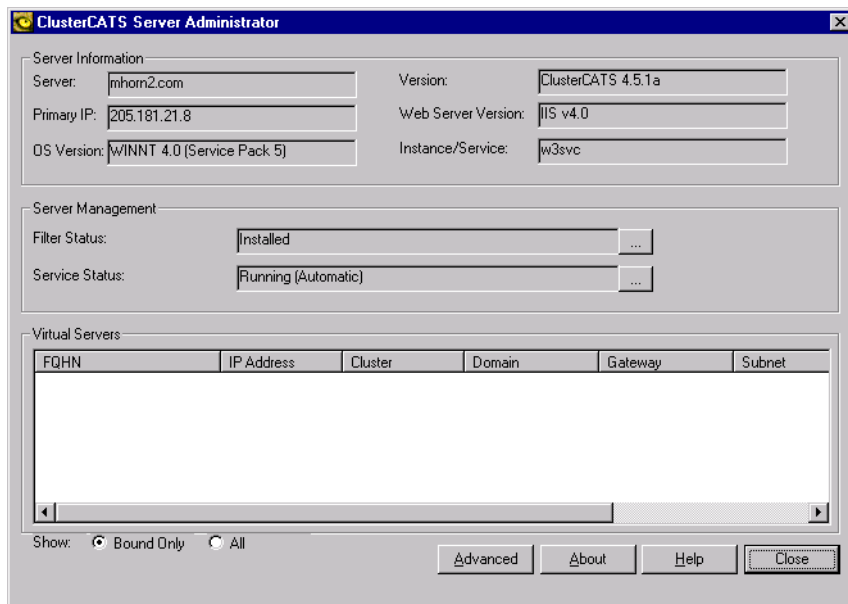
The ClusterCATS Server Administrator is a Windows-based utility that lets you perform server-specific maintenance activities for each server in a cluster. Unlike the ClusterCATS Explorer, which let you administer clusters from one central computer, you must run the ClusterCATS Server Administrator from each server in your cluster.

The Server Administrator lets you:

- Change installation settings
- Add and remove the ClusterCATS filter from the web server service
- Stop and start the ClusterCATS service
- Reset a clustered server's configuration to its preclustered state

The ClusterCATS Server Administrator lets you accomplish these tasks using an easy-to-use graphical user interface.

To open the ClusterCATS Server Administrator, select **Start > Programs > Macromedia > ClusterCATS Server Administrator**.



For more information on using the Server Administrator, see [Chapter 5](#).

btadmin

`btadmin` is a scriptable utility that lets you perform server-specific maintenance activities for each server in a cluster. `btadmin` is available on UNIX and Windows servers.

Unlike the ClusterCATS Web Explorer, which lets you administer your entire cluster from one central computer, you must use `btadmin` from each server in your cluster.

`btadmin` lets you:

- Add and remove the ClusterCATS filter from the web server service
- Stop and start the ClusterCATS service
- Place a cluster member in maintenance mode
- Reset a clustered server's configuration to its preclustered state

For more information, see [“Using btadmin” on page 122](#).

Creating clusters

If you have performed the tasks described in [“Before you install” on page 34](#) and you have successfully installed ClusterCATS, you are ready to create server clusters.

This section explains the following:

- [“Creating clusters in Windows” on page 54](#)
- [“Creating clusters in UNIX” on page 60](#)

Creating clusters in Windows

You can create clusters using the Cluster Setup Wizard or manually, using the ClusterCATS Explorer. It is easier and quicker to create and configure clusters completely with the Cluster Setup Wizard.

This section describes how to create clusters in both ways:

- [“Creating clusters with the Cluster Setup Wizard” on page 54](#)
- [“Manually creating clusters” on page 59](#)

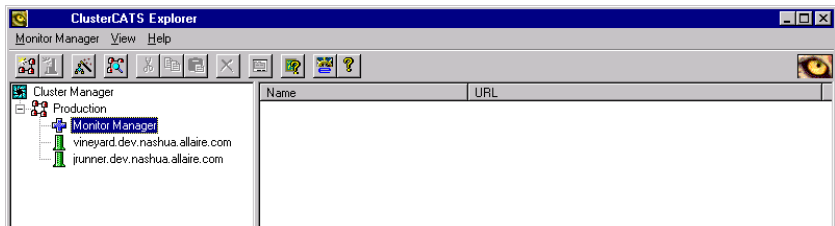
Creating clusters with the Cluster Setup Wizard

The ClusterCATS Explorer includes the Cluster Setup Wizard that makes creating and configuring clusters easy. The wizard steps you through the required definition and configuration steps. After creating a cluster with the wizard, you can use the ClusterCATS Explorer to make any necessary changes.

To create a server cluster using the Cluster Setup Wizard:

- 1 Select **Start > Programs > Macromedia > ClusterCATS Explorer**.

The ClusterCATS Explorer opens:

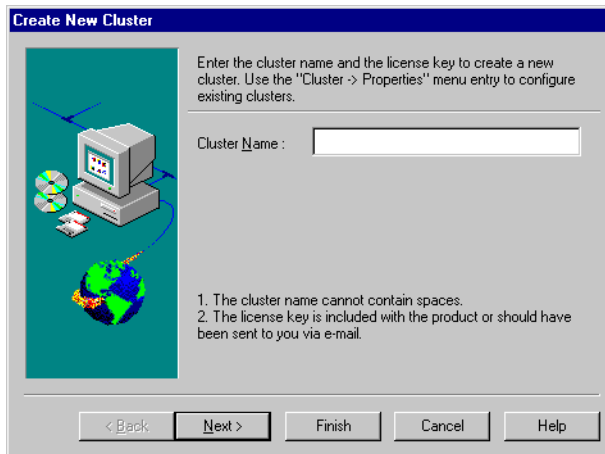


- 2 Select **Configure > Cluster Setup Wizard** or click the Cluster Setup Wizard icon



in the toolbar.

The Create New Cluster dialog box appears:



- 3 Enter a name for your cluster and click Next.

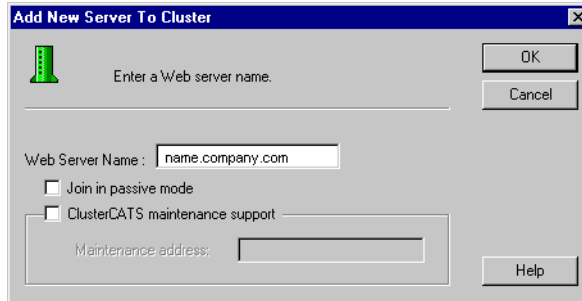
Make your cluster names logically consistent with their purpose. For example, Sales Web, Customer Support Web, and so on.

The List of Web Servers in the Cluster dialog box appears:



- 4 Click Add to add available web servers to your cluster.

The Add New Server to Cluster dialog box appears:



The "Add New Server To Cluster" dialog box has a title bar with a close button. It contains a green server icon and the text "Enter a Web server name." Below this is a text field with "name.company.com" entered. There are two checkboxes: "Join in passive mode" and "ClusterCATS maintenance support", both of which are unchecked. Below the checkboxes is a text field for "Maintenance address:". At the bottom right are buttons for "OK", "Cancel", and "Help".

- 5 Enter the fully qualified host name of a web server in the New Web Server Name field (for example, doc.macromedia.com).

- 6 If you use the ClusterCATS dynamic IP addressing scheme and the maintenance IP address is not bound to your NIC, select ClusterCATS Maintenance Support.

If you are *not* configuring this server for offline maintenance support, go to step 8.

Note: You can set the maintenance support option only when creating a cluster or adding a cluster member to a cluster. You cannot configure or modify this option after you have created and added the cluster member to the cluster.

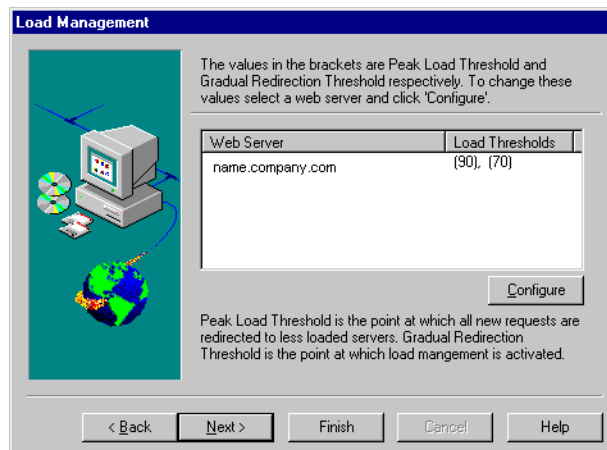
Enabling maintenance support for clusters requires that you configure your cluster for ClusterCATS dynamic IP addressing. For more information, see [“ClusterCATS dynamic IP addressing \(Windows only\)” on page 132](#).

- 7 Enter the fully qualified host name of the maintenance address (for example, serv1.yourcompany.com) in the Maintenance Address field.

- 8 Click OK.

- 9 Repeat steps 4 through 8 for each web server you want to add to the cluster, and then click Next to proceed.

The Load Management dialog box appears:

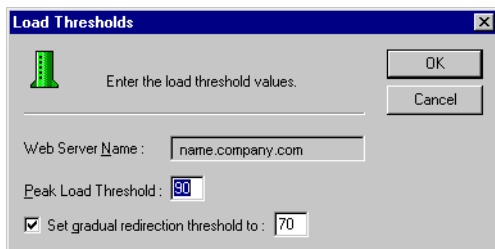


The "Load Management" dialog box has a title bar. On the left is a graphic showing a computer, a CD, and a globe. On the right, there is text explaining thresholds: "The values in the brackets are Peak Load Threshold and Gradual Redirection Threshold respectively. To change these values select a web server and click 'Configure'." Below this is a table with two columns: "Web Server" and "Load Thresholds". The table contains one row with "name.company.com" and "(90), (70)". Below the table is a "Configure" button. At the bottom, there is explanatory text: "Peak Load Threshold is the point at which all new requests are redirected to less loaded servers. Gradual Redirection Threshold is the point at which load management is activated." At the very bottom are buttons for "< Back", "Next >", "Finish", "Cancel", and "Help".

Web Server	Load Thresholds
name.company.com	(90), (70)

- 10 To use the default load threshold settings, click Next and go to step 13. If you do not want to use the defaults, select the server and click Configure to configure new peak and gradual redirect load thresholds for that cluster member.

The Load Thresholds dialog box appears:

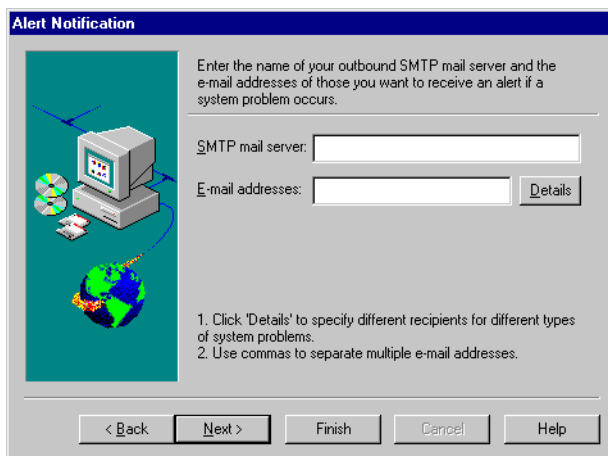
The 'Load Thresholds' dialog box has a blue title bar with a close button. It contains a green server icon and the text 'Enter the load threshold values.' with 'OK' and 'Cancel' buttons. Below is a 'Web Server Name' field with 'name.company.com'. The 'Peak Load Threshold' is set to '80'. A checked checkbox 'Set gradual redirection threshold to:' is followed by a field set to '70'.

- 11 Enter numerical values (not higher than 100%) in the Peak Load Threshold and Gradual Redirect fields and click OK.

Set the peak load threshold below 100%, to accommodate the server's processing needs. Set the gradual redirection threshold lower than the peak threshold.

- 12 Click Next.

The Alert Notification dialog box appears:

The 'Alert Notification' dialog box has a blue title bar. On the left is a graphic of a computer, CD, and globe. The main text says 'Enter the name of your outbound SMTP mail server and the e-mail addresses of those you want to receive an alert if a system problem occurs.' Below are fields for 'SMTP mail server:' and 'E-mail addresses:', with a 'Details' button next to the latter. At the bottom are instructions: '1. Click 'Details' to specify different recipients for different types of system problems.' and '2. Use commas to separate multiple e-mail addresses.' Navigation buttons at the bottom include '< Back', 'Next >', 'Finish', 'Cancel', and 'Help'.

- 13 Enter the name of your outbound SMTP mail server in the SMTP mail server field and the e-mail address for a recipient of cluster alerts in the E-mail addresses field. If multiple people will receive different alerts for different types of notification events, go to step 14. Otherwise, click Next and proceed to step 16.

- 14 To configure different types of alerts to go to different people, click Details in the Alert Notification dialog box.

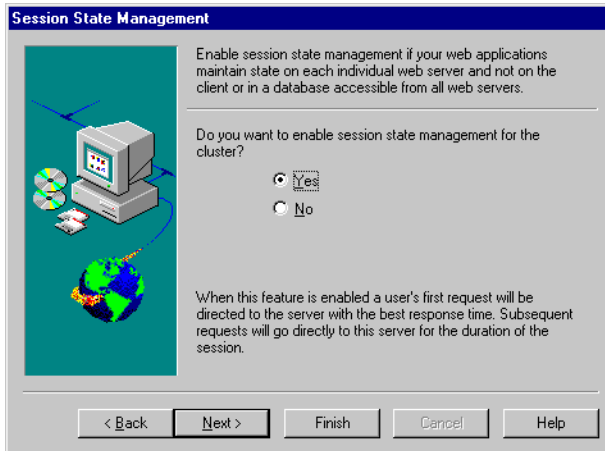
The Alarm Notification dialog box appears.

- 15 Select an alert event and enter the e-mail address of the recipient.

If you want one person to receive the majority of alerts, click Propagate to automatically fill each event's Recipient column with the same e-mail address. Then

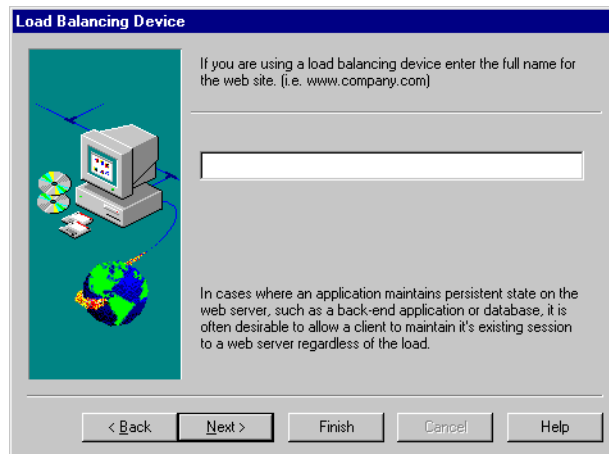
manually change the few recipients that are different. If there are multiple recipients for one alert event, separate e-mail address entries with commas. Click OK to return to the Alarm Notifications dialog box and then click Next to proceed.

The Session State Management dialog box appears:



- 16 If your server cluster supports a site that must maintain persistent state on the same web server during a user session, select Yes to enable session-aware load balancing. Otherwise, select No and click Next.

The Load Balancing Device dialog box appears:



- 17 If you use a hardware-based load-balancing device in addition to ClusterCATS to manage and distribute load, enter the name of the website that this device supports (for example, www.yourcompany.com) and click Next.
- 18 Click Finish.

ClusterCATS creates the cluster you configured and displays it in the ClusterCATS Explorer's left pane.

Manually creating clusters

If you do not want to create your clusters using the Cluster Setup Wizard, you can create them manually. If you manually create clusters, you must then add each cluster member to a cluster, using the ClusterCATS Explorer.

To manually add cluster members to a cluster, see [“Adding cluster members” on page 63](#).

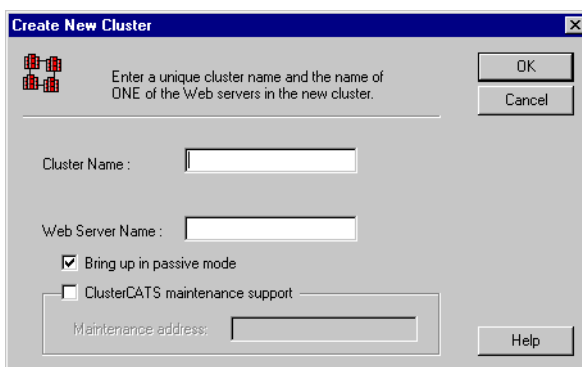
To manually create clusters:

- 1 Select **Start > Programs > Macromedia > ClusterCATS Explorer**.

The ClusterCATS Explorer opens.

- 2 Select **Cluster Manager > New Cluster**, or right-click the Cluster Manager icon and select New Cluster, or click the New Cluster button in the toolbar.

The Create New Cluster dialog box appears:



- 3 Add a new cluster using the fields as described in the following table:

Field	Description
Cluster Name	Enter a unique name for the cluster. Make cluster names logically consistent with their purpose. For example, Sales Web, or Customer Support Web.
Web Server Name	Enter the fully qualified host name (for example, doc.macromedia.com) for the first server you want to be a member of this cluster. You cannot create an empty cluster; you must specify a web server that will be part of the cluster. The first server that you add to a cluster is known as the Admin Manager. The remaining steps guide you in configuring the Admin Manager.
Bring up in passive mode	Select this checkbox to bring the Admin Manager up in Passive mode. If you do not select this checkbox, the server will be brought up in Active mode. For more information, see “Changing active/passive settings” on page 111 .

Field	Description
ClusterCATS maintenance support	<p>Select the ClusterCATS Maintenance Support check box to enable support for offline maintenance. The Admin Manager must be configured with a maintenance IP address.</p> <p>Using maintenance support requires that your cluster support ClusterCATS dynamic IP addressing. For more information, see “ClusterCATS dynamic IP addressing (Windows only)” on page 132.</p> <p>Offline maintenance support is available only on Windows NT server clusters. You can set the maintenance support option only when creating a cluster or adding a cluster member to a cluster. You cannot configure or modify this option after you create and added a cluster member to a cluster.</p>
Maintenance address	<p>Enter the fully qualified host name of the maintenance address (for example, serv1.yourcompany.com). This field is accessible only if you selected ClusterCATS Maintenance Support.</p>

- 4 Click OK.

The cluster appears below the Cluster Manager icon in the ClusterCATS Explorer left pane.

To manually add additional cluster members to your new cluster, see [“Adding cluster members” on page 63](#).

Creating clusters in UNIX

- 1 In the ClusterCATS Web Explorer, click the Create New Cluster link.

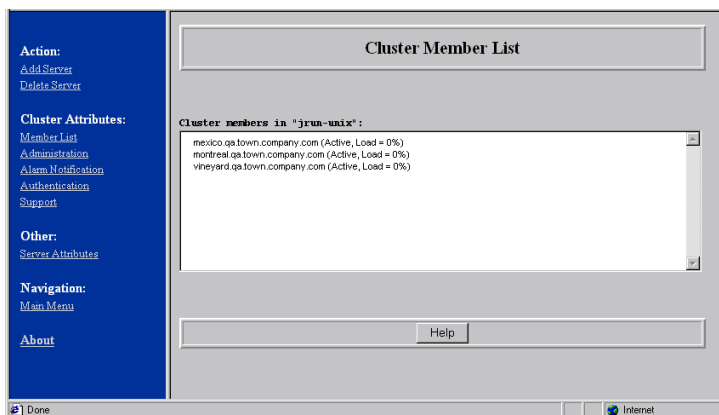
The Create New Cluster page appears:

2 Add a cluster using the fields as described in the following table:

Field	Description
Cluster Name	<p>Enter a unique name for the cluster.</p> <p>Make cluster names logically consistent with their purpose – for example, Sales Web, or Customer Support Web.</p>
Web Server Name	<p>Enter the fully qualified host name (for example, doc.macromedia.com) for the first server you want to be a member of this cluster.</p> <p>You cannot create an empty cluster; you must specify a web server that will be part of the cluster. The first server that you add to a cluster is known as the Admin Manager.</p> <p>You cannot create an empty cluster; you must specify a web server that will be part of the cluster.</p>

3 Click OK.

ClusterCATS creates the cluster and displays its members on the Cluster Member List page, as in the following figure:



Removing clusters

To delete a cluster, you must delete each member from the cluster individually, using the procedure described in [“Removing cluster members” on page 65](#).

Note: When deleting cluster members, you must delete the Admin Manager (Windows) or the Admin Agent (UNIX) last. This server is the first server you added to the cluster.

When the last cluster member has been removed, the cluster itself is deleted.

To determine which server is the Admin Manager in Windows:

- 1 Open the ClusterCATS Explorer.
- 2 Right-click on the cluster icon and select **Configure > Administration**.
The cluster's Properties dialog box opens displaying the Administration tab. The server designated as the Admin Manager is the active entry in the drop-down list.

To determine which server is the Admin Agent in UNIX:

- 1 In the ClusterCATS Web Explorer, click the Show Cluster link.
- 2 Enter the fully qualified host name of a server in the Web Server Name field.
- 3 Click OK.
The Cluster Member List page appears. If you get an "Error: Server <cluster_member_name> could not be found", ensure that you used the correct, fully qualified server name and that the server is running.
- 4 Click the Administration link. The Cluster Administration page appears. The Admin Agent is the currently selected host in the Admin Agent field.

Adding cluster members


You can add servers to a cluster at any time. This section describes the following:

- [“Adding cluster members in Windows” on page 63](#)
- [“Adding cluster members in UNIX” on page 64](#)

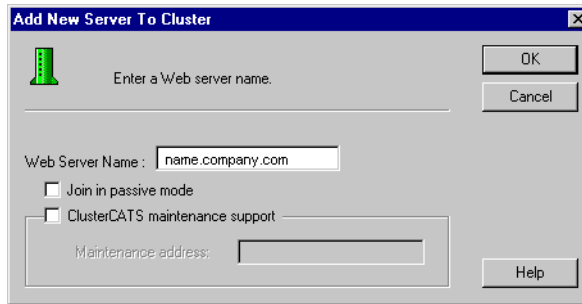
Adding cluster members in Windows

Use the ClusterCATS Explorer to add servers to a cluster. If you used the Cluster Setup Wizard (Windows only) to create a cluster and populate it with cluster members, you can also add clusters using the following procedure.

To add an additional cluster member to a cluster:

- 1 Open the ClusterCATS Explorer and select a cluster.
- 2 Select **Cluster > New > Cluster Member**. Alternatively, you can click the Add button  or right-click the cluster icon and select **New > Cluster Member**.

The Add New Server to Cluster dialog box appears:



- 3 In the Web Server Name field, enter the fully qualified host name of the web server (for example, doc.macromedia.com).
- 4 If you use the ClusterCATS dynamic IP addressing scheme and the maintenance IP address is not bound to your NIC, select ClusterCATS Maintenance Support. If you are *not* configuring this web server for offline maintenance support, go to step 6.

Note: You can set the maintenance support option only when creating a cluster or adding a cluster member to a cluster. You cannot configure or modify this option after you have created and added the cluster member to the cluster.

Enabling maintenance support for clusters requires that you configure your cluster for ClusterCATS dynamic IP addressing. For more information, see [“ClusterCATS dynamic IP addressing \(Windows only\)” on page 132](#).

- 5 Enter the fully qualified host name of the maintenance address (for example, serv1.yourcompany.com) in the Maintenance Address field.
- 6 Click OK.
- 7 Repeat steps 2 through 6 to add additional servers to the cluster manually.

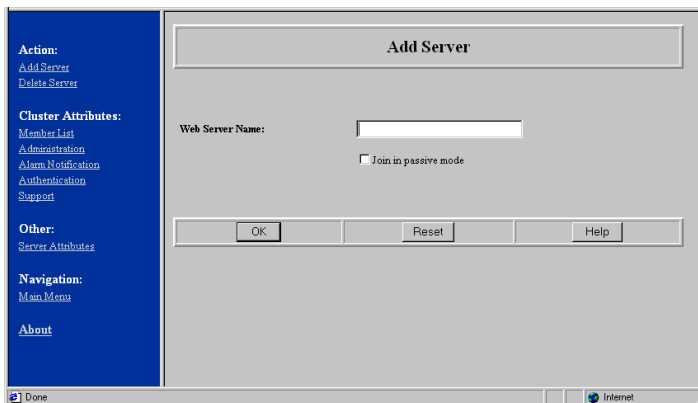
Adding cluster members in UNIX

Use the ClusterCATS Web Explorer to add cluster members.

To add a cluster member to a cluster:

- 1 Open the ClusterCATS Web Explorer if it is not already open.
- 2 Click the Add Server link.

The Add Server page appears:



The screenshot shows a web browser window displaying the 'Add Server' page. On the left is a blue navigation sidebar with the following links: **Action:** [Add Server](#), [Delete Server](#); **Cluster Attributes:** [Member List](#), [Administration](#), [Alarm Notification](#), [Authentication](#), [Support](#); **Other:** [Server Attributes](#); **Navigation:** [Main Menu](#), [About](#). The main content area is titled 'Add Server' and contains a 'Web Server Name:' label followed by a text input field. Below the input field is a checkbox labeled 'Join in passive mode'. At the bottom of the main area are three buttons: 'OK', 'Reset', and 'Help'. The browser's status bar at the bottom shows 'Done' and 'Internet'.

- 3 Enter the fully qualified host name (for example, doc.macromedia.com) in the Web Server Name field.
- 4 Click OK to add the cluster member to the existing cluster.

Removing cluster members

You can remove servers from a cluster at any time. This section describes the following:

- “Removing cluster members in Windows” on page 65
- “Removing cluster members in UNIX” on page 65

Removing cluster members in Windows

Use the ClusterCATS Explorer to remove cluster members.

To remove a cluster member from a cluster:

- 1 Open the ClusterCATS Explorer and select a cluster member.
- 2 Select **Server > Delete** or right-click the server name and select Delete.

The selected cluster member is deleted from the cluster you selected.

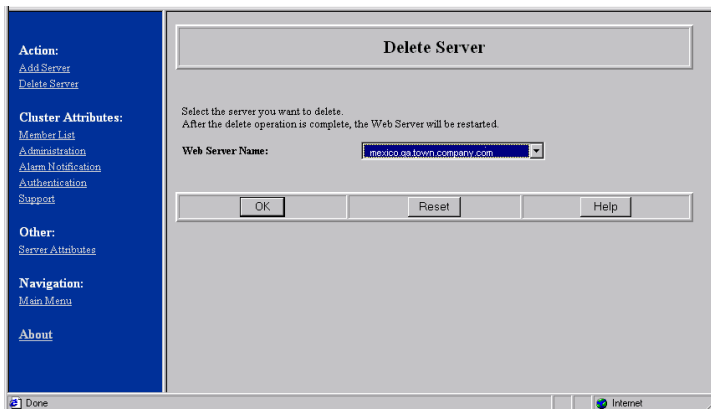
Removing cluster members in UNIX

Use the ClusterCATS Web Explorer to remove cluster members.

To remove a cluster member from a cluster:

- 1 In the ClusterCATS Web Explorer, click the Delete Server link.

The Delete Server page appears:



- 2 Select a cluster member to delete from the Web Server Name drop-down box.
A message appears, telling you that the selected server has been deleted.

Note: If you delete the last cluster member in a cluster, the cluster is also deleted and you are returned to the default page of the ClusterCATS Web Explorer.

- 3 Click OK.

Server load thresholds

ClusterCATS ensures that your web applications remain available and running at optimum performance by intelligently managing the HTTP traffic hitting your clustered servers. By setting load thresholds on each server in your cluster, you can control and manage your site's availability and performance. Many of your threshold configuration decisions hinge on your site's architecture and where the bulk of your processing resources must be allocated.

During an HTTP redirection, ClusterCATS evaluates the cluster's state according to the HTTP server state first, and then the JRun/ColdFusion server load. This policy is the same in centralized and distributed ClusterCATS configurations. In a centralized ClusterCATS cluster with all web servers at one site, ClusterCATS redirects only if the server is busy or restricted.

For each cluster member, you configure two load thresholds:

- Peak load threshold — the maximum load the server can handle before its performance degrades significantly or becomes unavailable.
- Gradual redirection threshold — the point at which HTTP requests begin to be redirected to other less-loaded members in a cluster so the server's performance does not degrade or become unavailable.

By default, the peak load threshold is 90% and the gradual redirection threshold is 10%. These default settings adequately handle HTTP traffic going across most websites. However, if your website is particularly processing intensive, you should lower both threshold settings to better accommodate the increased load.

If you want the server to handle as much load as possible, set the threshold values close to one another. However, if you want redirection to occur well in advance of the server nearing its peak threshold, set the values farther apart so there is a difference of at least 10% between the two threshold values.

This section shows you how to set the peak and gradual redirection load thresholds for ClusterCATS Servers in the following sections:

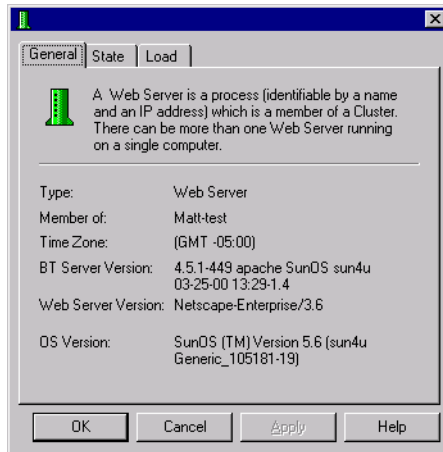
- [“Configuring load thresholds in Windows” on page 66](#)
- [“Configuring load thresholds on UNIX” on page 69](#)

Configuring load thresholds in Windows

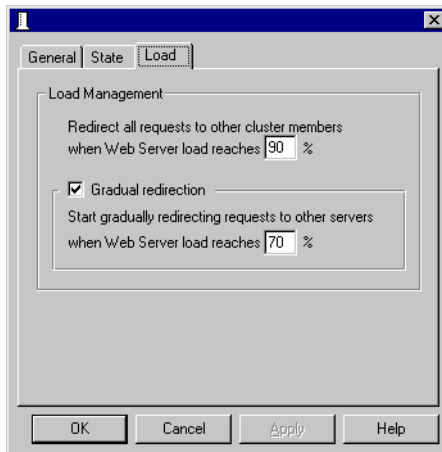
To adjust load thresholds for a cluster member:

- 1 Open the ClusterCATS Explorer and select a server.
- 2 Select **Server > Properties** or right-click the server and select Properties.

The server's Properties dialog box appears:



3 Click the Load tab.



- 4 Enter a numeric value (less than 100%) in the first Load Management field. This is referred to as the peak load threshold. In the example above, the peak load threshold is set to 90.
- 5 Enable the Gradual Redirection check box.
- 6 Enter a new value in the Gradual Redirection field. This value must be lower than the peak load threshold.
- 7 Click OK to apply your new threshold settings.

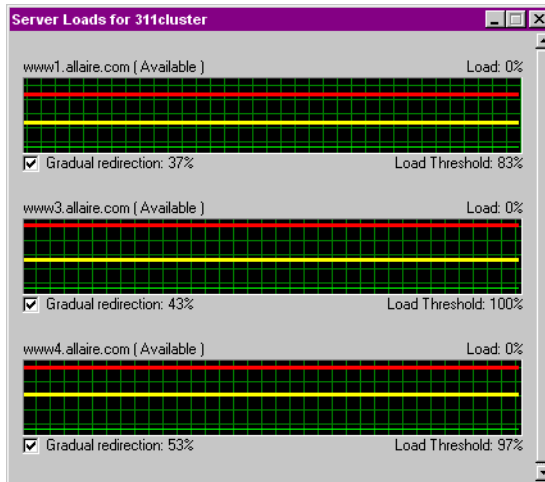
Viewing a cluster's load status

JRun/ColdFusion reports its load data directly to ClusterCATS. You can view the load on the servers at any time using the Server Load Monitor.

To view your cluster's current load levels:

- 1 Open the ClusterCATS Explorer and select a cluster.
- 2 Select **Monitor > Load** or right-click the cluster you have selected and select **Monitor > Load**.

The Server Loads dialog box appears, showing the current load status for each cluster member in the cluster that you selected:



The load monitor shows three lines:

- Top line (red): Peak load threshold
- Middle line (yellow): Gradual Redirection load threshold
- Bottom line (green): JRun/ColdFusion server load

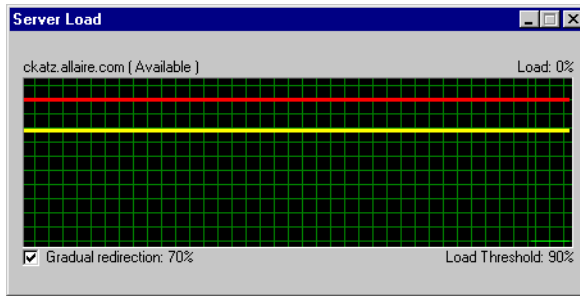
Adjusting load threshold settings graphically

You can view and set threshold settings of a cluster member using the Server Load Monitor's visual display. To set or change threshold settings, use the mouse to drag the Peak (red) and Gradual Redirection (yellow) threshold lines to their settings instead of entering numeric values in fields

To configure load threshold settings using the Server Load dialog box:

- 1 Open the ClusterCATS Explorer and select a server.
- 2 Select **Monitor > Load** or right-click the server and select **Monitor > Load**.

The Server Load dialog box appears:



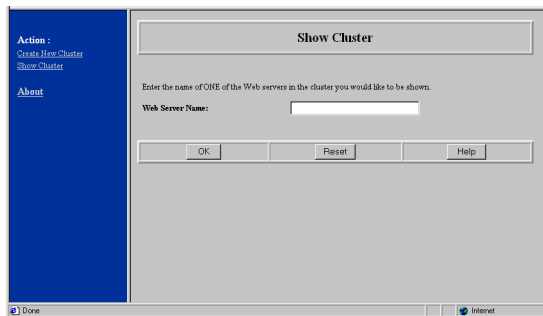
- 3 Use your mouse to drag the peak load threshold (red) up or down.
As you move the line, the peak load threshold percentage changes.
- 4 Enable gradual redirection by selecting the Gradual Redirection check box.
- 5 Drag the Gradual Redirection load threshold (yellow) to adjust it accordingly.
- 6 Close the dialog box to apply the load threshold settings you configured.

Configuring load thresholds on UNIX

To configure load thresholds for a cluster member:

- 1 In the ClusterCATS Web Explorer, click the Show Cluster link.

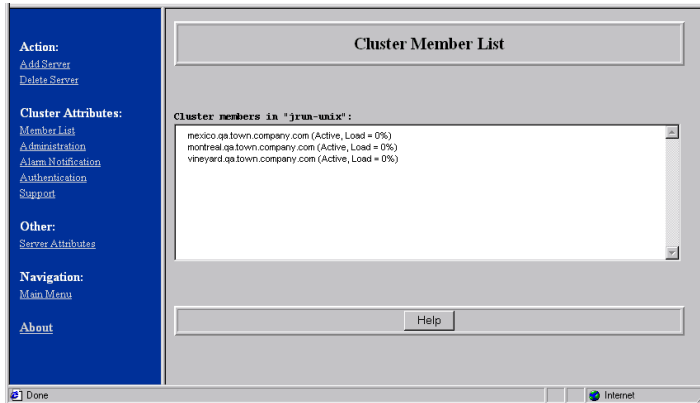
The Show Cluster page appears:



- 2 Enter the fully qualified host name of a server in the Web Server Name field.

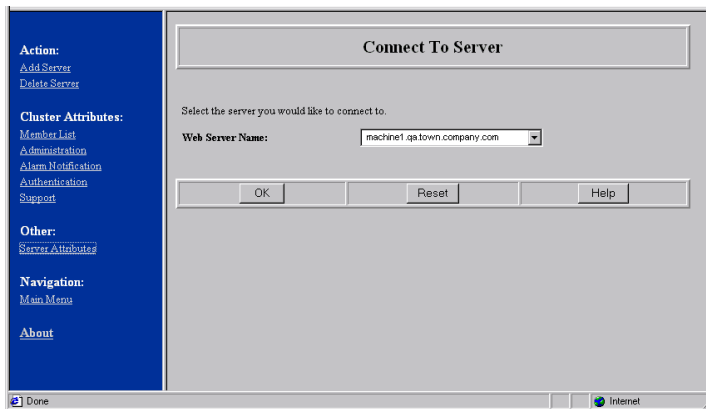
- 3 Click OK.

The Cluster Member List page appears. If you get an "Error: Server <cluster_member_name> could not be found", ensure that you used the correct, fully qualified server name and that the server is running.



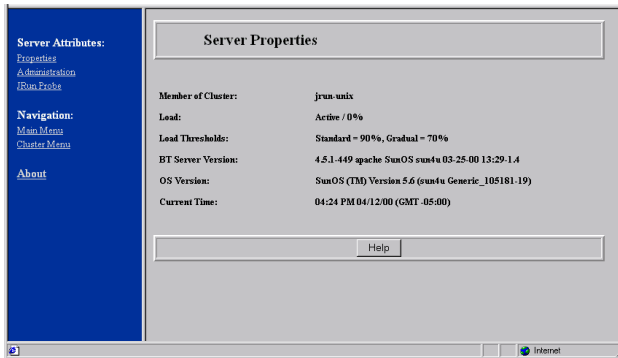
- 4 Click the Server Attributes link.

The Connect To Server page appears:



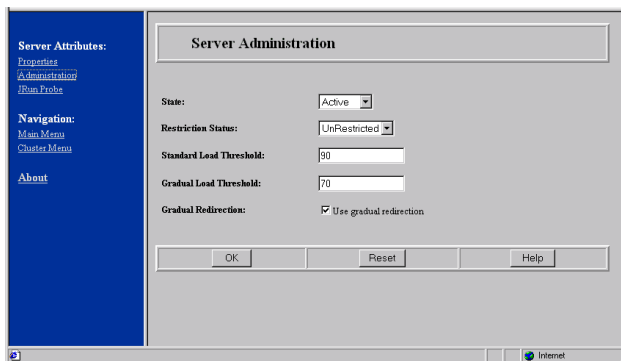
- 5 Select a server to connect to from the Web Server Name list box.
- 6 Click OK.

The selected server's Server Properties page appears:



- 7 Click the Administration link under Server Attributes.

The Server Administration page appears for the selected server:



- 8 To change the peak load threshold, enter a new numeric value (less than 100%) in the Standard Load Threshold field.
- 9 Enable the Gradual Redirection check box if it is not already enabled.
- 10 To change the Gradual Redirection load threshold, enter a new numeric value in the Gradual Load Threshold field. This value must be lower than the standard load threshold.
- 11 Click OK to apply your new load threshold settings.

Session-aware load balancing

Managing a web application's state in a clustered environment can be challenging. By default, web application, session, and server variables that are stored in memory or a repository during a user session do not persist during a server redirection. Consequently, the web server cannot maintain the application's state correctly.

To overcome this problem, ClusterCATS provides a session-aware load-balancing feature that lets you maintain application state in a clustered environment.

One way to maintain your web application's state is to create session variables that are stored on the web server. For an e-commerce website that is clustered, it is vital that users do not get redirected to another server in the middle of their session. If they did, their online transactions would be interrupted, making for an unsuccessful and frustrating user experience.

To ensure that users are not redirected from the server on which they start their session, ClusterCATS provides a built-in feature for enabling session-aware load balancing. Sometimes referred to as a "sticky" server, session-aware load balancing guarantees that users will not get bumped from the server on which they start their session until the session is complete, regardless of the load thresholds that have been defined for that server.

Note: Session-aware load balancing may not work if you use absolute hyperlinks in your web pages. Absolute links route the HTTP request back to the cluster entry point and redirect according to the current load threshold without regard to the state of the requesting client. To avoid inadvertent loss of state, use only relative linking in your web pages.

This section describes the following:

- ["Enabling session-aware load balancing on Windows" on page 72](#)
- ["Enabling session-aware load balancing on UNIX" on page 72](#)

Enabling session-aware load balancing on Windows

To enable session-aware load balancing:

- 1 Open the ClusterCATS Explorer and select a cluster.
- 2 Select **Configure > Administration** or right-click the cluster and select **Configure > Administration**.
The cluster Properties dialog box appears.
- 3 Select the Session State Management check box.
- 4 Click OK.

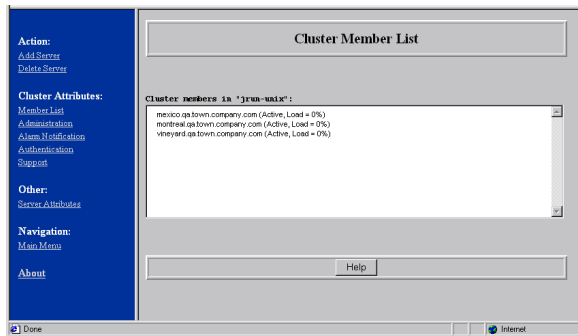
Enabling session-aware load balancing on UNIX

To enable session-aware load balancing:

- 1 In ClusterCATS Web Explorer, click the Show Cluster link.
- 2 Enter the fully qualified host name of a server for which you want to configure session-aware load balancing in the Web Server Name field.

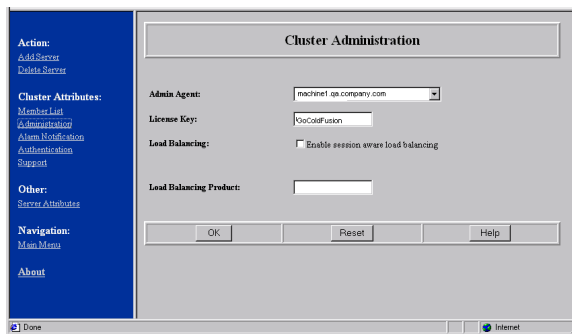
3 Click OK.

The Cluster Member List page appears:



4 Click the Administration link under Cluster Attributes.

The Cluster Administration page appears:



5 Select the Enable session-aware load balancing check box.

6 Click OK to enable session-aware load balancing for the selected cluster.

Persistent session failover in JRun

JRun can be configured to enable session persistence, meaning that all session data is saved (persisted) upon the completion of every request. When a server that is servicing a client's session goes down, the client's active session data can be retrieved intact from a common data store (such as a JDBC database) by another server. When the client attempts to continue its active session and presents its session ID to the replacement server, its session data is restored from the repository, completing the session failover. This feature is called **session swapping**. The client's session state is effectively swapped from one server to another when the first fails.

Session swapping overview

The following are required to use failover for persistent sessions with ClusterCATS:

- Session swapping must be enabled in JRun and ClusterCATS. For more information, see [“Configuring JRun for session swapping” on page 74](#) and [“Configuring ClusterCATS for session swapping” on page 74](#), respectively.
- Session-aware load balancing must be enabled. Because only one server can be responsible for a session at a time, JRun session swapping must be used in conjunction with the ClusterCATS *session-aware load-balancing* feature. This ensures that multiple servers do not have concurrent access to the same session data. For more information, see [“Session-aware load balancing” on page 72](#).
- A repository used for persistent session data must be shared among the JRun servers in a cluster. For information, see [“Using shared files for session swapping” on page 75](#). For an example of using JDBC to connect to a shared repository of session information, see [“Using JDBC for session swapping” on page 76](#).
- Cookies must have domain scope for proper session swapping. For more information, see [“Configuring JRun for session swapping” on page 74](#).

Configuring JRun for session swapping

To enable session swapping in JRun, the following properties must be set in the JRun server's `local.properties` file:

```
session.swapping=true  
session.maxresident=0
```

The `local.properties` file must enable domain scope for cookies by including the following property:

```
session.cookie.domain=yourdomain.com
```

The repository used for session swapping can be a shared file or a shared JDBC database. For information, see [“Using shared files for session swapping” on page 75](#) and [“Using JDBC for session swapping” on page 76](#).

Configuring ClusterCATS for session swapping

ClusterCATS must be configured to allow session swapping to function properly. The following platform-specific procedures explain how to enable session swapping in ClusterCATS.

To enable session swapping on Windows:

- 1 Edit the registry (using regedit) and open the following key:
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\BrightTiger\Parameters
- 2 Add the following REG_DWORD value:
SessionSwapping 1
- 3 Close the registry editor.
- 4 Repeat this procedure for every server in the cluster.

To enable session swapping on UNIX:

- 1 Log in as the super user (root).
- 2 Stop ClusterCATS by issuing the following command:
/usr/lib/btcats/btadmin stop all
- 3 Edit the file /usr/lib/btcats/database/bt.registry with a text editor.
- 4 Search for the following string:
hkey_local_machine\system\currentcontrolset\services\brighttiger\
parameters:5
Under the following entry:
Advertise: 0x2; REG_DWORD
Add the following line:
SessionSwapping: 0x1 ; REG_DWORD
- 5 Save the file and exit your text editor.
- 6 Restart ClusterCATS with the following command:
/usr/lib/btcats/btadmin start all
- 7 Repeat this procedure for every server in the cluster.

Using shared files for session swapping

To use file swapping, the JRun server's `local.properties` file should contain the following properties:

```
session.persistence.service=file
session.persistence.file.class=allaire.jrun.session.FileSessionStorage
# See the following paragraph for more on this property.
session.persistence.file.path=/mnt/myothermachine/sessionpool
```

The `session.persistence.file.path` property must specify a shared path that all computers can read and write to. For example, on UNIX, you must have `server1` export `/sessionpool` and `server1`'s `file.path=/sessionpool`. Now `server2` should mount `server1:/sessionpool` to some mount point — for example, `/mnt/sessionpool` — and set `server2`'s `file.path=/mnt/sessionpool`. This assumes that `server2` has write permission.

Note: If JRun runs as an NT service, and you use a mapped drive, JRun does not have permissions to write to the drive. To correct this problem, edit the properties for the JRun Service and change the user account for the service to be a user with the necessary privileges to write to the mapped drive.

Using JDBC for session swapping

To use JDBC for session swapping, the JRun server's `local.properties` file should contain the following properties:

```
session.persistence.service=jdbc
session.persistence.jdbc.class=allaire.jrun.session.JDBCSessionStorage
session.persistence.jdbc.JDBCDriver=sun.jdbc.odbc.JdbcOdbcDriver
session.persistence.jdbc.JDBCConnectionURL=jdbc:odbc:JRunSessions
session.persistence.jdbc.JDBCSessionTable=sessions
session.persistence.jdbc.JDBCSessionIDColumn=id
session.persistence.jdbc.JDBCSessionDataColumn=data
```

JDBC swapping requires that you have a valid JDBC driver that can successfully connect to the database. You must create a table in your database with an *id* column and a *data* column. This example uses a table named *sessions*, an *IDColumn* named *id*, and a *DataColumn* named *data*. Define the *id* column as `varchar(255)` and the data column as binary data.

Using ColdFusion probes

ClusterCATS provides load-balancing and failover support for your web applications in two ways. First, it automatically interprets and reacts to the load metric that the ColdFusion server generates. Second, ClusterCATS lets you create web application monitors. These monitors can have multiple probes that periodically test the health and operation of the websites that the servers process.

Note: Multiple probes are allowed per web server, and web applications can be restarted individually. However, each web application should have only one probe that restarts on a failure.

The probe is a high-availability feature that verifies that ColdFusion servers are running properly on clustered servers. It periodically tests specific URLs at specified intervals and verifies their validity against user-defined strings contained in the returned pages.

If the validation test succeeds, inbound HTTP requests continue to be sent to the server for which the probe exists. However, if a test fails (the URL fails, times out, or does not return the user-specified string in the page accessed), ClusterCATS restricts that server and redirects requests to other available servers in the cluster. ClusterCATS continues to test the restricted server; when the probe returns a valid value, the server is considered available.

If a ColdFusion server hangs or fails, ClusterCATS attempts to recover the failed service. When the service is recovered, the probe can restart the server and begin sending HTTP traffic to it again.

This section describes the following:

- [“Configuring ColdFusion probes in Windows” on page 77](#)
- [“Configuring ColdFusion probes in UNIX” on page 81](#)

Configuring ColdFusion probes in Windows

This section describes the following:

- [“Adding ColdFusion probes” on page 77](#)
- [“Removing ColdFusion probes” on page 81](#)

Adding ColdFusion probes

ClusterCATS lets you set up one probe monitor for each server in the cluster. Each monitor can have multiple probes associated with it. As a result, clusters will typically have multiple probe monitors (one for each server), and each monitor can have one or more probes.

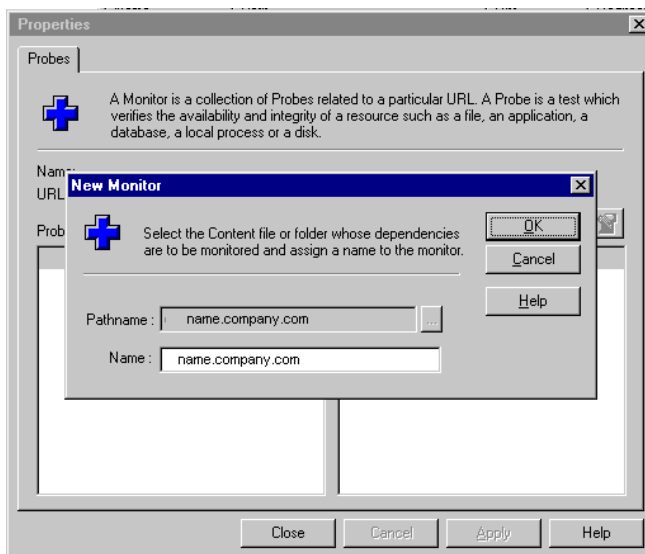
The procedure for adding a *new* monitor and probe is different from adding a probe to a server that already has a probe monitor. This section describes how to perform both activities.

Note: The ColdFusion service must be running on your server to add a probe.

To add a new monitor and ColdFusion probe:

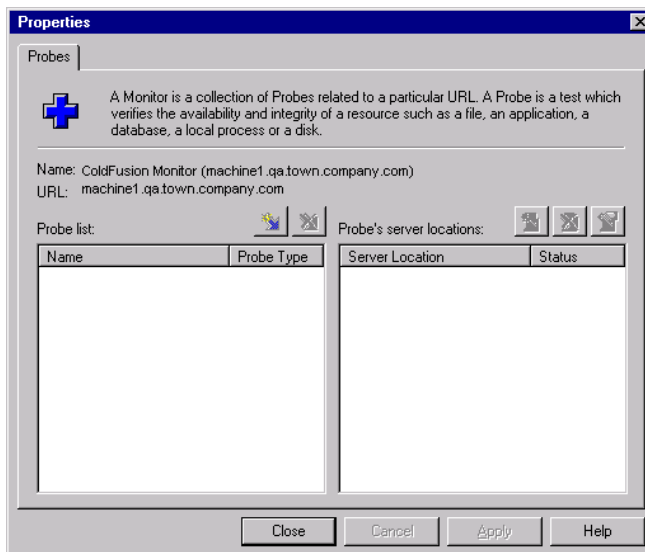
- 1 Open the ClusterCATS Explorer and select a server.
- 2 Select **Server > New Monitor**. Alternatively, you can right-click the server and select New Monitor.

The New Monitor dialog box appears:



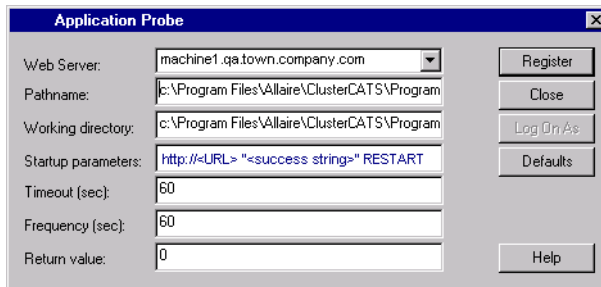
- 3 Enter a name to assign to this probe's monitor in the Name field and click OK.

The monitor's Properties dialog box appears:



- 4 Click the New Probe button .

The ColdFusion Web Application Probe settings dialog box appears:

The image shows the 'Application Probe' dialog box. It has a title bar with a close button. The dialog contains several input fields: 'Web Server' (a dropdown menu showing 'machine1.qa.town.company.com'), 'Pathname' (a text box with 'c:\Program Files\Allaire\ClusterCATS\Program'), 'Working directory' (a text box with 'c:\Program Files\Allaire\ClusterCATS\Program'), 'Startup parameters' (a text box with 'http://<URL> "<success strings>" RESTART'), 'Timeout (sec)' (a text box with '60'), 'Frequency (sec)' (a text box with '60'), and 'Return value' (a text box with '0'). On the right side, there are five buttons: 'Register', 'Close', 'Log On As...', 'Defaults', and 'Help'.

- 5 Configure the application probe settings as described in the following table:

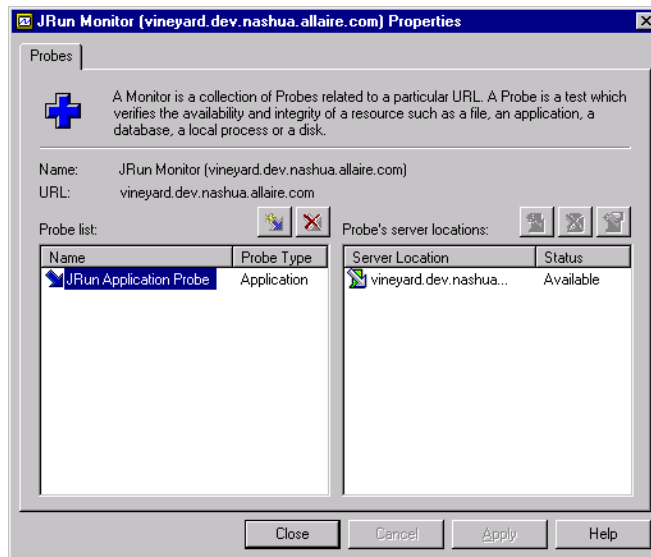
Field	Description
Web Server	Select the name of the server from the drop-down list.
Pathname	Enter the absolute path to the ColdFusion probe. Do not change the default selection unless you installed ColdFusion to a directory other than the default installation directory.
Working directory	Enter the absolute path to the probe's working directory. Do not change the default selection unless you installed ColdFusion to a directory other than the default installation directory.
Startup parameters	<p>Replace the <URL> with the actual URL of the site you want the probe to access, and replace <success string> with a text string that appears on a page on the site you are probing.</p> <p>Tips:</p> <ul style="list-style-type: none">• Be sure to include a space between the URL and the success string that you specify. The success string must be enclosed in quotation marks.• Do not modify the RESTART explicit parameter if you want the probe to automatically restart the ColdFusion Server upon detecting a failure. However, if you do not want ClusterCATS to automatically restart the ColdFusion Server upon detecting a failure, replace RESTART with NORESTART.
Timeout (sec)	<p>Enter a time, in seconds, to indicate how long ClusterCATS should wait before a ColdFusion server failure is registered.</p> <p>Do not set this value to less than 60 seconds because ClusterCATS might restart the ColdFusion server inadvertently (due to network congestion, for example), rather than detect an actual failure on the ColdFusion server.</p>


Field	Description
Frequency (sec)	<p>Enter a time, in seconds, to indicate how often the probe checks the ColdFusion server.</p> <p>Probes that restart web applications should be configured to run no more frequently than the time it takes to stop and restart ColdFusion. This time is highly site-specific, because it depends on the system resources available on the servers and the volume of traffic at the site.</p> <p>For probes that do not restart the web application, the Frequency depends on how long you can reasonably afford to have your web application off-line. A minimum Frequency of 15 seconds is recommended.</p>
Return Value	<p>Enter 0 so that the probe succeeds on a successful probing of the page. Enter a non-zero number to have the probe succeed on a failure.</p> <p>The default is 0. Only under rare circumstances would you change this to a non-zero number.</p>

- 6 Click Register to create the probe.
 - 7 Close all open dialog boxes.
- Icons for the monitor and probe appear under the Monitor Manager in the ClusterCATS Explorer.

To add a new probe to an existing probe monitor:

- 1 Open the ClusterCATS Explorer.
- 2 Select the *cluster_name* > **Monitor Manager** > *monitor_name* in the left pane.
- 3 Select **Monitor** > **Properties**. The monitor's Properties dialog box appears:



- 4 Click the New Probe button .
- The ColdFusion Web Application Probe settings dialog box appears:
- 5 Configure the application probe settings as described in the table on page 79.
- 6 Click Register to create the probe.
- 7 Close all open dialog boxes.
- An icon for the new probe appears under the Monitor Manager in the ClusterCATS Explorer.

Removing ColdFusion probes

To remove a ColdFusion probe:

- 1 Open the ClusterCATS Explorer.
- 2 Select the *cluster_name* > **Monitor Manager** > *monitor_name* > *probe_name* in the left pane.
- 3 Select **Probe** > **Delete**. Alternatively, you can right-click the probe and select Delete.

Configuring ColdFusion probes in UNIX

This section describes the following:

- [“Adding ColdFusion probes” on page 81](#)
- [“Editing and removing ColdFusion probes” on page 83](#)

Adding ColdFusion probes

To add a new ColdFusion probe:

- 1 Open the ClusterCATS Web Explorer if it is not already open.
- 2 Click the Show Cluster link.
The Show Cluster page appears.
- 3 In the Web Server Name field, enter the fully qualified host name of the server for which you want to configure the ColdFusion probe.
- 4 Click OK.
The Cluster Member List page appears.
- 5 Click the Server Attributes link.
The Connect To Server page appears.
- 6 Select the server to add a probe to from the Web Server Name listbox.
- 7 Click OK.
The selected server’s Properties page appears.
- 8 Click the ColdFusion Probe link.
If there are existing probes for this server, the Probe List page appears.

- 9 To create a new probe, click New.

The ColdFusion Application Probe page appears. If this is the first probe for this server or you clicked New to add another probe, the ColdFusion Application Probe page appears.

- 10 Configure the application probe settings as described in the following table:

Field	Description
Status	This is an informational field. If the probe is not registered, the Status displays <code>Not registered</code> . If the probe is registered, the Status displays <code>Succeeding</code> .
Pathname	Enter the path to the ColdFusion probe. Do not change the default selection unless you installed ClusterCATS for ColdFusion to a directory other than the default installation directory.
Working directory	Enter the path to the probe's working directory. Do not change the default selection unless you installed ClusterCATS for ColdFusion to a directory other than the default installation directory.
Startup Parameters	<p>Enter the actual URL of the site you want the probe to access followed by a text string that appears on a page within the site you are probing (cfprobe.cfm in the screen shown in step 9.)</p> <p>Note: Do not modify the RESTART explicit parameter if you want the probe to automatically restart the ColdFusion Server upon detecting a failure. However, if you do not want ClusterCATS to automatically restart the ColdFusion Server upon detecting a failure, replace RESTART with NORESTART.</p>
Timeout (sec)	<p>Enter a time, in seconds, to indicate how long ClusterCATS should wait before a ColdFusion server failure is registered.</p> <p>Do not set this value to less than 60 seconds because ClusterCATS might restart the ColdFusion server inadvertently (due to network congestion, for example), rather than detect an actual failure on the ColdFusion server.</p>
Frequency (sec)	<p>Enter a time, in seconds, to indicate how often the probe checks the ColdFusion server.</p> <p>Probes that restart web applications should be configured to run no more frequently than the time it takes to stop and restart ColdFusion. This time is highly site-specific, because it depends on the system resources available on the servers and the volume of traffic at the site.</p> <p>For probes that do not restart the web application, the Frequency depends on how long you can reasonably afford to have your web application off-line. A minimum Frequency of 15 seconds is recommended.</p>
Return value	<p>Enter 0 so that the probe succeeds on a successful probing of the page. Enter a non-zero number to have the probe succeed on a failure.</p> <p>The default is 0. Only under rare circumstances would you change this to a non-zero number.</p>

- 11 Click Register to create the probe. ClusterCATS begins to test the selected server immediately.

Editing and removing ColdFusion probes

To edit or remove a ColdFusion probe:

- 1 Open the ClusterCATS Web Explorer, if it is not already open.
- 2 Click the Show Cluster link.
The Show Cluster page appears.
- 3 Enter the fully qualified host name of the server for which you want to configure the ColdFusion probe in the Web Server Name field.
- 4 Click OK. The Cluster Member List page appears.
- 5 Click the Server Attributes link.
The Connect To Server page appears.
- 6 Select the server that hosts the probe in the Web Server Name list box.
- 7 Click OK.
The selected server's Properties page appears.
- 8 Click the ColdFusion Probe link.
The Probe List page appears.
- 9 Select the probe to edit or remove.
- 10 To remove the probe, click Delete.
ClusterCATS removes the ColdFusion probe.
- 11 To edit the probe, click Edit.
A page with all the available probes appears.
- 12 Edit the fields corresponding to the probe that you want to change, and click Register.

Using JRun probes

ClusterCATS provides load-balancing and failover support for your web applications in two ways. First, it automatically interprets and reacts to the load metric that the JRun servers generate. Second, ClusterCATS lets you create web application monitors. These monitors can have multiple probes that periodically test the health and operation of the websites that the JRun servers process.

Note: Multiple JRun probes are allowed per web server, and JRun web applications can be restarted individually. However, each web application should have only one probe that restarts on a failure.

The probe is a high-availability feature that verifies that JRun servers are running properly on clustered servers. It periodically tests specific JRun URLs at specified intervals and verifies their validity against user-defined strings contained in the returned pages.

If the validation test succeeds, inbound HTTP requests continue to be sent to the server for which the probe exists. However, if a test fails (the URL fails, times out, or does not return the user-specified string in the page accessed), ClusterCATS restricts that server and redirects requests to other available servers in the cluster. ClusterCATS continues to test the restricted server; when the probe returns a valid value, the server is considered available.

If the JRun server hangs or fails, ClusterCATS attempts to recover the failed service. When the JRun service is recovered, the probe can restart the JRun server and begin sending HTTP traffic to it again.

This section describes the following:

- [“Configuring JRun probes in Windows” on page 84](#)
- [“Configuring JRun probes in UNIX” on page 88](#)

Configuring JRun probes in Windows

This section describes the following:

- [“Adding JRun probes” on page 84](#)
- [“Removing JRun probes” on page 88](#)

Adding JRun probes

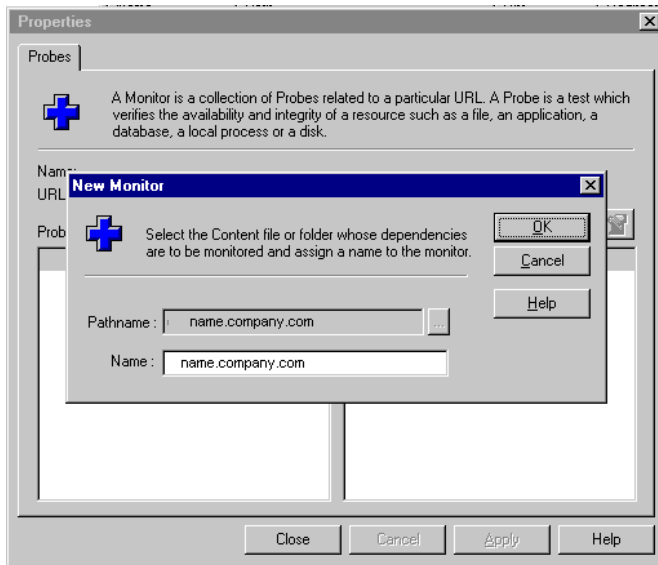
ClusterCATS lets you set up one probe monitor for each server in a cluster. Each monitor can have multiple probes associated with it. As a result, clusters typically have multiple probe monitors (one for each server), and each monitor may have one or more probes.

The procedure for adding a *new* monitor and probe is different from adding a probe to a server that already has a probe monitor. This section describes how to perform both activities.

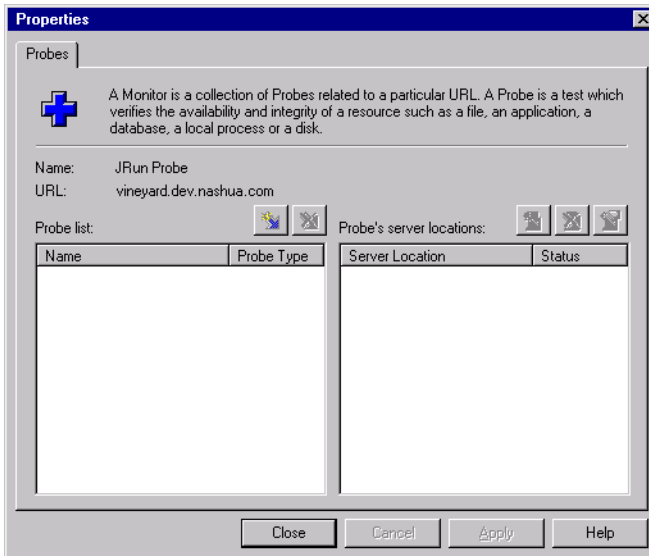
Note: The JRun service must be running on your server to add a probe.


To add a new monitor and JRun probe:

- 1 Open the ClusterCATS Explorer and select a server.
- 2 Select **Server > New Monitor** or right-click the server and select New Monitor.
The New Monitor dialog box appears:

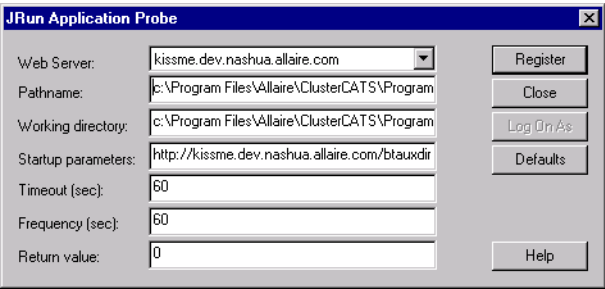


- 3 Enter a name to assign to this probe's monitor in the Name field on the New Monitor dialog box and click OK.
The monitor's Properties dialog box appears:



4 Click the New Probe button .

The JRun Application Probe settings dialog box appears:



5 Configure the application probe settings as described in the following table:

Field	Description
Web Server	Select the name of the server from the drop-down list.
Pathname	Enter the absolute path to the JRun probe. Do not change the default selection unless you installed JRun to a directory other than the default installation directory.
Working directory	Enter the absolute path to the probe's working directory. Do not change the default selection unless you installed JRun to a directory other than the default installation directory.
Startup parameters	<p>Enter parameters passed to the probe on execution using the following syntax:</p> <pre>URL success_string [RESTART NORESTART <JRun_server>] [LOG NOLOG]</pre> <p>URL – enter the actual URL of the page you want the probe to test. By default, this is <code>http://<your_server>/btauxdir/jrunprobe.jsp</code>. The probe opens the page and searches for the <code>success_string</code>.</p> <p><code>success_string</code> – enter a text string that appears at the page specified by the URL. If the <code>success_string</code> includes spaces, it must be enclosed in quotation marks.</p> <p><code>RESTART NORESTART</code> – enter <code>RESTART</code> to make the probe automatically restart the default JRun server on probe failure. Enter <code>NORESTART</code> if you do not want ClusterCATS to restart the JRun server on a failure. If you installed JRun as a service, replace <code>RESTART</code> with <code>Service-<service_name></code> to restart a particular service, or replace <code>RESTART</code> with the name of a specific JRun server (such as <code>admin</code>) to restart.</p> <p><code>LOG NOLOG</code> – enter <code>LOG</code> to enable logging for the ClusterCATS application probe or <code>NOLOG</code> to disable it. The probe logs are stored in the <code>/<CC_install_directory>/log/</code> directory.</p> <p>The default for Startup Parameters is <code>http://<your_server>/btauxdir/jrunprobe.jsp Hello NORESTART NOLOG</code></p>

Field	Description
Timeout (sec)	<p>Enter a time to indicate how long ClusterCATS waits before a JRun server failure is registered.</p> <p>Do not set this value to less than 60 seconds, because ClusterCATS may restart the JRun server inadvertently (due to network congestion, for example), rather than detect an actual failure on the JRun server.</p>
Frequency (sec)	<p>Enter a time to indicate how often the probe checks the JRun server.</p> <p>Probes that restart web applications should be configured to run no more frequently than the time it takes to stop and restart JRun. This is highly site-specific, because it depends on system resources available on servers and the volume of traffic at a site.</p> <p>For probes that do not restart the web application, the frequency depends on how long you can reasonably afford to have your web application offline. A minimum frequency of 15 seconds is recommended.</p>
Return value	<p>Enter 0 to make a probe succeed on a successful probing of the page. Enter a non-zero number to make a probe succeed on a failure.</p> <p>The default is 0. Only under rare circumstances would you change this to a non-zero number.</p>

6 Click Register to create the probe.

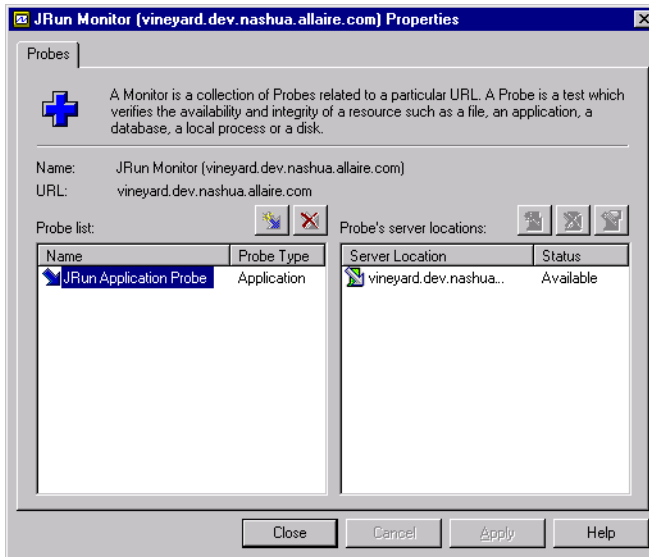
7 Close all open dialog boxes.

Icons for the monitor and probe appear under the Monitor Manager in the ClusterCATS Explorer.

To add a new probe to an existing probe monitor:

- 1 In the ClusterCATS Explorer, select the *cluster_name* > **Monitor Manager** > *monitor_name* in the left pane.
- 2 Select **Monitor** > **Properties**.

The monitor's Properties dialog box appears:



- 3 Click the New Probe button .

The JRun Application Probe settings dialog box displays.

- 4 Configure the application probe settings as described in the table in [“Using JRun probes” on page 84](#).
- 5 Click Register to create the probe.
- 6 Close all open dialog boxes.

An icon for the new probe appears under the Monitor Manager in the ClusterCATS Explorer.

Removing JRun probes

To remove a JRun probe:

- 1 Open the ClusterCATS Explorer.
- 2 Select the *cluster_name* > **Monitor Manager** > *monitor_name* > *probe_name* in the left pane.
- 3 Select **Probe > Delete** or right-click the probe and select Delete.

Configuring JRun probes in UNIX

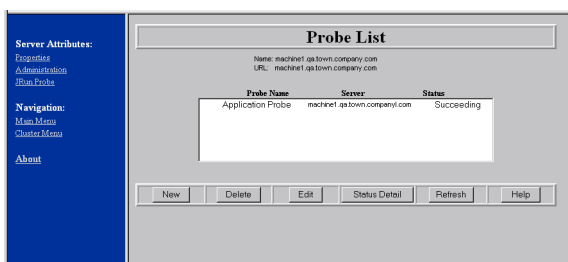
This section describes the following:

- [“Adding JRun probes” on page 89](#)
- [“Editing and removing JRun probes” on page 91](#)

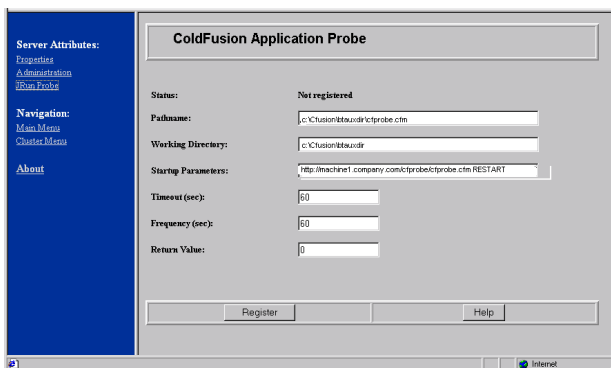
Adding JRun probes

To add a new JRun probe:

- 1 In the ClusterCATS Web Explorer, click the Show Cluster link.
The Show Cluster page appears.
- 2 In the Web Server Name field, enter the fully qualified host name of a server for which to configure the JRun probe.
- 3 Click OK.
The Cluster Member List page appears.
- 4 Click the Server Attributes link.
The Connect To Server page appears.
- 5 Select a server to add a probe to from the Web Server Name list box.
- 6 Click OK.
The selected server's Properties page appears.
- 7 Click the JRun Probe link.
If there are probes for this server, the Probe List page appears:



- 8 To create a new probe, click New. The JRun Application Probe page appears.
If this is the first probe for this server, or you clicked New to add a probe, the JRun Application Probe page appears:



9 Configure the application probe settings as described in the following table:

Field	Description
Status	This is an informational field. If the probe is not registered, the Status displays Not registered. If the probe is registered, the Status displays Succeeding.
Pathname	Enter the path to the JRun probe. Do not change the default selection unless you installed ClusterCATS for JRun to a directory other than the default installation directory. The default is <code>/usr/lib/btcats/program/jrunprobe</code> .
Working directory	Enter the path to the probe's working directory. Do not change the default selection unless you installed ClusterCATS for JRun to a directory other than the default installation directory. The default is <code>/usr/lib/btcats/program/</code> .
Startup Parameters	<p>Enter parameters passed to the probe on execution, using the following syntax:</p> <pre>URL success_string [RESTART NORESTART <JRun_server>] [LOG NOLOG]</pre> <p>URL – enter the actual URL of the page you want the probe to test. By default, this is <code>http://<your_server>/btauxdir/jrunprobe.jsp</code>. The probe opens the page and searches for the <code>success_string</code>.</p> <p><code>success_string</code> – enter a text string that appears at the page specified by the URL. If the <code>success_string</code> includes spaces, it must be enclosed in quotation marks.</p> <p><code>RESTART NORESTART</code> – enter <code>RESTART</code> to make the probe automatically restart the default JRun server on a failure. Enter <code>NORESTART</code> if you do not want ClusterCATS to restart the JRun server on a failure. You can replace <code>RESTART</code> with the name of a specific JRun server (such as <code>default</code>) to restart.</p> <p><code>LOG NOLOG</code> – enter <code>LOG</code> to enable logging for the ClusterCATS application probe or <code>NOLOG</code> to disable it. The probe logs are stored in the <code>/<CC_install_directory>/log/</code> directory.</p> <p>The default for Startup Parameters is <code>http://<your_server>/btauxdir/jrunprobe.jsp Hello NORESTART NOLOG</code></p>
Timeout (sec)	<p>Enter a time to indicate how long ClusterCATS waits before a JRun server failure is registered.</p> <p>Do not set this value to less than 60, because ClusterCATS might restart the JRun server inadvertently (due to network congestion, for example), rather than detect an actual failure on the JRun server.</p>

Field	Description
Frequency (sec)	<p>Enter a time to indicate how often the probe checks the JRun server.</p> <p>Probes that restart web applications should be configured to run no more frequently than the time it takes to stop and restart JRun. The time is highly site-specific, because it depends on the system resources available on the servers and the volume of traffic at the site.</p> <p>For probes that do not restart the web application, the frequency depends on how long you can reasonably afford to have your web application offline. A minimum frequency of 15 seconds is recommended.</p>
Return value	<p>Enter 0 to make the probe succeed on a successful probing of the page. Enter a non-zero number to make the probe succeed on a failure.</p> <p>The default is 0. Only under rare circumstances would you change this to a non-zero number.</p>

- 10 Click Register to create the probe. ClusterCATS begins to test the selected server immediately.

Editing and removing JRun probes

To edit or remove a JRun probe:

- 1 In the ClusterCATS Web Explorer, click the Show Cluster link.
The Show Cluster page appears.
- 2 Enter the fully qualified host name of a server for which to configure the JRun probe in the Web Server Name field.
- 3 Click OK.
The Cluster Member List page appears.
- 4 Click the Server Attributes link.
The Connect To Server page appears.
- 5 Select the server that hosts the probe in the Web Server Name list box.
- 6 Click OK.
The selected server's Properties page appears.
- 7 Click the JRun Probe link.
The Probe List page appears.
- 8 Select the probe to edit or remove.
- 9 To remove the probe, click Delete.
ClusterCATS removes the JRun probe.
- 10 To edit the probe, click Edit.
A page with all the available probes appears.
- 11 Edit the fields corresponding to the probe that you want to change, and click Register.

Load-balancing devices

You can configure ClusterCATS to work in conjunction with a third-party hardware load-balancing device or load-balancing software product to provide comprehensive load balancing and failover support for your server clusters.

This section describes the following:

- [“Using Cisco LocalDirector” on page 92](#)
- [“Using third-party load-balancing devices in Windows” on page 96](#)
- [“Using third-party load-balancing devices in UNIX” on page 97](#)

Using Cisco LocalDirector

Cisco LocalDirector is a network appliance with a secure, real-time, embedded operating system that intelligently load balances IP traffic across multiple servers. You can configure ClusterCATS to provide server availability and load information to LocalDirector using the Cisco Dynamic Feedback Protocol (DFP). LocalDirector then actively manages HTTP traffic across the cluster, based on the load information provided to it by ClusterCATS.

You can configure LocalDirector using the ClusterCATS Explorer on Windows only.

Note: You must use Cisco LocalDirector Version 3.1.4 software or later.

Before configuring ClusterCATS with LocalDirector, you must configure LocalDirector to manage your web servers. For more information, see the Cisco documentation.

LocalDirector considerations

Be aware of the following issues when using ClusterCATS with Cisco LocalDirector:

- When load balancing with LocalDirector, ClusterCATS sets the state of each cluster member to passive mode. For more information, see [“Changing active/passive settings” on page 111](#).
- Do not use round-robin DNS.
- Turn off the ClusterCATS gradual redirection load threshold. For more information, see [“Server load thresholds” on page 66](#).
- Do not use ClusterCATS dynamic IP addressing feature. If ClusterCATS performs dynamic IP failover, the LocalDirector cannot recover the failed-over IP address. For more information, see [“ClusterCATS dynamic IP addressing \(Windows only\)” on page 132](#).
- If two or more web servers on a system are in clusters using LocalDirector load balancing, each cluster must have the same DFP Agent Listen Port number configured. The ClusterCATS DFP agent can listen only on one port.

LocalDirector dynamic-feedback command settings

Use the LocalDirector `dynamic-feedback` command options as described in this section to optimize your LocalDirector setup.

Note: Do not use the `dynamic-feedback-pw` command. ClusterCATS does not support secure DFP hosts.

- `dynamic-feedback -timeout` — sets `timeout` to a value larger than the update frequency so LocalDirector does not prematurely terminate the connection with the cluster because of inactivity. We recommend that you set the value to at least twice the update frequency.
- `dynamic-feedback -retry`
Use the `dynamic-feedback -retry` — sets the `retry` value to zero to ensure that LocalDirector continues connection attempts to the ClusterCATS DFP agent in the event of a lengthy period of system unavailability.

For more information, see Cisco's LocalDirector Command Reference.

To integrate ClusterCATS with the Cisco LocalDirector:

- 1 Review all considerations before continuing with this procedure.
- 2 Complete the LocalDirector basic hardware installation and configuration. Ensure that you have defined an IP address for LocalDirector, and that the LocalDirector network interfaces are configured correctly. You can use the `ping` utility to test network connectivity.
- 3 Create a virtual server (`www.yourcompany.com`) in LocalDirector that corresponds to the cluster.
- 4 In LocalDirector, bind explicit (real) servers participating in the cluster with the virtual server.
- 5 Use LocalDirector's `dynamic-feedback` command to specify the IP addresses of each explicit server (cluster member) and port number that each server will use to listen for DFP requests from LocalDirector. The port number must be the same as the DFP Agent Listen Port configured in step 9.

For example:

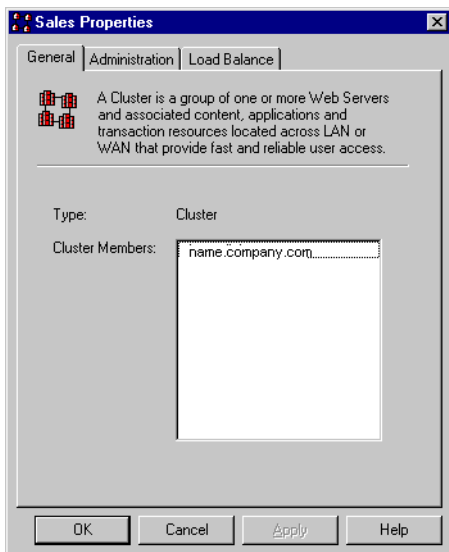
```
dynamic-feedback 192.168.64.22:9124 retry 0 attempts 30 timeout 60
```

The DFP protocol will connect to server 192.168.64.22 at port 9124. If the connection between LocalDirector and the server is closed for any reason, LocalDirector will attempt to reconnect, every 30 seconds, indefinitely. LocalDirector will close the connection if it is inactive for 60 seconds.

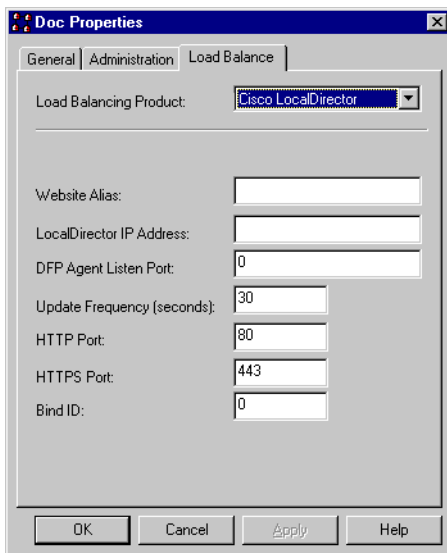
For more information, see [“LocalDirector dynamic-feedback command settings” on page 93](#).

- 6 Open the ClusterCATS Explorer and select a cluster.

- 7 Select **Cluster > Properties** or **Configure > Administration**. Both menu selections display the Cluster Properties dialog box, as the following figure shows:



- 8 Click the Load Balance tab and choose Cisco LocalDirector from the Load Balancing Product drop-down list.



9 Edit the cluster properties as described in the following table:

Field	Description
Website Alias	Enter the name of the virtual server (www.yourcompany.com) you created in step 3.
LocalDirector IP Address	Enter the IP address of Cisco LocalDirector.
DFP Agent Listen Port	Enter the port number on which the cluster's DFP agent should listen for incoming LocalDirector connection requests. This should be the same port specified in the LocalDirector <code>dynamic-feedback</code> as described in step 5.
Update Frequency (sec)	Enter the frequency with which you want ClusterCATS to update the LocalDirector with availability data. This is typically between 5 and 30. You can enter up to 120. As you add web servers to the cluster, set a larger value. This minimizes the overhead of traffic to LocalDirector.
HTTP Port	Enter the port number on which each cluster member listens for unsecured HTTP requests. Enter 0 if not applicable.
HTTPS Port	Enter the port number on which each cluster member listens for secured HTTP requests. Enter 0 if not applicable.
Bind ID	Enter the same Bind ID specified for the explicit (real) servers on the LocalDirector in step 4. For ClusterCATS/LocalDirector integration to work as intended, the server name, port number, and bind ID combination must be the same on this ClusterCATS Load Balance tab as on the LocalDirector box.

10 Click OK.

When configured, ClusterCATS automatically sets the state of each cluster member to Passive and provides the load-balancing and high availability data it acquires to LocalDirector. LocalDirector actively manages HTTP traffic across the cluster.

Using third-party load-balancing devices

Third-party load-balancing devices actively distribute load to the web servers based on packet flow while ClusterCATS monitors JRun and ColdFusion load and availability. If ClusterCATS detects that the server is becoming overloaded, it supersedes the load-balancing device and redirects traffic accordingly.

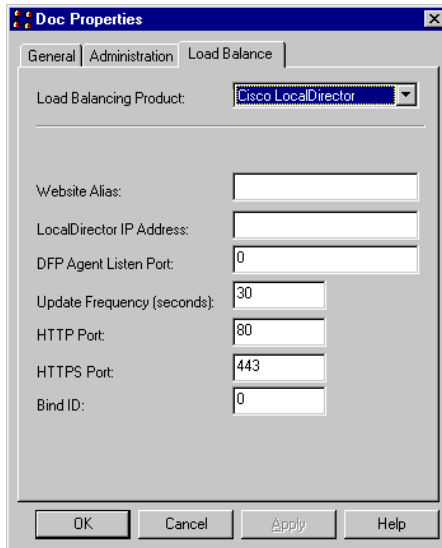
This section describes how to configure a third-party load-balancing device with ClusterCATS in the following sections:

- [“Using third-party load-balancing devices in Windows” on page 96](#)
- [“Using third-party load-balancing devices in UNIX” on page 97](#)

Using third-party load-balancing devices in Windows

To integrate ClusterCATS with a third-party load-balancing device:

- 1 Configure the load-balancing device or software product as recommended by the manufacturer.
- 2 Open the ClusterCATS Explorer and select a cluster.
- 3 Select **Configure > Administration** or right-click the cluster and select **Configure > Configure**. The Cluster Properties dialog box displays.
- 4 Click the Load Balance tab.



The selection in the Load Balancing Product drop-down list indicates how ClusterCATS will actively load balance HTTP traffic across the cluster.

- 5 Enter the name of the website in the Website Alias field.
- 6 Click OK to apply your changes.

Using third-party load-balancing devices in UNIX

You cannot take advantage of ClusterCATS support of Cisco LocalDirector with ClusterCATS Web Explorer. This capability is available only in the Windows-based ClusterCATS Explorer. You can, however, configure Cisco LocalDirector as a third-party load-balancing device to work with ClusterCATS.

To integrate ClusterCATS with a third-party load-balancing device:

- 1 In ClusterCATS Web Explorer, click the Show Cluster link.
- 2 Enter the fully qualified host name of a server to integrate with another load-balancing product in the Web Server Name field.
- 3 Click OK.
The Cluster Member List page appears.
- 4 Click the Administration link under Cluster Attributes.
- 5 In the Load Balancing Product field, enter the URL of the website for which the load-balancing product has been set up to manage HTTP traffic.
- 6 To apply your changes, click OK .

Administrator alarm notifications

The ClusterCATS alarm notification feature provides instant feedback about critical events that take place within a cluster. When an event triggers an alarm, ClusterCATS notifies one or more people by e-mail. The events that can trigger a notification are listed below.

If an event you selected occurs, ClusterCATS sends an e-mail message to the designated person. The following table explains the notification schedule for each event.

Event type	Notification occurs...
Disk Failure	Immediately
HTTP Server Failure	Immediately
Server Busy Warning	Every 24 hours
Server Unreachable	Immediately
Web Server Failover	Immediately
JRun Probe Failure	Immediately

This section describes the following:

- [“Configuring administrator alarm notifications on Windows” on page 98](#)
- [“Configuring administrator alarm notifications on UNIX” on page 99](#)

Configuring administrator alarm notifications on Windows

To configure an alarm notification:

- 1 In ClusterCATS Explorer, select a cluster.
- 2 Select **Configure > Alarm Notification** or right-click the cluster and select **Configure > Alarm Notification**.

The Alarm Notification dialog box displays.

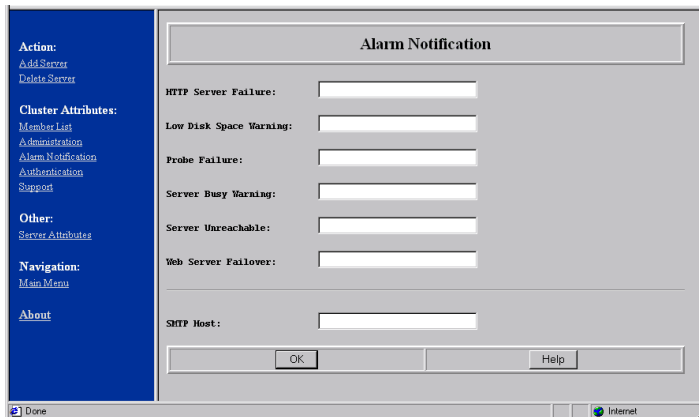
- 3 Select an event for which to trigger an alarm and enter the e-mail address of the person you want to receive an e-mail notification of the event.
If you want multiple people to receive an e-mail notification about an event, add more e-mail addresses to the field. Separate each e-mail address with a comma.
- 4 Repeat step 3 for each event you want to be notified about.
To send all notifications to one e-mail address, enter the address once and click Propagate.
- 5 Enter the name of the default SMTP mail server to which your mail is delivered, in the Default SMTP Host field.
- 6 Click OK.

Configuring administrator alarm notifications on UNIX

To configure administrator alarm notifications:

- 1 In ClusterCATS Web Explorer, click the Show Cluster link. The Show Cluster page appears.
- 2 Enter the fully qualified host name of a server for which to configure administrator alarm notifications in the Web Server Name field.
- 3 Click OK. The Cluster Member List page appears.
- 4 Click the Alarm Notification link.

The Alarm Notification page appears:



The screenshot shows a web browser window with a blue sidebar on the left and a main content area. The sidebar contains the following links:

- Action:**
 - [Add Server](#)
 - [Delete Server](#)
- Cluster Attributes:**
 - [Member List](#)
 - [Administration](#)
 - [Alarm Notification](#) (highlighted)
 - [Authentication](#)
 - [Support](#)
- Other:**
 - [Server Attributes](#)
- Navigation:**
 - [Main Menu](#)
- [About](#)

The main content area is titled "Alarm Notification" and contains the following fields:

- HTTP Server Failure:
- Low Disk Space Warning:
- Probe Failure:
- Server Busy Warning:
- Server Unreachable:
- Web Server Failover:
- SMTP Host:

At the bottom of the main content area are two buttons: "OK" and "Help". The browser's status bar at the bottom shows "Done" and "Internet".

- 5 Enter the e-mail address of the person you want to be notified about the occurrence of an event in the event's corresponding field.
If you want multiple people to receive an e-mail notification about one event, add more e-mail addresses to the field. Separate each address with a comma.
- 6 Enter the name of the default SMTP mail server to which your mail is delivered, in the SMTP Host field.
- 7 Click OK to apply your changes.

Administrator e-mail options

The ClusterCATS administration e-mail support feature reports vital statistics about your cluster to designated e-mail accounts in your organization. You can set up the following types of administration e-mail options:

- Report e-mail — lets you know each day how your server clusters are functioning. Daily e-mail reports include the following information:
 - Cluster name and each server's name and IP address in the cluster
 - Files — number of files in the web server's root directory
 - Disk space — amount of disk space used and remaining on the system drive that contains the web server's root directory
 - Log files — size and location of the log files
- Support e-mail — sends an automatic e-mail nightly to Macromedia's Technical Support team that contains basic configuration information about your cluster. This information enables Macromedia to provide optimal support by understanding your environment when you call a Technical Support representative. Support e-mail contains the following information:
 - Cluster name and the number of servers the cluster contains
 - Statistics for each server, including failover, redirection, and database statisticsYou can also have one or more people of your choice receive copies of this periodic e-mail.

This section describes the following:

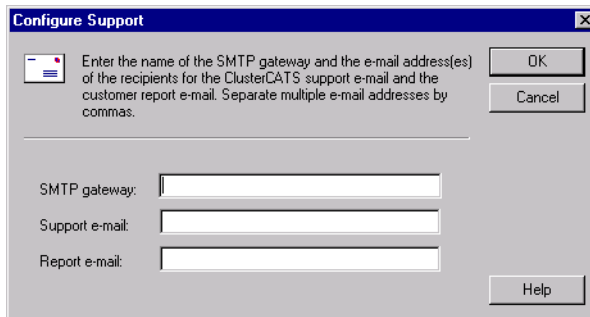
- [“Configuring administration e-mail options on Windows” on page 100](#)
- [“Configuring administration e-mail options on UNIX” on page 101](#)

Configuring administration e-mail options on Windows

To configure administration e-mail options:

- 1 In ClusterCATS Explorer, select a cluster.
- 2 Select **Configure > Support** or right-click a cluster and select **Configure > Support**.

The Support dialog box appears:



- 3 Edit the e-mail support options as described in the following table:

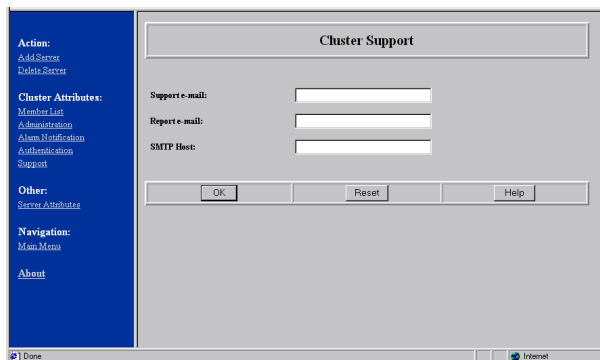
Field	Description
SMTP gateway	Enter the name of the server through which outgoing e-mail is sent.
Support e-mail	Enter the e-mail address of a person in your organization to receive a copy of the nightly technical support e-mail. If more than one person should receive the e-mail, separate the addresses with commas. You do not have to enter a Macromedia Technical Support address.
Report e-mail	Enter the e-mail address of a person in your organization to receive daily reports about your clusters. If more than one person is to receive the e-mail, separate the addresses with commas.

- 4 Click OK to enable the ClusterCATS Report and Support e-mail options.

Configuring administration e-mail options on UNIX

To configure administration e-mail options:

- 1 In ClusterCATS Web Explorer, click the Show Cluster link.
The Show Cluster page appears.
- 2 Enter the fully qualified host name of a server for which you want to configure administrator e-mail support in the Web Server Name field.
- 3 Click OK.
The Cluster Member List page appears.
- 4 Click the Support link.
The Cluster Support page appears:



- 5 Edit the e-mail support fields as described in the following table:

Field	Description
SMTP Gateway	Enter the name of the server through which outgoing e-mail is sent.
Support e-mail	Enter the e-mail address of a person in your organization to receive a copy of the nightly technical support e-mail. If more than one person should receive the e-mail, separate e-mail addresses with commas. You do not have to enter a Macromedia Technical Support address.
Report e-mail	Enter the e-mail address of the person at your organization that should receive daily reports about your clusters. If more than one person should receive the e-mail, separate the e-mail addresses with commas.

- 6 Click OK to enable the ClusterCATS Report and Support e-mail options.

Administering security

When you enable ClusterCATS administration security for a cluster, only authorized users are able to access and administer the cluster, using ClusterCATS Explorer (Windows) or the ClusterCATS Web Explorer (UNIX). ClusterCATS provides these administration security settings for securing your server cluster environment:

- **Disabled Authentication** — this is the default setting. It provides no security challenge, so anyone can access the server cluster with a ClusterCATS administration tool, or even a web browser, and modify the cluster environment.
- **Local User Authentication** — this is the recommended security setting for most clusters, residing in small to mid-sized organizations that have only a few administrators. This setting provides a security challenge for anyone accessing the server. The authentication is based on administrative privileges that you define for specific users on each server in the cluster.
- **Windows NT Domain Authentication (Windows NT Only)**— you may want to use this security setting if your organization is fairly large and contains many distributed administrator groups that need to access your server clusters. To use this setting, you must define your global administrators' group in the form "BT_*clustername*", where *clustername* is the exact name of the cluster you created with the ClusterCATS Explorer. The global administrators group must exist within the same domain as the clustered servers.

This section describes the following:

- [“Configuring authentication on Windows” on page 103](#)
- [“Configuring authentication on UNIX” on page 106](#)

Configuring authentication on Windows

The following sections describe how to enable authentication for your environment.

- [“Configuring local-user authentication” on page 103](#)
- [“Configuring Windows NT domain authentication” on page 105](#)

Configuring local-user authentication

Local-user authentication lets ClusterCATS authenticate specific users per server. Local users of a server must have an account on the server where the web server resides.

For example, if a cluster includes several web servers and you have an account on only one, then you can only administer that server.

To configure authentication modes for your clusters:

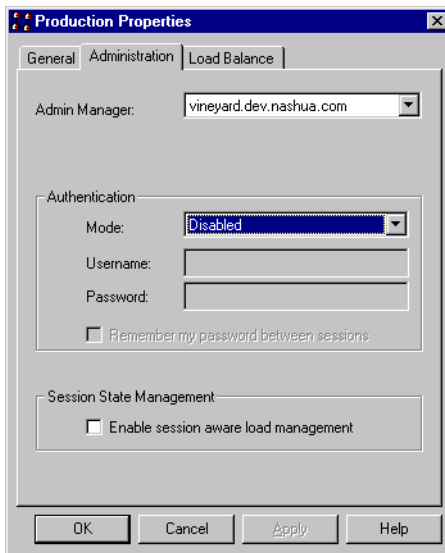
- 1 Create a user account on *each server* within your cluster for each administrator whom you want to be able to administer the servers using the ClusterCATS Explorer.

If your cluster members are NT servers, use the Windows User Manager utility to create your user accounts.

Note: If only one person will administer all cluster members in the cluster, be sure to create the same user account (identical user name and password) on each cluster member. The ClusterCATS Explorer will then prompt you only once for a user name and password. However, if you create multiple administrator accounts on each server, ClusterCATS Explorer will display user name and password prompts upon each attempt to access the servers from the ClusterCATS Explorer.

- 2 In ClusterCATS Explorer, select a cluster.
- 3 Select **Configure > Administration** or **Cluster > Properties** (both menu selections display the Properties dialog box); or right-click the cluster and select **Configure > Administration**.

The Properties dialog box appears:



- 4 Select Local User from the Mode drop-down box.
- 5 Enter a user name and password defined for a valid account.

Note: ClusterCATS requires you to enter a valid user name and password after selecting the authentication type, so you do not inadvertently lock yourself out of the cluster.

- 6 Click OK to enable local user authentication for the selected cluster. Only administrators who have accounts on each secured server can access and administer those cluster members using ClusterCATS Explorer.

Configuring Windows NT domain authentication

Windows NT Domain authentication lets ClusterCATS authenticate administrators who have been added to a Windows NT domain user group.

Note: This authentication mode can be used only on NT servers and on Windows 2000 servers if the domain is using the Windows NT compatible domain controller model rather than the Active Directory model.

Before you can enable NT domain authentication on a cluster, you must create an NT global user group within the domain you want to secure. You can do this using the Windows NT User Manager for Domains utility. After you create a user group, add users to it, and enable the NT Domain authentication mode from the ClusterCATS Explorer, all users you add to that group are automatically authenticated to view and change the cluster. All servers in the cluster must reside in the same Windows NT domain unless a trusted relationship is set up between two or more domains.

A global group must exist in the domain from which the ClusterCATS Explorer is executed. Cluster members in other domains need only the trust relationship. ClusterCATS Explorer determines what servers exist in which NT domain by communicating with any Windows NT domain controller for the domain. You can view the list of servers that exist in the Windows NT domain with the Network Neighborhood Windows NT utility. If no trust relationship exists, then cluster members must be from the same Windows NT domain.

To enable Windows NT domain authentication:

- 1 Select **Start > Programs > Administrative Tools > User Manager for Domains** to open the User Manager for Domains utility.
- 2 Select **User > New Global Group**.
The New Global Group dialog box appears.
- 3 Enter a name and description for the group in the applicable fields.
Your global group name must be `BT_clustername`, where *clustername* is the name of your ClusterCATS cluster.
- 4 Click **Add** to add the administrators whom you want to have privileges to your global group.
The Add Users and Groups dialog box appears.
- 5 Select the domain from the List Names drop-down box.
- 6 Select the users you want to add to the group and click **Add**.
- 7 Click **OK** in all open dialog boxes to apply your changes and to close the User Manager for Domains utility.
- 8 Open the ClusterCATS Explorer and select a cluster for which to configure authentication.
- 9 Select **Configure > Administration or Cluster > Properties** (both menu selections display the Properties dialog box) or right-click the cluster and select **Configure > Administration**.
The Properties dialog box appears.

10 Select NT Domain from the Mode drop-down box.

11 Enter a valid user name and password that participates in the domain.

Note: ClusterCATS requires you to enter a valid user name and password after selecting the authentication type, so you do not inadvertently lock yourself out of the cluster.

12 Click OK to enable Windows NT Domain authentication for the selected cluster.

Only users whom you added to the Global User Group of the domain can use ClusterCATS Explorer to view and administer clusters with ClusterCATS Explorer.

Disabling authentication

Disabling authentication lets any user employ ClusterCATS Explorer to create, configure, or administer clusters. When a cluster is added, administrators have unrestricted access to the content in that cluster. Therefore, you should choose disabled mode only if security is not a concern (for example, in a development or QA environment).

By default, ClusterCATS administrator security is disabled. However, if you have previously configured the security mode for your cluster and now want to turn it off, perform the following procedure.

To disable authentication:

- 1 Open the ClusterCATS Explorer and select a cluster with authentication enabled.
- 2 Select **Configure > Authentication** or select **Cluster > Properties** (both menu selections display the Properties dialog box.) or right-click the cluster and select **Configure > Administration**.
- 3 Select Disabled from the Mode drop-down box.
- 4 Click OK to apply your changes.

Configuring authentication on UNIX

To configure authentication modes for your clusters:

- 1 In ClusterCATS Web Explorer, click the Show Cluster link. The Show Cluster page appears.
- 2 Enter the fully qualified host name of the server for which to configure administrator authentication in the Web Server Name field.
- 3 Click OK.
The Cluster Member List page appears.
- 4 Click the Authentication link.

The Cluster Authentication page appears:

Action:
[Add Server](#)
[Delete Server](#)

Cluster Attributes:
[Member List](#)
[Administration](#)
[Alarm Notification](#)
[Authentication](#)
[Support](#)

Other:
[Server Attributes](#)

Navigation:
[Main Menu](#)

About

Cluster Authentication

Authentication:

User Name:

Password:

Done Internet

- 5 Select Local User from the Authentication drop-down box to enable local-user authentication.
- 6 Select Disabled to disable authentication.
- 7 If using local user authentication, enter a valid user name and password and click OK.

ClusterCATS requires you to enter a valid user name and password after selecting the authentication type, so you do not inadvertently lock yourself out of the cluster.

CHAPTER 5

Maintaining Cluster Members

After you have created your clusters, added servers to them, and configured them with load-balancing and high-availability features, they will probably run inconspicuously in your environment for quite some time. However, at some point you may need to update software and content or perform general maintenance tasks that are beyond the typical cluster creation and configuration activities.

Contents

- [Understanding ClusterCATS server modes](#) 110
- [Changing active/passive settings.....](#) 111
- [Changing restricted/unrestricted settings](#) 113
- [Using maintenance mode \(Windows only\)](#) 115
- [Updating a cluster member \(Windows only\).....](#) 118
- [Resetting cluster members](#) 120

Understanding ClusterCATS server modes

ClusterCATS lets you move cluster members into modes of operation depending on the tasks you want to perform on their server. The modes let you remove servers from clusters to perform maintenance activities without disturbing the current traffic flow, among other things.

The following table describes the modes of operation into which you can place cluster members:

Mode	Description
Active/Passive Setting	<p>Turns the ClusterCATS Server on and off. In active state, the ClusterCATS Server intercepts HTTP requests and processes them for load balancing and availability. In passive state, HTTP requests are passed directly to the web server without ClusterCATS Server interception.</p> <p>For more information, see “Changing active/passive settings” on page 111.</p>
Restricted/Unrestricted Setting	<p>Determines whether active cluster members receive HTTP traffic. Restricted ClusterCATS Servers do not receive HTTP traffic. Unrestricted ClusterCATS Servers are sent traffic.</p> <p>For more information, see “Changing restricted/unrestricted settings” on page 113.</p>
Maintenance Mode	<p>Lets you remove a server from a cluster by draining off all users without cutting connections. You use this when you want to upgrade a server or remove it entirely from a cluster.</p> <p>For more information, see “Using maintenance mode (Windows only)” on page 115.</p> <p>Only Windows cluster members can be put in Maintenance mode.</p>

Changing active/passive settings

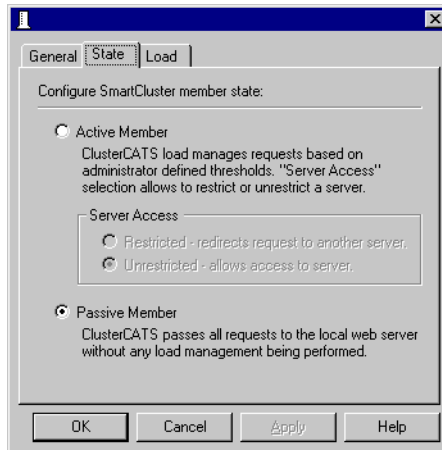
All cluster members are added to a cluster with the ClusterCATS Server in active state, by default. In active state, ClusterCATS Servers intercept requests to your web resources and provide availability and failover services. From time to time, you may want to turn off these load-balancing and failover services, to help troubleshoot problems. To do this, change the ClusterCATS Server's state from active to passive. In passive state, ClusterCATS Servers do not actively manage load nor protect against resource failures. HTTP requests sent to a server that is in the passive state are passed directly to the web server without ClusterCATS Server processing.

Changing active/passive settings in Windows

To change a cluster member's state:

- 1 Open the ClusterCATS Explorer and select a cluster member.
- 2 Select **Configure > State** or right-click a cluster member and select **Configure > State**.

The server Properties dialog box appears:



- 3 To make ClusterCATS Server ignore incoming HTTP requests and pass them directly to the web server, select Passive Member.
- 4 To make ClusterCATS Servers intercept requests to your web resources, select Active Member.
- 5 Click OK to apply your changes.
The cluster member's icon in the ClusterCATS Explorer turns white, indicating that the cluster is passive.
- 6 Repeat steps 1 through 5 to change other members in the cluster.

Changing active/passive settings in UNIX

To change a cluster member's state:

- 1 In ClusterCATS Web Explorer, click the Show Cluster link.
The Show Cluster page appears.
- 2 Enter the fully qualified host name of the server in the Web Server Name field.
- 3 Click OK.
The Cluster Member List page appears.
- 4 Click the Server Attributes link under Other.
The Connect To Server page appears.
- 5 Select a server to connect to from the Web Server Name drop-down box.
- 6 Click OK.
The selected server's Properties page appears.
- 7 Click the Administration link.
The Server Administration page appears for the selected server.
- 8 To make ClusterCATS Server ignore incoming HTTP requests and pass them directly to the web server, select Passive from the State drop-down box.
- 9 To make ClusterCATS Server intercept requests to your web resources, select Active from the State drop-down box.
- 10 Click OK.

Changing restricted/unrestricted settings

ClusterCATS lets you stop a cluster member from receiving HTTP requests by changing the restricted/unrestricted setting. You may want to restrict a server when performing server maintenance or software updates, verifying load configurations, or as an alternative method to managing load.

Only cluster members in active mode can be restricted, because cluster members in passive mode do not receive *any* ClusterCATS Server intervention.

This section describes the following:

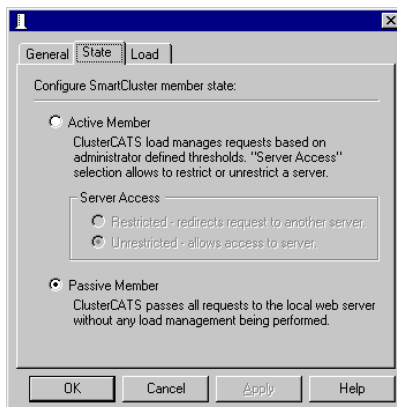
- [“Restricting/unrestricting servers in Windows” on page 113](#)
- [“Restricting/unrestricting servers in UNIX” on page 114](#)

Restricting/unrestricting servers in Windows


To change restriction settings for a cluster member:

- 1 Open the ClusterCATS Explorer and select a cluster member.
- 2 Select **Configure > State** or right-click a cluster member and select **Configure > State**.

The Server Properties dialog box appears:



- 3 Select the Active Member option if the server has been in passive state.
- 4 To ensure that HTTP requests sent explicitly to this cluster member are redirected to another server within the cluster, select Restricted in the Server Access area.

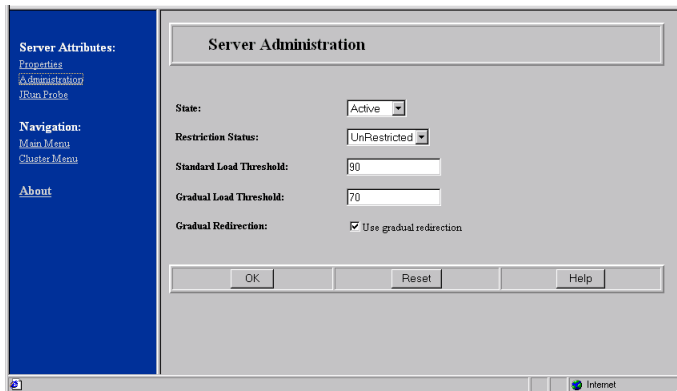
The cluster member icon changes to  in the ClusterCATS Explorer, indicating that the cluster is active but restricted.

- 5 To allow the server to participate in the cluster, select Unrestricted in the Server Access area.
- 6 Click OK.

Restricting/unrestricting servers in UNIX

To change restriction settings for a cluster member:

- 1 In ClusterCATS Web Explorer, click the Show Cluster link.
The Show Cluster page appears.
- 2 Enter the fully qualified host name of a server in the Web Server Name field.
- 3 Click OK.
The Cluster Member List page appears.
- 4 Click the Server Attributes link under Other.
The Connect To Server page appears.
- 5 Select a server to connect to from the Web Server Name drop-down box.
- 6 Click OK.
The selected server's Properties page appears.
- 7 Click the Administration link.
The Server Administration page appears for the selected server:



- 8 To ensure that HTTP requests sent explicitly to this cluster member are redirected to another server within the cluster, select Restricted from the Restriction Status drop-down box.
- 9 To allow the server to participate in the cluster, select Unrestricted from the Restriction Status drop-down box.
- 10 Click OK.

Using maintenance mode (Windows only)

Putting a ClusterCATS Server in maintenance mode lets you remove a server from an active cluster gracefully, so you can perform necessary updates or maintenance tasks without disrupting your users. Using the instructions in this section, you can take a server offline while allowing users to finish their current sessions.

Once the server is in maintenance mode, you can perform the following tasks that would normally disrupt users' experiences:

- Upgrading server software or applications
- Change content on the website
- Troubleshooting problems

When a server is in maintenance mode, inbound HTTP traffic for the affected server is redirected to the most available server in the cluster. After you complete your maintenance tasks and take the server out of maintenance mode, the servers that temporarily assumed the restricted server's IP address and HTTP traffic return the IP address to the affected server so it can receive and process HTTP requests.

Note: Macromedia recommends that you set up your clusters with ClusterCATS dynamic IP addressing for using maintenance mode. For more information, see ["Using server failover" on page 137](#).

When it is enabled, maintenance performs the following:

- The Clustered Web Server on the system is set to a busy state for a user-specified period of time. New traffic to the website is redirected to another server in the cluster.
- If you are running session-aware load balancing, users who have begun sessions can continue until the ClusterCATS service is shut down.
- When the timeout period has expired, the ClusterCATS service will be shut down.
- If ClusterCATS dynamic addressing is active, the IP addresses associated with cluster members for this server will be failed over to another server, thus allowing the site to continue to function while maintenance is performed.

To put a cluster member in maintenance mode:

- 1 In ClusterCATS Explorer, select a cluster member to update.
- 2 Select **Configure > Load** or right-click a cluster member and select **Configure > Load**.

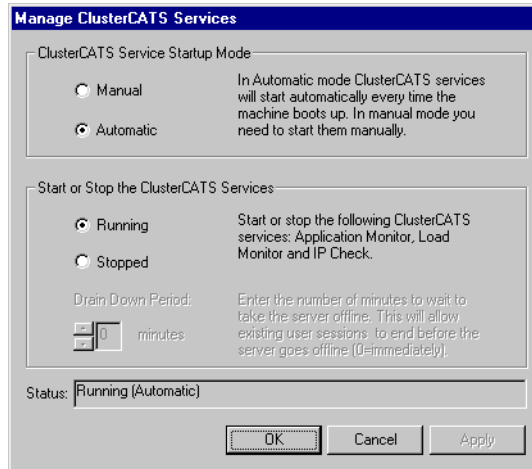
The Properties dialog box appears for the selected cluster member with the Load tab active:



- 3 Change the peak load threshold to 0% so any additional HTTP requests will be redirected to other servers in the cluster.
- 4 Click OK.
- 5 Physically go to the server you selected in step 1 and open the ClusterCATS Server Administrator on it, by selecting **Start > Programs > Macromedia > ClusterCATS Server Administrator**.

The ClusterCATS Server Administrator appears.

- Click the BT Service Status button to display the Manage ClusterCATS Services dialog box:



- Select the Stopped option to stop the ClusterCATS service, and enter a value, in minutes, in the Drain Down Period field. This option allows current users to conclude their sessions within the time indicated.

- Click OK.

When the drain-down period expires, the server fails over to another server in the cluster.

To take a cluster member out of maintenance mode:

- Physically go to the server and open the ClusterCATS Server Administrator by selecting **Start > Programs > Macromedia > ClusterCATS Server Administrator**.

The ClusterCATS Server Administrator appears.

- Click the BT Service Status button to display the Manage ClusterCATS Services dialog box.

- Select the Running option.

- Click OK.

- Open the ClusterCATS Explorer and select a cluster member to take out of maintenance mode.

- Select **Configure > Load** or right-click a cluster member and select **Configure > Load**.

The Properties dialog box appears for the selected cluster member with the Load tab active.

- Change the peak load threshold from 0 percent to an appropriate value.

- Click OK.

Updating a cluster member (Windows only)

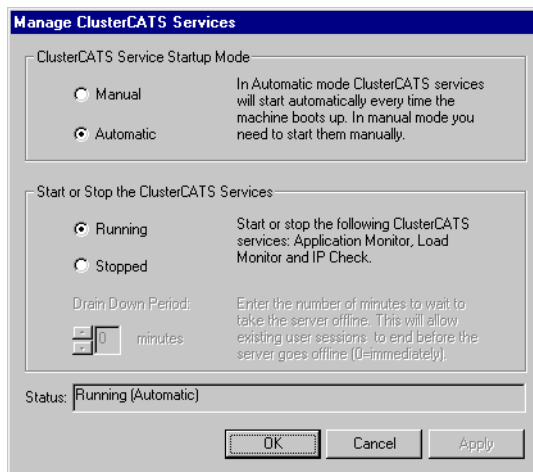
Periodically you will need to update software or content that resides on your cluster members. Software updates include new versions or patches of operating system software, web server software, new web applications, ClusterCATS software, or other third-party products.

ClusterCATS lets you put an active cluster member in maintenance mode and then bring it online slowly so you can verify that your changes do not introduce new problems. This section describes how to do this.

To update an existing cluster member with new software or content:

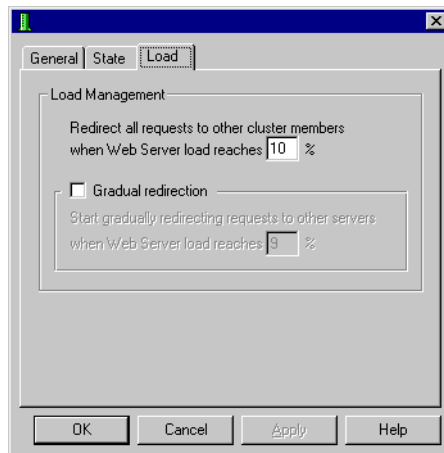
- 1 Put the server in maintenance mode using the instructions in [“Using maintenance mode \(Windows only\)” on page 115](#).
- 2 Make your updates to the inactive server.
- 3 Open a web browser on the cluster member and enter the server name associated with the maintenance address defined for the server. For example, `serv1.mycompany.com`. If you configured the maintenance address correctly as described in [“ClusterCATS dynamic IP addressing \(Windows only\)” on page 132](#), your site appears in the browser.
- 4 When you have verified your changes, exit the browser.
- 5 Open the ClusterCATS Server Administrator utility on the server by selecting **Start > Programs > Macromedia > ClusterCATS Server Administrator**.
- 6 Click the BT Service Status button.

The Manage ClusterCATS Services dialog box appears:

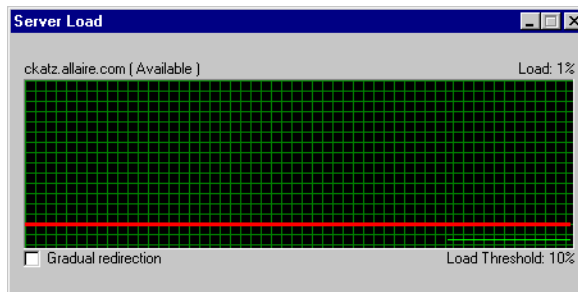


- 7 Select **Running**.
ClusterCATS adds the cluster member back into the cluster.

- 8 To initially limit the amount of HTTP traffic sent to the server, return to ClusterCATS Explorer and reconfigure the cluster member's peak load threshold to a low value such as 10%.



- 9 Click OK.
- 10 In ClusterCATS Explorer, right-click a cluster member and select **Monitor > Load**. The Server Load monitor appears:



- 11 Observe your cluster member at low usage levels until you are satisfied that your new changes are working properly.
- 12 When you are certain that the updates you made have not adversely affected the server's operation, set the peak and gradual redirection load thresholds back to their original values.

Resetting cluster members

ClusterCATS includes a utility for resetting cluster members to their preclustered state. You may want to do this for two reasons:

- To permanently remove a cluster member from a cluster
- To change a cluster member from one cluster to another

To perform these tasks, you must first reset each server's configuration to its original, preclustered state. This section describes the following:

- [“Resetting cluster members on Windows” on page 120](#)
- [“Resetting cluster members on UNIX” on page 120](#)

Resetting cluster members on Windows

You must use the ClusterCATS Server Administrator that is installed on each cluster member for the following reasons:

- Using ClusterCATS Explorer to delete cluster members from a cluster does not delete the server's ClusterCATS configuration, which is stored in the server's registry.
- Running the ClusterCATS uninstall program and reinstalling does not overwrite the server's ClusterCATS configuration.

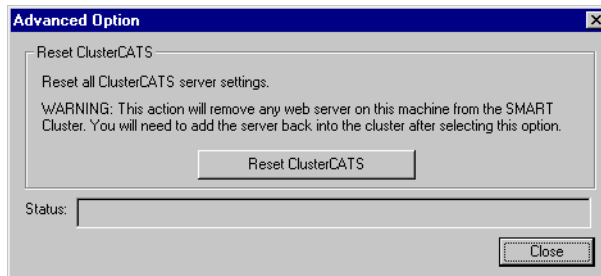
To reset a server to its preclustered state:

- 1 Open the ClusterCATS Server Administrator on this server by selecting **Start > Programs > Macromedia > ClusterCATS Server Administrator**.

The ClusterCATS Server Administrator appears.

- 2 Click **Advanced**.

The Advanced Option dialog box appears:



- 3 Click **Reset ClusterCATS** to remove the ClusterCATS configuration from the server. A message appears confirming that the server has been reset.
- 4 Exit the ClusterCATS Server Administrator.

Resetting cluster members on UNIX

Enter the following command at the server you want to reset:

```
btadmin -reset
```

CHAPTER 6

ClusterCATS Utilities

ClusterCATS ships with scriptable command-line utilities for configuring, administering, and troubleshooting ClusterCATS clusters. This chapter describes these utilities.

Contents

- Using `btadmin`..... 122
- Using `bt-start-server` and `bt-stop-server` (UNIX only) 125
- Using `btcfgchk` 126
- Using `hostinfo` 129
- Using `sniff` 130

Using btadmin

btadmin is a scriptable utility installed on each server in a cluster. It provides most of the functionality of the Windows-based ClusterCATS Server Administrator so UNIX and Windows administrators can include calls in automated scripts.

This section describes the following:

- [“Using btadmin on Windows” on page 122](#)
- [“Using btadmin on UNIX” on page 122](#)

Using btadmin on Windows

btadmin is a Windows executable invoked from the command line in the <CC_install_directory>/program directory.

The table below describes each of the options and their syntax for btadmin.

Option	Description
btadmin	Displays btadmin online help.
btadmin -v	Displays the current version of Microsoft’s IIS if it is bound to the ClusterCATS Server.
btadmin -f	Removes the ClusterCATS Web server filter and virtual directories.
btadmin +f	Adds the ClusterCATS filter to your web server.
btadmin -b	Stops ClusterCATS services.
btadmin +b	Starts ClusterCATS services.
btadmin +m	Reconfigures ClusterCATS services to manual start mode.
btadmin -m	Reconfigures ClusterCATS services to automatic start mode.
btadmin -r	Removes servers, delete database files and registry keys related to servers
btadmin -s <seconds>	Puts server into maintenance mode after a set delay (in seconds). This shuts down ClusterCATS services. For more information, see “Using maintenance mode (Windows only)” on page 115 .

You can invoke btadmin with more than one option. For example, to stop and restart ClusterCATS services, enter btadmin -b +b.

Using btadmin on UNIX

The btadmin utility on UNIX is a shell script invoked from the <CC_install_directory>/ directory. If you run btadmin on Red Hat Linux, the ksh shell must be installed.

The syntax for btadmin is:

```
btadmin [start | stop | restart <daemon>]
btadmin [enable | disable | add | delete | config <option><instance>]
btadmin [show | reset | help]
```

The following sections describe each of these options.

[start | stop | restart <daemon>]

You can start, stop and restart the following daemons with `btadmin`:

Daemon	Description
apmgr	Application manager daemon.
dfp	Cisco LocalDirector's Dynamic Feedback Protocol daemon.
failover	Failover daemon.
ipaliasd	ClusterCATS failover daemon.
ns-httpd	HTTP daemon.
reqmgr	Solaris-only authentication daemon.
teserver	ClusterCATS Server process on Apache. This option is an alias for <code>teserver_apache</code> . On NES, the ClusterCATS Server is run in the context of the web server.
wsprobe	Web server probe daemon.

Note: Stopping and starting some daemons can result in multiple daemons being stopped or started.

Following are examples of how you start and stop daemons with the `btadmin` utility:

```
btadmin start apmgr
btadmin stop failover
btadmin restart ns-httpd
```

[enable | disable | add | delete | config <option> _ <Web_server_instance>]

The following table describes the `btadmin` options for changing ClusterCATS settings:

Option	Description
enable	Enable the specified option for a web server instance.
disable	Disable the specified option for a web server instance.
add	Add a new web server instance.
delete	Delete an existing web server instance.
config	Configure a specified option for an instance. <code>btadmin</code> prompts you for additional information when using the <code>config</code> option.

For Netscape web servers, enter the web server instance as `https-<server>`. For Apache web servers, enter `https-<hostname>`.

You can enable, disable, and configure the following ClusterCATS options using the `btadmin` utility:

Option	Description
<code>btcats</code>	Configures the ClusterCATS Server.
<code>dfp</code>	Configures Cisco LocalDirector's Dynamic Feedback Protocol.
<code>failover</code>	Configures the ClusterCATS failover (<code>ipaliasd</code>) support.
<code>load</code>	Configures the load-balancing preferences.
<code>wsroot</code>	Configures a web server root directory in case you upgrade your installation or move the root directory.
<code>wsprobe</code>	Configures the web server probes.

The following examples show how to use the `btadmin` utility:

```
btadmin add https-myserver
btadmin enable btcats https-myserver
btadmin disable failover https-myserver
btadmin config load https-myserver
```

[show]

Use the `show` option to display the currently enabled ClusterCATS configuration settings.

[reset]

Use the `reset` option to reinitialize cluster configuration settings on the current server. For more information, see [“Resetting cluster members” on page 120](#).

[help]

Use the `help` option to get a list of the `btadmin` utility's features and syntax.

Using bt-start-server and bt-stop-server (UNIX only)

The `bt-start-server` and `bt-stop-server` utilities start and stop the web server that is bound to the ClusterCATS Server. This command starts or stops either the Netscape Enterprise Server or Apache Web Server.

`bt-start-server` and `bt-stop-server` are invoked from the command line in the `<CC_install_directory>/` directory using the following syntax:

```
bt-start-server
```

```
bt-stop-server [-f]
```

Use the `-f` option to stop the web server without being prompted for confirmation.

Using btcfgchk

The `btcfgchk` utility is a network management tool that displays information about your IP and DNS configurations. Use it to analyze and troubleshoot the servers and network.

Syntax

Invoke `btcfgchk` from the command line in the `<CC_install_directory>/` directory using the following syntax:

```
btcfgchk
```

Sample output

The following sample output shows how `btcfgchk` displays configuration information for a system with one network adapter and two IP addresses:

```
btcfgchk FQHN is hartford.brighttiger.com  
E190x1 [PRIMARY]:
```

```
hartford.brighttiger.com      192.168.0.31  
255.255.255.0  
hartford.brighttiger.com
```

```
hartford1.brighttiger.com     192.168.0.32  
255.255.255.0  
hartford1.brighttiger.com
```

btcfgchk DNS errors

The `btcfgchk` utility reports on DNS configuration problems. ClusterCATS requires that your DNS be configured with correct forward and reverse mappings. A forward mapping (AName record) translates the host name to an IP address. A reverse mapping

(PRT record) translates an IP address to its host name. ClusterCATS expects the mapping to be one-to-one (one host name to one IP address).

Error	Description
Host name does not map to a single IP address	<p>The main host name for this system is not mapping to one IP address. Possible problems are:</p> <ul style="list-style-type: none"> The main host name of the system could not be resolved to an IP address. Your fully qualified host name is the combination of the host name and the domain name. Ensure that no typos appear in these names in DNS definitions, on the DNS server and on each cluster member's DNS definition. To verify that the host name is correct, enter <code>nslookup <FQHN></code> at a command-line prompt (FQHN stands for fully qualified hostname). The host name is a round-robin DNS name. Run the ClusterCATS <code>hostinfo</code> utility to see whether more than one IP address is configured for the domain. For more information, see "Using hostinfo" on page 129.
No adapter associated with host name found	<code>btcfgchk</code> is unable to find the primary network adapter. The primary network adapter should be the network adapter containing the IP address of the main host name.
Duplicate Primary Adapter	<code>btcfgchk</code> found two network adapters with the same IP address. Use the <code>ifconfig -a</code> command to see information about your adapter.
Name lookup for <hostname> failed	<code>btcfgchk</code> was not able to determine the IP address for the specified host. Your DNS server may be down. Use <code>nslookup</code> to see whether it can contact your DNS server.
<IP_address1> reverse maps to <hostname> which then forward maps to <IP_address2>	<p><code>btcfgchk</code> did a lookup on <IP_address1> and found a host name to which it is mapped. It then attempted to verify that the host name maps back to the IP address specified, and the verification failed.</p> <p>There is likely an issue with your DNS configuration. Use the ClusterCATS <code>hostinfo</code> utility to get more information on how the names and IP addresses are configured. For more information, see "Using hostinfo" on page 129.</p>
Error looking up <hostname> by name	ClusterCATS could not resolve the given host name to an IP address. Use <code>nslookup</code> to look up the host name in DNS.
Host name a round-robin name, or does not map to configured IP address	<p>The host name maps to more than one IP address (round-robin DNS) or maps to an IP address not found on the computer. Use the ClusterCATS <code>hostinfo</code> utility to check the host name DNS configuration:</p> <pre>hostinfo <hostname></pre> <p>If more than one IP address is listed, round-robin DNS is in use. If one IP address is listed, check whether the address is configured on the computer. You can use the <code>ipconfig/all</code> command to view all IP addresses on this computer.</p>

Error	Description
Host name not found in any reverse mapping Probable forward mapping misconfiguration for <hostname>	For each IP address found on the system, an attempt was made to find the corresponding host name. None of the IP addresses on the system reverse mapped to the system's main fully qualified host name. The problem is either: <ul style="list-style-type: none"> • The host name maps to the wrong IP address. • The IP address that the host name maps to does not have an entry in the DNS table for the reverse map. Consequently, nslookup does not return the hostname.
Probable round robin configuration for <hostname>	The host name does not map to one IP address. Use the <code>hostinfo</code> tool to determine to which IP address it maps. For more information, see “Using hostinfo” on page 129 .

Using hostinfo

The `hostinfo` utility is a network management tool that displays information about a domain name. Use it to analyze and troubleshoot problems with DNS mappings to a domain.

Syntax

Invoke `hostinfo` from the command line in the `<CC_install_directory>/` directory using the following syntax:

```
hostinfo [fully_qualified_host_name]
```

Specifying a fully qualified host name is optional. If you do not specify one, then `hostinfo` returns information about the current host.

Sample output

The following sample output from the `hostinfo` utility provides information about a set of round-robin DNS host names.

```
>hostinfo macromedia.com
```

```
Information for host 'macromedia.com':
```

```
  FQHN: macromedia.com
```

```
  Primary Address: 0.0.0.0
```

```
  Domain: .com
```

```
  Aliases:
```

```
  macromedia.com
```

```
  www1.macromedia.com
```

```
  www2.macromedia.com
```

```
  www3.macromedia.com
```

```
  Addresses:
```

```
  205.181.25.81
```

```
  205.181.25.82
```

```
  205.181.25.83
```

The `hostinfo` utility displays the domain name, the primary IP address, and any IP aliases. If the primary IP address is set to `0.0.0.0`, the domain is using round-robin DNS. The round robin names appear under the Alias section of the DNS table and the round-robin addresses appear under the Addresses section.

Using sniff

The `sniff` utility is a network management tool that displays the packets that a specific network interface card (NIC) is hearing.

Syntax

Invoke `sniff` from the command line in the `<CC_install_directory>/program` directory using the following syntax:

```
sniff
```

Sample output

Below is sample output from the `sniff` utility:

Mail Test Environment Variables:

BTMailHost, BTSender, BTRecipients, BTSubject, BTText

Packet Test Environment Variables:

BTPort, BTMcastTTL, BTUcastCount, BTBcastCount, BTMcastCount
BTSendInterval, BTDoLocalBind, BTUcastAddress, BTBcastAddress
BTMcastAddress, BTLocalAddress, BTSendSize, BTRecvSize
BTConsole, BTLogFile, BTSystem

Press keys at run-time:

d - dump sniff configuration information
H - display this and more help
h - display this help
l - run load balance test thread
m - run mail test thread
p - toggle packet dump display
q, <ESC>, <ENTER> - quit all active threads and exit
r - run UDP listener thread
s - run packet test thread
x - execute system command

Use the "r" command within `sniff` to listen to intra-cluster packets:
Listen Thread thread running on 'any' interface...

```
[ SrvHello @ Tue Jun 30 17:01:57 1998] 192.168.0.213
boston1.brighttiger.com (192.168.0.118 ) (255.255.255.0 )
sales_automation Mcast V1.2 Available 2/90
[[ SrvHello @ Tue Jun 30 17:01:57 1998] 192.168.0.213
somewhere.brighttiger.com (192.168.0.213 ) (255.255.255.0 )
```

CHAPTER 7

Optimizing ClusterCATS

ClusterCATS provides enhanced capabilities that let you customize your ClusterCATS implementation. This chapter describes some of these options.

Contents

- ClusterCATS dynamic IP addressing (Windows only) 132
- Using server failover..... 137
- Configuring load-balancing metrics 138

ClusterCATS dynamic IP addressing (Windows only)

This section describes how to enable ClusterCATS dynamic IP addressing on your site. You do not have to configure your system on UNIX for dynamic IP addressing, because it is set up by default.

If your site is already configured so the IP address for the computer name is different from the IP address(es) for the websites configured on this server, you can skip [“Setting up maintenance IP addresses” on page 133](#) and continue with [“Enabling ClusterCATS dynamic IP addressing” on page 135](#).

Understanding static and dynamic IP address configurations

Each server that you add to a cluster must have an IP address defined for it. Because the Internet operates on a TCP/IP network protocol for sending and receiving packets of data to and from networked computers, you must correctly define your servers' IP addresses so they can send and receive network data as intended.

The static address must be assigned to the server itself — the physical box. You do so by making an entry in the server's IP stack. On Windows servers, you add the IP address using the Network icon in the Control Panel. The Network icon is also commonly referred to as your network interface card (NIC).

You must also ensure that the websites' static IP addresses that reside on the web server on this computer are removed from the IP stack (also with the Network icon in the Control Panel). Typically, someone added the website IP addresses to the server's IP stack before installing ClusterCATS and creating clusters. You must manually remove those IP addresses so ClusterCATS can *dynamically* create them in the IP stack according to server load and availability in the cluster.

There are generally two ways to move from static to dynamic addressing:

- Change the IP address and FQHN of the website
- Change the address and FQHN of the web server's computer

Because most webmasters cannot change the website address, the instructions in this section explain how to change the computer or machine name.

Note: Computer names associated with the ClusterCATS dynamic IP addresses must have fully qualified host names (FQHNs) in DNS and DNS forward and reverse entries.

The procedure for configuring ClusterCATS with dynamic IP addressing is as follows:

- 1 Set up your servers with maintenance addresses. See [“Setting up maintenance IP addresses” on page 133](#).
- 2 Install ClusterCATS. See [“Installing ClusterCATS on Windows” on page 41](#).
- 3 Enable ClusterCATS dynamic IP addressing. See [“Enabling ClusterCATS dynamic IP addressing” on page 135](#).
- 4 Create your clusters. See [“Creating clusters in Windows” on page 54](#).

Benefits of ClusterCATS dynamic IP addressing

There are several benefits to your using ClusterCATS dynamic IP addressing:

- Using maintenance mode — with dynamic IP addressing, cluster members put into maintenance mode on Windows clusters will fail over to another server and then gracefully return when brought out of maintenance mode. For more information, see [“Using maintenance mode \(Windows only\)” on page 115](#).
- Using maintenance IP addresses — if you use ClusterCATS dynamic IP addressing, you can remotely access servers in your cluster if they fail or become unavailable through the maintenance address. Maintenance addresses are statically bound to the server during the setup for ClusterCATS dynamic IP addressing. For more information, see [“Setting up maintenance IP addresses” on page 133](#).
- Optimizing Server failover — on Windows systems, when ClusterCATS is configured using static IP addresses, IP address conflicts occur when the failed server recovers from a failover and tries to re-claim its IP address. This IP conflict is cleared when the failed server automatically restarts. ClusterCATS dynamic IP addressing prevents this double-restart.

Setting up maintenance IP addresses

Setting up a maintenance IP address ensures that you have one static IP address on the system that is not assigned to a web server, virtual server, or website. This address, often referred to as the system’s “maintenance address,” provides administrators with a consistent way to access the system remotely at all times. It also lets ClusterCATS communicate with the server in case of a web server failure.

Note: You must have at least two IP addresses available for a computer in order to use one for a maintenance IP address.

This section explains how to add a maintenance address that supports ClusterCATS dynamic IP addressing. If your server has only one static address that corresponds to both the computer name and the website, you must reconfigure it to allow for a maintenance address.

Note: This procedure must be performed on each system in the cluster and must be done before installing ClusterCATS.

To set up a maintenance address before installing ClusterCATS:

- 1 Back up your system files.
- 2 Obtain a new IP address and new computer name. Be sure to configure your DNS so your new address has both forward and reverse DNS entries.
- 3 **For IIS 4.0 and 5.0:** Uninstall products that are configured as part of IIS, including JRun.
- 4 **For IIS 4.0:** Uninstall the Windows NT 4.0 Option Pack (which includes IIS) by selecting Start > Settings > Control Panel > Add/Remove Programs. Restart the server.
For IIS 5.0 or NES: Skip this step.

- 5 Open the Advanced IP Addressing dialog box by right-clicking Network Neighborhood. Select Properties. On the Protocols tab, select TCP/IP Protocol and click Properties and then click Advanced.



- 6 Select the computer's primary NIC in the Adapter field. Add the new IP address in the IP Addresses region. You will use this address as the maintenance address and machine address. Make a note of all IP addresses on the NIC.
- 7 Click OK, and OK again and click the Identification tab. Click Change.
- 8 Enter a new name for the computer in the Computer Name field. The name corresponds to the new IP address that you just added. Do not change the Domain field on this tab.

Note: The computer name on the Identification tab should only be a NetBIOS name, not a fully qualified host name (FQHN). For example, support1.macromedia.com is a possible FQHN. The first portion of this FQHN (support1) can be a NetBIOS name. Note that support1 would also appear as the host name under the DNS tab in Protocols. The domain under the DNS tab in this case would be macromedia.com. The Domain field on the Identification tab is different; it has nothing to do with DNS but only corresponds to your NT domain.

- 9 Close all open dialog boxes and restart the server.
- 10 **For IIS 4.0:** Reinstall the NT 4.0 Option Pack and then restart the server.
For IIS 5.0 or NES: Skip this step.
- 11 **For IIS 4.0:** You may need to reconfigure your websites using the Internet Service Manager.
For IIS 5.0 or NES: Skip this step.

- 12 Reinstall products that are configured as part of IIS, including JRun/ColdFusion and ClusterCATS. This should include any products you uninstalled in step 3.
When you install ClusterCATS, you must select the "Server Failover" option during the installation procedure.
Note: Do not create any clusters at this time.
- 13 Enable the ClusterCATS dynamic IP addressing scheme using the procedure described in ["Enabling ClusterCATS dynamic IP addressing" on page 135](#).

Enabling ClusterCATS dynamic IP addressing

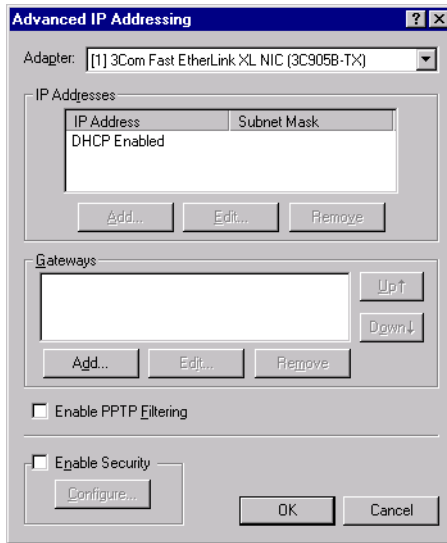
Before enabling the ClusterCATS dynamic IP addressing, you must have set up a maintenance IP address for each web server in the cluster, as described in ["Setting up maintenance IP addresses" on page 133](#), and bound any websites to the appropriate IP addresses. The maintenance IP address must be different from the IP address associated with the website.

This section instructs you to create the cluster while the website is still bound to the IP address. When creating a cluster, you should not specify the maintenance address. When you test the cluster, you can remove the websites from the IP addresses and restart. ClusterCATS creates the address dynamically when the server restarts.

To enable dynamic addressing:

- 1 Verify that you can access your server by its maintenance address. If not, assign one to the server using the procedure described in ["Setting up maintenance IP addresses" on page 133](#).
- 2 Configure your web server to support ClusterCATS dynamic IP addressing.
For Netscape Enterprise Server: Verify that the IP addresses associated with the primary web server and hardware virtual servers are configured on your system by the Network Control Panel. If the addresses are not configured, the Netscape Enterprise Server will fail to start. In order for failover to work properly, the primary web server cannot be bound to a specific IP address. If it is, remove the binding using the Netscape Administrative Server.
For IIS: Verify that you have a unique IP address (or addresses) assigned to each website on the web server in the MMC. If IP addresses are not assigned to your web server yet, assign them now. With IIS 4.0, you may have to manually enter the IP address, if it does not appear in the drop-down list on the Web Site properties tab.
- 3 Restart your server to apply these changes.
- 4 Create a cluster using the Cluster Setup Wizard.
Note: Do not specify a maintenance address when adding cluster members. Because the IP addresses for the cluster members are still bound to their NICs, there is no need to do this. For more information, see ["Creating clusters with the Cluster Setup Wizard" on page 54](#).
- 5 Verify that your cluster is functioning properly.

- 6 Open the Advanced IP Addressing dialog box by right-clicking Network Neighborhood, and select Properties. On the Protocols tab, select TCP/IP Protocol and click Properties. Click Advanced.



- 7 Unbind the IP addresses from the web server's NIC by selecting each IP address in the IP Addresses region and clicking Remove. This step removes the IP addresses corresponding to the website.
- 8 Click OK, three times.
- 9 Restart all the systems in the cluster.
ClusterCATS assigns the IP addresses dynamically to your web servers.

Using server failover

The ability to fail over servers that have become unavailable to redundant servers is a cornerstone of any mission-critical application, one that ensures continuous, reliable operation. Server failover is an option to select during installation. If you did not select it, you must reinstall ClusterCATS to enable it.

Static versus ClusterCATS dynamic IP addressing

There are two schemes with which you implement server failover:

- Static IP addressing — when a computer fails, IP address(es) bound to its web server are reassigned to the most available cluster member's web server. When the failed-over server comes back online, it must claim the IP address and then restart again.
- Dynamic IP addressing — ClusterCATS can be configured to dynamically assign IP addresses so that when a server fails, its IP address(es) can be assigned to other servers. When the failed-over server comes back online, ClusterCATS returns the IP addresses to it without conflict.

On Windows clusters, you should use server failover with the ClusterCATS dynamic IP address scheme. To configure ClusterCATS dynamic IP addresses, the IP address associated with the computer name must be different from the IP addresses associated with the websites. ClusterCATS refers to the IP address associated with the computer name as the maintenance address. For more information, see [“ClusterCATS dynamic IP addressing \(Windows only\)” on page 132](#).

Windows domain controllers

If you use Windows NT Domain server authentication, each web server in a cluster must participate as a member NT server in a domain. Do not set a server in your cluster the primary domain controller (PDC). Server failover will interfere with the function of the PDC. You can set an NT server as a backup domain controller, but this configuration is not recommended.

Configuring load-balancing metrics

You have the option to customize the load-balancing metrics of web servers clustered with ClusterCATS software. This section describes how to customize metrics to your specific website implementation.

Overview of metrics

The JRun and ColdFusion servers record how long it takes to process each request, and can return metrics derived from this data, upon request. These metrics include the following:

- **Average Request Time** (default) — the average processing time of all requests within a one-minute moving window. The use of an average smooths the effects of brief spikes in request volume, and in a mixture of short- and long-running requests.
- **Round Trip Time** — the round-trip time it takes to process a request.
- **Last Request Time** (ColdFusion only) — the time it takes to process the last request to the server. Because it is a single, undiluted snapshot of the request time, it immediately reflects peaks and troughs in request processing time.

To translate these metrics into a single load value for the web server, they must be weighed against a subjective measure of server performance — a maximum acceptable response time. The maximum indicates the upper threshold of performance at which a server is declared “busy” for load-balancing purposes. When a server reaches this critical threshold, ClusterCATS redirects further service requests away from the server until it becomes more responsive to its clients.

ClusterCATS software provides a further enhancement in load-balancing options. A ClusterCATS agent process probes a special page, `getsimpleload.jsp` for JRun and `getsimpleload.cfm` for ColdFusion, every five seconds, and records the round-trip time (RTT) for each request. From this data, it computes its own average RTT over a one-minute moving window. You can edit this file to customize load-balancing options for your application.

This external view of request time accounts for the processing time of the request itself, but, more importantly, for other system overhead included in reaching the web server and receiving an acceptable response. By factoring in external influences on web server responsiveness, such as network load, scheduling load, and disk I/O load, the ClusterCATS probe agent can adjust the load reported by JRun or ColdFusion, to generate a more realistic picture of the web server's performance for its clients.

For example, if the JRun server reports a light load of requests, but the probe agent finds significant round-trip times to and from the web server, then it reports a proportionally higher load for server than JRun reports.

Load types

The page ClusterCATS probes for load-balancing options, `getsimpleload.jsp` for JRun and `getsimpleload.cfm` for ColdFusion, is located in `<CC_install_directory>/btauxdir`. The probe agent responds to output generated by this page and uses it to calculate the overall load, based on the weighting of the metrics set in the `LOADTYPE` variable:

- `AVG_REQ_TIME` — calculates load based on the average service request time. The load is derived by dividing the request time by the maximum acceptable request time. This is the default metric.
- `ROUND_TRIP_TIME` — calculates load based on the round-trip time for the request. This metric leaves all load calculation to the probe agent.
- `PREV_REQ_TIME` (ColdFusion only) — calculates load based on the time to process the last request to the server.

For servers that process database-intensive requests, `ROUND_TRIP_TIME` is not a good indication of load, because JRun/ColdFusion processes the threads that calculate `ROUND_TRIP_TIME` differently than queued database connection requests. With this in mind, if your web server uses many concurrent connections to a database, either use `AVG_REQ_TIME` as your load type, or include a database call in `getsimpleload.jsp/getsimpleload.cfm` to make this load type's results more indicative of actual conditions.

Output variables

During processing, `getsimpleload.jsp/getsimpleload.cfm` generates three significant output variables that are sent in response to the probe agent's HTTP query:

- `CCLOADVALUE` — the load calculated by `getsimpleload.jsp/getsimpleload.cfm` using one of the available load metrics. The load value identifies how busy the server is as a percentage of its total capacity.
- `CCLOADMAX` — the maximum acceptable time (in milliseconds) for a request to complete; marks the "busy threshold" for this server. The load percentage calculation is based on it, given the results of the `AVG_REQ_TIME` metric. The default maximum is 8 seconds (8000 ms), but this value is arbitrary and you should customize it to fit the capacity and expectations of your particular website.

If you increase the value of `CCLOADMAX`, the server can take longer (on average) for each request before the server is declared busy. If you decrease `CCLOADMAX`, the server's average request is shorter when the server is declared busy.

- `CCRTTPercent` — the percentage of the calculated average `ROUND_TRIP_TIME` that the probe agent applies to the load metric supplied by `CCLOADVALUE`. `CCRTTPercent` is the second variable that you might change to customize your server's load metrics. It lets you "tune" the amount of external influence on server performance to calculate into the server's overall load value.

For example, increase `CCRTTPercent` to apply a greater weighting to the `ROUND_TRIP_TIME` metric in the overall load calculations. The default value of `CCRTTPercent` is 0 (disabled). If you change the load type to `ROUND_TRIP_TIME`, the default value of `CCRTTPercent` is 100, which gives `ROUND_TRIP_TIME` the maximum weighting.

Troubleshooting the load-balancing metrics

If ClusterCATS gets an exception every time it processes `getsimpleload.jsp`, you may have installed ClusterCATS before installing JRun. In this case, verify that the following is true:

- `JRunMetricThread.class` file is located in `/jrun/lib/ext`
- The virtual directory `/btauxdir` is configured on your web server. (This was created during installation, but you might have removed it.)

INDEX

A

- A records 19
- absolute hyperlinks 72
- active mode 110
- active/passive mode
 - changing 111
 - changing in UNIX 112
 - changing in Windows 111
- adding cluster members
 - UNIX 64
 - Windows 63
- Admin Agent 62
- Admin Manager 62
- administering ClusterCATS
 - alarm notifications 98
 - Apache considerations 50
 - btadmin 53, 122
 - bt-start-server 125
 - bt-stop-server 125
 - ClusterCATS Explorer 48
 - ClusterCATS Web Explorer 49
 - e-mail support options 100
 - introduction 48
 - Netscape considerations 50
 - opening the Web Explorer 51
 - scripting 122
 - security 103
 - Server Administrator 52
 - server load threshold 66
- administrator alarm
 - notifications 98
- after you install 45
- alarm notifications
 - configuring on UNIX 99
 - configuring on Windows 98
 - overview 98
 - types 98

alarms

- See* alarm notifications

Apache 50

applications

- database locking 14
- load management 3
- load testing 20
- partitioning 15
- scalability bottlenecks 16
- state management 13

authentication

- configuring on UNIX 106
- configuring on Windows 103
- disabling 106
- domain 105
- local user 103
- NT Domain 105

availability and reliability

- common failures 24
- defined 23
- elements of 23
- failover considerations 25
- sample scenario 25

average request time 138

AVG_REQ_TIME 139

avoiding bottlenecks 16

avoiding double-reboot 133

B

backup servers 26

before you install 34

- configuring DNS 34
- domain controllers 40
- dynamic IP addressing 38
- firewalls 38
- maintenance IP addresses 38, 133

server failover 38

- website content 39

bottlenecks, avoiding 16

btadmin

- described 53
- use 122
- Windows syntax 122

btcfgchk

- DNS Errors 126
- sample output 126
- syntax 126

bt-start-server 125

bt-stop-server 125

btweb 49

busy state 115

C

CCLOADMAX 139

CCLOADVALUE 139

CCRTTPercent 139

Cisco LocalDirector 92

- DFP Agent Listen Port 92

- dynamic IP addressing 92

- dynamic-feedback command 93

- gradual redirection 92

- integrating with

- ClusterCATS 93

- passive mode 92

- round-robin DNS 92

cluster maintenance mode 115

cluster members

- adding (UNIX) 64

- adding (Windows) 63

- busy state 115

- changing state 111

- gradual redirection threshold 66

- load status 68

- load threshold, adjusting 68

- load thresholds 66
 - maintenance mode 115
 - maintenance support 60
 - moving to cluster 120
 - peak load threshold 66
 - preclustered state 120
 - probes and monitors 77, 84
 - removing (UNIX) 65
 - removing (Windows) 65
 - restricting 113
 - updating 118
 - Cluster Setup Wizard 54
 - ClusterCATS administration 53
 - ClusterCATS components
 - btadmin 53
 - Explorer 48
 - overview 6
 - Server 48
 - Server Administrator 52
 - Web Explorer 49
 - ClusterCATS Explorer
 - defined 48
 - icon legend 49
 - interface 49
 - ClusterCATS Server
 - Administrator 52
 - ClusterCATS Server, defined 48
 - ClusterCATS Web Explorer 49
 - Apache considerations 50
 - Netscape considerations 50
 - opening 51
 - clustering
 - defined 28
 - hardware considerations 30
 - hardware-based advantages 29
 - hardware-based solutions 29
 - illustrated 28
 - intelligent vs. nonintelligent 28
 - software considerations 31
 - software-based advantages 31
 - software-based solutions 30
 - techniques 28
 - viewing server load 68
 - clusters
 - adding members 63
 - adding members (UNIX) 64
 - adding members (Windows) 63
 - alarm notifications 98
 - creating 54
 - creating manually 59
 - creating UNIX 60
 - creating Windows 54
 - creating with Cluster Setup Wizard 54
 - moving members among 120
 - removing members (UNIX) 65
 - removing members (Windows) 65
 - restricting members 113
 - clusters, defined 30
 - CNAME records 19
 - Macromedia ColdFusion
 - See also* ColdFusion
 - Macromedia JRun
 - See also* JRun
 - com port on web server 50
 - common failures 24
 - concurrency 14
 - creating clusters 54
 - in UNIX 60
 - in Windows 54
 - manually 59
 - Windows 54
 - with hardware solutions 29
 - with software solutions 30
- D**
- databases
 - concurrency issues 14
 - locking mechanisms 14
 - deleting clusters 62
 - DFP Agent Listen Port 92
 - DFP hosts 93
 - DHCP 34
 - disable mode 106
 - disabling authentication 106
 - disk failures 98
 - DNS
 - before you install 34
 - btcfgchk 126
 - configuring 36
 - core elements 18
 - defined 17, 34
 - domains 18
 - forward DNS entries 37
 - local and primary servers 34
 - local servers 35
 - name servers 19
 - primary servers 34
 - record types 19
 - round-robin 37
 - round-robin, example 20
 - sample table entries 20
 - scalability 17
 - server aliases 19
 - tables, examples 37
 - troubleshooting 129
 - zones 18
 - domain authentication for
 - clustering 105
 - domain controllers 40
 - domains
 - DNS 18
 - hostinfo 129
 - double-reboot, avoiding 133
 - Dynamic Host Configuration Protocol 34
 - dynamic IP addressing
 - before you install 38
 - benefits 133
 - choosing during installation 42
 - enabling 135
 - maintenance IP addresses 133
 - maintenance mode 133
 - optimizing failover 133
 - static vs. dynamic addressing 132
 - with LocalDirector 92
 - dynamic-feedback command 93
 - dynamic-feedback-pw 93
- E**
- e-mail
 - alarm notifications 98
 - reports 100
 - e-mail support
 - configuring on UNIX 101
 - configuring on Windows 100
 - options 100
 - events, alarm notifications 98
- F**
- failover
 - backup servers 26
 - before you install 38
 - choosing during installation (UNIX) 43
 - considerations 25
 - corrective actions 26
 - described 137
 - domain controllers 137
 - hardware planning for 26

- optimizing with dynamic IP addressing 133
 - parallel servers 26
 - persistent sessions 74
 - static vs. dynamic IP addressing 137
 - systems monitoring 26
 - Web server alarm notification 98
- failures
 - alarm notifications 98
 - common 24
 - disk 98
 - HTTP server 98
 - probes 98
 - server busy 98
 - server unreachable 98
 - web server failover 98
- firewalls
 - before you install 38
 - configuring 38
 - port 9123 39
 - port 9129 39
 - scalability 16
- forward DNS entries 37
- G**
 - getsimpleload.jsp
 - description 138, 139
 - gradual redirection
 - LocalDirector 92
 - threshold 66
- H**
 - hardware planning for failover 26
 - hardware-based clustering
 - advantages 29
 - considerations 30
 - illustrated 29
 - solutions 29
 - high availability, probes and monitors 77, 84
 - hostinfo 129
 - HTTP redirection 66
 - HTTP server failure 98
 - hyperlinks, relative 72
- I**
 - icon legend 49
 - installation 41
 - after you install 45
 - before you install 34
 - UNIX 42
 - UNIX vs. Windows 6
 - Windows NT 41
 - installation, support viii
 - integrating ClusterCATS with LocalDirector 93
 - IP aliasing
 - See dynamic IP addressing
- J**
 - JDBC, session swapping 76
 - JRun probes 77, 84
- L**
 - linear scalability 11
 - load balancing
 - combining hardware and software 93
 - configuring load thresholds 66
 - configuring metrics 138
 - configuring probes 77, 84
 - integrating ClusterCATS with other devices 92
 - issues related to scalability 12
 - metrics 138
 - session-aware 72
 - session-aware on UNIX 72
 - session-aware on Windows 72
 - software-based 30
 - third-party devices in UNIX 97
 - third-party devices in Windows 96
 - using a hardware solution 29
 - using round-robin DNS 30
 - load levels 68
 - load management 3
 - load management method 42
 - load metrics
 - output variables 139
 - troubleshooting 140
 - load monitor 68
 - load status, monitoring 68
 - load testing
 - available web tools 21
 - considerations 22
 - minimizing problems 22
 - reasons to perform 20
 - web applications 20
 - load thresholds
 - adjusting 68
 - configuring 66
 - configuring in UNIX 69
 - configuring in Windows 66
 - load status 68
 - LocalDirector 92
 - peak 66
 - status 68
 - load-balancing devices 92
 - third-party 95
 - UNIX 97
 - Windows 96
 - local DNS servers 35
 - local user authentication 103
- M**
 - Macromedia
 - headquarters x
 - sales x
 - website viii
 - Macromedia ColdFusion
 - developer resources viii
 - documentation, about ix
 - training resources viii
 - Macromedia JRun
 - developer resources viii
 - documentation, about ix
 - training resources viii
 - maintenance IP addresses 133
 - before you install 38
 - setting up 133
 - maintenance mode 110
 - btadmin 122
 - dynamic IP addressing 133
 - upgrading cluster members 118
 - using 115
 - maintenance support 60
 - metrics
 - average request time 138
 - configuring 138
 - last request time 138
 - load-balancing 138
 - output variables 139
 - overview 138
 - round-trip request time 138
 - troubleshooting 140
 - modes 110
 - active/passive 110
 - active/passive settings 111
 - disabled 106
 - maintenance 115
 - restricted/unrestricted 110
 - upgrading cluster members 118

- monitoring load status 68
- monitors 77, 84
 - adding new 78, 85
 - removing in Windows 81, 88

N

- name servers 19
- Netscape, Web Explorer considerations 50
- NT domain authentication 105

O

- optimizing server failover 133

P

- parallel servers 26
- passive mode 110
- passive mode, LocalDirector 92
- peak load threshold 66
- performance, scalability 10
- persistent session failover 74
- PREV_REQ_TIME 139
- primary DNS server 34
- probe monitors
 - adding 78
- probe monitors, adding 85
- probes 77, 84
 - adding in UNIX 81, 89
 - adding in Windows 77, 84
 - adding to existing monitor 80, 87
 - alarm notification 98
 - editing and removing in UNIX 83, 91
 - failure 98
 - multiple 77, 84
 - removing in Windows 81, 88
 - startup parameters 79, 86
- PTR records 19

R

- rebooting, avoiding
 - double-reboot 133
- redirecting traffic 66
- maintenance mode 115
- redundancy
 - ensuring corrective actions 26
 - planning 26
 - systems monitoring 26
- relative hyperlinks 72
- relative vs. absolute hyperlinks 72

- removing cluster members
 - in UNIX 65
 - in Windows 65
- removing clusters 62
- resetting members 120
- reports, e-mail 100
- requests
 - average request time 138
 - round-trip request time 138
- resetting servers to preclustered state
 - btadmin 124
 - description 120
 - UNIX 120
 - Windows 120
- response time 138
- restricted mode 110
- restricted/unrestricted mode 110
- restricted/unrestricted state,
 - changing 113
- restricting cluster members
 - UNIX 114
 - Windows 113
- ROUND_TRIP_TIME 139
- round-robin DNS 30
 - configuration 37
 - LocalDirector 92
 - reverse entries 37
 - tables, example 37
 - using with ClusterCATS 37
- round-trip request time 138
- routers 92
 - Cisco LocalDirector 92
 - load balancing 29
 - load balancing devices 95

S

- scalability
 - application partitioning 15
 - business services 15
 - common bottlenecks 16
 - data services 15
 - databases 17
 - defined 10
 - DNS 17
 - linear 11
 - load management factors 12
 - performance 10
 - presentation services 15
- scalable applications
 - database locking 14
 - session and state 13

- scripting ClusterCATS
 - administration 53
- security
 - authentication 103
 - configuring authentication on UNIX 106
 - configuring authentication on Windows 103
 - configuring domain authentication 105
 - disabling authentication 106
 - local user authentication 103
- server busy warning 98
- server failover
 - before you install 38
 - choosing during installation (UNIX) 43
 - described 137
 - domain controllers 137
 - static vs. dynamic IP addressing 137
- server load
 - adjusting 68
 - monitoring 68
- server load balancing, configuring metrics 138
- server load management 3
- server load thresholds 66
 - configuring in UNIX 69
 - configuring in Windows 66
- server modes 110
- server state, changing 111
- server unreachable 98
- session management 4, 13
 - See also* session swapping
- session management, persistent sessions 74
- session swapping
 - configuring ClusterCATS 74
 - configuring JRun 74
 - JDBC 76
 - overview 74
 - shared files 75
- session-aware load balancing 72
 - enabling on UNIX 72
 - enabling on Windows 72
 - relative vs. absolute hyperlinks 72
- Setup Wizard 54

- sniff
 - sample output 130
 - syntax 130
 - using 130
- software-based clustering
 - advantages 31
 - considerations 31
 - solutions 30
- state management 13
- static vs. dynamic addressing 132
- sticky servers 72
- support options
 - e-mail 100
 - e-mail support on UNIX 101
 - e-mail support on Windows 100
- system requirements 7
 - server on Linux 7
 - server on NT 7
 - server on Solaris 7
 - Web Explorer 8
 - Windows Explorer 8
- systems monitoring for failover 26

T

- technical support, e-mail 100
- testing website load 21
- thresholds 66
 - gradual redirection 66
- training. *See* ColdFusion
- training. *See* JRun
- troubleshooting
 - e-mail support 100
 - load-balancing metrics 140
 - sniff 130
- troubleshooting DNS
 - btcfgchk 126
 - hostinfo 129

U

- unrestricted mode 110
- updating cluster members 118
- upgrading servers 115

V

- virtual servers, hardware-based
 - clustering 29

W

- Web applications
 - database locking mechanisms 14
 - load testing 20
 - managing state 13
 - partitioning 15
 - scalability bottlenecks 16
- Web Explorer
 - Apache considerations 50
 - configuring com port 50
 - limitations 49
 - Netscape considerations 50
 - opening 51
- Web server failover 98
- Web servers
 - configuring com port 50
 - content 39
 - DNS concerns 17
 - responsiveness 138
 - stopping and starting 125
- Website availability and reliability
 - defined 23
 - example 25
 - failover considerations 25
- Website content 39
- Website scalability
 - defined 10
 - implementations 13
 - linear 11
 - load management factors 12
 - performance factors 10

Z

- zones, DNS 18

