



# Configuring and Administering ColdFusion MX



## **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, Drumbear, Drumbear 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 © 1999–2003 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 ZCF61M200

## **Acknowledgments**

Project Management: Randy Nielsen

Writing: Randy Nielsen

Editing: Linda Adler, Noreen Maher

First Edition: August 2003

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

# CONTENTS

<b>INTRODUCTION</b> .....	9
About Macromedia ColdFusion MX documentation .....	9
Documentation set .....	9
Viewing online documentation .....	9

## **PART I: Administering ColdFusion MX**

---

<b>CHAPTER 1: Administering ColdFusion MX</b> .....	13
About the ColdFusion MX Administrator .....	13
Accessing user assistance .....	14
Administrator layout .....	14
Server Settings section .....	14
Data & Services section .....	15
Debugging & Logging section .....	15
Extensions section .....	16
Security section .....	16
<b>CHAPTER 2: Basic ColdFusion MX Administration</b> .....	17
Initial administration tasks .....	17
Server Settings section .....	18
Settings page .....	18
Caching page .....	19
Client Variables page .....	20
Memory Variables page .....	22
Mappings page .....	22
Mail Server page .....	23
Charting Settings page .....	24
Java and JVM Settings page .....	25
Archives and Deployment page .....	25
Settings Summary page .....	26
Data & Services section .....	26
Data Sources page .....	26
Verity Collections page .....	26

Verity K2 Server page . . . . .	27
Web Services page. . . . .	27
Debugging & Logging section. . . . .	27
Debugging Settings page . . . . .	28
Debugging IP Addresses page . . . . .	30
Logging Settings page. . . . .	31
Log Files page . . . . .	31
Scheduled Tasks page . . . . .	32
System Probes page. . . . .	32
Code Compatibility Analyzer page . . . . .	33
Extensions section . . . . .	33
Java Applets page . . . . .	33
CFX Tags page . . . . .	33
Custom Tag Paths page. . . . .	33
CORBA Connectors page . . . . .	33
Security section . . . . .	34
CF Admin Password page. . . . .	34
RDS Password page . . . . .	34
Sandbox Security page . . . . .	35
Custom Extensions section . . . . .	35
<b>CHAPTER 3: Data Source Management . . . . .</b>	<b>37</b>
About JDBC . . . . .	37
Supplied drivers . . . . .	38
Adding data sources. . . . .	39
Adding data sources in the Administrator . . . . .	39
Connecting to DB2 Universal Database 6.x, 7.2, and OS/390 . . . . .	41
Connecting to Informix 9.x. . . . .	43
Connecting to Microsoft Access . . . . .	44
Connecting to Microsoft Access with Unicode . . . . .	46
Connecting to Microsoft SQL Server 7.x, 2000. . . . .	47
Connecting to MySQL . . . . .	48
Connecting to ODBC Socket . . . . .	50
Connecting to Oracle R3 (8.1.7), Oracle 9i. . . . .	51
Connecting to other data sources. . . . .	52
Connecting to Sybase 11.5, 11.9, 12.0, and 12.5 . . . . .	54
<b>CHAPTER 4: Web Server Management . . . . .</b>	<b>57</b>
Understanding web servers in ColdFusion MX . . . . .	57
Using the built-in web server. . . . .	58
Using an external web server . . . . .	59
Web server configuration. . . . .	59
Using GUI mode . . . . .	60
Using the command-line interface . . . . .	60
Configuration files . . . . .	62
Advanced configurations . . . . .	65
Multihoming . . . . .	65
SSL. . . . .	68

<b>CHAPTER 5: Administering Security</b> . . . . .	69
About ColdFusion MX security. . . . .	69
Security and edition differences . . . . .	70
Using sandbox security . . . . .	70
Using multiple sandboxes (Enterprise Edition only) . . . . .	70
Resources that can be restricted . . . . .	71
About directories and permissions . . . . .	71
Adding a sandbox (Enterprise Edition only) . . . . .	72
Configuring a sandbox . . . . .	72
<b>CHAPTER 6: Using Multiple Server Instances</b> . . . . .	75
Overview of multiple server instances . . . . .	75
Defining additional server instances . . . . .	76
Deploying ColdFusion MX . . . . .	76
Enabling application isolation . . . . .	77
Web server configuration for application isolation . . . . .	78
Enabling load balancing and failover . . . . .	81

## **PART II: Administering Verity**

---

<b>CHAPTER 7: Introducing Verity Tools</b> . . . . .	85
About the Verity utilities . . . . .	85
ColdFusion MX OEM restrictions . . . . .	86
Collection structure and ColdFusion MX . . . . .	86
Verity search modes in ColdFusion MX . . . . .	87
How ColdFusion MX determines which mode to use. . . . .	87
Verity information storage . . . . .	88
About K2 Server . . . . .	88
<b>CHAPTER 8: Managing Collections with the mkvdk Utility</b> . . . . .	89
About the Verity mkvdk utility . . . . .	89
The mkvdk utility syntax . . . . .	89
Getting started with the Verity mkvdk utility . . . . .	91
Creating a collection . . . . .	91
Collection setup options . . . . .	92
General processing options . . . . .	93
Date format options . . . . .	95
Service-level keywords . . . . .	95
Message options . . . . .	96
Document processing options . . . . .	96
Bulk submit options . . . . .	97
Using bulk insert and delete options . . . . .	97
Collection maintenance options . . . . .	97
Examples: maintaining collections . . . . .	98
Deleting a collection . . . . .	99
Optimization keywords . . . . .	99

About squeezing deleted documents . . . . .	99
About optimized Verity databases . . . . .	100
Performance tuning options . . . . .	100
<b>CHAPTER 9: Indexing Collections with Verity Spider . . . . .</b>	<b>101</b>
About Verity Spider . . . . .	101
Web standard support . . . . .	102
Restart capability . . . . .	102
State maintenance through a persistent store . . . . .	102
Performance . . . . .	102
About Verity Spider syntax . . . . .	103
The Verity Spider command . . . . .	103
Using a command file . . . . .	104
Command-line option reference . . . . .	104
Core options . . . . .	106
Processing options . . . . .	107
Networking options . . . . .	112
Path and URL options . . . . .	115
Content options . . . . .	119
Locale options . . . . .	126
Logging options . . . . .	127
Maintenance options . . . . .	128
Setting MIME types . . . . .	129
Syntax restrictions . . . . .	129
MIME types and web crawling . . . . .	129
MIME types and file system indexing . . . . .	130
Indexing unknown MIME types . . . . .	130
Known MIME types for file system indexing . . . . .	131
<b>CHAPTER 10: Searching Collections with K2 Server . . . . .</b>	<b>133</b>
Using K2 Server . . . . .	133
Editing the k2server.ini file . . . . .	133
Starting K2 Server . . . . .	135
Specifying K2 Server parameters in the ColdFusion MX Administrator . . . . .	135
Stopping K2 Server . . . . .	135
Stopping K2 Server when run as a service . . . . .	136
Stopping K2 Server when run as an application . . . . .	136
Stopping K2 Server on UNIX . . . . .	136
The k2server.ini parameter reference . . . . .	136
Server section . . . . .	136
Search thread keywords . . . . .	137
Collection sections . . . . .	138
Using the rck2 utility to search K2 Server documents . . . . .	140
rck2 syntax . . . . .	140
rck2 command options . . . . .	140

<b>CHAPTER 11: Searching Collections with the rcvdk Utility.</b>	143
Using the Verity rcvdk utility.	143
Attaching to a collection using the rcvdk utility.	144
Basic searching	144
Viewing results of the rcvdk utility	145
Displaying more fields	146
<b>CHAPTER 12: Troubleshooting Collections with Verity Utilities.</b>	149
Overview of Verity utilities	149
Using the Verity didump utility.	150
Viewing the word list with the didump utility	150
Viewing the zone list with the didump utility	151
Viewing the zone attribute list with the didump utility.	151
Using the Verity browse utility	152
Using menu options with the browse utility.	152
Displaying fields.	153
Using the Verity merge utility	153
Merging collections using the merge utility	154
Splitting collections using the merge utility	154
<b>CHAPTER 13: Verity Error Messages.</b>	155
VDK mode error codes	155
Generic error codes.	155
Usage error codes	155
Runtime error codes.	156
Data error codes	156
Query error codes.	157
Licensing error codes	157
Security error codes	158
Remote connection error codes.	158
Filtering error codes	158
Dispatch error codes.	159
Warning error codes	159
K2 mode error codes	159
Generic error codes.	159
Usage error codes	159
Runtime error codes.	160
Data error codes	160
Query error codes.	161
Security error codes	161
Remote connection error codes.	161
File handling error codes.	161
Dispatch error codes.	161
Warning error codes	162
TCP/IP error codes	162
<b>INDEX</b>	163





# INTRODUCTION

*Configuring and Administering ColdFusion MX* is intended for anyone who needs to configure and manage their ColdFusion development environment.

## About Macromedia ColdFusion MX documentation

The ColdFusion documentation is designed to provide support for the complete spectrum of participants.

### Documentation set

The ColdFusion documentation set includes the following titles:

<b>Book</b>	<b>Description</b>
<i>Installing and Using ColdFusion MX</i>	Describes system installation and basic configuration for Windows, Solaris, Linux, and HP-UX.
<i>Configuring and Administering ColdFusion MX</i>	Part I describes how to manage the ColdFusion environment, including connecting to your data sources and configuring security for your applications. Part II describes Verity search tools and utilities that you can use for configuring the Verity K2 Server search engine, as well as creating, managing, and troubleshooting Verity collections.
<i>Developing ColdFusion MX Applications</i>	Describes how to develop your dynamic web applications, including retrieving and updating your data, using structures, and forms.
<i>Getting Started Building ColdFusion MX Applications</i>	Contains an overview of ColdFusion features and application development procedures. Includes a tutorial that guides you through the process of developing an example ColdFusion application.
<i>CFML Reference</i>	Provides descriptions, syntax, usage, and code examples for all ColdFusion tags, functions, and variables.
<i>CFML Quick Reference</i>	A brief guide that shows the syntax of ColdFusion tags, functions, and variables.

### Viewing online documentation

All ColdFusion MX documentation is available online in HTML and Adobe Acrobat Portable Document Format (PDF) files. Go to the documentation home page for ColdFusion MX on the Macromedia website: [www.macromedia.com](http://www.macromedia.com).



# PART I

## Administering ColdFusion MX

This part describes how to use the ColdFusion MX Administrator to manage the ColdFusion environment, including connecting to your data sources and configuring security for your applications

Chapter 1: Administering ColdFusion MX . . . . .	13
Chapter 2: Basic ColdFusion MX Administration . . . . .	17
Chapter 3: Data Source Management . . . . .	37
Chapter 4: Web Server Management . . . . .	57
Chapter 5: Administering Security . . . . .	69
Chapter 6: Using Multiple Server Instances . . . . .	75



# CHAPTER 1

## Administering ColdFusion MX

This chapter presents an overview of the ColdFusion MX Administrator and how you can use it to manage your development environment. For procedures, see the ColdFusion MX Administrator online Help.

### Contents

About the ColdFusion MX Administrator . . . . .	13
Accessing user assistance. . . . .	14
Administrator layout . . . . .	14

### About the ColdFusion MX Administrator

The ColdFusion MX Administrator provides a browser-based interface for managing your ColdFusion environment. You can configure many settings to provide optimal levels of security and functionality. The available options are based on your edition of ColdFusion: Standard or Enterprise, as well as your configuration: server or J2EE.

The default location for the ColdFusion MX Administrator login page is:

`http://servername/CFIDE/administrator/index.cfm`

In the previous URL, *servername* is the fully qualified domain name of your web server. Common values for *servername* are localhost or 127.0.0.1 (each refers to the web server on the local computer).

If you are using the ColdFusion built-in web server, include the port number as part of the servername. The default port number is 8500. For example, `http://servername:8500/CFIDE/administrator/index.cfm`.

If you are using the J2EE configuration, include the port number used the J2EE application server's web server. For example, `http://servername:8100/CFIDE/administrator/index.cfm`.

If your ColdFusion MX Administrator is on a remote computer, use the DNS name or IP address of the remote host.

To access the ColdFusion MX Administrator, enter the password specified when you installed ColdFusion MX.

## Accessing user assistance

You can obtain assistance from the ColdFusion MX Administrator in the following ways:

- **Online Help** You access the context-sensitive online Help by clicking the question-mark icon on any ColdFusion MX Administrator page. The online Help has procedural and brief overview content for the ColdFusion MX Administrator page that you are viewing. This information appears in a new browser window and contains standard Contents, Index, and Search tabs.
- **Documentation** Click the link to access the entire ColdFusion MX documentation set online.
- **Examples** The example applications provide samples for you to learn about ColdFusion MX.
- **Tech notes** You can access the collection of articles about ColdFusion MX from the Macromedia website.

## Administrator layout

The home page of the ColdFusion MX Administrator includes links to Documentation, the Macromedia Servers TechNotes Knowledge Base, Release Notes, System Information, online Help, and Code Examples.

The tasks that you perform in the ColdFusion MX Administrator are grouped into the following sections. Each section contains links to pages for managing aspects of the system.

- **Server Settings** Manage whitespace, client and memory variables, locking, and mappings. Register a mail server and configure mail logging. Configure your JVM, the ColdFusion charting and graphing engine, and create and manage archives.
- **Data & Services** Configure data sources, Verity collections, and the Verity K2 Server. Define mappings to web services.
- **Debugging & Logging** Manage options that can assist you in troubleshooting your ColdFusion applications. Manage scheduled tasks, system probes, and a variety of log files and server statistics. Run the Code Compatibility Analyzer to assist you in migrating older ColdFusion applications.
- **Extensions** Configure and register Java Applets, CORBA ORBs, and CFX Tags.
- **Security** Control passwords for ColdFusion MX Administrator and Remote Development Services (RDS) access. Restrict the use of resources, such as data sources.

For more information about each section, see [Chapter 2, “Basic ColdFusion MX Administration,”](#) on page 17.

## Server Settings section

The Server Settings section contains the following areas:

- **Settings** Manage the number of simultaneous requests, request timeouts, whitespace, and handlers.
- **Caching** Manage caching options for memory, database connection time, the number of cached queries, and using a trusted template cache.

- **Client Variables** Configure an external data source, the operating system registry, or web browser cookies to store client variables. These can use and store information about a client browsing your site to provide customized page content.
- **Memory Variables** Specify timeout values for Application and Session variables. These variables are stored in RAM and maintain information throughout a ColdFusion session.
- **Mappings** Create logical aliases for physical directories on your server. One of your first tasks after installing ColdFusion is to configure the mapping for your web server.
- **Mail Server** Configure the mail server that ColdFusion uses to send dynamic mail messages using SMTP (Simple Mail Transfer Protocol). Specify backup mail servers for failover and manage concurrent threads (Enterprise Edition only).
- **Charting** Specify caching and thread settings for the ColdFusion charting and graphing engine.
- **Java and JVM Settings** Manage Java Virtual Machine settings such as paths, heap sizes, and implementation options. Not available in the J2EE configuration.
- **Archives and Deployment** Create and deploy application archives.
- **Settings Summary** View the status of all ColdFusion configuration settings. You can navigate to a particular area of the ColdFusion MX Administrator by clicking its name.

## Data & Services section

The Data & Services section contains the following pages:

- **Data Sources** Create and manage your data sources. You can specify login parameters, connection information, and restrict certain SQL operations. For more information, see [Chapter 3, “Data Source Management,” on page 37](#).
- **Verity Collections** Create and manage your Verity collections. Search engines for your ColdFusion applications use these indexes of various files within specified directories.
- **Verity K2 Server** Configure the Host Name and Port settings for your K2 Server. This specialized server is optimized for high-performance Verity searches.
- **Web Services** Define a mapping to the location of a web service.

## Debugging & Logging section

The Debugging & Logging section contains the following pages:

- **Debugging Settings** Enable and configure information to help you diagnose ColdFusion page failures. You can return information on items such as template stack, database activity, and variable values.
- **Debugging IP Addresses** Control which IP addresses receive debug messages.
- **Logging Settings** Specify the directory for your log files, and whether to write some ColdFusion log messages to the operating system’s logging facility (such as EventLog for Windows and syslog for UNIX).
- **Log Files** Search, view, download, schedule, archive, or delete a file from a list of all available log files.
- **Scheduled Tasks** Add, edit, or delete scheduled tasks. These tasks are helpful for such items as daily reports, inventories, and statistical reports.

- **System Probes** Manage probes that monitor your application's status. If a potential problem is detected, a system probe can send an alert e-mail message and execute a recovery script.
- **Code Analyzer** Evaluate application code for potential incompatibilities between ColdFusion MX and ColdFusion Server 5.

## Extensions section

The Extensions section contains the following pages:

- **Java Applets** Register, edit, or delete Java applets. You must register a Java applet prior to adding it to your CFFORM forms using the `cfapplet` tag.
- **CFX Tags** Register, edit, or delete C++ and Java custom tags.
- **Custom Tag Paths** Register the paths that contain your custom tags.
- **CORBA Connectors** Register, edit, or delete CORBA connectors. You can also specify ORB initialization options.

## Security section

The Security section contains the following pages:

- **CF Admin Password** Set the password for the administrator.
- **RDS Password** Set the password for Dreamweaver MX and CF Studio users connecting to ColdFusion.
- **Sandbox Security** Restrict access to ColdFusion resources such as data sources, tags, functions, files and directories, and IP addresses. This is called Resource Security in ColdFusion MX Standard Edition.

For more information, see [Chapter 5, "Administering Security,"](#) on page 69.



# CHAPTER 2

## Basic ColdFusion MX Administration

This chapter explains the basic ColdFusion MX administration tasks, following the structure of the ColdFusion MX Administrator sections.

### Contents

Initial administration tasks . . . . .	17
Server Settings section . . . . .	18
Data & Services section . . . . .	26
Debugging & Logging section . . . . .	27
Extensions section . . . . .	33
Security section . . . . .	34
Custom Extensions section . . . . .	35

### Initial administration tasks

Immediately after installing ColdFusion MX, you might have to perform some or all of the administrative tasks described in the following table:

Task	Description
Establish database connections	ColdFusion applications require data source connections to query and write to databases. To create, verify, edit, and delete database connections, use the Data Sources pages in the Administrator. For more information, see <a href="#">Chapter 3, “Data Source Management,” on page 37</a> .
Specify directory mappings	Directory mappings redirect relative file paths to physical directories on your server. To specify server-wide directory aliases, use the Mappings page in the Administrator. For more information, see <a href="#">“Mappings page” on page 22</a> .
Configure debugging settings	Debugging information provides important data about CFML page processing. To choose the debugging information to display, and to designate an IP address to receive debugging information, use the Debugging & Logging section of the Administrator. For more information, see <a href="#">“Debugging Settings page” on page 28</a> .

Task	Description
Set up e-mail	E-mail lets ColdFusion MX and ColdFusion applications send automated mail messages. To configure an e-mail server and mail options, use the Mail Server page of the Administrator. For more information, see <a href="#">“Mail Server page” on page 23</a> .
Change passwords	You might have to change the passwords that you set for the ColdFusion MX Administrator and RDS during ColdFusion MX installation. To change passwords, use the Security section of the Administrator. For more information, see <a href="#">“CF Admin Password page” on page 34</a> and <a href="#">“RDS Password page” on page 34</a> .
Configure Java settings	Java and Java applets require configuring Java settings, such as JVM paths. To change Java settings, use the Java and JVM page of the Administrator. For more information, see <a href="#">“Extensions section” on page 33</a> .
Restrict tag access	Some CFML tags might present a potential security risk for your server. To disable certain tags, use the Sandbox Security page of the Administrator. For more information, see <a href="#">“Administering Security” on page 69</a> .

## Server Settings section

The Server Settings section lets you manage client and memory variables, mappings, charting, and archiving. You also configure mail and Java settings in this section.

### Settings page

The Settings page of the ColdFusion MX Administrator contains configuration options that you can set or enable to manage ColdFusion MX. These options can significantly affect server performance. The following table describes the settings:

Setting	Description
Limit simultaneous requests (Server configuration only)	Enter a number to limit simultaneous requests to ColdFusion MX. When the server reaches the limit, requests are queued and handled in the order received. Limiting the number of simultaneous requests can improve performance.
Timeout requests after [n] seconds	Enable this option to prevent unusually lengthy requests from using up server resources. Enter a limit to the time that ColdFusion MX waits before terminating a request. Requests that take longer than the timeout period are terminated.
Use UUID for cftoken	Specify whether to use a universally unique identifier (UUID), rather than a random number, for a cftoken.
Enable HTTP status codes	Select this option to configure ColdFusion MX to set a status code of 500 Internal Server Error for an unhandled error. Disable this option to configure ColdFusion MX to set a status code of 200 OK for everything, including unhandled errors.
Enable Whitespace Management (Server configuration only)	Enable this option to compress runs of spaces, tabs and carriage return/line feeds. Compressing whitespace can significantly compact the output of a ColdFusion page.

Setting	Description
Missing Template Handler	Specify a page to execute when ColdFusion MX cannot find a requested page. This specification is relative to the web root. If the user is running Internet Explorer with "Show Friendly HTTP error messages" enabled in advanced settings (the default), Internet Explorer will only display this page if it contains more than 512 bytes.
Site-wide Error Handler	Specify a page to execute when ColdFusion MX encounters an error while processing a request. This specification is relative to the web root. If the user is running Internet Explorer with "Show Friendly HTTP error messages" enabled in advanced settings (the default), Internet Explorer will only display this page if it contains more than 512 bytes.

## Caching page

The Caching page of the Administrator contains configuration options that you can set or enable to cache templates, queries, and data sources. These options can significantly affect server performance. The following table describes the settings:

Setting	Description
Template cache size (number of templates)	Enable this option to limit the memory reserved for template caching. For best performance, set this to a value that is large enough to contain your application's commonly accessed ColdFusion pages, yet small enough to avoid excessive reloading. You can experiment with a range of values on your development server; a suitable starting point is one page per MB of JVM size.
Trusted cache	Enable this option if you want ColdFusion MX to use cached templates without checking whether they changed. For sites that are not updated frequently, using this option minimizes file system overhead.
Save Class Files	Select this option to save to disk the class files generated by the ColdFusion bytecode compiler. During the development phase, it is typically faster to disable this option.
Cache web server paths (Server configuration only)	Select this option to cache ColdFusion page paths for a single server. Deselect this option if ColdFusion MX connects to a web server with multiple websites or multiple virtual websites.
Limit the maximum number of cached queries on the server to [n] queries	Enable this option by entering a value to limit the maximum number of cached queries that the server maintains. Cached queries allow retrieval of result sets from memory rather than through a database transaction. Because queries reside in memory, and query result set sizes differ, you must provide a limit for the number of cached queries. You enable cached queries with the <code>cachedwithin</code> or <code>cachedafter</code> attributes of the <code>cfquery</code> tag.

## Client Variables page

Client variables let you store user information and preferences between sessions. Using information from client variables, you can customize page content for individual users.

You enable client variable default settings in ColdFusion MX on the Client Variables page of the Administrator. ColdFusion MX lets you store client variables in the following ways:

- In database tables  
If your data source uses one of the JDBC drivers bundled with ColdFusion, ColdFusion can automatically create the necessary tables. If your data source uses the ODBC Socket or a third-party JDBC driver, you must manually create the necessary CDATA and CGLOBAL database tables. For more information, see *Developing ColdFusion MX Applications*.
- As cookies in users' web browsers
- In the operating system registry

**Caution:** Macromedia recommends that you do not store client variables in the registry because it can critically degrade performance of the server. If you do use the registry to store client variables, you must allocate sufficient memory and disk space.

You can override settings specified in the Client Variables page using the attributes of the `cfapplication` tag. For more information, see *Developing ColdFusion MX Applications*.

The following table compares these storage options:

Storage type	Advantages	Disadvantages
Data source	<ul style="list-style-type: none"><li>• Can use existing data source</li><li>• Portable: not tied to the host system or operating system</li></ul>	<ul style="list-style-type: none"><li>• Requires database transaction to read/write variables</li><li>• More complex to implement</li></ul>
Browser cookies	<ul style="list-style-type: none"><li>• Simple implementation</li><li>• Good performance</li><li>• Can be set to expire automatically</li><li>• Client-side control</li></ul>	<ul style="list-style-type: none"><li>• Users can configure browsers to disallow cookies</li><li>• Cookie data is limited to 4 KB</li><li>• Netscape Navigator allows only 20 cookies from one host; ColdFusion MX uses three cookies to store read-only data, leaving only 17 cookies available</li></ul>
System registry	<ul style="list-style-type: none"><li>• Simple implementation</li><li>• Good performance</li><li>• Registry can be exported easily to other systems</li><li>• Server-side control</li></ul>	<ul style="list-style-type: none"><li>• Possible restriction of the registry's maximum size limit in Windows in the Control Panel</li><li>• Integrated with the host system: not practical for clustered servers</li><li>• Solaris, Linux, and HP-UX registries are text files. Their registries deliver slow performance and low scalability.</li></ul>

## Migrating client variable data

To migrate your client variable data to another data source, you should know the structure of the database tables that store this information. Client variables stored externally use two simple database tables, like those shown in the following tables:

---

### CDATA Table

Column	Data type
cfid	CHAR(64), TEXT, VARCHAR, or equivalent
app	CHAR(64), TEXT, VARCHAR, or equivalent
data	MEMO, LONGTEXT, LONG VARCHAR, or equivalent

---

### CGLOBAL Table

Column	Data type
cfid	CHAR(64), TEXT, VARCHAR, or equivalent
data	MEMO, LONGTEXT, LONG VARCHAR, or equivalent
lvisit	TIMESTAMP, DATETIME, DATE, or equivalent

---

## Creating client variable tables

Use the following sample ColdFusion page as a model for creating client variable database tables in your own database. However, keep in mind that not all databases support the same column data type names. For the proper data type, see your database documentation.

**Tip:** The ColdFusion MX Administrator can create client variable tables for data sources that use bundled JDBC drivers. For more information, see the online Help.

### Sample table creation page

```
<!--- Create the Client variable storage tables in a datasource.
This example applies to Microsoft Access databases --->
```

```
<cfquery name="data1" datasource="#DSN#">
CREATE TABLE CDATA
(
    cfid char(20),
    app char(64),
    data memo
)
</cfquery>
```

```
<cfquery name="data2" datasource="#DSN#">
CREATE UNIQUE INDEX id1
ON CDATA (cfid,app)
</cfquery>
```

```
<cfquery name="global1" datasource="#DSN#">
CREATE TABLE CGLOBAL
(
    cfid char(20),
    data memo,
    lvisit date
)
```

```

)
</cfquery>

<cfquery name="global2" datasource="#DSN#">
  CREATE INDEX id2
  ON CGLOBAL (cfid)
</cfquery>

<cfquery name="global2" datasource="#DSN#">
  CREATE INDEX id3
  ON CGLOBAL (lvisit)
</cfquery>

```

## Memory Variables page

You use the Memory Variables page of the ColdFusion Administrator to enable application and session variables server-wide. By default, application and session variables are enabled when you install ColdFusion MX. If you disable either type of variable in the Memory Variables page, you cannot use them in a ColdFusion application.

You can specify maximum and default timeout values for session and application variables. Unless you define a timeout value in `Application.cfm`, application variables expire in two days. Session variables expire when user sessions end. To change these behaviors, enter new default and maximum timeout values on the Memory Variables page of the Administrator.

**Note:** Timeout values that you specify for application variables override the timeout values set in `Application.cfm`.

You can also specify whether to use J2EE session variables. When you enable the J2EE session variables, ColdFusion creates an identifier for each session and does not use the `CFToken` or `CFID` cookie value. For more information, see *Developing ColdFusion MX Applications*.

## Mappings page

You use the Mappings page of the ColdFusion MX Administrator to add, update, and delete logical aliases for paths to directories on your server. ColdFusion mappings apply only to pages processed by ColdFusion MX with the `cfinclude` and `cfmodule` tags. If you save CFML pages outside of the `web_root` (or whatever directory is mapped to `/`), you must add a mapping to the location of those files on your server.

Assume that the `/` mapping on your server points to `C:\CFusionMX\wwwroot`, but all your ColdFusion header pages reside in `c:\2002\newpages\headers`. In order for ColdFusion MX to find your header pages, you must add a mapping in the ColdFusion Administrator that points to `c:\2002\newpages\headers` (for example, add a mapping for `/headers` that points to `c:\2002\newpages\headers`). In the ColdFusion pages located in `C:\CFusionMX\wwwroot`, you reference these header pages using `/headers` in your `cfinclude` and `cfmodule` tags.

**Note:** ColdFusion mappings are different from web server virtual directories. For information on creating a virtual directory to access a given directory using a URL in your web browser, consult your web server's documentation.

## Mail Server page

You use the Mail Server page of the ColdFusion MX Administrator to specify a mail server to send automated e-mail messages. ColdFusion MX supports the Simple Mail Transfer Protocol (SMTP) for sending e-mail messages and the Post Office Protocol (POP) for retrieving e-mail messages from your mail server. To use e-mail messaging in your ColdFusion applications, you must have access to an SMTP server and/or a POP account.

The ColdFusion MX Enterprise edition supports mail server failover as well as additional mail delivery options.

The ColdFusion implementation of SMTP mail uses a spooled architecture. This means that when a `cfmail` tag is processed in an application page, the messages generated might not be sent immediately. If ColdFusion is extremely busy or has a large queue, delivery could occur after some delay.

**Note:** For more information about the `cfmail` tag, see *Developing ColdFusion MX Applications*.

## Mail Connection Settings area

Select preferences for handling mail logs, as described in the following table:

Setting	Description
Mail Server	Lets you enter a valid mail server for sending dynamic SMTP mail messages in the text box. You can enter an Internet address, such as <code>mail.company.com</code> or the IP address of the mail server, such as <code>127.0.0.1</code> .
Server Port	Enter the number of the port on which the mail server is running. Contact your server administrator if you are unsure of the appropriate port number.
Verify Mail Server Connection	Select this option to verify that ColdFusion MX can connect to your specified mail server after you submit this form. Whether or not you use this option, you should verify that your mail server connection works by sending a test message.
Backup Mail Servers (Enterprise Edition only)	Enter zero or more backup servers for sending SMTP mail messages. You can enter an Internet address, such as <code>mail.company.com</code> , or the IP address of the mail server, such as <code>127.0.0.1</code> . Separate multiple servers with a comma. If the mail server requires authentication, prefix the mail server with the username and password, as follows: <code>username:password@mailserveraddress</code> . To use a port number other than the default (25), specify <code>mailserveraddress:portnumber</code> .
Maintain Connection to Mail Server (Enterprise Edition only)	Select this option to keep mail server connections open after sending a mail message. Enabling this option can enhance performance when delivering multiple messages.
Connection Timeout (seconds)	Enter the number of seconds that ColdFusion MX should wait for a response from the mail server.
Spool Interval (seconds)	Enter the number of seconds at which you want the mail server to process spooled mail.
Mail Delivery Threads (Enterprise Edition only)	The maximum number of simultaneous threads used to deliver spooled mail.

Setting	Description
Spool mail messages for delivery (Memory spooling available for Enterprise Edition only)	Select this option to route outgoing mail messages to the mail spooler. If you disable this option, ColdFusion MX delivers outgoing mail messages immediately. In ColdFusion MX Enterprise Edition, you can spool messages either to disk (slower, but messages persist across shutdowns) or to memory (faster, but messages do not persist). You can override this setting in the <code>cfmail</code> tag.
Maximum number of messages spooled to memory (Enterprise Edition only)	Enter the maximum number of messages ColdFusion MX will spool to memory before switching to disk spooling.

## Mail Logging Settings area

Select preferences for handling mail logs, as described in the following table:

Setting	Description
Error Log Severity	From the drop-down list box, select the type of SMTP-related error message to write to a log file. The options are: Debug, Warning, Debug, Information, and Error.
Log all e-mail messages sent by ColdFusion MX	Enable this option to save to a log file the To, From, and Subject fields of all e-mail messages.

ColdFusion MX writes sent mail and mail error logs to either of the following directories:

- `\CFusionMX\logs`, in Windows
- `/opt/coldfusionmx/log`, on Solaris, Linux, and HP-UX

The following table describes the e-mail log files:

Log	Description
<code>mailex.log</code>	Records sent e-mail messages
<code>mail.log</code>	Records general e-mail errors

## Charting Settings page

The ColdFusion charting and graphing engine lets you produce highly customizable business graphics, in a variety of formats, using the `cfquery` tag. You use the Charting page in the Administrator to control characteristics of the engine.

The following table describes the caching and thread settings for the ColdFusion charting and graphing engine:

Setting	Description
Cache Type	Set the cache type. Charts can be cached either in memory or to disk. In memory caching is faster, but more memory intensive.
Maximum number of images in cache	Specify the maximum number of charts to store in the cache. After the cache is full, if you generate a new chart, ColdFusion discards the oldest chart in the cache.



Setting	Description
Max number of charting threads	Specify the maximum number of chart requests that can be processed concurrently. The minimum number is 1 and the maximum is 5. Higher numbers are more memory intensive.
Disk cache location	When caching to disk, specify the directory in which to store the generated charts.

## Java and JVM Settings page

The Java and JVM Settings page lets you specify the following settings, which enable ColdFusion MX to work with Java:

Setting	Description
Java Virtual Machine Path	The absolute file path to the location of the Java virtual machine (JVM) root directory. Default is <code>cf_root/runtime/jre</code> .
Initial Memory Size	The JVM initial heap size. Default is 8196 MB.
Maximum Memory Size	The JVM maximum heap size. Default is 512 MB.
Class Path	The file paths to the directories that contain the JAR files used by ColdFusion MX. Specify either the fully qualified name of a directory that contains your JAR files or a fully qualified JAR file name. Use a comma to separate multiple entries.
JVM Arguments	The arguments to the JVM. Use a space to separate multiple entries; for example, <code>-Xint -Xincg</code>

Before ColdFusion saves your changes, it saves a copy of the current `cf_root/runtime/bin/jvm.config` file as `jvm.bak`. If your changes prevent ColdFusion from restarting, use `jvm.bak` to restore your system. For more information, see the online Help.

**Note:** This page is not enabled in the J2EE configuration.

## Archives and Deployment page

The Archives and Deployment page includes tools that let you archive and deploy ColdFusion applications, configuration settings, data source information, and other types of information to back up your files quickly and easily. The complete list of archivable information includes the following:

- Name and file location
- Server settings
- ColdFusion mappings
- Data sources
- Verity collections
- Scheduled tasks
- Java applets
- CFX tags
- Archive to do lists

After you archive the information, you can use the Administrator to deploy your web applications to the same ColdFusion MX server or to a ColdFusion MX server running on a different computer. Additionally, you can use these features to deploy and receive any ColdFusion archive file electronically.

The Archive Settings page in the Administrator lets you configure various archive system settings that apply to all archive and deploy operations. For more information, see the online Help.

## Settings Summary page

The Settings Summary page shows all ColdFusion configuration settings. Click a group name to open that group's Administrator section, where you can edit settings. This page is not enabled in the Standard Edition.

## Data & Services section

The Data & Services section of the Administrator is the interface between you, ColdFusion MX, data sources, and Verity search and indexing features. The following table describes some common tasks that you can perform in the Data & Services section of the Administrator:

Task	Description
Create and manage JDBC data sources	The Data Sources page lets you establish, edit, and delete JDBC data source connections for ColdFusion MX. For more information, see <a href="#">Chapter 3, "Data Source Management," on page 37</a> .
Create and maintain Verity collections	The Verity Collections page lets you create and delete Verity collections and perform maintenance operations on collections that you create. For more information, see <a href="#">"Verity Collections page" on page 26</a> .
Register a Verity K2 Server with ColdFusion MX	The Verity K2 Server page lets you register a K2 Server to use with ColdFusion MX. For more information, see <a href="#">"Verity K2 Server page" on page 27</a> .
Define mappings for web services	Web services let you produce and consume remote application functionality over the internet. For more information, see <a href="#">"Web Services page" on page 27</a> .

## Data Sources page

The Data Sources page lets you create, edit, and delete data sources. Before you can use a database in a ColdFusion application, you must register the data source in the ColdFusion MX Administrator. For more information, see [Chapter 3, "Data Source Management," on page 37](#).

## Verity Collections page

The Verity Development Kit (VDK) provides indexing and searching technology to create, populate, and manage *collections* of indexed data that are optimized for fast and efficient site searches. It is available on the Verity Collections page.

A *collection* is a *logical group* of documents and *metadata* about the documents. The *metadata* includes word indexes, an internal documents table of document field information, and logical pointers to the document files.

For more information about building search interfaces, see the chapters about the `cfindex`, `cfsearch`, and `cfcollection` tags in *Developing ColdFusion MX Applications*.

ColdFusion lets you manage your collections from the Administrator. You can index, repair, optimize, purge, or delete Verity collections that are connected to ColdFusion. You use the buttons along the bottom of the Connected Verity Collections table to perform the following actions:

---

Action	Description
Index	Analyzes the files in a collection and assembles metadata and pointers to the files.
Repair	Re-indexes a collection to fix broken links and update indexes.
Optimize	Reclaims space left by deleted and changed files by consolidating collection indexes for faster searching. You should optimize collections regularly.
Purge	Deletes all documents in a collection, but not the collection itself. Leaves the collection directory structure intact.
Delete	Deletes a collection.

---

**Note:** Before performing management operations, ensure that the K2 Server is not using the collections. For more information, see [“Administering Verity” on page 83](#).

## Verity K2 Server page

For faster searching, configure a K2 Server in the ColdFusion MX Administrator. The high-performance K2 Server caches collection information so that your searches retrieve documents more quickly. The Verity K2 Server delivers rapid search results in a highly efficient and scalable architecture.

For more information on configuring and using K2 Server with ColdFusion, see [“Administering Verity” on page 83](#).

## Web Services page

You can use web services to produce and consume remote application functionality over the Internet. The ColdFusion MX Administrator lets you register web services so that you do not have to specify the entire Web Services Description Language (WSDL) URL when you reference the web service. The first time you reference a web service, ColdFusion MX automatically registers it in the Administrator.

When you register a web service, you can shorten your code and change a web service’s URL without editing your code. For more information, see *Developing ColdFusion MX Applications*.

## Debugging & Logging section

You use the Debugging Settings and Debugging IPs pages of the Administrator to configure ColdFusion MX to provide debugging information for every application page requested by a browser. You specify debugging preferences using the pages as follows:

- On the Debugging Settings page, select debugging output options. If debugging is enabled, the output appears in block format after normal page output.
- On the Debugging IPs page, restrict access to debugging output. If a debugging option is enabled, debugging output is visible to all users by default.

This section also includes pages for managing your Log Files, Scheduled Tasks, System Probes, and the Code Compatibility Analyzer.

## Debugging Settings page

The Debugging Settings page provides the following debugging options:

Setting	Description
Enable Robust Exception Information	Displays detailed information in the exceptions page, including the template's physical path and URI, the line number and snippet, the SQL statement used (if any), the data source name (if any), and the Java stack trace.
Enable Debugging	Enables the ColdFusion debugging service.
Select Debugging Output Format	Select a format of: <ul style="list-style-type: none"><li>• classic.cfm - The format available in ColdFusion 5 and earlier. It provides a basic view and few browser restrictions.</li><li>• dockable.cfm - A dockable tree-based debugging panel. For details about the panel and browser restrictions, see the online Help.</li></ul>
Report stack trace to a depth of [n] rows	Reports execution times. The stack trace shows a hierarchical tree of executed templates, includes, modules, and custom tags that were executing at the time of the exception. The default is 5. A blank value or 0 implies no limit.
Database Activity	Shows the database activity for the SQL Query events and Stored Procedure events in the debugging output.
Exception Information	Shows all ColdFusion exceptions raised for the request in the debugging output.
Tracing Information	Shows trace event information in the debugging output. Tracing lets you track program flow and efficiency through the use of the <code>cftrace</code> tag.
Variables	Displays information about parameters, URL parameters, cookies, session, and CGI variables in the debugging output.
Enable Robust Exception Information	Lets visitors view detailed information in the exceptions page, including: the template's physical path and URI, the line number and snippet, the SQL statement used (if any), the Data Source Name (if any), and the Java stack trace.
Enable Performance Monitoring* (Server configuration only)	Enables the standard NT Performance Monitor application to display information about a running ColdFusion Application Server.
Enable CFSTAT* (Server configuration only)	Shows performance information on platforms that do not support the NT Performance Monitor. For more information, see <a href="#">"Using the cfstat utility" on page 29</a> .

\* Restart ColdFusion MX after changing this setting.

## Using the cfstat utility

The cfstat command-line utility provides real-time performance metrics for ColdFusion MX. Using a socket connection to obtain metric data, the cfstat utility displays the information that ColdFusion MX writes to the System Monitor without actually using the System Monitor application. The following table lists the metrics that cfstat returns:

---

<b>Metric abbreviation</b>	<b>Metric name</b>	<b>Description</b>
Pg/Sec	Page hits per second	The number of ColdFusion pages processed per second. You can reduce this by moving static content to HTML pages.
DB/Sec	Database accesses per second	The number of database accesses per second made by ColdFusion MX. Any difference in complexity and resource load between calls is ignored.
CP/Sec	Cache pops per second	The number of ColdFusion template cache pops per second. A <i>cache pop</i> occurs when ColdFusion MX ejects a cached template from the template cache to make room for a new template.
Req Q'ed	Number of queued requests	The number of requests that are currently waiting for ColdFusion MX to process them. Lower values, which you can achieve with efficient CFML, are better.
Req Run'g	Number of running requests	The number of requests that ColdFusion MX is currently actively processing.
Req TO'ed	Number of timed out requests	The total number of ColdFusion requests that have timed out. Lower values, which you can achieve by aggressive caching, removing unnecessary dynamic operations and third-party events, are better.
AvgQ Time	Average queue time	A running average of the time, in milliseconds, that requests spend waiting for ColdFusion MX to process them. Lower values, which you can achieve with efficient CFML and enhanced caching, are better.
AvgReq Time	Average request time	A running average of the time, in milliseconds, that ColdFusion MX spends to process a request (including queued time). Lower values, which you can achieve with efficient CFML, are better.
AvgDB Time	Average database transaction time	A running average of the time that ColdFusion MX spends on database-related processing of ColdFusion requests.

---

<b>Metric abbreviation</b>	<b>Metric name</b>	<b>Description</b>
Bytes In/Sec	Bytes incoming per second	The number of bytes that ColdFusion MX read in the last second (not an average).
Bytes Out/Sec	Bytes outgoing per second	The number of bytes that ColdFusion MX wrote in the last second (not an average).

Before you use the cfstat utility, ensure that you selected the Enable Performance Monitoring check box in the ColdFusion MX Administrator (on the **Debugging & Logging > Debugging Settings** page). If you select this check box, you must restart ColdFusion MX for this change to take effect.

Your cfusionmx\bin directory contains the cfstat utility. From that directory, type **cfstat** and use the following available switches:

<b>Switch</b>	<b>Description/Comment</b>
-n	Suppress column headers (useful for saving output to a file).
-s	Display output in a single line (delay display of the first line so cfstat can display meaningful values in the per-second counters).
#	Where # is an integer, delay display output by # seconds. If you do not specify an integer, cfstat returns one line.
-h	Web server hostname (localhost is the default).
-p	Web server listening port number (80 is the default).

## Debugging IP Addresses page

You use the Debugging IP Addresses page to restrict debugging output to one or more IP addresses. You can add and remove IP addresses.

**Note:** If you do not specify IP addresses, and debugging options are active, debugging output is displayed for all users.

## Logging Settings page

You use the Logging Settings page of the Administrator to change ColdFusion MX logging options. The following table describes the settings:

Setting	Description
Log directory*	Directory to which error log files are written.
Maximum file size (kb)	Set the maximum file size for log files. Once a file hits this size, it will be automatically archived.
Maximum number of archives	Set the maximum number of log archives to create. After reaching this limit, files will be deleted in order of oldest to newest.
Log slow pages taking longer than [n] seconds	Log the names of pages that take longer than the specified interval to process. Logging slow pages can help you diagnose potential problems or bottlenecks in your ColdFusion applications. Entries are written to server.log.
Log all CORBA calls	Log all CORBA calls.
Enable logging for scheduled tasks	Log ColdFusion Executive task scheduling.

\* Restart ColdFusion MX after changing this setting.

## Log Files page

The Log Files page of the Administrator lets you perform operations on log files, such as searching, viewing, downloading, archiving, and deleting.

Click on a Log File icon, located in the Actions column of the Available Log Files table, to search, view, download, archive, or delete a log file.

For more information, see the online Help.

The following table describes the ColdFusion MX log files:

Log	Description
rdservice.log	Records errors occurring in the ColdFusion Remote Development Services (RDS). This service provides remote HTTP-based access to files and databases.
application.log	Records every ColdFusion MX error reported to a user. Application page errors, including ColdFusion MX syntax, ODBC, and SQL errors are written to the log file.
webserver.log	Records errors that occur in the web server and the ColdFusion MX stub.
exceptions.log	Records stacktraces for exceptions that occur in the server.
scheduler.log	Records scheduled events that have been submitted for execution. Indicates whether task submission was initiated and whether it succeeded. Provides the scheduled page URL, the date and time executed, and a task ID.
server.log	Records errors for ColdFusion MX.
customtag.log	Records errors generated in custom tag processing.

Log	Description
car.log	Records errors associated with Site Archive and Restore operations.
mail.log	Records errors generated by an SMTP mail server.
mailex.log	Records messages sent by ColdFusion MX.
flash.log	Records entries for Flash Remoting.

## Scheduled Tasks page

You use the Scheduled Tasks page to schedule the execution of local and remote web pages and to generate static HTML pages. The scheduling facility is useful for applications that do not require user interactions or customized output. ColdFusion developers use this facility to schedule daily sales reports, corporate directories, statistical reports, and so on.

Information that is read more often than written is a good candidate for scheduled tasks. Instead of executing a query to a database every time the page is requested, ColdFusion MX renders the static page with information generated by the scheduled event. Response time is faster because no database transaction takes place.

You can run scheduled tasks once; on a specified date; or at a specified time, daily, weekly, or monthly. You can run a scheduled task daily, at a specified interval, or between specified dates.

The Schedule Task page lets you create, edit, and delete scheduled tasks. For more information, see the online Help.

## System Probes page

System probes help you evaluate the status of your ColdFusion applications. Like scheduled tasks, they access a URL at a specified interval, but they can also check for the presence or absence of a string in the URL. If the URL contents are unexpected, or if an error occurred while accessing the URL, the probe can send an e-mail alert to the address specified in the System Probes page. The probe can also execute a script to perform a recovery action, such as restarting the server. All probe actions are logged in logs/probes.log. The System Probes page also displays the status of each probe.

You use the buttons in the Actions column in the System Probes table to perform the following actions:

Action	Description
Edit	Lets you edit the probe.
Run	Runs the probe immediately, even if it was previously disabled.
Enable/ Disable	Starts and stops the probe from automatically executing at its specified interval.
Delete	Deletes the probe.

Because probes run as scheduled ColdFusion tasks, they will not run if the ColdFusion MX server on which they are hosted crashes, or if the host web server crashes or otherwise does not respond.



## Code Compatibility Analyzer page

The Code Compatibility Analyzer evaluates your ColdFusion pages for potential incompatibilities between ColdFusion MX and ColdFusion Server 5.

## Extensions section

You use the Extensions section of the Administrator to configure ColdFusion MX to work with other technologies, such as Java and CORBA. This section contains the Java Applets, CFX Tags, Custom Tag Paths, and CORBA Connectors pages.

### Java Applets page

The Java Applets page of the Administrator lets you register applets and edit and delete applet registrations. Before you can use Java applets in your ColdFusion applications, you must register them in the Java Applets page.

When your applet is registered with ColdFusion MX, using the `cfapplet` tag in your CFML code is very simple, because all parameters are predefined. Simply enter the applet source and the form variable name you want to use.

**Note:** Parameters set in the `cfapplet` tag override parameters defined in the Java Applets page.

For more information, see the online Help.

### CFX Tags page

Before you can use a CFX tag in ColdFusion applications, you must register it. You use the CFX Tags page to register and manage ColdFusion custom tags built with C++ and Java.

You can build CFX tags in the following two ways:

- Using C++ as a dynamic link library (DLL) in Windows; as shared objects (so/sl) on Solaris, Linux, and HP-UX
- Using Java interfaces defined within the `cfx.jar` file

For more information, see the online Help.

### Custom Tag Paths page

You use the Custom Tag Paths page of the Administrator to add, edit, and delete custom tag directory paths. The default custom tag path is under the installation directory. To use custom tags in another path, register the path on this Administrator page.

For more information, see the online Help.

### CORBA Connectors page

You use the CORBA Connectors page of the Administrator to register, edit, and delete CORBA connectors. You must register CORBA connectors before using them in your ColdFusion applications. You must also restart the server when you are done with the CORBA Connector configuration.

ColdFusion MX loads ORB libraries dynamically using a connector, which does not restrict ColdFusion developers to a specific ORB vendor. The connectors depend on the ORB runtime libraries provided by the vendor. A connector for Borland Visibroker is embedded within ColdFusion MX. Make sure that the ORB runtime libraries are in *cf\_root/runtime/lib*.

The following table contains information about the libraries and connectors:

Operating System	Vendor	ORB	ColdFusion connector	ORB library
WindowsNT and later	Borland	VisiBroker 4.5	coldfusion.runtime.corba.VisibrokerConnector (embedded)	vbjorb.jar
Solaris	Borland	VisiBroker 4.5	coldfusion.runtime.corba.VisibrokerConnector (embedded)	vbjorb.jar
HP-UX	Borland	VisiBroker 4.5	coldfusion.runtime.corba.VisibrokerConnector (embedded)	vbjorb.jar

**Note:** Macromedia will provide implementations of the connectors for some of the popular ORBs. For those that are not supported, Macromedia will make the source available under NDA to a select group of third-party candidates and/or ORB vendors.

The following lines are an example of a CORBA connector configuration for VisiBroker:

```
ORB Name      visibroker
ORB Class Name coldfusion.runtime.corba.VisibrokerConnector
ORB Property File:\cfusionmx\runtime\cfusion\lib\vbjorb.properties
Classpath     [blank]
```

ColdFusion includes the vbjorb.properties file, which contains the following properties that configure the ORB:

```
org.omg.CORBA.ORBClass=com.inprise.vbroker.orb.ORB
org.omg.CORBA.ORBSingletonClass=com.inprise.vbroker.orb.ORB
SVcnameroot=namingroot
```

## Security section

The Security section of the Administrator lets you configure the security frameworks of ColdFusion MX.

For more information on security, see [Chapter 5, “Administering Security,” on page 69](#).

## CF Admin Password page

You use the CF Admin Password page of the Administrator to enable and disable password-restricted access to the Administrator, and to change the Administrator password.

## RDS Password page

You use the RDS Password page to enable and disable password-restricted access to server resources from Macromedia Dreamweaver MX or Macromedia HomeSite+ using Remote Development Services (RDS), and to change the password.

## Sandbox Security page

You use the Sandbox Security page (called Resource Security in the Standard Edition) to specify security permissions for data sources, tags, functions, files, and directories.

Sandbox security uses the location of your ColdFusion pages to determine functionality. A *sandbox* is a designated area (CFM files or directories containing CFM files) of your site to which you apply security restrictions. By default, a subdirectory (or *child* directory) inherits the sandbox settings of the directory one level above it (the *parent* directory). If you define sandbox settings for a subdirectory, you override the sandbox settings inherited from the parent directory.

Use sandbox security to control access to:

- Data sources
- Tags
- Functions
- Files and directories
- IP addresses and ports

## Custom Extensions section

You can extend the functionality of the ColdFusion Administrator by adding links to other web applications and sites. These links appear under the Custom Extensions section in the left navigation pane of the Administrator.

**Note:** You must create a link for the Custom Extensions section to appear in the Administrator.

To extend the Administrator, create a file that contains the HTML link code, followed by a `<BR>`, with a separate line for each link. Do not include other HTML code, such as `<head>` or `<body>` tags. Save this file as `extensionscustom.cfm` in the Administrator root directory (`/CFIDE/administrator/`).

For example, the following file adds to the Administrator links for Bowdoin College, Universidad Complutense de Madrid, and La Sapienza:

```
<a href="http://www.bowdoin.edu/" target="content">Bowdoin College</a><br>
<a href="http://www.http://www.ucm.es/" target="_blank">Universidad
  Complutense de Madrid</a><br>
<a href="http://www.uniroma1.it/" target="_blank">La Sapienza</a><br>
```

When you click a link, the page displays. The target attribute is required; if you specify `target="content"`, the page appears in the main pane of the Administrator. If you specify any other value for the target attribute, the page appears in a new window.



# CHAPTER 3

## Data Source Management

This chapter describes the configuration options for ColdFusion MX data sources. For basic information on data sources and connecting to databases, see *Developing ColdFusion MX Applications*.

### Contents

About JDBC . . . . .	37
Adding data sources . . . . .	39
Connecting to DB2 Universal Database 6.x, 7.2, and OS/390 . . . . .	41
Connecting to Informix 9.x . . . . .	43
Connecting to Microsoft Access . . . . .	44
Connecting to Microsoft Access with Unicode . . . . .	46
Connecting to Microsoft SQL Server 7.x, 2000 . . . . .	47
Connecting to MySQL . . . . .	48
Connecting to ODBC Socket . . . . .	50
Connecting to Oracle R3 (8.1.7), Oracle 9i . . . . .	51
Connecting to other data sources . . . . .	52
Connecting to Sybase 11.5, 11.9, 12.0, and 12.5 . . . . .	54

### About JDBC

JDBC is a Java Application Programming Interface (API) that you use to execute SQL statements. JDBC enables an application, such as ColdFusion MX, to interact with a variety of relational databases, without using interfaces that are database- and platform-specific.

The following table describes the four types of JDBC drivers:

Type	Name	Description
1	JDBC-ODBC bridge	<p>Translates JDBC calls into ODBC calls, and sends them to the ODBC driver.</p> <p><b>Advantages</b> Allows access to many different databases.</p> <p><b>Disadvantages</b> The ODBC driver, and possibly the client database libraries, must reside on the ColdFusion server computer. Performance is also below par.</p> <p>Macromedia does not recommend this driver type unless your application requires specific features of these drivers.</p>
2	Native-API/partly Java driver	<p>Converts JDBC calls into database-specific calls.</p> <p><b>Advantages</b> Better performance than Type 1 Driver.</p> <p><b>Disadvantages</b> The vendor's client database libraries must reside on the same computer as ColdFusion.</p> <p>Macromedia does not recommend this driver type unless your application requires specific features of these drivers, such as the Unicode support offered by the ColdFusion MX Microsoft Access with Unicode driver.</p>
3	JDBC-Net pure Java driver	<p>Translates JDBC calls to the middle-tier server, which then translates the request to the database-specific native-connectivity interface.</p> <p><b>Advantages</b> No need for vendor's database libraries to be present on client computer. Can be tailored for small size (faster loading).</p> <p><b>Disadvantages</b> Database-specific code must be executed in the middle-tier.</p> <p>ColdFusion MX includes an ODBC socket type 3 driver for use with Microsoft Access databases and ODBC data sources.</p>
4	Native-protocol/all-Java driver	<p>Converts JDBC calls into the network protocol used directly by the database.</p> <p><b>Advantages</b> Fast performance. No special software needed on the computer on which you run ColdFusion MX.</p> <p><b>Disadvantages</b> Many of these protocols are proprietary, requiring a different driver for each database.</p> <p>ColdFusion MX includes type 4 drivers for many popular DBMSs, however, not all DBMSs are supported in ColdFusion MX Standard Edition.</p>

## Supplied drivers

The following table shows the database drivers supplied with ColdFusion MX and where you can find more information:

Driver	Type	Reference
DB2 Universal Database 6.x, 7.2, and OS/390	4	<a href="#">"Connecting to DB2 Universal Database 6.x, 7.2, and OS/390" on page 41</a>
Informix 9.x	4	<a href="#">"Connecting to Informix 9.x" on page 43</a>
Microsoft Access	3	<a href="#">"Connecting to Microsoft Access" on page 44</a>

Driver	Type	Reference
Microsoft Access with Unicode support	2	<a href="#">“Connecting to Microsoft Access with Unicode” on page 46</a>
Microsoft SQL Server 7.x, 2000	4	<a href="#">“Connecting to Microsoft SQL Server 7.x, 2000” on page 47</a>
MySQL	4	<a href="#">“Connecting to MySQL” on page 48</a>
ODBC Socket	3	<a href="#">“Connecting to ODBC Socket” on page 50</a>
Oracle R3 (8.1.7), Oracle 9i	4	<a href="#">“Connecting to Oracle R3 (8.1.7), Oracle 9i” on page 51</a>
Other		<a href="#">“Connecting to other data sources” on page 52</a>
Sybase 11.5, 11.9, 12.0, 12.5	4	<a href="#">“Connecting to Sybase 11.5, 11.9, 12.0, and 12.5” on page 54</a>

## Adding data sources

In the ColdFusion MX Administrator, you configure your data sources to communicate with ColdFusion. Once you add a data source to the Administrator, you access it by name in any CFML tag that establishes database connections; for example, `cfquery`. During a query, the data source tells ColdFusion which database to connect to and what parameters to use for the connection.

The ColdFusion MX Administrator organizes all the information about a ColdFusion MX server’s database connections in a single, easy-to-manage location. In addition to adding new data sources, you can use the Administrator to specify changes to your database configuration, such as relocation, renaming, or changes in security permissions.

### Adding data sources in the Administrator

You use the ColdFusion MX Administrator to quickly add a data source for use in your ColdFusion applications. When you add a data source, you assign it a data source name (DSN) and set all information required to establish a connection.

**Note:** ColdFusion MX includes several data sources that are configured by default, including `cfsnippets`, `CompanyInfo`, and `exampleapps`. This procedure should not be necessary to work with these data sources.

#### To add a data source:

- 1 In the ColdFusion MX Administrator, select **Data & Services > Data Sources**.
- 2 Under Add New Data Source, enter a Data Source Name; for example, `MyTestDSN`. The following names are reserved. You cannot use them for data source names:
  - `service`
  - `jms_provider`
  - `comp`
  - `jms`
- 3 Select a Driver from the drop-down list box; for example, Microsoft SQL Server.

#### 4 Click Add.

A form for additional DSN information appears. The available fields in this form depend on the Driver that you selected.

#### 5 In the Database field, enter the name of the database; for example, Northwind.

#### 6 In the Server field, enter the network name or IP address of the server that hosts the database, and enter any required Port value; for example, the bullwinkle server on the default port.

#### 7 If your database requires login information, enter your Username and Password.

**Tip:** The omission of required username and password information is a common reason why a data source fails to verify.

#### 8 (Optional) Enter a Description.

#### 9 (Optional) Click Show Advanced Settings to specify any ColdFusion-specific settings; for example, to configure which SQL commands can interact with this data source.

#### 10 Create Submit to create the data source.

ColdFusion MX automatically verifies that it can connect to the data source.

#### 11 (Optional) To verify this data source later, click the verify icon:

**Note:** To check the status of all data sources available to ColdFusion MX, click Verify All Connections.

## Specifying connection string arguments

You can use the ColdFusion MX Administrator to specify connection string arguments for data sources that use the Microsoft Access, ODBC Socket, MYSQL, or DB2 drivers. In the Advanced Settings page for one of these drivers, in the Connection String field, enter name=value pairs separated by a semicolon. The Administrator configures the following ODBC connection string:

```
DSN=odbcdsnname;APP=RaiseGenerator;WSID=TWriter01
```

In the preceding string, `odbcdsnname` is the name of the ODBC DSN. This is the string that the Microsoft Access or ODBC Socket driver uses to connect to the data source at runtime.

**Note:** The `connectstring` tag attribute (`cfquery` tag) is not supported in ColdFusion MX.

## Adding data source notes and considerations

When adding data sources to ColdFusion MX, keep these guidelines in mind:

- Data source names should be all one word.
- Data source names can contain only letters, numbers, hyphens, and the underscore.
- Data source names should not contain special characters or spaces.
- Although data source names are not case-sensitive, you should use a consistent capitalization scheme.
- Depending on the JDBC driver, connection strings and JDBC URLs might be case-sensitive.
- Ensure that you use the Administrator to verify that ColdFusion MX can connect to the data source.

A data source must exist in the ColdFusion MX Administrator before you use it on an application page to retrieve data.



## Connecting to DB2 Universal Database 6.x, 7.2, and OS/390

Use the settings in the following table to connect ColdFusion to DB2 Universal Database 6.x, 7.2, and OS/390 data sources:

Setting	Description
CF Data Source Name	The data source name (DSN) used by ColdFusion MX to connect to the data source.
Database	The name of the database.
Server	The name of the server that hosts the database that you want to use. If the database is local, enclose the word <i>local</i> in parentheses.
Port	The number of the TCP/IP port that the server monitors for connections.
Username	<p>The user name that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a user name; for example, in a <code>cfquery</code> tag.</p> <p>The username must have CREATE PACKAGE privileges for the database, or the database administrator must create a package. Consult the database administrator when configuring this type of data source.</p>
Password	The password (16-character limit) that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a password—for example, in a <code>cfquery</code> tag.
Description	(Optional) A description for this connection.
Connection String	<p>A field that passes database-specific parameters, such as login credentials, to the data source.</p> <p>For UDB on the initial connection, specify DatabaseName, PackageName, CreateDefaultPackage, and ReplacePackage, as shown in the following example:</p> <pre>DatabaseName=SAMPLE;PackageName=pkgname; CreateDefaultPackage=TRUE;ReplacePackage=TRUE</pre> <p>For UDB on subsequent connections, specify DatabaseName and PackageName, as shown in the following example:</p> <pre>DatabaseName=SAMPLE;PackageName=pkgname</pre> <p>For OS/390 on the initial connection, specify LocationName, CollectionId, PackageName, and CreateDefaultPackage, as shown in the following example:</p> <pre>LocationName=SAMPLE;CollectionId=DEFAULT; PackageName=pkgname;CreateDefaultPackage=TRUE</pre> <p>For OS/390 on subsequent connections, specify LocationName, CollectionId, and PackageName, as shown in the following example:</p> <pre>LocationName=SAMPLE;CollectionId=DEFAULT; PackageName=pkgname</pre> <p>Where <i>pkgname</i> is the name of the package (maximum of 7 characters) that the driver uses to process SQL statements.</p> <p>Your user ID must have CREATE PACKAGE privileges on the database, or your database administrator must create a package for you.</p>

Setting	Description
Limit Connections	Specifies whether ColdFusion MX limits the number of database connections for the data source. If you enable this option, use the Restrict Connections to field to specify the maximum.
Restrict Connections to	Specifies the maximum number of database connections for the data source. To use this restriction, you must enable Limit Connections.
Maintain Connections	ColdFusion establishes a connection to a data source for every operation that requires one. Enable this option to improve performance by caching the data source connection.
String Format	Enable this option if your application uses Unicode data in DBMS-specific Unicode datatypes such as National Character or nchar.
Max Pooled Statements	Enables reuse of prepared statements (that is, stored procedures and queries that use the <code>cfqueryparam</code> tag). Although you tune this setting based on your application, start by setting it to the sum of the following: <ul style="list-style-type: none"> <li>• Unique <code>cfquery</code> tags that use <code>cfqueryparam</code></li> <li>• Unique <code>cfstoredproc</code> tags</li> </ul>
Timeout (min)	The maximum number of minutes after the data source connection is made that you want ColdFusion to cache a connection after it is used.
Interval (min)	The time (in minutes) that the server waits between cycles to check for expired data source connections to close.
Disable Connections	If selected, suspends all client connections.
Login Timeout (sec)	The number of seconds before ColdFusion times out the data source connection login attempt.
CLOB	Select to return the entire contents of any CLOB/Text columns in the database for this data source. If unchecked, ColdFusion retrieves the amount specified in the Long Text Buffer setting.
BLOB	Select to return the entire contents of any BLOB/Image columns in the database for this data source. If unchecked, ColdFusion retrieves the amount specified in the Blob Buffer setting.
LongText Buffer (chr)	The default buffer size, used if Enable Long Text Retrieval (CLOB) is not selected. Default is 64000 bytes.
BLOB Buffer (bytes)	The default buffer size, used if Enable binary large object retrieval (BLOB) is not selected. Default is 64000 bytes.
Allowed SQL	The SQL operations that can interact with the current data source.

## Connecting to Informix 9.x

Use the settings in the following table to connect ColdFusion MX to Informix 9.x data sources:

Setting	Description
CF Data Source Name	The data source name (DSN) used by ColdFusion MX to connect to the data source.
Database	The database to which this data source connects.
Informix Server	The name of the Informix database server to which you want to connect.
Server	The name of the server that hosts the database that you want to use. If the database is local, enclose the word <i>local</i> in parentheses.
Port	The number of the TCP/IP port that the server monitors for connections.
Username	The user name that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a user name; for example, in a <code>cfquery</code> tag.
Password	The password (16-character limit) that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a password; for example, in a <code>cfquery</code> tag.
Description	(Optional) A description for this connection.
Limit Connections	Specifies whether ColdFusion MX limits the number of database connections for the data source. If you enable this option, use the Restrict Connections to field to specify the maximum.
Restrict Connections to	Specifies the maximum number of database connections for the data source. To use this restriction, you must enable Limit Connections.
Maintain Connections	ColdFusion MX establishes a connection to a data source for every operation that requires one. Enable this option to improve performance by caching the data source connection.
String Format	Enable Unicode for data sources that are configured for non-Latin characters.
Max Pooled Statements	Enables reuse of prepared statements (that is, stored procedures and queries that use the <code>cfqueryparam</code> tag). Although you tune this setting based on your application, start by setting it to the sum of the following: <ul style="list-style-type: none"><li>• Unique <code>cfquery</code> tags that use <code>cfqueryparam</code></li><li>• Unique <code>cfstoredproc</code> tags</li></ul>
Timeout (min)	The maximum number of minutes after the data source connection is made that you want ColdFusion MX to cache a connection after it is used.
Interval (min)	The time (in minutes) that the server waits between cycles to check for expired data source connections to close.
Disable Connections	If selected, suspends all client connections.
Login Timeout (sec)	The number of seconds before ColdFusion MX times out the data source connection login attempt.

Setting	Description
CLOB	Select to return the entire contents of any CLOB/Text columns in the database for this data source. If not selected, ColdFusion MX retrieves the amount specified in the Long Text Buffer setting.
BLOB	Select to return the entire contents of any BLOB/Image columns in the database for this data source. If not selected, ColdFusion MX retrieves the amount specified in the Blob Buffer setting.
LongText Buffer	The default buffer size, used if Enable Long Text Retrieval(CLOB) is not selected. Default is 64000 bytes.
BLOB Buffer	The default buffer size; used if Enable binary large object retrieval (BLOB) is not selected. Default is 64000 bytes.
Allowed SQL	The SQL operations that can interact with the current data source.

## Connecting to Microsoft Access

Use the settings in the following table to connect ColdFusion MX to Microsoft Access data sources:

Setting	Description
CF Data Source Name	The data source name (DSN) used by ColdFusion MX to connect to the data source.
Database File	The file that contains the database.
System Database File	To secure access to the specified database file, click Browse Server to locate and enter a database that contains database security information. The system database is usually located in winnt\system32\system.mdw.
Use Default Username	If selected, ColdFusion MX does not pass a user name or password when requesting a connection. The Microsoft Access driver uses the default user name and password. (specified in Advanced Settings).
ColdFusion Username	The user name that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a user name; for example, in a <code>cfquery</code> tag.
ColdFusion Password	The password (16-character limit) that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a password; for example, in a <code>cfquery</code> tag.
Description	(Optional) A description for this connection.
Page Timeout	The time (in tenths of a second) before a request for a ColdFusion page times out.
Max Buffer Size	The total number of bytes that ColdFusion MX uses to cache application pages. Enter a value to optimize ColdFusion performance.
Connection String	A field that passes database-specific parameters, such as login credentials, to the data source.
Default Username	The user name that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a user name-for example, in a <code>cfquery</code> tag.

<b>Setting</b>	<b>Description</b>
Default Password	The password (16-character limit) that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a password—for example, in a <code>cfquery</code> tag.
Return Timestamp as String	Enable this setting if your application retrieves Date/Time data and then re-uses it in SQL statements without applying formatting (using functions such as <code>DateFormat</code> , <code>TimeFormat</code> , and <code>CreateODBCDateTime</code> ).
Limit Connections	Specifies whether ColdFusion MX limits the number of database connections for the data source. If you enable this option, use the <code>Restrict Connections</code> to field to specify the maximum.
Restrict Connections to	Specifies the maximum number of database connections for the data source. To use this restriction, you must enable <code>Limit Connections</code> .
Maintain Connections	ColdFusion MX establishes a connection to a data source for every operation that requires one. Enable this option to improve performance by caching the data source connection.
Timeout (min)	The maximum number of minutes after the data source connection is made that you want ColdFusion MX to cache a connection after it is used. The <code>Timeout</code> setting does not return a connection to the cache after a specified period of time, regardless of how infrequently it is used. The default is "" or 0, which means that the connection timeout is never enforced.
Interval (min)	The time (in minutes) that the server waits between cycles to check for expired data source connections to close.
Disable Connections	If selected, suspends all client connections.
Login Timeout (sec)	The number of seconds before ColdFusion MX times out the data source connection login attempt.
CLOB	Select to return the entire contents of any CLOB/Text columns in the database for this data source. If not selected, ColdFusion MX retrieves the amount specified in the <code>Long Text Buffer</code> setting.
BLOB	Select to return the entire contents of any BLOB/Image columns in the database for this data source. If unchecked, ColdFusion retrieves the amount specified in the <code>Blob Buffer</code> setting.
LongText Buffer	The default buffer size, used if <code>Enable Long Text Retrieval(CLOB)</code> is not selected. Default is 64000 bytes.
BLOB Buffer	The default buffer size, used if <code>Enable binary large object retrieval (BLOB)</code> is not selected. Default is 64000 bytes.
Allowed SQL	The SQL operations that can interact with the current data source.

## Connecting to Microsoft Access with Unicode

Type 2 driver. Use the settings in the following table to connect ColdFusion MX to Microsoft Access with Unicode data sources:

Setting	Description
CF Data Source Name	The data source name (DSN) used by ColdFusion MX to connect to the data source.
Database File	The file that contains the database.
Description	(Optional) A description for this connection.
ColdFusion Username	The user name that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a user name; for example, in a <code>cfquery</code> tag.
ColdFusion Password	The password (16-character limit) that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a password; for example, in a <code>cfquery</code> tag.
Page Timeout	The time (in tenths of a second) before a request for a ColdFusion page times out.
Max Buffer Size	The total number of bytes that ColdFusion MX uses to cache application pages. Enter a value to optimize ColdFusion performance.
Connection String	A field that passes database-specific parameters, such as login credentials, to the data source.
Limit Connections	Specifies whether ColdFusion MX limits the number of database connections for the data source. If you enable this option, use the Restrict Connections to field to specify the maximum.
Restrict Connections to	Specifies the maximum number of database connections for the data source. To use this restriction, you must enable Limit Connections.
Maintain Connections	ColdFusion MX establishes a connection to a data source for every operation that requires one. Enable this option to improve performance by caching the data source connection.
Timeout (min)	The maximum number of minutes after the data source connection is made that you want ColdFusion MX to cache a connection after it is used. The Timeout setting does not return a connection to the cache after a specified period of time, regardless of how infrequently it is used. The default is "" or 0, which means that the connection timeout is never enforced.
Interval (min)	The time (in minutes) that the server waits between cycles to check for expired data source connections to close.
Disable Connections	If selected, suspends all client connections.
Login Timeout (sec)	The number of seconds before ColdFusion MX times out the data source connection login attempt.
CLOB	Select to return the entire contents of any CLOB/Text columns in the database for this data source. If not selected, ColdFusion MX retrieves the amount specified in the Long Text Buffer setting.

Setting	Description
BLOB	Select to return the entire contents of any BLOB/Image columns in the database for this data source. If not selected, ColdFusion MX retrieves the amount specified in the Blob Buffer setting.
LongText Buffer	The default buffer size, used if Enable Long Text Retrieval(CLOB) is not selected. Default is 64000 bytes.
BLOB Buffer	The default buffer size, used if Enable binary large object retrieval (BLOB) is not selected. Default is 64000 bytes.
Allowed SQL	The SQL operations that can interact with the current data source.

## Connecting to Microsoft SQL Server 7.x, 2000

Use the settings in the following table to connect ColdFusion MX to Microsoft SQL Server 7.x, 2000 data sources:

Setting	Description
CF Data Source Name	The data source name (DSN) used by ColdFusion MX to connect to the data source.
Database	The database to which this data source connects.
Server	The name of the server that hosts the database that you want to use. If the database is local, enclose the word <i>local</i> in parentheses.
Port	The number of the TCP/IP port that the server monitors for connections.
Username	The user name that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a user name; for example, in a <code>cfquery</code> tag.
Password	The password (16-character limit) that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a password—for example, in a <code>cfquery</code> tag.
Description	(Optional) A description for this connection.
Select Method	Determines whether server cursors are used for SQL queries.  The Direct method provides more efficient retrieval of data when you retrieve record sets in a forward-only direction and you limit your SQL Server connection to a single open SQL statement at a time. This is typical for ColdFusion applications. The Cursor method lets you have multiple open SQL statements on a connection. This is not typical for ColdFusion applications, unless you use pooled statements.
Limit Connections	Specifies whether ColdFusion MX limits the number of database connections for the data source. If you enable this option, use the Restrict Connections to field to specify the maximum.
Restrict Connections to	Specifies the maximum number of database connections for the data source. To use this restriction, you must enable Limit Connections.

Setting	Description
Maintain Connections	ColdFusion MX establishes a connection to a data source for every operation that requires one. Enable this option to improve performance by caching the data source connection.
String Format	Enable this option if your application uses Unicode data in DBMS-specific Unicode datatypes such as National Character or nchar.
Max Pooled Statements	Enables reuse of prepared statements (that is, stored procedures and queries that use the <code>cfqueryparam</code> tag). Although you tune this setting based on your application, start by setting it to the sum of the following: <ul style="list-style-type: none"> <li>• Unique <code>cfquery</code> tags that use <code>cfqueryparam</code></li> <li>• Unique <code>cfstoredproc</code> tags</li> </ul>
Timeout (min)	The maximum number of minutes after the data source connection is made that you want ColdFusion MX to cache a connection after it is used.
Interval (min)	The time (in minutes) that the server waits between cycles to check for expired data source connections to close.
Disable Connections	If selected, suspends all client connections.
Login Timeout (sec)	The number of seconds before ColdFusion MX times out the data source connection login attempt.
CLOB	Select to return the entire contents of any CLOB/Text columns in the database for this data source. If not selected, ColdFusion MX retrieves the amount specified in the Long Text Buffer setting.
BLOB	Select to return the entire contents of any BLOB/Image columns in the database for this data source. If not selected, ColdFusion MX retrieves the amount specified in the Blob Buffer setting.
LongText Buffer	The default buffer size, used if Enable Long Text Retrieval(CLOB) is not selected. Default is 64000 bytes.
BLOB Buffer	The default buffer size, used if Enable binary large object retrieval (BLOB) is not selected. Default is 64000 bytes.
Allowed SQL	The SQL operations that can interact with the current data source.

## Connecting to MySQL

Use the settings in the following table to connect ColdFusion MX to MySQL data sources:

Setting	Description
CF Data Source Name	The data source name (DSN) used by ColdFusion MX to connect to the data source.
Database	The database to which this data source connects.
Server	The name of the server that hosts the database that you want to use. If the database is local, enclose the word <i>local</i> in parentheses.
Port	The number of the TCP/IP port that the server monitors for connections.



Setting	Description
Username	The user name that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a user name; for example, in a <code>cfquery</code> tag.
Password	The password (16-character limit) that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a password; for example, in a <code>cfquery</code> tag.
Description	(Optional) A description for this connection.
Connection String	A field that passes database-specific parameters, such as login credentials, to the data source.
Limit Connections	Specifies whether ColdFusion MX limits the number of database connections for the data source. If you enable this option, use the Restrict Connections to field to specify the maximum.
Restrict Connections to	Specifies the maximum number of database connections for the data source. To use this restriction, you must enable Limit Connections.
Maintain Connections	ColdFusion MX establishes a connection to a data source for every operation that requires one. Enable this option to improve performance by caching the data source connection.
Timeout (min)	The maximum number of minutes after the data source connection is made that you want ColdFusion MX to cache a connection after it is used.
Interval (min)	The time (in minutes) that the server waits between cycles to check for expired data source connections to close.
Disable Connections	If selected, suspends all client connections.
Login Timeout (sec)	The number of seconds before ColdFusion MX times out the data source connection login attempt.
CLOB	Select to return the entire contents of any CLOB/Text columns in the database for this data source. If not selected, ColdFusion MX retrieves the amount specified in the Long Text Buffer setting.
BLOB	Select to return the entire contents of any BLOB/Image columns in the database for this data source. If not selected, ColdFusion MX retrieves the amount specified in the Blob Buffer setting.
LongText Buffer	The default buffer size, used if Enable Long Text Retrieval(CLOB) is not selected. Default is 64000 bytes.
BLOB Buffer	The default buffer size, used if Enable binary large object retrieval (BLOB) is not selected. Default is 64000 bytes.
Allowed SQL	The SQL operations that can interact with the current data source.

## Connecting to ODBC Socket

Type 3 driver. Use the settings in the following table to connect ColdFusion MX to ODBC Socket data sources:

Setting	Description
CF Data Source Name	The data source name (DSN) used by ColdFusion MX to connect to the data source.
ODBC DSN	Select the ODBC DSN to which you want ColdFusion MX to connect.
Trusted Connection	Specifies whether to use domain user account access to the database. Only valid for SQL Server.
Username	The user name that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a user name; for example, in a <code>cfquery</code> tag.
Password	The password (16-character limit) that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a password; for example, in a <code>cfquery</code> tag.
Description	(Optional) A description for this connection.
Connection String	A field that passes database-specific parameters, such as login credentials, to the data source.
Limit Connections	Specifies whether ColdFusion MX limits the number of database connections for the data source. If you enable this option, use the Restrict Connections to field to specify the maximum.
Restrict Connections to	Specifies the maximum number of database connections for the data source. To use this restriction, you must enable Limit Connections.
Maintain Connections	ColdFusion MX establishes a connection to a data source for every operation that requires one. Enable this option to improve performance by caching the data source connection.
Timeout (min)	The maximum number of minutes after the data source connection is made that you want ColdFusion MX to cache a connection after it is used.
Interval (min)	The time (in minutes) that the server waits between cycles to check for expired data source connections to close.
Disable Connections	If selected, suspends all client connections.
Login Timeout (sec)	The number of seconds before ColdFusion MX times out the data source connection login attempt.
CLOB	Select to return the entire contents of any CLOB/Text columns in the database for this data source. If not selected, ColdFusion MX retrieves the amount specified in the Long Text Buffer setting.
BLOB	Select to return the entire contents of any BLOB/Image columns in the database for this data source. If not selected, ColdFusion MX retrieves the amount specified in the Blob Buffer setting.
LongText Buffer	The default buffer size, used if Enable Long Text Retrieval(CLOB) is not selected. Default is 64000 bytes.

Setting	Description
BLOB Buffer	The default buffer size, used if Enable binary large object retrieval (BLOB) is not selected. Default is 64000 bytes.
Allowed SQL	The SQL operations that can interact with the current data source.

## Connecting to Oracle R3 (8.1.7), Oracle 9i

Use the settings in the following table to connect ColdFusion MX to Oracle R3 (8.1.7), Oracle 9i data sources:

Setting	Description
CF Data Source Name	The data source name (DSN) used by ColdFusion MX to connect to the data source.
SID Name	The Oracle System Identifier that refers to the instance of the Oracle database software running on the server. 'ORCL' is the default.
Server	The name of the server that hosts the database that you want to use. If the database is local, enclose the word <i>local</i> in parentheses.
Port	The number of the TCP/IP port that the server monitors for connections.
Username	The user name that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a user name; for example, in a <code>cfquery</code> tag.
Password	The password (16-character limit) that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a password; for example, in a <code>cfquery</code> tag.
Description	(Optional) A description for this connection.
Limit Connections	Specifies whether ColdFusion MX limits the number of database connections for the data source. If you enable this option, use the Restrict Connections to field to specify the maximum.
Restrict Connections to	Specifies the maximum number of database connections for the data source. To use this restriction, you must enable Limit Connections.
Maintain Connections	ColdFusion MX establishes a connection to a data source for every operation that requires one. Enable this option to improve performance by caching the data source connection.
String Format	Enable this option if your application uses Unicode data in DBMS-specific Unicode datatypes such as National Character or nchar.
Max Pooled Statements	Enables reuse of prepared statements (that is, stored procedures and queries that use the <code>cfqueryparam</code> tag). Although you tune this setting based on your application, start by setting it to the sum of the following: <ul style="list-style-type: none"> <li>• Unique <code>cfquery</code> tags that use <code>cfqueryparam</code></li> <li>• Unique <code>cfstoredproc</code> tags</li> </ul>
Timeout (min)	The maximum number of minutes after the data source connection is made that you want ColdFusion MX to cache a connection after it is used.

Setting	Description
Interval (min)	The time (in minutes) that the server waits between cycles to check for expired data source connections to close.
Disable Connections	If selected, suspends all client connections.
Login Timeout (sec)	The number of seconds before ColdFusion MX times out the data source connection login attempt.
CLOB	Select to return the entire contents of any CLOB/Text columns in the database for this data source. If not selected, ColdFusion MX retrieves the amount specified in the Long Text Buffer setting.
BLOB	Select to return the entire contents of any BLOB/Image columns in the database for this data source. If not selected, ColdFusion MX retrieves the amount specified in the Blob Buffer setting.
LongText Buffer	The default buffer size, used if Enable Long Text Retrieval(CLOB) is not selected. Default is 64000 bytes.
BLOB Buffer	The default buffer size, used if Enable binary large object retrieval (BLOB) is not selected. Default is 64000 bytes.
Allowed SQL	The SQL operations that can interact with the current data source.

## Connecting to other data sources

Use the settings in the following table to connect ColdFusion MX to data sources through JDBC drivers that do not appear in the drop-down list of drivers:

Setting	Description
CF Data Source Name	The data source name (DSN) used by ColdFusion MX to connect to the data source.
JDBC URL	The JDBC Connection URL for this data source.
Driver Class	The fully qualified class name of the driver. For example, com.inet.tds.TdsDriver. The JAR file that contains this class must be in a directory defined to the ColdFusion classpath.
Driver Name	(Optional) The name of the driver.
Username	The user name that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a user name; for example, in a <code>cfquery</code> tag.
Password	The password (16-character limit) that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a password; for example, in a <code>cfquery</code> tag.
Description	(Optional) A description for this connection.
Limit Connections	Specifies whether ColdFusion MX limits the number of database connections for the data source. If you enable this option, use the Restrict Connections to field to specify the maximum.
Restrict Connections to	Specifies the maximum number of database connections for the data source. To use this restriction, you must enable Limit Connections.

<b>Setting</b>	<b>Description</b>
Maintain Connections	ColdFusion MX establishes a connection to a data source for every operation that requires one. Enable this option to improve performance by caching the data source connection.
Timeout (min)	The maximum number of minutes after the data source connection is made that you want ColdFusion MX to cache a connection after it is used.
Interval (min)	The time (in minutes) that the server waits between cycles to check for expired data source connections to close.
Disable Connections	If selected, suspends all client connections.
Login Timeout (sec)	The number of seconds before ColdFusion MX times out the data source connection login attempt.
CLOB	Select to return the entire contents of any CLOB/Text columns in the database for this data source. If not selected, ColdFusion MX retrieves the amount specified in the Long Text Buffer setting.
BLOB	Select to return the entire contents of any BLOB/Image columns in the database for this data source. If not selected, ColdFusion MX retrieves the amount specified in the Blob Buffer setting.
LongText Buffer	The default buffer size, used if Enable Long Text Retrieval(CLOB) is not selected. Default is 64000 bytes.
BLOB Buffer	The default buffer size, used if Enable binary large object retrieval (BLOB) is not selected. Default is 64000 bytes.
Allowed SQL	The SQL operations that can interact with the current data source.

## Connecting to Sybase 11.5, 11.9, 12.0, and 12.5

Use the settings in the following table to connect ColdFusion MX to Sybase 11.5, 11.9, 12.0, and 12.5 data sources:

Setting	Description
CF Data Source Name	The data source name (DSN) used by ColdFusion MX to connect to the data source.
Database	The database to which this data source connects.
Server	The name of the server that hosts the database that you want to use. If the database is local, enclose the word <i>local</i> in parentheses. This name must be either a fully qualified domain name (resolvable through DNS) or an IP address. It cannot be a netbios name (even if you are running NBT), or an alias you set up using the client connectivity wizard (both of these approaches worked in earlier ColdFusion versions).
Port	The number of the TCP/IP port that the server monitors for connections.
Username	The user name that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a user name; for example, in a <code>cfquery</code> tag.
Password	The password (16-character limit) that ColdFusion MX passes to the JDBC driver to connect to the data source if a ColdFusion application does not supply a password; for example, in a <code>cfquery</code> tag.
Description	(Optional) A description for this connection.
Select Method	Determines whether server cursors are used for SQL queries.  The Direct method provides more efficient retrieval of data when you retrieve record sets in a forward-only direction and you limit your Sybase connection to a single open SQL statement at a time. This is typical for ColdFusion applications.  The Cursor method lets you have multiple open SQL statements on a connection. This is not typical for ColdFusion applications, unless you use pooled statements.
Limit Connections	Specifies whether ColdFusion MX limits the number of database connections for the data source. If you enable this option, use the Restrict Connections to field to specify the maximum.
Restrict Connections to	Specifies the maximum number of database connections for the data source. To use this restriction, you must enable Limit Connections.
Maintain Connections	ColdFusion MX establishes a connection to a data source for every operation that requires one. Enable this option to improve performance by caching the data source connection.
String Format	Enable this option if your application uses Unicode data in DBMS-specific Unicode datatypes such as National Character or nchar.

Setting	Description
Max Pooled Statements	Enables reuse of prepared statements (that is, stored procedures and queries that use the <code>cfqueryparam</code> tag). Although you tune this setting based on your application, start by setting it to the sum of the following: <ul style="list-style-type: none"> <li>• Unique <code>cfquery</code> tags that use <code>cfqueryparam</code></li> <li>• Unique <code>cfstoredproc</code> tags</li> </ul>
Timeout (min)	The maximum number of minutes after the data source connection is made that you want ColdFusion MX to cache a connection after it is used.
Interval (min)	The time (in minutes) that the server waits between cycles to check for expired data source connections to close.
Disable Connections	If selected, suspends all client connections.
Login Timeout (sec)	The number of seconds before ColdFusion MX times out the data source connection login attempt.
CLOB	Select to return the entire contents of any CLOB/Text columns in the database for this data source. If not selected, ColdFusion MX retrieves the amount specified in the Long Text Buffer setting.
BLOB	Select to return the entire contents of any BLOB/Image columns in the database for this data source. If not selected, ColdFusion MX retrieves the amount specified in the Blob Buffer setting.
LongText Buffer	The default buffer size, used if Enable Long Text Retrieval(CLOB) is not selected. Default is 64000 bytes.
BLOB Buffer	The default buffer size, used if Enable binary large object retrieval (BLOB) is not selected. Default is 64000 bytes.
Allowed SQL	The SQL operations that can interact with the current data source.





# CHAPTER 4

## Web Server Management

This chapter discusses connecting ColdFusion MX to the built-in web server and to external web servers, such as Apache, IIS, and SunONE Web Server (formerly known as iPlanet). It explores common scenarios, security, multi-hosting, and other issues that you might find helpful.

The discussions in this chapter apply when running ColdFusion MX in the server configuration; they do not apply when running ColdFusion MX in the J2EE configuration. However, certain discussions may apply when running ColdFusion MX as an EAR or WAR on JRun 4. Additionally, some J2EE application servers include web server plug-ins that provide similar functionality.

### Contents

Understanding web servers in ColdFusion MX .....	57
Using the built-in web server .....	58
Using an external web server .....	59
Web server configuration .....	59
Advanced configurations .....	65

## Understanding web servers in ColdFusion MX

The web server is a critical component in your ColdFusion MX environment and understanding how ColdFusion interacts with web servers can help you administer your site. ColdFusion MX provides the following options with regard to web servers:

- **Built-in web server** Lightweight, all-Java, HTTP 1.0 web server. Suitable for development but not intended for use in production applications. For more information, see “Using the built-in web server” on page 58.
- **External web server** Customized web server connector module that forwards requests for ColdFusion pages from an external web server to ColdFusion MX. For more information, see “Using an external web server” on page 59.

## Using the built-in web server

The ColdFusion MX server configuration is built on top of JRun, which includes the JRun web server, also called the built-in web server. Although not intended for use in a production environment, the built-in web server is particularly useful in the following cases:

- **Coexistence/transition** The built-in web server lets you run a previous version of ColdFusion (using an external web server) and ColdFusion MX (using the built-in web server) on the same machine while you migrate your existing applications to ColdFusion MX.
- **Development** If your workstation runs ColdFusion MX but does not run an external web server, you can still develop and test ColdFusion applications locally through the built-in web server.

All web servers listen on a TCP/IP port and this port can be specified in the URL. By default, web servers listen for HTTP request on port 80 (for example, `http://www.macromedia.com` and `http://www.macromedia.com:80` are the same). Similarly, 443 is the default port for HTTPS requests.

By default in the server configuration, the built-in web server listens on port 8500. For example, to access the ColdFusion MX Administrator through the built-in web server, specify `http://servername:8500/CFIDE/administrator/index.cfm`.

If you enable the built-in web server during the installation process and port 8500 is already in use, the installer automatically finds the next available port above 8500 and configures the built-in web server to use that port. If you think that your built-in web server is using a port other than 8500, open `cf_root/runtime/servers/default/SERVER-INF/jrun.xml` in a text editor and examine the `port` attribute of the `WebService` service.

**Note:** When installing ColdFusion MX Enterprise Edition using the option that also installs JRun 4, the installation wizard always configures the built-in web server, even if you select an external web server.

The following list outlines additional facts related to the built-in web server.

- Whenever possible, you should choose to configure your external web server as part of ColdFusion MX installation, except for the two cases mentioned at the beginning of this discussion (coexistence with a previous ColdFusion version and when there is no web server on the computer). If you select the built-in web server by mistake, you must run the Web Server Configuration Tool manually to configure your external web server after the installation. The Web Server Configuration Tool is described in [“Web server configuration” on page 59](#).
- The default web root when using the built-in web server is `cf_root/wwwroot`.
- When using the built-in web server, the ColdFusion MX Administrator is in the `cf_root/wwwroot/CFIDE` directory by default.
- If you want the built-in web server to serve pages from a different web root directory, define a virtual mapping in the `cf_root/wwwroot/WEB-INF/jrun-web.xml` file, as the following example shows:

```
<virtual-mapping>
  <resource-path>/*</resource-path>
  <system-path>C:/myApps/wwwroot</system-path>
</virtual-mapping>
```

**Warning:** If you have CFML pages under your external web server's root, ensure that ColdFusion MX has been configured to serve these pages through the external web server. If you have not configured ColdFusion MX to use an external web server, your external web server will serve CFML source code for ColdFusion pages saved under its web root.

## Using an external web server

ColdFusion MX uses the JRun web server connector to forward requests from an external web server to the ColdFusion MX runtime system.

When a request is made for a CFM page, the connector on the web server opens a network connection to the JRun proxy service. The ColdFusion MX runtime system handles the request and sends its reply back through the proxy service and connector. The web server connector uses web server-specific plug-in modules, as the following table shows:

Web server	Connector details
Apache	<p>The Web Server Configuration Tool adds the following elements to the Apache httpd.conf file:</p> <ul style="list-style-type: none"><li>• A LoadModule directive defines the connector.</li><li>• An AddHandler directive tells Apache to route requests for ColdFusion pages through the connector.</li></ul> <p>For Apache 1.3.x, the connection module is mod_jrun.so; for Apache 2.x, the connection module is mod_jrun20.so.</p>
IIS	<p>The Web Server Configuration Tool adds the following elements at either the global level (default) or web server level:</p> <ul style="list-style-type: none"><li>• An ISAPI filter.</li><li>• Extension mappings tell IIS to route requests for ColdFusion pages through the connector.</li></ul> <p>With IIS 4 and 5, the IIS connection module is jrun.dll. IIS 6 uses a connection module named jrun_iis6.dll and a helper DLL named jrun_iis6_wildcard.dll.</p>
SunONE Web Server, includes iPlanet and Netscape Enterprise Server (NES)	<p>The Web Server Configuration Tool adds the following elements to SunONE Web Server configuration files:</p> <ul style="list-style-type: none"><li>• obj.conf A NameTrans directive for the JRun filter and ObjectType directives to route requests for ColdFusion pages through the connector.</li><li>• magnus.conf Init directives to load and initialize the connector.</li></ul> <p>In Windows, the SunONE web server connection module is jrun_nsapi35.dll; on UNIX and Linux, the SunONE web server connection module is libjrun_nsapi35.so; on AIX, the module name is libjrun_nsapi40.so.</p> <p>With NES 3.5 and iPlanet 4.x, the Web Server Configuration Tool places all settings in the obj.conf file.</p>

## Web server configuration

ColdFusion MX uses the Web Server Configuration Tool to configure an external web server with the modules and settings the connector needs to connect to ColdFusion MX. You can run the Web Server Configuration Tool through either the command-line interface or the GUI mode. In either case, the Web Server Configuration Tool configures your external web server to interact with a ColdFusion MX server, as explained in the following discussions:

- [Using GUI mode](#)
- [Using the command-line interface](#)
- [Configuration files](#)

## Using GUI mode

The Web Server Configuration Tool includes a GUI mode, which you can use to specify external web server configuration settings through a graphical interface.

**Note:** When using the Web Server Configuration Tool in GUI mode, it is critical that you select the Configure web server for ColdFusion MX applications checkbox.

### To run the Web Server Configuration Tool in GUI mode:

- 1 Open a console window.

**Tip:** In Windows, you can start the Web Server Configuration Tool by selecting Start > Programs > Macromedia ColdFusion MX > Web Server Configuration Tool.

- 2 Change to the *cf\_root/runtime/lib* (server configuration) or *jrun\_root/lib* (JRun J2EE configuration) directory.

- 3 Start the Web Server Configuration Tool using the following command:

```
java_home/bin/java -jar wsconfig.jar
```

The Web Server Configuration Tool window appears.

- 4 Click the Add button.
- 5 Select the Configure web server for ColdFusion MX applications option.
- 6 In the Server drop-down list box, select the server or cluster name that you want to configure. Individual server names in a cluster do not appear. Clustering support is only available on the JRun J2EE configuration.

**Note:** The server or cluster does not have to reside on the web server computer. In this case, enter the IP address or server name of the remote computer in the JRun Host field.

- 7 In the Web Server Properties area, enter web-server-specific information, and click OK.
- 8 Move the CFIDE and cfdocs directories from *cf\_root/wwwroot* to your web server root directory. In addition, copy your application's CFM pages from *cf\_root/wwwroot* to your web server root directory.

## Using the command-line interface

You can also run the Web Server Configuration Tool through a command-line interface. To run the command-line interface, open a console window, change to the *cf\_root/runtime/lib* (server configuration) or *jrun\_root/lib* (J2EE configuration with JRun) directory, and use the following command-line syntax:

```
java_home/bin/java -jar wsconfig.jar [-options]
```

The following table lists the options:

Option	Description
-ws	Specifies the web server, as follows: <ul style="list-style-type: none"><li>• IIS</li><li>• Apache</li><li>• NES</li><li>• iPlanet</li></ul>
-dir	Path to the configuration directory (Apache conf or NES/iPlanet config)

Option	Description
-site	Specifies the IIS website name. Specify All or 0 to configure the connector at a global level, which applies to all IIS websites.
-host	Specifies the ColdFusion server address. The default is localhost.
-server	Specifies the ColdFusion server name. The default is default.
-username	Specifies a username defined to the JRun server. The default is guest account.
-password	Specifies a password that corresponds to -username. The default is guest account.
-norestart	Do not restart the web server.
-cluster	Specifies the JRun cluster name. Use this option to define a connection to a JRun cluster instead of a single server.
-l	Enables verbose logging for the connector.
-a	Enables native OS memory allocation.
-s	Enables SSL between the connector and JRun server.
-map .cfm,.cfc,.cfml,.jsp,.jws	Specifies the extension mappings list (to use the webserver connector with ColdFusion MX, you should specify .cfm,.cfc,.cfml,.jsp,.jws)
-filter-prefix-only	Sets ignoresuffixmap=true in the jrun.ini file. This means that the connector module runs as an IIS extension. (IIS only).
-coldfusion	In conjunction with -upgrade, forces the replace of the connector module regardless of version or date stamp and starts the web server if it is down. Always use this option when configuring a web server for use with ColdFusion MX.
-upgrade	Upgrades existing configured connectors with newer modules from a newer wsconfig.jar.
-service	Specifies the Apache Windows service name. The default is <i>Apache</i> .
-bin	Path to Apache server binary file (apache.exe in Windows, httpd on Unix).
-script	Path to Apache UNIX control script file (apachectl, slightly different with certain Apache variants, such as Stronghold).
-v	Enables verbose output from the Web Server Configuration Tool.
-list	Lists all configured web servers.
-list -host server-host	Lists all JRun servers on the specified host.
-remove	Removes a configuration. Requires -ws and either -dir or -site.
-uninstall	Uninstalls all configured connectors.
-h	Lists all parameters.

## Using the batch files and shell scripts

ColdFusion MX ships with batch files and shell scripts that implement typical command-line connector configurations. These files are in *cf\_root/bin/connectors*. For example, *IIS\_connector.bat* configures all sites in IIS to site 0, which establishes a globally defined connector so that all sites inherit the filter and mappings.

If you use Apache or iPlanet, use these files as prototypes, editing and saving them, as appropriate for your site.

## Command-line interface examples

This section provides examples of multiple use-cases for different web servers:

- Configure a specific IIS site:

```
java_home/bin/java -jar wsconfig.jar -ws iis -site "web31" -filter-prefix-only -map .cfm,.cfc,.cfml,.jsp,.jws -coldfusion -v
```

On systems where all sites are .cfm, there is generally no need to configure an individual site.

- Configure all existing IIS sites: (ISPs):

```
java_home/bin/java -jar wsconfig.jar -ws iis -site 0 -filter-prefix-only -map .cfm,.cfc,.cfml,.jsp,.jws -coldfusion -v
```

This does not automatically configure newly added sites after the first "-site 0" run but you can rerun with "-site 0" at a later time and the Web Server Configuration Tool configures new sites only.

- Netscape on UNIX:

```
java_home/bin/java -jar /opt/coldfusionmx/runtime/lib/wsconfig.jar -ws nes -dir [path to config] -map .cfm,.cfc,.cfml,.jsp,.jws -coldfusion -v
```

- SunOne Web Server on UNIX:

```
java_home/bin/java -jar /opt/coldfusionmx/runtime/lib/wsconfig.jar -ws iplanet -dir [path to config] -map .cfm,.cfc,.cfml,.jsp,.jws -coldfusion -v
```

- Apache on UNIX:

```
java_home/bin/java -jar /opt/coldfusionmx/runtime/lib/wsconfig.jar -ws Apache -bin /opt/apache2/bin/httpd -script /opt/apache2/bin/apachectl -dir /opt/apache2/conf -map .cfm,.cfc,.cfml,.jsp,.jws -coldfusion -v
```

- Apache in Windows:

```
java_home/bin/java -jar wsconfig.jar -ws apache -dir "c:\program files\apache group\apache2\conf" -map .cfm,.cfc,.cfml,.jsp,.jws -coldfusion -v
```

## Configuration files

The Web Server Configuration Tool stores properties in configuration files, as follows:

- **IIS** In the *jrunit.ini* file, typically found in a subdirectory of the *cf\_root/runtime/lib/wsconfig* directory. It also defines a filter and extension mappings in the IIS metabase.
- **Apache** In the *httpd.conf* file, typically found in the *apache\_root/conf* directory.
- **Netscape/iPlanet** In the *obj.conf* and *magnus.conf* files, typically found in the *server-http-xxx/config* directory.

The following table describes the web server connector properties in the web server configuration files. The web server connector uses these settings to help it find the ColdFusion server and know which servers to connect to.

Property	Description
bootstrap	The IP address and port on which the JRun server's proxy service is listening for connector requests. JRun must also be configured to listen on this port/address combination, the ProxyService must be activated, and the JRun server must be running. For example, 127.0.0.1:51010.
serverstore	The name of file that contains information on for the associated JRun server. The connector creates this file automatically. The default is jrunserver.store.
verbose	Creates more detailed web server log file entries for the connector. Enabling this option can cause the web server's log files to fill quickly. Specify true or false; the default is false. In Apache, the connector writes to the error log; on iPlanet, to errors; and on IIS, the connector writes to its own log in the related wsconfig subdirectory.
scriptpath	IIS only. Points to the virtual /JRunScripts directory on the web server.
errorurl	(Optional) Specifies the URL to a file containing a customized error message. This property is commented out by default.
ssl	(Optional) Enables SSL between the web server and the JRun server. Specify true or false. Because most web servers are already inside a firewall, you typically leave this property set to false (the default).
apialloc	Enables native OS memory allocation rather than the web server's allocator (for use on Solaris with iPlanet at the direction of Macromedia Support staff).

Each time you run the Web Server Configuration Tool, it creates a new directory beneath *cf\_root/runtime/lib/wsconfig*. For example, the first time you run the tool, it creates files under *cf\_root/runtime/lib/wsconfig/1*, the second time, it creates *cf\_root/runtime/lib/wsconfig/2*, and so on. Each of these subdirectories contains the appropriate platform-specific connector module and web server-specific supporting files.

## Example configuration files

To help describe the web server configuration file parameters, this section provides examples of connector-specific web server properties. These examples assume that JRun and the web server are on the same computer.

### Apache configuration file

A typical httpd.conf file for an installation of ColdFusion on the same machine as an Apache 2.0 web server follows.

```
...
LoadModule jrun_module /opt/coldfusionmx/runtime/lib/wsconfig/1/mod_jrun20.so
<IfModule mod_jrun20.c>
    JRunConfig Verbose false
    JRunConfig Apialloc false
    JRunConfig Ssl false
    JRunConfig Ignoresuffixmap false
    JRunConfig Serverstore "/opt/coldfusionmx/runtime/lib/wsconfig/1/
    jrunserver.store"
    JRunConfig Bootstrap 127.0.0.1:51010
```

```

    #JRunConfig Errorurl <optionally redirect to this URL on errors>
    AddHandler jrun-handler .cfm .cfc .cfml .jsp .jws
</IfModule>

```

## IIS configuration file

For IIS, JRun uses the `jrun.ini` file to initialize `jrun.dll` (`jrun_iis6.dll` on IIS 6). A typical `jrun.ini` file follows:

```

verbose=false
scriptpath=/JRunScripts/jrun.dll
serverstore=C:/CFusionMX/runtime/lib/wsconfig/1/jrunserver.store
bootstrap=127.0.0.1:51010
apialloc=false
ssl=false
ignoresuffixmap=true
#errorurl=<optionally redirect to this URL on errors>

```

## Netscape/iPlanet configuration file

A typical `obj.conf` file for Netscape/iPlanet web servers follows:

```

...
<Object name=default>
NameTrans fn="jrunfilter" serverstore="C:/CFusionMX/runtime/lib/wsconfig/2/
    jrunserver.store" bootstrap="127.0.0.1:51010"
...
ObjectType fn=type-by-exp exp=*.cfm type="jrun-internal/"
ObjectType fn=type-by-exp exp=*.cfc type="jrun-internal/"
ObjectType fn=type-by-exp exp=*.cfml type="jrun-internal/"
ObjectType fn=type-by-exp exp=*.jsp type="jrun-internal/"
ObjectType fn=type-by-exp exp=*.jws type="jrun-internal/"
ObjectType fn=type-by-extension
ObjectType fn=force-type type=text/plain
Service method=(GET|POST) type="jrun-internal/" fn="jrunservice"
Service type="magnus-internal/jsp" fn="NSServletService"
Service method=(GET|HEAD) type=magnus-internal/imagemap fn=imagemap
Service method=(GET|HEAD) type=magnus-internal/directory fn=index-common
Service method=(GET|HEAD|POST) type=*~magnus-internal/* fn=send-file
AddLog fn=flex-log name="access"
</Object>
...
<Object name="jrun">
PathCheck fn="jrunfilter"
Service fn="jrunservice"
</Object>
...

```

A typical `magnus.conf` file for Netscape/iPlanet web servers follows:

```

...
Init fn="load-modules" shlib="C:/CFusionMX/runtime/lib/wsconfig/2/
    jrun_nsapi35.dll"
funcs="jruninit,jrunfilter,jrunservice"
Init fn="jruninit" serverstore="C:/CFusionMX/runtime/lib/wsconfig/2/
    jrunserver.store"
bootstrap="127.0.0.1:51010" verbose="false" apialloc="false" ssl="false"
    ignoresuffixmap="false"

```



## Advanced configurations

You typically use the Web Server Configuration Tool to configure a connection between the web server and ColdFusion server running on the same computer. However, you can use the web server connector to route requests to multiple virtual sites to a single ColdFusion server. This section also describes how to configure SSL between the web server and ColdFusion MX.

### Multihoming

In a multihomed environment, you have multiple virtual hosts (also known as virtual sites) connected to a single ColdFusion server. You might use these virtual hosts for separate applications, such as HR, payroll, and marketing or for separate users in a hosting environment.

**Note:** You use web-server-specific methods to create separate virtual websites for each use.

Two important multihoming configuration tasks are copying the `cform.js` file and disabling the `cacheRealPath` attribute:

- **Copying the `cform.js` file** If any of the applications under a virtual host use the `cform` tag, you must enable the virtual website to find the JavaScript files under the `CFIDE/scripts` directory. You can either copy the *original\_web\_root*/`CFIDE/scripts` directory to a `CFIDE/scripts` directory on your virtual website or modify the `cform` tags to use the `scriptsrc` attribute to specify the location of the `cform.js` file.
- **Disabling `cacheRealPath`** To ensure that ColdFusion MX always returns pages from the correct server, ensure that Cache Web Server Paths is disabled in the Caching page of the ColdFusion MX Administrator (when using the J2EE configuration on JRun, set the `cacheRealPath` attribute to `false` for the `ProxyService` in the `jrun.xml` file).

The procedures you perform to enable multihoming differ for each web server:

- [IIS](#)
- [Apache](#)
- [Sun ONE Web Server \(iPlanet\)](#)

### IIS

When using IIS, you use the IIS Administrator to create additional websites and run the Web Server Configuration Tool. You store CFM pages under the web root of each virtual website.

**To connect multiple virtual sites on IIS to a single ColdFusion server:**

- 1 Use the IIS Administrator to create virtual websites, as necessary. The web root directory should enable read, write, and execute access. For more information, see your IIS documentation.
- 2 Configure DNS for each virtual website, as described in your IIS documentation.
- 3 Test each virtual website to ensure that HTML pages are served correctly.
- 4 Run the Web Server Configuration Tool, as follows:
  - **GUI** Specify IIS for the Web Server, All for the IIS Web Site dropdown listbox and select the Configure web server for ColdFusion MX applications field.
  - **Command line** Specify `-site 0` and `-map` options, as shown in the following example:

```
java_home/bin/java -jar wsconfig.jar -ws iis -site 0 -filter-prefix-only  
-map .cfm,.cfc,.cfml,.jsp,.jws -coldfusion -v
```

- 5 The JavaScript validation used by the `cform` tag references the `CFIDE/scripts/cform.js` file. However in a multi-homed environment, each virtual website may not contain this directory and file. Either copy this file and store it in your virtual website's web root in a `CFIDE/scripts` directory or modify all `cform` tags to use the `scriptsrc` attribute to specify the location of the `cform.js` file.
- 6 Ensure that Cache Web Server Paths is disabled in the Caching page of the ColdFusion MX Administrator (J2EE configuration on JRun, set the `cacheRealPath` attribute to `false` for the `ProxyService` in the `jrun.xml` file).
- 7 Test each virtual website to ensure that CFM pages are served correctly.

## Apache

When using Apache, you modify the `apache_root/conf/httpd.conf` file to create virtual hosts and run the Web Server Configuration Tool. You store CFM pages under the web root of each virtual website.

### To connect multiple virtual hosts on a web server to a single ColdFusion server:

- 1 Open the `apache_root/conf/httpd.conf` file in a text editor and create virtual hosts, as necessary. For more information, see your Apache documentation. For example:

```
...
# JRun Settings
LoadModule jrun_module "C:/CFusionMX/runtime/lib/wsconfig/2/mod_jrun20.so"
<IfModule mod_jrun20.c>
    JRunConfig Verbose false
    JRunConfig Apialloc false
    JRunConfig Ssl false
    JRunConfig Ignoresuffixmap false
    JRunConfig Serverstore "C:/CFusionMX/runtime/lib/wsconfig/2/
    jrunserver.store"
    JRunConfig Bootstrap 127.0.0.1:51020
    #JRunConfig Errorurl <optionally redirect to this URL on errors>
    AddHandler jrun-handler .jsp .jws .cfm .cfml .cfc
</IfModule>

NameVirtualHost 127.0.0.1
<VirtualHost 127.0.0.1>
    ServerAdmin admin@yoursite.com
    DocumentRoot "C:/Program Files/Apache Group/Apache2/htdocs"
    ServerName SERVER02
    ErrorLog logs/error.log
</VirtualHost>
<VirtualHost 127.0.0.1>
    ServerAdmin admin@yoursite.com
    DocumentRoot "C:/Program Files/Apache Group/Apache2/htdocs2"
    ServerName mystore
    ErrorLog logs/error-store.log
</VirtualHost>
<VirtualHost 127.0.0.1>
    ServerAdmin admin@yoursite.com
    DocumentRoot "C:/Program Files/Apache Group/Apache2/htdocs3"
    ServerName myemployee
    ErrorLog logs/error-employee.log
</VirtualHost>
...
```

- 2 Configure DNS for each virtual website, as described in your web server documentation.

- 3 Restart Apache to ensure that the virtual hosts are defined correctly. You store CFM files for each virtual host in the directory specified by the DocumentRoot directive.
- 4 Test each virtual host to ensure that HTML pages are served correctly.
- 5 Run the Web Server Configuration Tool, as follows:
  - **GUI** Specify Apache for the Web Server, specify the directory containing the httpd.conf file, and select the Configure web server for ColdFusion MX applications option.
  - **Command line** Specify `-ws apache` and the directory containing the httpd.conf file, as shown in the following example:
 

```
java_home/bin/java -jar wsconfig.jar -ws apache -dir
"c:\program files\apache group\apache2\conf" -map
.cfm,.cfc,.cfml,.jsp,.jws -coldfusion -v
```

For additional Unix command-line examples, see [“Using the command-line interface” on page 60](#).
- 6 The JavaScript validation used by the `cform` tag references the CFIDE/scripts/cform.js file. However in a multi-homed environment, each virtual website may not contain this directory and file. Either copy this file and store it in your virtual website’s web root in a CFIDE/scripts directory or modify all `cform` tags to use the `scriptsrc` attribute to specify the location of the `cform.js` file.
- 7 Ensure that Cache Web Server Paths is disabled in the Caching page of the ColdFusion MX Administrator (in the J2EE configuration on JRun, set the `cacheRealPath` attribute to false for the ProxyService in the `jrun.xml` file).
- 8 Test each virtual host to ensure that CFM pages are served correctly.

## Sun ONE Web Server (iPlanet)

When using Sun ONE Web Server version 6, you use the Server Administrator to create virtual servers and run the Web Server Configuration Tool. You store CFM pages under the web root of each virtual website.

**Note:** For earlier versions of iPlanet and Netscape Enterprise Server (NES), you must create separate server instances for each site and run the Web Server Configuration Tool once for each site.

### To connect multiple virtual hosts on a web server to a single ColdFusion server:

- 1 Using the Sun ONE Web Server Administrator, create virtual web servers for use by ColdFusion MX. For more information, see your Sun ONE Web Server documentation.
- 2 Configure DNS for each virtual website, as described in your web server documentation.
- 3 Test each virtual server to ensure that HTML pages are served correctly.
- 4 Run the Web Server Configuration Tool, as follows:
  - **GUI** Specify Netscape Enterprise Server/iPlanet for the Web Server, specify the directory containing the `obj.conf` and `magnus.conf` files, and select the Configure web server for ColdFusion MX applications option.
  - **Command line** Specify `-ws iplanet` and the directory containing the `obj.conf` file, as shown in the following example:
 

```
/opt/coldfusionmx/runtime/jre/bin/java -jar /opt/coldfusionmx/runtime/lib/
wsconfig.jar -ws iplanet -dir [path to config] -map
.cfm,.cfc,.cfml,.jsp,.jws -coldfusion -v
```

- 5 The JavaScript validation used by the `cfform` tag references the `CFIDE/scripts/cfform.js` file. However in a multi-homed environment, each virtual website may not contain this directory and file. Either copy this file and store it in your virtual website's web root in a `CFIDE/scripts` directory or modify all `cfform` tags to use the `scriptsrc` attribute to specify the location of the `cfform.js` file.
- 6 Ensure that Cache Web Server Paths is disabled in the Caching page of the ColdFusion MX Administrator (in the J2EE configuration on JRun, set the `cacheRealPath` attribute to `false` for the `ProxyService` in the `jrun.xml` file).
- 7 Test each virtual server to ensure that CFM pages are served correctly.

## SSL

The web server connectors supports the use of secure sockets layer (SSL) between the web server and a ColdFusion server. This is typically not necessary, since the web server is behind a firewall in most production configurations. However, for maximum security, you can use SSL with the web server connector.

### To enable SSL for the web server connector:

- 1 Generate a keystore using the following Java `keytool` command. For example:
 

```
keytool -genkey -dname "cn=<server name or IP address>, ou=CFEngineering,
o=Macromedia, L=Newton, ST=MA, C=US" -keyalg rsa -keystore <keystore name>
```
- 2 When prompted, enter appropriate passwords that are six or more characters in length.
- 3 Rerun `keytool` to add certificates to the keystore.
 

**Note:** In a production environment you would obtain a signed certificate from a certificate authority.
- 4 Open the `cf_root/runtime/servers/default/SERVER-INF/jrun.xml` file in a text editor and set the `ProxyService` `keyStore`, `keyStorePassword`, and `trustStore` (optional) attributes to appropriate values. The `keyStore` and `trustStore` attributes should be the paths and filenames of the keystore and truststore files.
- 5 Download and build OpenSSL. The OpenSSL distribution is available at <http://openssl.org> in a `tar.gz` file. You must download the distribution and build it for your operating system based on the included installation instructions. Place the compiled OpenSSL code in a directory that is in your system path, such as `cf_root/runtime/servers/lib`.
- 6 Open the web server connector configuration file (for example, `jrun.ini`, `httpd.conf`, or `magnus.conf`) and set the `ssl` property to `true`.
 

**Note:** If using Apache virtual hosts, the `ssl` property must be outside of any `VirtualHost` directives.

To use SSL with the built-in web server, enable the `SSLService` in the `cf_root/runtime/servers/default/SERVER-INF/jrun.xml` file and set the `keyStore`, `keyStorePassword`, and `trustStore` attributes to appropriate values.

# CHAPTER 5

## Administering Security

You can secure a number of ColdFusion MX resources with password access and configure sandbox security. This chapter describes configuration options for ColdFusion security.

### Contents

About ColdFusion MX security . . . . .	69
Using sandbox security . . . . .	70

### About ColdFusion MX security

Security is especially important in web-based applications, such as those you develop in ColdFusion MX. ColdFusion developers and administrators must fully understand the security risks that could affect their development and runtime environments so they can enable and restrict access appropriately.

You implement *development security* by requiring a password to use the ColdFusion MX Administrator and a password for Remote Development Services (RDS), which allows developers to develop CFML pages remotely. You implement *runtime security* in your CFML pages and in the ColdFusion MX Administrator. ColdFusion MX has the following runtime security categories:

- **User security** Programmatically determine the logged-in user and allow or disallow restricted functionality based on the roles assigned to that user. For more information about user security, see *Developing ColdFusion MX Applications*.
- **Sandbox security** Using the Administrator, define the actions and resources that the ColdFusion pages in and below a specified directory can use.

The Security area in the Administrator lets you do the following tasks:

- Configure password protection for the Administrator.
- Configure password protection for RDS access.
- Enable, disable, and customize ColdFusion security, on the Security > Sandbox Security page (called Resource Security page in the Standard edition).

## Security and edition differences

If you have the Enterprise Edition of ColdFusion MX, you can configure multiple security sandboxes. If you have the Standard Edition of ColdFusion MX, you can only configure a single security sandbox. For more on sandbox security, see [“Using sandbox security” on page 70](#).

### ColdFusion MX Administrator password protection

The Administrator installs with secure access enabled. The password that you enter during installation is saved as the default. You are prompted to enter this password whenever you open the Administrator.

Password protection for accessing the Administrator helps guard against unauthorized modifications of ColdFusion MX, and Macromedia highly recommends using passwords. You can disable or change the Administrator password on the Security > CF Admin Password page.

### RDS password protection

If you configured password protection for RDS access when you installed ColdFusion, you are prompted for the password when you attempt to access ColdFusion MX from Macromedia Dreamweaver MX or Macromedia HomeSite+.

You can disable or change the RDS password on the Security > RDS Password page.

If you use RDS security, you rely on web server and operating system security settings to set permissions for ColdFusion application and document directories.

## Using sandbox security

Sandbox security (called Resource security in the Standard Edition) uses the location of your ColdFusion pages to control access to ColdFusion resources. A *sandbox* is a designated directory of your site to which you apply security restrictions. Sandbox security lets you specify which tags, functions, and resources (for example, files, directories, and data sources) can be used by ColdFusion pages located in and below the designated directory.

**Note:** Sandbox security is not enabled by default. You must enable it on the **Security > Sandbox Security** page before ColdFusion enforces the settings.

### Using multiple sandboxes (Enterprise Edition only)

By default, a subdirectory of a sandbox inherits the settings of the directory one level above it. However, if you define a sandbox for a subdirectory, the subdirectory no longer inherits settings from the parent, completely overriding the parent directory's sandbox settings. For example, consider the following directories:

```
C:\Inetpub\wwwroot
C:\Inetpub\wwwroot\sales
C:\Inetpub\wwwroot\rnd
C:\Inetpub\wwwroot\rnd\dev
C:\Inetpub\wwwroot\rnd\qa
```

If you define a sandbox for the wwwroot directory, the settings also apply to the sales, and rnd directories. If you also define a sandbox for the rnd directory, the rnd sandbox settings also apply to the dev and qa directories; the wwwroot and sales directories maintain their original settings.

This hierarchical arrangement of security permits the configuration of personalized sandboxes for users with different security levels. For example, if you are a web hosting administrator who hosts several clients on a ColdFusion shared server, you can configure a sandbox for each customer. This prevents one customer from accessing the data sources or files of another customer.

## Resources that can be restricted

You can restrict the following resources:

- **Data Sources** Restrict the usage of ColdFusion data sources.
- **CF Tags** Restrict usage of the ColdFusion tags that manipulate resources on the server (or on an external server), such as files, the registry, LDAP, mail, and the log.
- **CF Functions** Restrict usage of the ColdFusion functions that access the file system.
- **Files/Dirs** Enable tags and functions in the sandbox to access files and directories outside of the sandbox.
- **IP/Ports** Specify the IP addresses, ports, and port ranges that the ColdFusion tags that call third-party resources can use.

For more information, see the Administrator online Help.

**Note:** When running ColdFusion MX in the J2EE configuration on IBM WebSphere, file/directory security and IP/port security are not enabled.

## About directories and permissions

When enabling access to files outside of the sandbox, you specify the filename. When enabling access to directories outside of the sandbox, you specify *directoryname\indicator*, where *indicator* is a dash or asterisk, as follows:

- A backslash followed by a dash (\-) lets tags and functions access all files in the specified directory and recursively allows access to all files in subdirectories.
- A backslash followed by an asterisk (\\*) lets tags and functions access all files in the specified directory and a list of subdirectories, but denies access to files in any subdirectories.

You can also specify the actions that ColdFusion tags and functions are allowed to perform on files and directories outside the sandbox. The following table shows the relationship between permissions of a file and a directory:

Permission	Affect on files	Affect on directories
Read	View the file	List all files in the directory
Write	Write to the file	Not applicable
Execute	Execute the file	Not applicable
Delete	Delete the file	Delete the directory

## Adding a sandbox (Enterprise Edition only)

ColdFusion MX Enterprise Edition lets you define multiple security sandboxes.

### To add a sandbox:

- 1 Open the **Security > Sandbox Security** page in the ColdFusion MX Administrator. The Sandbox Security Permissions page appears.
- 2 In the Add Security Sandbox box, enter the name of the new sandbox. This name must be either a ColdFusion mapping (defined in the Administrator) or an absolute path.
- 3 Select New Sandbox from the drop-down list to create a sandbox based on the default sandbox, or select an existing sandbox to copy its settings to your new sandbox.
- 4 Click Add.  
The new sandbox appears in the list of Defined Directory Permissions.

## Configuring a sandbox

Before you begin security sandbox configuration, analyze your application and its usage to determine the tags, functions, and resources that it requires. You can then configure the sandbox to enable access to the required resources and disable usage of the appropriate tags and functions. For example, if the applications in the sandbox do not use the `cfregistry` tag, you can safely disable it.

**Note:** In the Standard Edition, the Root Security Context is the only sandbox. There is no initial list of defined directory permissions.

### To configure a sandbox:

- 1 Open the **Security > Sandbox Security page (Security > Resource Security** in Standard Edition) in the ColdFusion MX Administrator.
- 2 (Enterprise Edition only) In the list of Defined Directory Permissions, click the name or Edit icon for the directory.  
A screen with several tabs appears. This is the initial screen in Standard Edition. The remaining steps describe the use of each tab.
- 3 To disable a data source, in the left column of the Datasources tab, highlight the data source, and click the right arrow.  
By default, ColdFusion pages in this sandbox can access all data sources.  
**Note:** If «ALL DATASOURCES» is in the Enabled Datasources column, any data source that you add when creating this sandbox is enabled. If you move «ALL DATASOURCES» to the Disabled Datasources column, any new data source is disabled.
- 4 Click the CFTags tab.
- 5 To disable tags, in the left column of the CFTags tab, highlight the tags, and click the right arrow.  
By default, ColdFusion pages in this sandbox can access all listed tags.
- 6 Click the CFFunctions tab.
- 7 To disable functions, in the left column of the CFFunctions tab, highlight the functions, and click the right arrow.  
By default, ColdFusion pages in this sandbox can access all listed functions.
- 8 Click the Files/Dirs tab.



- 9 To enable files or directories, in the File Path box, enter or browse to the files or directories; for example, C:\pix. A file path consisting of the special token <<ALL FILES>> matches any file. For information on using the \- and \\* wildcard characters, see [“About directories and permissions” on page 71](#).
- 10 Select the permissions.  
For example, select the Read check box to let ColdFusion pages in the mytestapps sandbox read files in the C:\pix directory.
- 11 Click Add Files/Paths. When editing an existing sandbox, this button reads Edit Files/Paths.  
The file path and its permissions appear in the Secured Files and Directories list.
- 12 In the Secured Files and Directories list, verify that the file path is correct.  
The character after the backslash is important. For information, see [“About directories and permissions” on page 71](#).  
**Note:** The Files/Dirs tab works together with the file-based permissions of the operating system. To restrict a user from browsing another user’s directory, you must use file-based permissions.
- 13 Click the IP/Port tab.
- 14 To turn off default behavior (global access to all IP addresses and ports) enter the IP addresses and port numbers that pages in this sandbox can connect to using tags that access external resources (for example, cfmail, cfpop, cfldap, cfhttp, and so on). You can specify an IP address, a server name (such as www.someservername.com), or a domain name (such as someservername.com). Specifying a port restriction is optional.  
**Note:** This behavior differs from other tabs, such as CFTags, where you select items to disable. If you set any values in this tab, external-resource tags executed in this sandbox can access only the specified servers and ports.  
For example, to allow this sandbox access to 207.88.220.3 on ports 80 and lower, perform the following steps:
  - a In the IP Address field, enter 207.88.220.3.
  - b In the Port field, enter 80, and click This Port and Lower.**Tip:** To deny access by these ColdFusion tags to an entire site, enable access for a local resource, such as your local mail server, ftp server, and so on.
- 15 Click Finish to save changes to the sandbox.



# CHAPTER 6

## Using Multiple Server Instances

When you install ColdFusion MX Enterprise using a J2EE deployment, you can use J2EE application-server-specific functionality to create multiple server instances. Deploying ColdFusion MX on multiple server instances lets you isolate individual applications and leverage clustering functionality.

### Contents

Overview of multiple server instances .....	75
Defining additional server instances .....	76
Deploying ColdFusion MX .....	76
Enabling application isolation .....	77
Enabling load balancing and failover .....	81

### Overview of multiple server instances

When using the J2EE install, you can define multiple server instances on a single machine, each running ColdFusion MX. Running multiple instances of ColdFusion MX has the following advantages:

- **Application isolation** You deploy an independent application to each server instance. Each server instance has separate settings and because each server instance runs in its own JVM, problems encountered by one application have no effect on other applications.
- **Load balancing and failover** You deploy the same application to each server instance and add the instances to a cluster. The web server connector optimizes performance and stability by automatically balancing load and by switching requests to another server instance when a server instance stops running.

The remaining discussions in this chapter assume that you have installed JRun. If you have not installed JRun, rerun the ColdFusion MX Enterprise install, selecting the JRun with CFMX option. This installs JRun and deploys ColdFusion MX as an expanded EAR in the cfusion JRun server.

**Note:** Discussions in this chapter apply to running ColdFusion MX Enterprise as a J2EE application deployed on top of a J2EE application server. Although the examples in this chapter describe using JRun 4, other J2EE application servers provide equivalent capabilities, and most of the concepts described in this chapter apply when deploying ColdFusion MX Enterprise on those J2EE servers.

## File location considerations

In the J2EE configuration, you can store CFM pages either under the external web server root or under the ColdFusion web application root. ColdFusion MX first looks for CFM files in the web application root and then looks in the external web server root.

The discussions in this chapter assume that you are using an external web server and that you store your CFM pages under the external web server root.

## Defining additional server instances

A single installation of JRun supports multiple server instances (also called JRun servers) running on the same machine. Each server instance has associated with it a separate running Java Virtual Machine (JVM), which executes all ColdFusion pages for that instance. The JVM, also known as the Java Runtime Environment (JRE), is the software implementation of a CPU. It contains everything necessary to run programs written for the Java platform. Additionally, you can define classpaths, data sources, and other resources for each application at the server instance level.

You use the J2EE application server's management console to define and manage server instances. For JRun, this is the JRun Management Console (JMC). For more information on the JMC, see *Getting Started with JRun* in the JRun documentation set.

### To define a server instance:

- 1 Ensure that the admin JRun server is running by starting the JRun Launcher (*jrun\_root/bin/jrun.exe* in Windows, *jrun\_root/bin/jrun* on UNIX). Start the admin server if it is not running.
- 2 Open the JMC by typing `http://hostname:8000` in a browser. You are prompted for the user name and password specified during the installation.
- 3 Select Create New Server, and specify a host name, server name, and server directory. In most cases you can accept the suggested host name and server directory.
- 4 Click Create Server. JRun shows the ports to be used by the new server instance.
- 5 (Optional) Specify different port numbers and click Update Port Numbers. The JMC automatically looks for unused ports so you don't typically change the suggested port numbers. Make a note of the JRun web server (JWS) port number; you use it later in this procedure to ensure a successful server startup.
- 6 Click Finish. JRun creates a server that includes a default enterprise application.
- 7 Start the server instance using the JMC, the JRun Launcher, or the command line (*jrun\_root/bin/jrun start servername*).
- 8 Ensure that the server instance is running by typing `http://hostname:jwsportnumber/index.jsp` in a browser. A Welcome page appears.

## Deploying ColdFusion MX

After you create the new server instance, you must deploy the ColdFusion MX application. When using JRun, you must deploy an expanded directory structure. If you already have ColdFusion MX deployed on a JRun server, you can copy the ColdFusion application to the new JRun server and it will deploy automatically. Depending on your requirements, you might have to modify server settings, such as the data sources defined to each server instance.

For more information on deploying ColdFusion MX, see *Installing and Using ColdFusion MX*.

## Enabling application isolation

When you install the J2EE version of ColdFusion MX Enterprise on top of JRun, you can use the JMC to create multiple server instances and deploy ColdFusion MX on each instance. This configuration provides multiple ColdFusion MX web applications in fully independent processes, with no shared ColdFusion or J2EE server resources. In this configuration, you typically have a single web server with multiple virtual hosts (or sites) and multiple server instances on one computer.

**Note:** Although this discussion describes using JRun 4, other J2EE application servers provide equivalent capabilities, and most of the concepts apply when deploying ColdFusion MX Enterprise on those J2EE servers.

Running independent applications this way has several advantages, including the following:

- Errors at the levels of the ColdFusion application or the JRun server do not affect any other ColdFusion applications.
- You can support multihomed servers, where a single web server supports multiple IP addresses or domain names, such as `www.mycompany.com` and `services.anothercompany.com`, each running out of a separate web root.
- Individual applications can use different JVM configurations, or even different JVM implementations. This feature is particularly useful if one application requires a particularly large Java heap. To specify customized JVM options, start the JRun server instance from the command line using the `-config` option of the `jrun` command, which specifies a customized `jvm.config` file. This is explained in the “Starting and stopping JRun servers” discussion in *Installing JRun*.

**Note:** These instructions describe creating multiple server instances on a single computer. To create multiple server instances on separate computers, each computer requires a separate license of ColdFusion MX Enterprise Edition.

To achieve complete application isolation, you use web-server-specific functionality to create a separate website for each application. Web servers have different terminology for this concept. For example, in IIS, you define separate websites (available in Windows server editions only) and in Apache, you create multiple virtual hosts.

These instructions apply when running ColdFusion MX on JRun. The principles apply when running ColdFusion MX on other J2EE application servers. However, not all J2EE application servers integrate with external web servers. For more information, see [“Multihoming” on page 65](#).

These instructions assume that you deploy each application at the context root of `/`, which enables users to access CFM pages by specifying `http://hostname/pagename.cfm`. If other web applications are running in the server instance, another web application may already use the context root of `/` and you must deploy ColdFusion MX using a different context root, such as `/cfusion`, which requires that users access CFM pages by specifying `http://hostname/cfusion/pagename.cfm`. For more information on using a context root, see *Installing and Using ColdFusion MX*.

**Note:** Although `cfusion` is the context root, it does not relate to your web root directory structure and you still store CFM pages in the web root directory.

### To use multiple server instances for application isolation:

- 1 Create a separate server instance.
- 2 Deploy ColdFusion MX on the server instance.

- 3 Open the ColdFusion MX Administrator on the server instance using the built-in web server (*hostname:portnumber/CFIDE/administrator/index.cfm*) and define the resources (such as data sources and Verity collections) required for the application. Performing this step also ensures that ColdFusion MX was deployed successfully.
- 4 Using your web-server-specific method, create a virtual website (or separate website) for the application.

This is different for each web server; for more information, see [“Multihoming” on page 65](#) or consult your web server documentation.
- 5 Test each virtual website to ensure that HTML pages are served correctly.
- 6 Follow the instructions for your web server to configure the connection between your virtual website and the server instance. For more information, see [“Web server configuration for application isolation” on page 78](#).
- 7 Store your application’s CFM files in the web root of the virtual website.
- 8 Test your application using the virtual website.
- 9 Test the ColdFusion MX Administrator. If you configured your web server during installation, the CFIDE directory is under the original web root and you must copy it to each virtual website or create a web server mapping to the original CFIDE directory.
- 10 Repeat these steps for each server instance.

## Web server configuration for application isolation

When using multiple server instances for application isolation, the steps you perform to configure communication between the website and the server instance differ by web server. This section contains the following discussions:

- [Configuring application isolation in IIS](#)
- [Configuring application isolation in Apache](#)
- [Configuring application isolation in SunONE Web Server](#)

### Configuring application isolation in IIS

When using multiple virtual websites with multiple server instances under IIS, you define separate filters and mappings for each virtual website/server instance combination.

This discussion assumes that you have already created server instances and virtual websites, as described in [“Enabling application isolation” on page 77](#).

#### To configure multiple server instances for application isolation when using IIS:

- Run the Web Server Configuration Tool multiple times, once for each virtual website, specifying a different site and server instance each time. For more information on running the Web Server Configuration Tool, see [“Using an external web server” on page 59](#).

### Configuring application isolation in Apache

When using multiple virtual hosts with multiple server instances under Apache, you edit the `httpd.conf` file manually.

This discussion assumes that you have already created server instances and virtual websites, as described in [“Enabling application isolation” on page 77](#).

## To configure multiple server instances for application isolation when using Apache:

- 1 Run the Web Server Configuration Tool once, specifying the location of the Apache httpd.conf file and any other required information.
- 2 The Web Server Configuration Tool creates a sequentially numbered subdirectory under *jrun\_root/lib/wsconfig*. You can use the subdirectory created by the Web Server Configuration Tool for one of your virtual hosts but you must create additional subdirectories for all other virtual hosts. For example, the first time you run the Web Server Configuration Tool, it creates *jrun\_root/lib/wsconfig/1*; if you have two other virtual hosts, you must manually create two other directories, *jrun\_root/lib/wsconfig/mystore* and *jrun\_root/lib/wsconfig/myemp* directories. These directories can be empty.
- 3 Open the *jrun\_root/servers/servername/SERVER-INF/jrun.xml* file for each of your server instances, ensure that the `deactivated` element is set to `false`, and note the value of the `port` element for the `ProxyService` service. For example:

```
...
<service class="jrun.servlet.jrpp.JRunProxyService" name="ProxyService">
  <attribute name="activeHandlerThreads">25</attribute>
  <attribute name="backlog">500</attribute>
  <attribute name="deactivated">false</attribute>
  <attribute name="interface">*</attribute>
  <attribute name="maxHandlerThreads">1000</attribute>
  <attribute name="minHandlerThreads">1</attribute>
  <attribute name="port">51002</attribute>
...
```

- 4 Open the *apache\_root/conf/httpd.conf* file in a text edit and find the `VirtualHost` directives. The settings added by the Web Server Configuration Tool are after the last `</IfModule>` directive, as the following example shows:

```
...
# JRun Settings
LoadModule jrun_module "C:/JRun4/lib/wsconfig/1/mod_jrun20.so"
<IfModule mod_jrun20.c>
  JRunConfig Verbose false
  JRunConfig Apialloc false
  JRunConfig Ssl false
  JRunConfig Ignoresuffixmap false
  JRunConfig Serverstore "C:/JRun4/lib/wsconfig/1/jrunserver.store"
  JRunConfig Bootstrap 127.0.0.1:51020
  #JRunConfig Errorurl <optionally redirect to this URL on errors>
  AddHandler jrun-handler .jsp .jws .cfm .cfml .cfc
</IfModule>

NameVirtualHost 127.0.0.1
<VirtualHost 127.0.0.1>
  ServerAdmin admin@mysite.com
  DocumentRoot "C:/Program Files/Apache Group/Apache2/htdocs"
  ServerName SERVER02
  ErrorLog logs/error.log
</VirtualHost>
<VirtualHost 127.0.0.1>
  ServerAdmin admin@mysite.com
  DocumentRoot "C:/Program Files/Apache Group/Apache2/htdocs2"
  ServerName mystore
  ErrorLog logs/error-store.log
</VirtualHost>
<VirtualHost 127.0.0.1>
  ServerAdmin admin@mysite.com
  DocumentRoot "C:/Program Files/Apache Group/Apache2/htdocs3"
```

```

    ServerName myemployee
    ErrorLog logs/error-employee.log
</VirtualHost>
...

```

- 5 For each `VirtualHost` directive, copy the `IfModule` directive from its default location outside the `VirtualHost` directive to the last element in the `VirtualHost` directive.
- 6 Delete the `Apialloc`, `Ssl`, and `Ignore suffixmap` elements in the `IfModule` directive for each virtual host. Modify the `Serverstore` and `Bootstrap` elements to point to the appropriate proxy port (from the `jrunit.xml` file) and `jrunit_root/lib/wsconfig/subdirectory/jrunserver.store` file, which the web server connector creates automatically. Do not modify the `jrunit-handler` line.
- 7 In the original `IfModule` directive, remove the `Serverstore` and `Bootstrap` lines. The following example shows three virtual hosts, two of which are configured for ColdFusion MX:

```

...
# JRun Settings
LoadModule jrun_module "C:/JRun4/lib/wsconfig/1/mod_jrun20.so"
<IfModule mod_jrun20.c>
    JRunConfig Verbose false
    JRunConfig Apialloc false
    JRunConfig Ssl false
    JRunConfig Ignore suffixmap false
    #JRunConfig Serverstore "C:/JRun4/lib/wsconfig/1/jrunserver.store"
    #JRunConfig Bootstrap 127.0.0.1:51020
    AddHandler jrun-handler .jsp .jws .cfm .cfml .cfc
</IfModule>
NameVirtualHost 127.0.0.1
<VirtualHost 127.0.0.1>
    ServerAdmin admin@mysite.com
    DocumentRoot "C:/Program Files/Apache Group/Apache2/htdocs"
    ServerName RNIELSEN02
    ErrorLog logs/error.log
</VirtualHost>
<VirtualHost 127.0.0.1>
    ServerAdmin admin@mysite.com
    DocumentRoot "C:/Program Files/Apache Group/Apache2/htdocs2"
    ServerName rnielsenstore
    ErrorLog logs/error-store.log
    <IfModule mod_jrun20.c>
        JRunConfig Verbose true
        JRunConfig Serverstore "C:/JRun4/lib/wsconfig/mystore/jrunserver.store"
        JRunConfig Bootstrap 127.0.0.1:51002
        AddHandler jrun-handler .jsp .jws .cfm .cfml .cfc
    </IfModule>
</VirtualHost>
<VirtualHost 127.0.0.1>
    ServerAdmin admin@mysite.com
    DocumentRoot "C:/Program Files/Apache Group/Apache2/htdocs3"
    ServerName rnielsenemployee
    ErrorLog logs/error-employee.log
    <IfModule mod_jrun20.c>
        JRunConfig Verbose true
        JRunConfig Serverstore "C:/JRun4/lib/wsconfig/myemp/jrunserver.store"
        JRunConfig Bootstrap 127.0.0.1:51003
        AddHandler jrun-handler .jsp .jws .cfm .cfml .cfc
    </IfModule>
</VirtualHost>
...

```

- 8 Restart Apache.



## Configuring application isolation in SunONE Web Server

When using multiple virtual hosts with multiple server instances under SunONE Web Server, you create multiple SunONE Web Server instances, one for each ColdFusion server instance.

This discussion assumes that you have already created server instances, as described in [“Enabling application isolation” on page 77](#).

**To configure multiple server instances for application isolation when using SunONE Web Server:**

- Run the Web Server Configuration Tool multiple times, once for each SunONE Web Server server instance, specifying a different configuration directory and ColdFusion server instance each time.

## Enabling load balancing and failover

Load balancing is an enterprise-level feature in which the application server automatically alternates requests among the server instances in a cluster. Clustering also enables application servers to route requests to a running server instance when the original server instance goes down.

**Note:** These instructions apply only when running ColdFusion MX on JRun. When deploying ColdFusion MX on other J2EE application servers, consult the application server documentation for information on enabling session replication.

You can get load balancing and failover by deploying the ColdFusion application to multiple server instances and adding the instances to a cluster. The web server connector optimizes performance and stability by automatically balancing load and by switching requests to another server instance when a server instance stops running.

For maximum failover protection, use multiple computers in a cluster. However, you must purchase a separate ColdFusion MX license for each computer.

**Note:** If you set up and test multiple server instances while running the 30-day Trial version, the cluster may not continue to function appropriately when the Trial version reverts to Developer version after 30 days.

To implement failover for the server instances in a cluster, you must enable session replication. Session replication coordinates session information in real-time among the server instances in a cluster. Enabling session replication lets a request be automatically routed to a running server if the current server is unavailable.

**To configure a cluster of server instances for load balancing and failover:**

- 1 Create server instances for the cluster as described in [“Defining additional server instances” on page 76](#).
- 2 Deploy ColdFusion on each server instance as described in [“Deploying ColdFusion MX” on page 76](#).
- 3 Start each server instance.
- 4 Open the ColdFusion MX Administrator on each server instance using the built-in web server. Define the resources (such as data sources and Verity collections) required for the application. If using failover, go to the Memory Variables pages, and enable J2EE sessions. You must do this for all server instances in the cluster.

**Note:** Session variables are the only memory variables that support failover. In particular, ColdFusion components do not support failover.

- 5 Open the `jrun_root/lib/security.properties` file and add the IP addresses of the other JRun servers in the cluster to `jrun.trusted.hosts`.
- 6 Open the JMC and create a cluster that contains your server instances.  
**Note:** Do not add the admin JRun server to a cluster.
- 7 If using failover, perform the following steps in the JMC:
  - a Open the cluster by clicking the cluster name in the left panel.
  - b Open the first server instance by clicking its name in the list.
  - c Open the Macromedia ColdFusion MX application.
  - d Specify the context path (usually `/`).
  - e Select Enable Session Replication.
  - f In the New Replication Buddy field, enter the names of the other servers in the cluster one-by-one, and click Add.
  - g Click Apply.
  - h Perform these steps for every server instance in the cluster.
- 8 Run the Web Server Configuration Tool. Choose your website, but instead of choosing a single server instance, select the cluster. For more information, see [“Web server configuration” on page 59](#).
- 9 Store the application’s CFM files in your external web server root directory.
- 10 Test the application to ensure that load balancing and failover work as expected.

# PART II

## Administering Verity

This part describes the Verity search tools and utilities that you can use for configuring the Verity K2 Server search engine, as well as creating, managing, and troubleshooting Verity collection.

Chapter 7: Introducing Verity Tools . . . . .	85
Chapter 8: Managing Collections with the mkvdk Utility . . . . .	89
Chapter 9: Indexing Collections with Verity Spider . . . . .	101
Chapter 11: Searching Collections with the rcvdk Utility . . . . .	143
Chapter 10: Searching Collections with K2 Server . . . . .	133
Chapter 12: Troubleshooting Collections with Verity Utilities . . . . .	149
Chapter 13: Verity Error Messages . . . . .	155



# CHAPTER 7

## Introducing Verity Tools

This chapter provides an overview of the advanced Verity features included in ColdFusion MX. These include several utilities that you can use to configure, manage, and troubleshoot search functionality in your ColdFusion applications. This chapter also introduces the Verity K2 Server, which lets you provide high-performance search capabilities for your ColdFusion applications.

### Contents

About the Verity utilities . . . . .	85
Collection structure and ColdFusion MX . . . . .	86
Verity search modes in ColdFusion MX . . . . .	87
About K2 Server . . . . .	88

### About the Verity utilities

ColdFusion MX includes several Verity utilities to diagnose and manage your collections. These tools include the mkvdk, rcvdk, rck2, and vspider utilities.

The following table describes the relationship between the major Verity utilities and the corresponding `cfcollection`, `cfsearch`, and `cfindex` ColdFusion tags (the `cfcollection` tag operates on the entire collection; the `cfindex` tag operates on records within a collection). For more information, see [Chapter 12, “Troubleshooting Collections with Verity Utilities,”](#) on page 149.

utility	cfcollection				cfindex				cfsearch
	create	repair	delete	optimize	update	delete	purge	refresh	search
mkvdk	X	X		X	X	X	X	X	
rcvdk									VDK mode search
rck2									K2 mode search
vspider	X	X		X	X		X	X	

**Note:** Collections created with ColdFusion MX and those created externally using native Verity tools differ in structure. When performing operations on Verity collections created with ColdFusion MX, you may be required to include the full path to the collection. For more information, see [“Collection structure and ColdFusion MX”](#) on page 86.

## ColdFusion MX OEM restrictions

ColdFusion MX includes an OEM-restricted version of the Verity Server. The version of Verity Server that is part of ColdFusion MX is restricted in the following areas:

- ColdFusion MX can only interact with one Verity Server at a time.
- Verity Server has the following document search limits (limits are for all collections registered to Verity Server):
  - 10,000 documents for ColdFusion MX Developer
  - 125,000 documents for ColdFusion MX Professional
  - 250,000 documents for ColdFusion MX Enterprise
  - 750,000 documents for Macromedia Spectra sites

**Note:** Each row in a database table is considered a document.

If you install a fully licensed version of Verity Server and you configure ColdFusion MX to use it, ColdFusion MX will not restrict document searches.

- The Verity Spider that is included with ColdFusion MX is licensed for local host indexing only. Contact Verity Sales for licensing options regarding the use of the Verity Spider for remote host indexing.

## Collection structure and ColdFusion MX

Collections created in ColdFusion MX, either through the ColdFusion MX Administrator or by using the `cfcollection` tag, have different directory structures than external collections. An *external* collection is one created by a tool other than ColdFusion MX, such as the native Verity indexing tool `mkvdk`. For more information on `mkvdk`, see [Chapter 8, “Managing Collections with the `mkvdk` Utility,”](#) on page 89.

The directory structure of a collection that was created with ColdFusion MX consists of two subdirectories—`custom` and `file`—that are not present in external collections.

The `type` of index used dictates which folder is populated with index data. Based on the `type` attribute of the `cfindex` tag, the `file` folder is used for `type="File"` and for `type="Path"`; the `custom` folder is used for `type="Custom"`. For more information on indexing, see *Developing ColdFusion MX Applications*.

The `type` information is important when you configure the `collPath` attribute of a collection in your `k2server.ini` file. The name of the external collection (for example, `col_01`) is `C:\myColls\col_01`. In contrast, the collection created by ColdFusion MX (`cfdocumentation`) actually contains two collections—`C:\CFusionMX\Verity\Collections\cfdocumentation\file` and `C:\CFusionMX\Verity\Collections\cfdocumentation\custom`. Using CFML tags, you only need to refer to *cfdocumentation* to access both the `file` and `custom` collections. However, since Verity tools, such as K2 Server, do not understand the ColdFusion MX collection structure, you must explicitly specify both the `file` collection and the `custom` collection in order for K2 Server to search collections created with ColdFusion MX.

For more information about configuring the `collPath` attribute, see [“Editing the `k2server.ini` file”](#) on page 133.

# Verity search modes in ColdFusion MX

Your ColdFusion MX applications can search Verity collections using two modes:

- **VDK mode** The default ColdFusion MX search mode. You register a collection with ColdFusion MX by using the `cfcollection` tag or by using the Verity Collections page in the ColdFusion MX Administrator (which also uses the `cfcollection` tag).
- **K2 mode** The high-performance K2 Server mode. Use the ColdFusion MX Administrator Verity Server page to configure ColdFusion MX to also search using K2 Server. Once you add the existing collections to `k2server.ini` and start K2 Server, the ColdFusion MX Administrator Verity Collections page indicates these K2 Server-registered collections. For more information, see [“Using K2 Server” on page 133](#).

By default, unless you configure ColdFusion MX to use K2 Server, ColdFusion MX uses VDK mode to search collections. The `cfsearch` tag is functionally identical between the two modes.

For more information about the benefits and restrictions of K2 Server, see [“About K2 Server” on page 88](#).

For more information on using VDK mode (the default Verity search mode), see *Developing ColdFusion MX Applications*.

## How ColdFusion MX determines which mode to use

ColdFusion MX determines which search mode to use by examining which server (ColdFusion or K2 Server) has registered the collection name(s) that you specified in your `cfsearch` tag.

**Note:** You cannot combine collections registered with ColdFusion MX and with K2 Server in a single `cfsearch` tag. Use two `cfsearch` tags to search both collection types from the same ColdFusion page.

Your server may contain several Verity collections. You can register a collection with the ColdFusion server (for VDK mode searches) and with the K2 Server (for K2 mode searches). To register a collection for VDK mode searches, you use a `cfcollection` tag, either directly in CFML or indirectly with the ColdFusion MX Administrator. To register a collection for K2 mode searches, edit the `k2server.ini` file. For more information, see [“Editing the k2server.ini file” on page 133](#).

In the following example, the `plants` collection has been registered with ColdFusion MX and is not listed in the `k2server.ini` file. ColdFusion MX uses the VDK mode to search this collection:

```
<cfsearch
  collection="plants"
  name="getData"
  criteria="#form.criteria#">
```

In the following example, `plants_al` has been listed in `k2server.ini` and is a unique alias. That is, the collection name, `plants_al`, is different than any Verity collections that are configured for use by ColdFusion MX. ColdFusion MX uses K2 mode to search this collection:

```
<cfsearch
  collection="plants_al"
  name="getData"
  criteria="#form.criteria#">
```

**Tip:** Check the Verity Collections page in the ColdFusion MX Administrator for possible naming conflicts between collection and collection alias names. If you have a collection named `plants` that is registered with ColdFusion MX, you must have a unique alias in the `k2server.ini` file to run a K2 mode search.

## Verity information storage

All Verity configuration data and collection name registration information are stored in an XML file (`neo-verity.xml`), which is used solely by the ColdFusion server. This XML file, which is located in `cf_root/lib`, contains two collection lists. One list contains collections that are registered with ColdFusion MX; ColdFusion MX uses the VDK mode to search these collections. The second list contains collections that are registered with K2 Server; ColdFusion MX uses the K2 mode to search these collections. You do not need to edit this XML file.

ColdFusion MX updates `neo-verity.xml` whenever one of the following occurs:

- ColdFusion starts.
- You change Verity or K2 information in the ColdFusion MX Administrator.
- You change the list of registered collections in the `cfcollection` tag.
- ColdFusion stops.

Before ColdFusion updates `neo-verity.xml`, it copies the file, using the BAK extension.

**Tip:** If the `neo-verity.xml` and `neo-verity.bak` files become damaged, use the `neo-verity.org` file. This file is a valid `neo-verity.xml` file that has not been modified since you installed ColdFusion MX.

## About K2 Server

The Verity K2 Server is a high-performance search engine designed to process searches quickly in a high-performance, distributed system. The K2 search system has a client/server model. K2 client applications, such as ColdFusion server, provide users access to document indexes stored in Verity collections. K2 Server supports simultaneous indexing of distributed enterprise repositories and handles hundreds of concurrent queries and users. You will see considerable performance improvements when using K2 Server to search Verity collections.

The K2 search system takes advantage of the latest advances in hardware and software technology, and provides the following features:

- Multithreaded architecture
- Support for Verity knowledge retrieval features, including topics
- Continuous operation support
- High scalability

ColdFusion MX installs K2 Server by default. You must make minor changes to configure K2 Server to work with ColdFusion MX.



# CHAPTER 8

## Managing Collections with the mkvdk Utility

The `mkvdk` utility is a command-line utility installed with ColdFusion MX. You can use it to perform maintenance operations on Verity collections.

### Contents

<a href="#">About the Verity <code>mkvdk</code> utility</a> .....	89
<a href="#">Getting started with the Verity <code>mkvdk</code> utility</a> .....	91

### About the Verity `mkvdk` utility

The `mkvdk` utility is an indexing application, provided with other Verity utilities, that you can use to create and maintain collections. It is a command-line utility that you can use within other applications or shell scripts to provide more sophisticated scheduling and other capabilities.

The `mkvdk.exe` file, which starts the `mkvdk` utility, is located in the `cf_root\lib\_nti40\bin` directory in Windows, and in the `cf_root/lib/platform/bin` directory on UNIX.

In these pathnames, *cf\_root* refers to the ColdFusion MX root directory. In Windows, this is typically `C:\CFusionMX`; on UNIX, this is typically `/opt/coldfusionmx`. On UNIX, *platform* refers to the UNIX version of the server that runs ColdFusion: `_solaris`, `_hpux11`, or `_ilnx21`.

### The `mkvdk` utility syntax

The following is the basic syntax of the `mkvdk` command:

```
mkvdk -collection path [option] [dockey]
```

Multiple options and dockeys can be included, as needed. If `dockey` is a list of files, it should consist of an at sign (`@`) followed by the filename that contains a simple list of files (for example, `@filelist`). For more information about the options for the `mkvdk` utility, see [“Getting started with the Verity `mkvdk` utility” on page 91](#).

The following operations occur when you use the `mkvdk` utility to create a new collection:

- 1 New collection directories are created and the specified style files are copied to the style subdirectory.
- 2 The style file settings are read and the required information is passed to the Verity search engine.
- 3 The gateway is used to open the document files, which are parsed according to the settings in various style files.

- 4 A new partition is created, which includes an index and an attribute table.
- 5 Assist data is generated, which might include a spanning word list.

When problems occur during an operation, the `mkvdk` utility writes error messages to the system log file (`sysinfo.log`). You can direct error and other messages to the console by using the `mkvdk` command with the `-outlevel` option. You can direct messages to a file of your choice by using the `-loglevel` and `-logfile` options.

The log file contains the following fields:

- Date
- Time
- Level
- Code
- Component
- Description

You can use the log file to view details about what happens during the collection creation process. Use the `mkvdk -loglevel` command and specify the numeric identifier for the message level you want, as summarized in the following table:

Type	Number
Fatal	1
Error	2
Warning	4
Status	8
Info	16
Verbose	32
Debug	64

To calculate the numeric parameter, add the numbers for the message types you want to include. The default for both `-outlevel` and `-loglevel` is 15, which selects fatal, error, warning, and status messages (1+2+4+8).

## Getting started with the Verity mkvdk utility

The following is the basic mkvdk syntax:

```
mkvdk -collection path [option] [...] [filespec] [...]
```

Where:

- Square brackets ( `[]` ) indicate optional items.
- An ellipsis (...) indicates repetition of the previous item. Thus, `[filespec] [...]` indicates an optional series of filespec items.
- `filespec` represents a document filename or a list of document filenames. If `filespec` is a list of files, it should consist of an at sign (@) followed by the filename containing the list (for example, `@filelist`).
- The `-collection path` argument creates or opens a collection. This argument is required.

Numerous optional syntax options are listed below. All syntax options must precede the first `filespec` parameter.

### Creating a collection

Creating a collection with the mkvdk utility involves setting up a collection directory structure and inserting documents into this structure. You can create a collection in two steps, using two separate commands.

#### To create a collection:

- 1 Set up a collection using the following syntax:

```
mkvdk -create -collection collectionname
```

Where `collectionname` is the path to the collection directory. Running this command creates a collection directory that includes style files with configuration information.

- 2 Insert documents using the following syntax:

```
mkvdk -collection collectionname -bulk -insert filespec
```

Where `filespec` is the name of a bulk insert file that specifies which documents to index and insert into the collection.

Alternatively, you can set up a collection and insert documents in one command, using the following syntax:

```
mkvdk -create -collection collectionname -bulk -insert filespec
```

**Note:** You can use the `-create` option only once to create the collection directory structure. After a collection directory structure has been created, do not to use the `-create` option to update the collection.

### Accessing online Help for the mkvdk utility

To display a list of mkvdk command-line options, enter the following command:

```
mkvdk -help
```

## Collection setup options

The `mkvdk` utility has a variety of collection setup options, which the following table describes:

Option	Description
<code>-create</code>	Creates a collection in the specified collection directory. It creates the directory structure, determines the index contents and sets up the document's table schema according to the style files used. If the specified collection already exists, the <code>mkvdk</code> utility exits rather than overwriting the existing collection.
<code>-style dir</code>	Specifies the style directory that contains the style files to use to create a collection. This option can only be used with the <code>-create</code> option. If you do not specify this option when you use the <code>mkvdk</code> utility to create a collection, the <code>mkvdk</code> utility uses the style files in the <code>common/style</code> directory.
<code>-description desc</code>	Sets the collection's description. Enter alphanumeric text, such as "This collection contains electronic mail from ABC Company." Include the quotation marks.
<code>-words</code>	Builds the word list for all partitions in the collection.

### Examples: setting up collections

The following examples show the commands for creating a collection and building the word list.

#### Creating a collection

The following command creates a collection in `path_2` using the style files in `path_1`, and submits and indexes the document(s) in `filespec`:

```
mkvdk -create -style path_1 -collection path_2 filespec
```

#### Building the word list

The following command builds the word list in the collection residing in the `path` directory:

```
mkvdk -words -collection path
```

## General processing options

The mkvdk utility provides a variety of general processing options, which the following table describes:

Option	Description
-collection path	Specifies the path of the collection to create or open. This option is required to execute the mkvdk utility.
-nolock	Turns off file locking. Locking is on by default.
-synch	Performs work immediately. If this option is not used, indexing work is done in the background, as time permits.
-about	Shows information about the collection, such as its description and the date when it was last modified.
-datapath path	Specifies the datapath to use to find documents that are added to the specified collection. All relative document paths are relative to this setting. If you do not set this option, the mkvdk utility looks for documents next to the collection directory.
-topicset path	Creates a topic index for the collection, based on the specified topic set, and stores it in the collection directory. This facilitates quick and efficient searches over the collection data when using topics.
-mode mode	Sets the indexing mode. Values are case-insensitive. The following are the valid settings: <ul style="list-style-type: none"><li>• Generic</li><li>• FastSearch</li><li>• NewsfeedIdx</li><li>• NewsfeedOpt</li><li>• BulkLoad</li><li>• ReadOnly</li><li>• Any custom mode defined in the style.plc file.</li></ul> The default is Generic mode.
-common	Specifies the path of the Verity common directory. If you do not use this option, the Verity engine looks for the common directory in the directory containing the mkvdk executable, and then along the executable search path. The executable search path is determined by your operating system environment settings. It is the path used by the OS to find the programs you run.
-help	Displays the mkvdk utility syntax options.
-debug	Runs the mkvdk command in debugging mode.
-nooptimize	Prevents optimization by this instance of the mkvdk utility. Using this option turns off the service-level VdkServiceType_Optimize. The service types determine the type of work the Verity engine and its self-administration features will execute on a collection.
-nohousekeep	Prevents housekeeping by this instance of the mkvdk utility. Housekeeping includes deleting files that are no longer needed. Using this option turns off the service-level VdkServiceType_DBA. (Service types are described under -nooptimize.)

Option	Description
-noindex	Prevents indexing by this instance of mkvdk. Documents are not inserted or deleted. Using this option turns off the service-level VdkServiceType_Index. (Service types are described under <code>-nooptimize</code> .)
-charmap name	Specifies the name of the character set to which to map all strings for your application. Set this to a character set that your system can display properly. Using the search engine with the English locale, the character set that any version of Windows displays is 8859. This is NOT the name of the character set of documents being indexed, it is only the name of the character set that your display can handle properly. (The character set of the document is set in the style.dft file using the <code>/charmap</code> option.) Valid options are 850 and 8859. The default is no mapping.
-locale name	Specifies the name of the Verity locale to be used by the mkvdk utility. The locale name must correspond to the name of an existing locale directory, which must exist in the <code>install_dir/common/locale</code> directory. Valid options are <code>english</code> , <code>deutsch</code> , and <code>francais</code> . The default is <code>english</code> .
-datefmt format	Converts a date field value into Verity's internal data representation. You can use this option in conjunction with the mkvdk options <code>-extract</code> (for the field extraction feature) and <code>-bulk</code> (for the bulk submit feature). The named format string identifies to the date parsing routines in what order dates are written when the date string only consists of a sequence of numbers (for example, 03/03/96). Valid options are described in <a href="#">"Date format options" on page 95</a> . The default is MDY.
-servlev level	Specifies service level. The specifier, <code>level</code> , is a string consisting of keywords separated by hyphens, such as <code>search-index-optimize</code> . Valid keywords are described in <a href="#">"Service-level keywords" on page 95</a> .

## Examples: processing documents

The following examples show the commands for processing documents.

### Using the default options

By default, the mkvdk command submits and indexes documents specified in the command, and services the specified collection. The following command executes the default options:

```
mkvdk -collection path filespec
```

### Servicing only

The following command performs servicing only. Use this command to only index submitted documents and service the collection:

```
mkvdk -collection path
```

### Deleting documents from a collection

The following command deletes documents from a collection:

```
mkvdk -delete -collection path filespec
```

## Bulk inserting or deleting

The following command specifies bulk insertion of a list of documents:

```
mkvdk -collection coll -bulk -insert filespec
```

Where `filespec` is the list of files to insert. Since `insert` is the default, the following command is equivalent to the preceding command:

```
mkvdk -collection coll -bulk filespec
```

The following command specifies bulk deletion of a list of documents:

```
mkvdk -collection coll -bulk -delete filespec
```

Where `filespec` is the list of files to delete. It can be the same file used to insert documents; the only difference is that `-delete` is specified instead of `-insert` (or no specification).

## Date format options

The Verity engine supports many import date formats, including many textual date formats, and the numeric date formats listed in the following table:

Format variable	Description
MDY	Dates written as month-day-year (US format, the default)
DMY	Dates written as day-month-year (European format)
YMD	Dates written as year-month-day (ISO international format)
YDM	Dates written as year-day-month (Swedish format)
USA	Dates written in US format (the same as MDY)
EUR	Dates written in European format (the same as DMY)

## Service-level keywords

The following table describes the valid keywords for the `-servicelevel` keyword:

Keyword	Description
search	Enables search and retrieval
insert	Enables adding and updating documents
optimize	Enables opportunistic collection optimization
assist	Enables building of word list
housekeep	Enables housekeeping of unneeded files
delete	Enables document deletion
backup	Enables backup
purge	Enables background purging
repair	Enables collection repair
dataprep	Same as search-index-optimize-assist-housekeep
index	Same as insert-delete

## Message options

The `mkvdk` utility provides a variety of messaging options, as described in the following table:

Option	Description
<code>-quiet</code>	Displays only fatal and error messages to the console. It overrides the <code>-outlevel</code> setting. For a list of message types, see the table in <a href="#">“The <code>mkvdk</code> utility syntax” on page 89</a> .
<code>-outlevel (num)</code>	Indicates which message types to display to the console. Valid values are determined by adding together the numbers that correspond to the desired message types. The default value is 15. For more information, see the table in <a href="#">“The <code>mkvdk</code> utility syntax” on page 89</a> .
<code>-logfile filename</code>	Saves messages in the specified file.
<code>-loglevel (num)</code>	Indicates which message types to route to the optional log file. Valid values are determined by adding numbers together that correspond to the desired message types. The default value is 15. For more information, see the table in <a href="#">“The <code>mkvdk</code> utility syntax” on page 89</a> .

## Document processing options

The `mkvdk` utility provides a variety of document processing options, as the following table describes:

Option	Description
<code>-extract</code>	Extracts field values from documents, using the field extraction rules specified in the <code>style.tde</code> file.
<code>-insert</code>	Adds documents to the collection. This is the default option for the <code>mkvdk</code> command.
<code>-update</code>	Adds documents to the collection by replacing all previous information about the specified documents.
<code>-delete</code>	Marks the specified documents as deleted, and makes them unavailable for searches. To actually remove deleted documents from the collection’s internal documents table and word indexes, use the <code>squeeze</code> keyword (see <a href="#">“About squeezing deleted documents” on page 99</a> ).
<code>-nosave</code>	Specifies that a work list, which is generated by the <code>mkvdk</code> utility automatically when you use the <code>-extract</code> option, will not be saved in the collection directory in a file called <code>worklist</code> (in the Verity bulk submit file format). By default, the <code>mkvdk</code> utility saves the worklist in the <code>worklist</code> file.
<code>-nosubmit</code>	Specifies that a work list, which is generated by the <code>mkvdk</code> utility automatically when you use the <code>-extract</code> option, will not be submitted to the indexing engine and will be saved in the collection directory in a file called <code>worklist</code> (in the Verity bulk submit file format). This option allows the <code>mkvdk</code> utility to process field extraction separately from other indexing tasks.



## Bulk submit options

The `mkvdk` utility provides a variety of bulk submit options, as described in the following table:

Option	Description
<code>-bulk</code>	Interprets <code>filespec</code> as a bulk submit file. You can use this option with the <code>-insert</code> , <code>-update</code> , and <code>-delete</code> options.
<code>-offset num</code>	Specifies the offset into a bulk submit file or files. If you specify multiple bulk submit files and use the <code>-offset</code> option, the offset is applied to all of the bulk submit files.
<code>-numdocs num</code>	Specifies the number of documents to insert or delete from the bulk insert file or files. If you specify multiple bulk insert or delete files and use the <code>-numdocs</code> option, the <code>-numdocs</code> setting is applied to all of the bulk insert or delete files.
<code>-autodel</code>	Deletes the bulk submit file or files when the bulk submission work is finished.

## Using bulk insert and delete options

The bulk submit feature supports the insertion of documents and related field values into collections.

**To use the bulk submit feature to populate fields:**

- 1 Define the fields in the `style.sfl` and `style.ufl` file, as appropriate.
- 2 Create a bulk submit file that specifies the documents to insert and the field values for each document.
- 3 Run the `mkvdk` utility using the `-bulk` option and specifying the bulk submit file or files.

## Collection maintenance options

The `mkvdk` utility provides a variety of collection maintenance options, as described in the following table:

Option	Description
<code>-backup dir</code>	Backs up the collection into the specified directory. The backup does not include the <code>tde</code> subdirectory. The <code>tde</code> subdirectory is created by and for Topic Document Entry if Topic Document Entry is used to create or maintain the collection.
<code>-repair</code>	Repairs the collection, performed by an API call.
<code>-purge</code>	Waits the amount of time specified by the <code>-purgewait</code> option and then deletes all documents in the collection, but not the collection itself. It leaves the collection directory structure intact. To specify a different wait period, use the <code>-purgewait</code> option instead of the <code>-purge</code> option. If you do not use the <code>-purgewait</code> option, the default is 600 seconds.
<code>-purgeback</code>	Used with the <code>-purge</code> option, performs a purge in the background.
<code>-purgewait sec</code>	Specifies to the <code>-purge</code> option how many seconds to wait. If you do not specify <code>sec</code> , the default is 600.
<code>-noservice</code>	Prevents collection servicing, which includes indexing, by this instance of the <code>mkvdk</code> command, performed by an API call.

---

Option	Description
-persist	Serves the collection repeatedly, at default intervals of 30 seconds. Use the -sleeptime option to set a different interval.
-sleeptime sec	Specifies the interval between service calls when the mkvdk utility is run with the -persist option.
-optimize spec	Performs various optimizations on the collection, depending on the value of spec. The specifier, spec, is a string consisting of keywords separated by hyphens, such as maxmerge-squeeze-readonly. For valid keywords, see <a href="#">“Optimization keywords” on page 99</a> .
-noexit	Windows only. Causes the I/O window to remain after the program is finished. By default, the window closes and the program exits, so that scripts calling the mkvdk utility do not hang.

---

## Examples: maintaining collections

The following examples show the commands for maintaining a collection.

### Repairing a collection

The following command automatically repairs a collection, or enables it after manual repairs:

```
mkvdk -repair -collection path
```

### Backing up a collection

The following command backs up a collection to the specified directory:

```
mkvdk -backup path_1 -collection path_2
```

### Deleting a collection

To delete a collection, use the appropriate command for your operating system. For example, to remove the collection directory structure and control files on a UNIX system, use the following command:

```
rm -r -collection_path
```

### Purging a collection

The following command deletes all documents from a collection, but does not delete the collection itself:

```
mkvdk -purge -collection path
```

### Purging a collection in the background

The following command purges the specified collection in the background:

```
mkvdk -purge -purgeback -collection path
```

### Specifying persistent service

The following command runs the mkvdk command as a persistent process, so that servicing is performed repeatedly after num idle seconds:

```
mkvdk -persist -sleeptime num -collection path
```

## Deleting a collection

The `-purge` option deletes all documents in a collection, but does not delete the collection itself. To delete a collection, use operating system commands, such as the `rm` command on UNIX, to remove the collection directory structure and control files.

## Optimization keywords

The following table describes the optimization keywords for the `-optimize` option:

Keyword	Description
<code>maxclean</code>	Performs the most comprehensive housekeeping possible, and removes out-of-date collection files. Macromedia recommends this optimization only when you are preparing an isolated collection for publication. When using this type, if the collection is being searched, files sometimes get deleted too early, which can affect search results.
<code>maxmerge</code>	Performs maximal merging on the partitions to create partitions that are as large as possible. This creates partitions that can have up to 64000 documents in them.
<code>readonly</code>	Marks the collection as read-only and unchanged after the function call is done. This is appropriate for CD-ROM collections.
<code>spanword</code>	Creates a spanning word list across all the collection's partitions. A collection consists of numerous smaller units, called partitions, each of which includes a word list. Optionally, a spanning word list can be built with an ngram index.
<code>ngramindex</code>	Builds an ngram index for the collection. An ngram index is designed to improve the search performance for queries with the <code>&lt;TYPO&gt;</code> and <code>&lt;WILDCARD&gt;</code> operators. An ngram index cannot be built without a spanning word list. You can build a spanning word list and ngram index in the same command, for example: <code>mkvdk -collection collname -optimize spanword -ngramindex</code>
<code>squeeze</code>	Squeezes deleted documents from the collection. Squeezing deleted documents recovers space in a collection, and improves search performance. (For more information about squeeze, see <a href="#">"About squeezing deleted documents" on page 99</a> .) Using this option invalidates the search results.
<code>vdbopt</code>	Configures the collection's Verity databases (VDBs). Each collection consists of smaller units called VDBs. This keyword has the effect of linearizing the data in a VDB, and making the collection metadata contained in the VDB more streamlined. It also lets the VDB grow to a much larger size.
<code>tuneup</code>	Performs the same as combining the <code>maxmerge</code> , <code>vdbopt</code> , and <code>spanword</code> keywords.
<code>publish</code>	Performs the same as all of the optimization types combined. Use this keyword to optimize the collection for the best possible retrieval performance, such as for publication to a network on a server or on a CD-ROM.

## About squeezing deleted documents

When a document is deleted from a collection, its space is not recovered. It is merely marked as deleted and not available for subsequent searches. Squeezing actually removes deleted documents from the collection's internal documents table and word indexes, thus creating a smaller collection and reducing the collection's disk space. A smaller collection has a more efficient structure that makes searching slightly faster and uses slightly less memory.

You can safely squeeze deleted documents for a collection at anytime, because the mkvdk utility ensures that the collection is available for searching and servicing through its self-administration features. The application does not need to temporarily disable a collection to squeeze deleted documents, because when a squeeze request is made, the mkvdk utility assigns a new revision code to the collection. After a squeeze has occurred, the next time the application accesses the collection, the Verity engine notifies the application that dramatic changes have been made, and points the application to the new collection data.

Squeezing deleted documents out of a collection is a significant update to the collection. If users are reviewing search results at the time when squeezing occurs, the search results might be invalidated after the squeeze operation.

## About optimized Verity databases

The Verity database (VDB) is the fundamental storage mechanism responsible for supporting dynamic access to documents in collections. A VDB consists of simple tables with rows and columns that relate to each other by row position. VDB tables are not relational, and their architecture supports quick and efficient searching over textual data. A VDB consists of segments that are packed into a single file. One of the advantages of having one packed VDB file is optimized search performance. The fewer files that need to be opened during search processing, the faster the search performance.

The VDB optimization option optimizes the packing of a collection's VDBs. When VDBs are built during normal indexing operations, the segments are not stored sequentially in the one-file VDB file system. As a result of VDB optimization, performance can be improved by reserializing the packed segments in the VDBs so that all segments are contiguous, and VDBs can grow in size. Optimized VDBs can grow up to 2 gigabytes in size, as opposed to the maximum 64 megabytes for an unoptimized one.

Using this option might degrade your indexing performance when certain indexing modes are set for the collection.

## Performance tuning options

The mkvdk utility provides performance tuning options, as the following table describes:

Option	Description
-maxfiles num	Sets the maximum number of files that the mkvdk utility can have open at once. The default is 50.
-diskcache num	Sets the size of the mkvdk disk cache in kilobytes. The default is 128.

# CHAPTER 9

## Indexing Collections with Verity Spider

This chapter contains basic Verity Spider information and explains how to index documents on your website.

### Contents

About Verity Spider . . . . .	101
About Verity Spider syntax. . . . .	103
Core options . . . . .	106
Processing options . . . . .	107
Networking options . . . . .	112
Path and URL options . . . . .	115
Content options. . . . .	119
Locale options . . . . .	126
Logging options. . . . .	127
Maintenance options . . . . .	128
Setting MIME types. . . . .	129

### About Verity Spider

Verity Spider enables you to index web-based and file system documents throughout your enterprise. Verity Spider works in conjunction with the Verity KeyView document filtering technology, so that you can index more than two hundred of the most popular application document formats, including Microsoft Office2000, WordPerfect, ASCII text, HTML, SGML, XML and PDF (Adobe Acrobat) documents.

**Note:** The Verity Spider that is included with ColdFusion MX is licensed for websites that are defined and reside on the same machine on which ColdFusion MX is installed. Contact Verity Sales for licensing options regarding the use of Verity Spider for external websites.

## Web standard support

Verity Spider supports key web standards used by Internet and intranet sites. Standard HREF links and frames pointers are recognized, so that navigation through them is supported. Redirected pages are followed so that the real underlying document is indexed. Verity Spider adheres to the robots exclusion standard specified in robots.txt files, so that administrators can maintain friendly visits to remote websites. HTTP Basic Authentication mechanism is supported so that password-protected sites can be indexed.

Unlike other web crawlers, Verity Spider does not need to maintain complete local copies of remote documents. When documents are viewed through Verity Information Server, documents are read from their native location with optional highlights.

## Restart capability

When an indexing job fails, or for some reason the Verity Spider cannot index a significant number or type of URLs, you can now restart the indexing job to update the collection. Only those URLs that were not successfully indexed previously are processed.

## State maintenance through a persistent store

Verity Spider V3.7 stores the state of gathered and indexed URLs in a persistent store, which lets it track progress for the purposes of gracefully and efficiently restarting halted indexing jobs.

Previous versions of Verity Spider only held state information in memory, which meant that any stoppage of spidering resulted in lost work. This also meant that larger target sites required significantly more memory for spidering. The information in the persistent store can help report information, such as the number of indexed pages, visited pages, rejected pages, and broken links.

## Performance

Spidering performance is greatly improved over previous versions, because of low memory requirements, flow control, and the help of multithreading and efficient Domain Name System (DNS) lookups.

### Flow control

When indexing websites, Verity Spider distributes requests to web servers in a round-robin manner. This means that one URL is fetched from each web server in turn. With flow control, a faster website can finish before a slower one. The Verity Spider optimizes indexing on every web server.

Verity Spider V3.7 adjusts the number of connections per server depending on the download bandwidth. When the download bandwidth from a web server falls below a certain value, Verity Spider automatically scales back the number of connections to that web server. There will always be at least one connection to a web server. When the download bandwidth increases to an acceptable level, Verity Spider reallocates connections (per the value of the `-connections` option, which is 4 by default). You can turn off flow control with the `-noflowctrl` option.

## Multithreading

Since version 3.1, Verity Spider has separated the gathering and indexing jobs into multiple threads for concurrence. Verity Spider V3.7 can create concurrent connections to web servers for fetching documents, and have concurrent indexing threads for maximum utilization. This translates to an overall improvement in throughput. In previous releases, work was done in a round-robin manner, so that at any given time, only one job was running. Spider attends to the websites within an indexing job in a round-robin manner.

## Efficient DNS lookups

Verity Spider V3.7 significantly reduces DNS lookups, which means great improvements to spidering throughput. If spidering is limited by domain or host, then no DNS lookups are made on hosts that fall outside of that range. In earlier versions, DNS lookups were made on all candidate URLs.

## Proxy handling efficiency

To allow for greater flexibility when dealing with indexing jobs that involve proxy servers and firewalls, use the following options:

- **-noproxy** To reduce proxy checking for certain hosts
- **-proxyauth** To authenticate on proxy servers

**Note:** Information Server V3.7 does not support retrieving documents for viewing through secure proxy servers. Do not use the `-proxyauth` option for indexing documents that you will view through Information Server V3.7.

## About Verity Spider syntax

Before you create an indexing task for a new collection, make copies of the relevant default style files to ensure that you have a set of template style files in a known, stable state.

Running multiple simultaneous Verity Spider jobs on the Information Server host can cause performance problems for searches. This does not mean that you should never run indexing jobs when users might be searching, because your collections are available for searching even while indexing jobs are running. To optimize performance, try staggering your indexing jobs to avoid overloading your server.

## The Verity Spider command

The `vspider` executable, which starts the `vspider` application, is located in the `cf_root\lib\_nti40\bin` directory in Windows, and in the `cf_root/lib/platform/bin` directory on UNIX.

In these pathnames, `cf_root` refers to the ColdFusion MX root directory. In Windows, this is typically `C:\CFusionMX`; on UNIX, this is typically `/opt/coldfusionmx`. On UNIX, `platform` refers to the UNIX version of the server that runs ColdFusion: `_solaris`, `_hpux11`, or `_ilnx21`.

At its most basic level, a Verity Spider command consists of the following:

```
vspider -initialize -collection coll [options]
```

Where `-initialize` is `-start` or `-refresh` (when starting points have changed), and `-collection` is required to provide a target for the Verity Spider, and `[options]` can be a near-limitless combination of the options described later in this chapter.

For example:

```
c:\cfusionmx\lib\_nti40\bin\vspider -common c:\cfusionmx\lib\common  
-collection c:\new -start http://localhost -indinclude *
```

There are dependencies for other options, depending on the nature of the indexing task. The following are some examples:

- To build a new collection, you must use `-style`.
- To control how Verity Spider operates, including which documents it indexes, use some Verity Spider options.

If you do not run the Verity Spider executable from its default installation directory, you must include that directory in your path. This is because the Verity Spider executable depends on other files to run properly.

## Using a command file

For simpler reuse and archiving of your indexing commands, use the `-cmdfile` option for abstraction. By using an ASCII text file to store a task's options, you avoid the potential problem of requiring special characters in an option's parameter value. For example, the `-processbif` option requires the use of `"!*"` and therefore any task using that option must also use the `-cmdfile` option.

## Command-line option reference

The following sections describe the Verity Spider V3.7 command-line options. Option names are case-sensitive.

### **-start**

Specifies a starting point for an indexing job. You can specify multiple instances, or use multiple values in a single instance.

When you execute an indexing job from a command line, and you do not use a command file (with the `-cmdfile` option), you must URL-escape any special characters in the starting point. To URL-escape a special character, use `"%hex-ASCII-character-number"` in place of the character. For example, use `/time%26/` instead of `/time&/`. This allows the operating system to properly process the command string.



If an indexing task halts, you can rerun the task as-is. The persistent store for the specified collection is read, and only those candidate URLs that are in the queue but not yet processed are parsed. Candidate URLs correspond to URLs of the following status, as reported by vsdb:

cand, used, inse, upda, dele, fail

---

Repository type	Starting point
Web	The URL or URLs from which Verity Spider is to begin indexing. Use other options, such as the <code>-jumps</code> option, to control how far from the starting point Verity Spider goes.
File	The starting directory or directories in which Verity Spider will start indexing. All subdirectories beneath the starting point will be indexed, unless you use the <code>-pathlen</code> option or any of the inclusion or exclusion criteria.

---

**Note:** By using the `-start` option with the `-refresh` option, you provide a starting point for Verity Spider and therefore do not need to use at least one of the following options: `-host`, `-domain`, `-nofollow`, or `-unlimited`.

## **-refresh**

Used for updating a collection, specifies that Verity Spider process only those documents that qualify, as follows:

- They are new documents in the repository, and they qualify for indexing under the criteria.
- They exist in the collection and are recorded in the Verity Spider persistent store with a status of done. If Verity Spider determines that these indexed documents have been updated in the repository, then they are retrieved again to be reparsed and reindexed. The document `VdkVgwKey` values do not change.
- They are deleted in the collection. If Verity Spider determines that documents have been deleted from the repository, then they are also deleted from the persistent store and the collection. The exception to this rule is when you use the `-nooptimize` option with the `-refresh` option. In this case, any document deleted from the repository is marked for deletion in the collection. It will be removed from the collection and the persistent store when the next indexing task is run for the collection.

When you rerun an existing indexing job, Verity Spider automatically refreshes the collection. If you add or remove any of the starting points, however, you must manually specify the `-refresh` option to refresh existing documents.

**Note:** You can also use the `-start` option to provide a starting point for Verity Spider. If you do not use the `-start` option, use at least one of the following options: `-host`, `-domain`, or `-nofollow`. For further control, also see the `-refreshtime` option. If you do not use any constraint criteria, Verity Spider operates without limits and will likely index far more than you intended.

## Core options

The following sections describe the Verity Spider core options.

### **-cmdfile**

**Syntax:** `-cmdfile path_and_filename`

Specifies that Verity Spider reads command-line syntax from a file, in addition to the options passed in the command-line. This option includes the pathname to the file that contains the command-line syntax. The `-cmdfile` option circumvents command-line length limits.

The syntax for the command-file is:

```
option optional_parameters
```

For better readability, put each option and any parameters on a single line. Verity Spider can properly parse the lines.

**Note:** Macromedia strongly recommends that you take advantage of the abstraction offered by this option. This can greatly reduce user error in erroneously including or omitting options in subsequent indexing jobs.

### **-collection**

Specifies the full path to the collection to create or update.

**Note:** You receive an error if you specify a filename with an extension of CLM. Meta collections are not supported.

### **-help**

Displays Verity Spider syntax options.

### **-jobpath**

**Syntax:** `-jobpath path`

Specifies the location of the Verity Spider databases and the indexing job-related files and directories.

The following are the job-related directories and their contents:

- **log** All Verity Spider log files. For descriptions of the log files, see `-loglevel`.
- **bif** Bulk insert files.
- **temp** Web pages cached for indexing. You can also specify the temp directory using the `-temp` option.
- **admin** Files created by the Information Server Admin Tool.

These directories are created for you under the last directory specified in path.

Path values must be unique for all indexing jobs. If you do not use the `-jobpath` option, Verity Spider creates a `/spider/job` directory within the collection. For multiple-collection tasks, the first collection specified is used.

**Note:** You cannot use multiple job paths for multiple simultaneous indexing tasks for the same collection. Only one indexing task at a time can run for a given collection.

## **-style**

**Syntax:** `-style path`

Specifies the path to the style files to use when creating a new collection.

If the `-style` option is not specified, Verity Spider uses the default style files in `cf_root/lib/common/style`.

**Note:** You can safely omit the `-style` option when resubmitting an indexing job, as the style information will already be part of the collection. If you are using the `-cmdfile` option, you can leave it there.

## **Processing options**

The following sections describe the Verity Spider processing options.

### **-abspath**

**Type:** File system only

Generates absolute paths for files. Use this option when the document locations are not going to change, but the collection might be moved around.

When you index a web server's contents through the file system, use the `-prefixmap` option with the `-abspath` option to map the absolute file paths to URLs.

See also [-prefixmap](#).

### **-detectdupfile**

**Type:** File system only

Enables checksum-based detection of duplicates when indexing file systems.

By default, a document checksum is not computed on indexed files. By using the `-detectdupfile` option, a checksum is computed based on the CRC-32 algorithm. The checksum combined with the document size is used to determine if the document is a duplicate.

### **-indexers**

**Syntax:** `-indexers num_indexers`

Specifies the maximum number of indexing threads to run on a collection.

The default value is 2. Increasing the value for the `-indexers` option requires additional CPU and memory resources.

See also [-maxindmem](#).

### **-license**

**Syntax:** `-license path_and_filename`

Specifies the license file to use.

By default, the `ind.lic` file is used, from the `cf_root/lib/platform/bin` directory; where *platform* represents the platform directory.

## **-maxindmem**

**Syntax:** `-maxindmem kilobytes`

Specifies the maximum amount of memory, in kilobytes, used by each indexing thread. Specify the number of threads with the `-indexers` option.

By default, each indexing thread uses as much memory as is available from the system.

## **-maxnumdoc**

**Syntax:** `-maxnumdoc num_docs`

Specifies the maximum number of documents to download or submit for indexing. The value for `num_docs` does not necessarily correspond to the number of documents indexed. The following factors affect the actual number:

- Whether the value of `num_docs` falls within a block of documents dictated by the `-submitsize` option. If it does, the entire block of documents must be processed.
- Whether documents retrieved are actually indexed, because they are invalid or corrupt.

## **-mimemap**

**Syntax:** `-mimemap path_and_filename`

Specifies a control file (simple ASCII text) that maps file extensions to MIME-types. This lets you make custom associations and override defaults.

The following is the format for the control file:

```
#file_ext_no_dot      mime-type
abc                   application/word
```

## **-nocache**

**Type:** Web crawling only

Used with the `-noindex` or `-nosubmit` options, this option disables the caching of files during website indexing. This has the effect of decreasing the demands on your disk space.

Normally, Verity Spider downloads URLs, then writes them to a bulk insert file and downloads the documents themselves. When indexing occurs, once the `-submitsize` option has been reached, the cached files are indexed and then deleted. If you use the `-noindex` option, the bulk insert file is submitted but not processed by Verity Spider, and so the documents are not deleted until indexing occurs. This will usually be `mkvdk` or `collsvc`, or you can use Verity Spider again with the `-processbif` option.

By using the `-nocache` option in conjunction with the `-noindex` or `-nosubmit` option, you avoid storing files locally. Files are downloaded only when indexing actually occurs.

See also [-noindex](#).

## **-nodupdetect**

**Type:** Web crawling only

Disables checksum-based detection of duplicates when indexing websites. URL-based duplicate detection is still performed.

By default, a document checksum is computed based on the CRC-32 algorithm. The checksum combined with the document size is used to determine if the document is a duplicate.

See also [-followdup](#).

## **-noindex**

Specifies that Verity Spider gathers document locations without indexing them. The document locations are stored in a bulk insert file (BIF), which is then submitted to the collection. This option is typically used in conjunction with a separate indexing process, such as `mkvdk` or collection servicers (`collsvc`). The BIF will be processed by the next indexing process run for the collection, whether it is Verity Spider, `mkvdk`, or collection servicers (`collsvc`).

Do not try to start Verity Spider and another process at the same time. You must allow Verity Spider time to generate enough work for the secondary indexing process. If you are using `mkvdk`, you can run it in persistent mode to ensure it will act upon work generated by Verity Spider.

**Note:** When you execute an indexing job for a collection and you use the `-noindex` option, the persistent store for the collection is not updated.

See also [-nocache](#) and [-nosubmit](#).

For more information on the `mkvdk` utility, see [Chapter 8, “Managing Collections with the `mkvdk` Utility,” on page 89](#).

## **-nosubmit**

Specifies that Verity Spider gathers document locations without submitting them. The document locations are stored in a bulk insert file (BIF), which is not submitted to the collection. This option is typically used in conjunction with a separate indexing process, such as `mkvdk` or collection servicers (`collsvc`). You can also use Verity Spider again with the `-processbif` option. With an indexing process other than Verity Spider, you must specify the name and path for the BIF, because the collection has no record of it.

## **-persist**

**Syntax:** `-persist num_seconds`

Enables the Verity Spider to run in persistent mode, checking for updates every `num_seconds` seconds until it is stopped.

While Verity Spider is running in persistent mode, there is no optimization. After Verity Spider is taken out of persistent mode, you need to perform optimization on the collection. For more information about using the `mkvdk` utility, see [Chapter 8, “Managing Collections with the `mkvdk` Utility,” on page 89](#).

**Note:** Do not run more than one Verity Spider process in persistent mode. As the Verity Spider is a resource-intensive process, only run it in persistent mode with an interval of less than one day. For time intervals greater than twelve hours, use some form of scheduling. Some examples are cron jobs for UNIX, and the AT command for Windows server.

## **-preferred**

**Type:** Web crawling only

**Syntax:** `-preferred exp_1 [exp_n] ...`

Specifies a list of hosts or domains that are preferred when retrieving documents for viewing. You can use wildcard expressions, where the asterisk (\*) is for text strings and the question mark (?) is for single characters. To use regular expressions, also specify the `-regexp` option. Use this option when you leave duplicate detection enabled and do not specify the `-nodupdetect` option.

When indexing, you might encounter a nonpreferred host first. In that case, documents are parsed and followed and stored as candidates. When duplicates are encountered on another server, which is preferred, the duplicate documents from the nonpreferred server are skipped. When documents are requested for viewing, they will be retrieved from the preferred server.

In Windows, include double-quotation marks around the argument to protect the special characters, such as the asterisk (\*). On UNIX, use single-quotation marks. This is only required when you run the indexing job from a command line. Quotation marks are not necessary within a command file (the `-cmdfile` option).

See also [-regexp](#).

## **-prefixmap**

**Type:** File system only

**Syntax:** `-prefixmap path_and_filename`

Specifies a control file (simple ASCII text) that maps file system paths to web aliases.

In conjunction with the `-abspath` option, this option is typically used to create a URL field that is the web equivalent of a file system path. File system indexing is faster than web crawling over the network. If you use the `-prefixmap` option to replace the file system path with the web URL, relative hyperlinks in the HTML pages are kept intact when viewed through Information Server.

The following is the format for the control file:

```
src_field src_prefix dest_field dest_prefix
```

If you use backslashes, you must double them so that they are properly escaped; for example:

```
C:\\test\\docs\\path
```

For example, to map the filepath `/usr/pub/docs` to `http://web/~verity`, use the following:

```
vdkgwkey /usr/pub URL http://web/~verity
```

See also [-abspath](#).

## **-processbif**

**Syntax:** `-processbif 'command_string !*'`

Specifies a command string in which you can call a program or script that operates on BIFs generated by Verity Spider.

Due to the use of special characters, which represent the bulk insert file (BIF), you must run Verity Spider with a command file using the `-cmdfile` option.

For example, if you want to use a script called `fix_bif` to add customized information to BIF files, use the following command:

```
vspider -cmdfile filename
```

Where `filename` is the text-only command file that contains the following (along with any other necessary options):

```
-processbif 'fix_bif !*'
```

Your command file will include other options as well.

### **-regexp**

Specifies the use of regular expressions rather than the default wildcard expressions for the following options: `-exclude`, `-indexclude`, `-include`, `-indinclude`, `-skip`, `-indskip`, `-preferred`, and `-nofollow`.

Wildcard expressions allow the use of the asterisk (\*) for text strings, and the question mark (?) for single characters, as the following table shows:

Wildcard expression	Text string
<code>a*t</code>	although, attitude, audit
<code>a?t</code>	ant, art
<code>file?.htm</code>	files.htm, file1.htm, filer.htm
<code>name??.*</code>	names.txt, named.blank, names.ext

Regular expressions allow for more powerful and flexible matching of alphanumeric strings; for example, to match "ab11" or "ab34" but not "abcd" or "ab11cd," you could use the following regular expression:

```
^ab[0-9][0-9]$
```

The full extent to which regular expressions can be employed is beyond the scope of this description. For more information on regular expressions, refer to a book devoted to the subject.

### **-submitsize**

**Syntax:** `-submitsize num_documents`

Specifies the number of documents submitted for indexing at one time. The default value is 128. The upper limit is 64,000.

**Note:** Although larger values mean more efficient processing by the indexer, smaller values allow more parallelism on multi-CPU systems. In the event of a halt during indexing, a smaller value means fewer documents will be lost.

If a halt occurs during indexing, the chunk of documents specified by the `-submitsize` option is lost because there is no transactional rollback for indexing and the documents are no longer in the queue for indexing. When you rerun the indexing task, Verity Spider can only continue with URLs and documents that are enqueued.

## **-temp**

**Syntax:** `-temp path`

Specifies the directory for temporary files (disk cache). By default, the temp directory is under the job directory (optionally specified with the `-jobpath` option).

If you do not specify a value for this option, Verity Spider creates a `/spider/temp` directory within the collection. For multiple-collection tasks, the first collection specified is used.

**Note:** Make sure the location you specify contains enough disk space to handle the documents that are downloaded and held before indexing. The documents are deleted from the hard disk after they are indexed.

See also `-jobpath`, for specifying the location of all indexing job directories and files, one of which is the temp directory.

## **Networking options**

The following sections describe the Verity Spider networking options.

### **-agentname**

**Type:** Web crawling only

**Syntax:** `-agentname string`

Specifies the value for the agent name field that is part of the HTTP request. Since web servers can be configured to return different versions of the same page depending on the requesting agent, you can use the `-agentname` option to impersonate a browser client.

Use double-quotation marks if the name contains a space. Use the `-cmdfile` option if the agent name you want to use contains forbidden characters, such as slashes or backslashes.

### **-connections**

**Syntax:** `-connections num_connections`

Specifies the maximum number of simultaneous socket connections to make to websites for indexing. Each connection implies a separate thread.

The default value is 6.

**Note:** The Verity Spider dynamic flow control makes the most use of all available connections when indexing websites. If you are indexing multiple sites, you might want to increase this number. Increasing the number of connections does not always help, because of such dependencies as your network connection and the capabilities of the remote hosts.

### **-delay**

**Type:** Web crawling only

**Syntax:** `-delay num_milliseconds`

Specifies the minimum time between HTTP requests, in milliseconds. The default value is 0 milliseconds for no delay.



## **-header**

**Type:** Web crawling only

**Syntax:** `-header string`

Specifies an HTTP header to add to the spidering request; for example:

```
-header "Referer: http://www.verity.com/"
```

Verity Spider sends some predefined headers, such as Accept and User-Agent, by default. Special headers are sometimes necessary to correctly index a site.

For example, earlier versions of Verity Spider did not support the Host header, which is needed for Virtual Host indexing. Also, a Proxy-authentication header was needed to pass a username and password to a proxy server.

In Verity Spider V3.7, the Host header is supported by default, and the `-proxyauth` option is available for proxy server authentication. Therefore, the `-header` option is maintained only for backwards compatibility and possible future enhancements.

**Note:** Misuse of this option causes spider failure. If this happens, rerun the indexing task with modified `-header` values.

## **-hostcache**

**Syntax:** `-hostcache num_hostnames`

Specifies the number of host names to cache to avoid DNS lookups. Without this option, the host cache continues to grow.

The default value is 256.

## **-noflowctrl**

**Type:** Web crawling only

Disables round-robin indexing of websites with network flow control.

By default, Verity Spider uses round-robin indexing of websites to avoid overwhelming a web server and to improve indexing performance. Verity Spider connects to each web server in a round-robin manner, using up to the value for the `-connections` option. This means that one URL is fetched from each web server, in turn.

**Note:** Using the `-noflowctrl` option can result in a significant drop in performance.

## **-noproxy**

**Type:** Web crawling only

**Syntax:** `-noproxy name_1 [name_n] ...`

Used in conjunction with the `-proxy` option, the `-noproxy` option specifies that Verity Spider directly access the hosts whose names match those specified. By default, when you specify the `-proxy` option, Verity Spider first tries to access every host with the proxy information. To improve performance, use the `-noproxy` option for the hosts you know can be accessed without a proxy host. For the name variable, you can use the asterisk (\*) wildcard for text strings; for example:

```
'*.verity.com'
```

You cannot use the question mark (?) wildcard, and the `-regexp` option does not let you use regular expressions.

In Windows, include double-quotation marks around the argument to protect the asterisk special character (\*). On UNIX, use single-quotation marks. This is only required when you run the indexing job from a command line. Quotation marks are not necessary within a command file (the `-cmdfile` option).

**Note:** You must have valid Verity Spider licensing capability to use this option.

### **-proxy**

**Type:** Web crawling only

**Syntax:** `-proxy proxyhost:port`

Specifies host and port for proxy server.

**Note:** You must have valid Verity Spider licensing capability to use this option.

See also `-proxyauth` for proxy servers that require authentication, and `-noproxy` for hosts that you know are accessible without having to go through a proxy server.

### **-proxyauth**

**Type:** Web crawling only

**Syntax:** `-proxyauth login:password`

Specifies login information for proxy server connections that require authorization to get outside the firewall. Use this option in conjunction with the `-proxy` option.

**Note:** You must have valid Verity Spider licensing capability to use this option. Information Server V3.7 does not support retrieving documents for viewing through secure proxy servers. Do not use the `-proxyauth` option for indexing documents that are viewed through Information Server V3.7

### **-retry**

**Type:** Web crawling only

**Syntax:** `-retry num_retries`

Specifies the number of times that Verity Spider should attempt to access a URL. Use the `-retry` option when it is likely that an unstable network connection will give false rejections.

The default value is 4.

### **-timeout**

**Type:** Web crawling only

**Syntax:** `-timeout num_seconds`

Specifies the time period, in seconds, that Verity Spider should wait before timing out on a network connection and on accessing data. The data access value is automatically twice the value you specify for the network connection timeout.

The default value for the network connection time-out is 30 seconds, and therefore the default value for the data access time-out is 60 seconds.

## Path and URL options

The following sections describe the Verity Spider path and URL options.

### **-auth**

**Syntax:** `-auth path_and_filename`

Specifies an authorization file to support authentication for secure paths.

Use the `-auth` option to specify the authorization file. The file contains one record per line. Each line consists of server, realm, username, and password, separated by whitespace.

The following is a sample authorization file:

```
# This is the Authorization file for HTTP's Basic Authentication
#server realm username password
doleary MACR my_username my_password
```

### **-cgiok**

**Type:** Web crawling only

Lets you index URLs containing the question mark (?). This typically means that the URL leads to a CGI or other processing program.

The return document produced by the web server is indexed and parsed for document links, which are followed and in turn indexed and parsed. However, if the web server does not return a page, perhaps because the URL is missing parameters that are required for processing in order to produce a page, nothing happens. There is no page to index and parse.

### **Example**

The following is a URL without parameters:

```
http://server.com/cgi-bin/program?
```

If you include parameters in the URL to be indexed, as specified with the `-start` option, those parameters are processed and any resulting pages are indexed and parsed.

By default, a URL with a question mark (?) is skipped.

### **-domain**

**Type:** Web crawling only

**Syntax:** `-domain name_1 [name_n] ...`

Limits indexing to the specified domain(s). You must use only complete text strings for domains. You cannot use wildcard expressions. URLs not in the specified domain(s) are not downloaded or parsed.

You can list multiple domains by separating each one with a single space.

**Note:** You must have the appropriate Verity Spider licensing capability to use this option. The Verity Spider that is included with ColdFusion MX is licensed for websites that are defined and reside on the same machine on which ColdFusion MX is installed. Contact Verity Sales for licensing options regarding the use of Verity Spider for external websites.

## **-followdup**

Specifies that Verity Spider follows links within duplicate documents, although only the first instance of any duplicate documents is indexed.

You might find this option useful if you use the same home page on multiple sites. By default, only the first instance of the document is indexed, while subsequent instances are skipped. If you have different secondary documents on the different sites, using the `-followdup` option lets you get to them for indexing, while still indexing the common home page only once.

## **-followsymlink**

**Type:** File system only

Specifies that Verity Spider follows symbolic links when indexing UNIX file systems.

## **-host**

**Type:** Web crawling only

**Syntax:** `-host name_1 [name_n] ...`

Limits indexing to the specified host or hosts. You must use only complete text strings for hosts. You cannot use wildcard expressions.

You can list multiple hosts by separating each one with a single space. URLs not on the specified host(s) are not downloaded or parsed.

## **-https**

**Type:** Web crawling only

Lets you index SSL-enabled websites.

**Note:** You must have the Verity SSL Option Pack installed to use the `-https` option. The Verity SSL Option Pack is a Verity Spider add-on available separately from a Verity salesperson.

## **-jumps**

**Type:** Web crawling only

**Syntax:** `-jumps num_jumps`

Specifies the maximum number of levels an indexing job can go from the starting URL. Specify a number between 0 and 254.

The default value is unlimited. If you see extremely large numbers of documents in a collection where you do not expect them, consider experimenting with this option, in conjunction with the Content options, to pare down your collection.

## **-nodocrobo**

Specifies to ignore ROBOT META tag directives.

In HTML 3.0 and earlier, robot directives could only be given as the file `robots.txt` under the root directory of a website. In HTML 4.0, every document can have robot directives embedded in the META field. Use this option to ignore them. Use this option with discretion.

## **-nofollow**

**Type:** Web crawling only

**Syntax:** `-nofollow "exp"`

Specifies that Verity Spider cannot follow any URLs that match the `exp` expression. If you do not specify an `exp` value for the `-nofollow` option, Verity Spider assumes a value of `"*"`, where no documents are followed.

You can use wildcard expressions, where the asterisk (\*) is for text strings and the question mark (?) is for single characters. Always encapsulate the `exp` values in double-quotation marks to ensure that they are properly interpreted.

If you use backslashes, you must double them so that they are properly escaped; for example:

```
C:\\test\\docs\\path
```

To use regular expressions, also specify the `-regex` option.

Earlier versions of Verity Spider did not allow the use of an expression. This meant that for each starting point URL, only the first document would be indexed. With the addition of the expression functionality, you can now selectively skip URLs, even within documents.

See also `-regex`

## **-norobo**

**Type:** Web crawling only

Specifies to ignore any `robots.txt` files encountered. The `robots.txt` file is used on many websites to specify what parts of the site indexers should avoid. The default is to honor any `robots.txt` files.

If you are re-indexing a site and the `robots.txt` file has changed, Verity Spider deletes documents that have been newly disallowed by the `robots.txt` file.

Use this option with discretion and extreme care, especially in conjunction with the `-cgiok` option.

See also `-nodocrobo` and <http://info.webcrawler.com/mak/projects/robots/norobots.html>.

## **-pathlen**

**Syntax:** `-pathlen num_pathsegments`

Limits indexing to the specified number of path segments in the URL or file system path. The path length is determined as follows:

- The host name and drive letter are not included; for example, neither `www.spider.com:80/` nor `C:\` would be included in determining the path length.
- All elements following the host name are included.
- The actual filename, if present, is included; for example, `/world.html` would be included in determining the path length.
- Any directory paths between the host and the actual filename are included.

## Example

For the following URL, the path length would be four:

```
http://www.spider:80/comics/fun/funny/world.html
    <-1->          <2>  <-3-> <---4--->
```

For the following file system path, the path length would be three:

```
C:\files\docs\datasheets
    <-1-><-2-><---3--->
```

The default value is 100 path segments.

## -refreshtime

**Syntax:** `-refreshtime timeunits`

Specifies not to refresh any documents that have been indexed since the `timeunits` value began.

The following is the syntax for `timeunits`:

```
n day n hour n min n sec
```

Where `n` is a positive integer. You must include spaces, and since the first three letters of each time unit are parsed, you can use the singular or plural form of the word.

If you specify the following:

```
-refreshtime 1 day 6 hours
```

Only those documents that were last indexed at least 30 hours and 1 second ago, are refreshed.

**Note:** This option is valid only with the `-refresh` option. When you use `vsdb -recreate`, the last indexed date is cleared.

## -reparse

**Type:** Web crawling only

Forces parsing of all HTML documents already in the collection. You must specify a starting point with the `-start` option when you use the `-reparse` option.

You can use the `-reparse` option when you want to include paths and documents that were previously skipped due to exclusion or inclusion criteria. Remember to change the criteria, or there will be little for Verity Spider to do. This can be easy to overlook when you are using the `-cmdfile` option.

## -unlimited

Specifies that no limits are placed on Verity Spider if neither the `-host` nor the `-domain` option is specified. The default is to limit based on the host of the first starting point listed.

## -virtualhost

**Syntax:** `-virtualhost name_1 [name_n] ...`

Specifies that DNS lookups are avoided for the hosts listed. You must use only complete text strings for hosts. You cannot use wildcard expressions. This lets you index by alias, such as when multiple web servers are running on the same host. You can use regular expressions.

Normally, when Verity Spider resolves host names, it uses DNS lookups to convert the names to canonical names, of which there can be only one per machine. This allows for the detection of duplicate documents, to prevent results from being diluted. In the case of multiple aliased hosts, however, duplication is not a barrier as documents can be referred to by more than one alias and yet remain distinct because of the different alias names.

### Example

You can have both `marketing.verity.com` and `sales.verity.com` running on the same host. Each alias has a different document root, although document names such as `index.htm` can occur for both. With the `-virtualhost` option, both server aliases can be indexed as distinct sites. Without the `-virtualhost` option, they would both be resolved to the same host name, and only the first document encountered from any duplicate pair would be indexed.

**Note:** If you are using Netscape Enterprise Server, and you have specified only the host name as a virtual host, Verity Spider will not be able to index the virtual host site. This is because Verity Spider always adds the domain name to the document key.

## Content options

The following sections describe the Verity Spider content options.

### **-casesen**

Makes processing case-sensitive by specifying that the spider separately process keys that differ only in case. Use only for indexing UNIX servers.

### **-exclude**

**Syntax:** `-exclude exp_1 [exp_n] ...`

Specifies that files, paths, and URLs matching the specified expression(s) will not be followed. If you use backslashes, you must double them so that they are properly escaped; for example:

```
C:\\test\\docs\\path
```

You can use wildcard expressions, where the asterisk (\*) is for text strings and the question mark (?) is for single characters; for example:

```
'/my_doc*/year199?'
```

In Windows, include double-quotation marks around the argument to protect special characters, such as the asterisk (\*). On UNIX, use single-quotation marks. This is only required when you run the indexing job from a command line. Quotation marks are not necessary within a command file (the `-cmdfile` option).

To use regular expressions, also specify the `-regexp` option.

To specify a file, path, or URL that you want followed but not indexed, use the `-indexexclude` option. For document types, use the `-mimeexclude` option instead; for example, specify `-mimeexclude application/pdf` rather than `-exclude *.pdf`.

**Note:** When specifying a URL, you must use full, absolute paths using the same format that appears in the HTML hyperlink. If the link is relative, you must change it to absolute to use it with the `-exclude` option.

See also [-regexp](#).

## -include

Specifies that only those files, paths, and URLs that match the specified expression or expressions will be followed. If you use backslashes, you must double them so that they are properly escaped; for example:

```
C:\\test\\docs\\path
```

You can use wildcard expressions, where the asterisk (\*) is for text strings and the question mark (?) is for single characters; for example:

```
'/my_doc*/year199?'
```

In Windows, include double-quotation marks around the argument to protect the special characters, such as the asterisk (\*). On UNIX, use single-quotation marks. This is only required when you run the indexing job from a command line. Quotation marks are not necessary within a command file (the `-cmdfile` option).

To use regular expressions, also specify the `-regex` option.

If your starting points do not contain the specified `-include` expressions, nothing will be indexed. The `-include` option prevents Verity Spider from even following anything that does not match the specified expressions. You might want to use the `-indinclude` option instead. Where the `-include` option prevents Verity Spider from even following anything that does not match the specified expressions, the `-indinclude` option allows Verity Spider to follow what matches the specified expressions, while not indexing.

For document types, use the `-mimeinclude` option instead; for example, specify `-mimeinclude text/html` rather than `-include *.htm`.

**Note:** When specifying a URL, you must use full, absolute paths using the same format that appears in the HTML hyperlink. If the link is relative, you must change it to absolute to use it with the `-include` option.

See also [-regex](#).

## -indexclude

**Syntax:** `-indexclude exp_1 [exp_n] ...`

Specifies that the files and paths in URLs that match the expressions are not indexed. They are, however, still followed. If you use backslashes, you must double them so that they are properly escaped; for example:

```
C:\\test\\docs\\path
```

You can use wildcard expressions, where the asterisk (\*) is for text strings and the question mark (?) is for single characters; for example:

```
'/my_doc*/year199?'
```

In Windows, include double-quotation marks around the argument to protect the special characters, such as the asterisk (\*). On UNIX, use single-quotation marks. This is only required when you run the indexing job from a command line. Quotation marks are not necessary within a command file (the `-cmdfile` option).

To use regular expressions, also specify the `-regex` option.

You would use this option to gather some documents, such as HTML tables of contents, to gain access to other documents for indexing.



Where the `-exclude` option prevents Verity Spider from even following anything that matches the specified expressions, the `-indexexclude` option allows Verity Spider to follow anything while only skipping that which matches the specified expressions.

For document types, use the `-indmimeexclude` option instead.

**Note:** When specifying a URL, you must use full, absolute paths using the same format as appears in the HTML hyperlink. If the link is relative, you must change it to absolute to use it with `-indexexclude`.

See also [-regex](#).

## **-indinclude**

**Syntax:** `-indinclude exp_1 [exp_n] ...`

Specifies that only those files and paths in URLs that match the expressions be followed and indexed. If you use backslashes, you must double them so that they are properly escaped; for example:

```
C:\\test\\docs\\path
```

You can use wildcard expressions, where the asterisk (\*) is for text strings and the question mark (?) is for single characters; for example:

```
'/my_doc*/year199?'
```

In Windows, include double-quotation marks around the argument to protect the special characters, such as the asterisk (\*). On UNIX, use single-quotation marks. This is only required when you run the indexing job from a command line. Quotation marks are not necessary within a command file (the `-cmdfile` option).

To use regular expressions, also specify the `-regex` option.

Where the `-include` option prevents Verity Spider from even following anything that does not match the specified expressions, the `-indinclude` option allows Verity Spider to follow anything while only indexing that which matches the specified expressions.

## **Example**

If you want to index all documents that include "search" in the URL at `http://web.verity.com`, you cannot use the following:

```
vspider -collection collname -start http://web.verity.com  
-include '*search*'
```

This is because the starting point does not match the `-include` option criteria. Instead, use the `-indinclude` option to follow all documents (unless you have specified any of the `exclude` options) and index only those documents that match your criteria. Replace the `-include` option with the `-indinclude` option in the preceding example.

**Note:** When specifying a URL, you must use full, absolute paths using the same format that appears in the HTML hyperlink. If the link is relative, you must change it to absolute to use it with the `-indinclude` option.

See also [-regex](#).

## **-indmimeexclude**

**Syntax:** `-indmimeexclude mime_1 [mime_n] ...`

Specifies that only those MIME types that match the expressions be followed but not indexed.

In Windows, include double-quotation marks around the argument to protect the special characters, such as the asterisk (\*). On UNIX, use single-quotation marks. This is only required when you run the indexing job from a command line. Quotation marks are not necessary within a command file (the `-cmdfile` option).

Use this option to gather some documents, such as HTML tables of contents, to gain access to other documents for indexing. The `-mimeexclude` option, on the other hand, prevents specified documents from being followed at all. For the mime variable, you can include the asterisk (\*) wildcard for text strings; for example:

```
'text/*'
```

You cannot use the question mark (?) wildcard, and the `-regex` option does not let you use regular expressions.

## **-indmimeinclude**

**Syntax:** `-indmimeinclude mime_1 [mime_n] ...`

Specifies that only those MIME types that match the expressions be followed and indexed.

The `-mimeinclude` option does not let you index desired documents if the starting URL is not followed. For the mime variable, you can include the asterisk (\*) wildcard for text strings; for example:

```
'text/*'
```

In Windows, include double-quotation marks around the argument to protect the special character (\*). On UNIX, use single-quotation marks. This is only required when you run the indexing job from a command line. Quotation marks are not necessary within a command file (the `-cmdfile` option).

You cannot use the question mark (?) wildcard, and the `-regex` option does not allow you to use regular expressions.

## **Example**

If you want to index all Word documents at `http://web.verity.com`, you cannot use:

```
vspider -collection collname -style style_dir -start  
http://web.verity.com -mimeinclude 'application/msword'
```

This is because the starting point does not match the `-mimeinclude` criteria. You can use the `-indmimeinclude` option to follow all documents (unless you have specified any of the exclude options) and index only those documents that match your criteria. Replace the `-mimeinclude` option with the `-indmimeinclude` option in the preceding example.

## **-indskip**

**Syntax:** `-indskip HTML_tag "exp"`

**Type:** Web crawling only

Specifies that Verity Spider follow and parse links, but not index, any HTML document that contains the text of `exp` within the given `HTML_tag`. For multiple `HTML_tag` and `exp` combinations, use multiple instances of the `-skip` option.

You can use wildcard expressions, where the asterisk (\*) is for text strings and the question mark (?) is for single characters; for example:

```
'/my_doc*/year199?'
```

In Windows, include double-quotation marks around the argument to protect the special characters, such as the asterisk (\*). On UNIX, use single-quotation marks. This is only required when you run the indexing job from a command line. Quotation marks are not necessary within a command file (the `-cmdfile` option).

If you use backslashes, you must double them so that they are properly escaped; for example:

```
C:\\test\\docs\\path
```

To use regular expressions, also specify the `-regexp` option.

### **Example**

To skip all HTML documents that contain the word "personnel" in the Title element, while still parsing those documents for links to other documents, use the following:

```
-indskip title "personnel"
```

### **Example**

To avoid indexing directory listing pages, while still parsing the document and path links except for the link to the parent directory, use one of the following, depending on the web server being indexed:

- For Netscape web servers, use the following:

```
-indskip title "*Index of*"
-nofollow "*parent directory*"
```
- For Microsoft Internet Information Server, use the following:

```
-indskip a "**to parent directory*"
-nofollow "*parent directory*"
```

## **-maxdocsize**

**Syntax:** `-maxdocsize integer`

Specifies the maximum size, in kilobytes, for documents to be indexed. Any documents larger than the value specified by the `-maxdocsize` option are ignored.

The default is to index documents of any size.

## -metafile

**Type:** Web crawling only

**Syntax:** -metafile path\_and\_filename

Allows you to use a text file to map custom meta tags to valid HTTP header fields. If you use backslashes, you must double them so that they are properly escaped; for example:

```
C:\\test\\docs\\path
```

This means that you can use your own meta tag, in the document, to replace what is returned by the web server, or to insert it if nothing is returned. Currently, the only header fields of real value are "Last-Modified" and "Content-Length." Future enhancements could allow for greater variety.

The following is the syntax for entries in the text file:

```
name Last-Modified y|n
```

or

```
name Content-Length y|n
```

Where y|n is an override flag, which can be yes or no.

## Example

A mapping file for the -metafile option might include the following:

```
Doc_Last_Touched Last-Modified n  
Doc_Size Content-Length y
```

If you use the y override flag, the value for the custom meta tag overrides the value for the valid field, even if both values are present and differ. This can be useful when the valid field value is always sent, but you want to specify your own value with a custom meta tag.

If you use the n override flag, the value for the custom meta tag is used only if there is no value for the valid field returned by the server. If a value for the valid field exists, it is given precedence.

**Note:** If you have several entries mapping to the same valid field, only the last entry takes effect.

## -mimeexclude

**Syntax:** -mimeexclude mime\_1 [mime\_n] ...

Specifies MIME types that are neither followed nor indexed.

In Windows, include double-quotation marks around the argument to protect the special characters, such as the asterisk (\*). On UNIX, use single-quotation marks. This is only required when you run the indexing job from a command line. Quotation marks are not necessary within a command file (the -cmdfile option).

The default is to include all MIME types. For the mime variable, you can include the asterisk (\*) wildcard for text strings; for example:

```
'text/*'
```

You cannot use the question mark (?) wildcard, and the -regex option does not let you use regular expressions.

Use the -indmimeexclude option to allow Verity Spider to follow documents, without indexing them, to gain access to other desirable document types.

## **-mimeinclude**

**Syntax:** `-mimeinclude mime_1 [mime_n] ...`

Specifies MIME types to be included.

In Windows, include double-quotation marks around the argument to protect the special characters, such as the asterisk (\*). On UNIX, use single-quotation marks. This is only required when you run the indexing job from a command line. Quotation marks are not necessary within a command file (the `-cmdfile` option).

The default is to include all MIME types. For the mime variable, you can include the asterisk (\*) wildcard for text strings; for example:

```
'text/*'
```

You cannot use the question mark (?) wildcard, and the `-regex` option does not let you use regular expressions.

## **-mindocsize**

**Syntax:** `-mindocsize integer`

Specifies the minimum size, in kilobytes, for documents to be indexed. Any documents smaller than the value specified by the `-mindocsize` option are ignored.

The default is to index documents of any sizes.

## **-skip**

**Type:** Web crawling only

**Syntax:** `-skip HTML_tag "exp"`

Specifies that Verity Spider not index any HTML document that contains the text of `exp` within the given `HTML_tag`. For multiple `HTML_tag` and `exp` combinations, use multiple instances of the `-skip` option.

You can use wildcard expressions, where the asterisk (\*) is for text strings and the question mark (?) is for single characters; for example:

```
'/my_doc*/year199?'
```

In Windows, include double-quotation marks around the argument to protect the special characters, such as the asterisk (\*). On UNIX, use single-quotation marks. This is only required when you run the indexing job from a command line. Quotation marks are not necessary within a command file (the `-cmdfile` option).

If you use backslashes, you must double them so that they are properly escaped; for example:

```
C:\\test\\docs\\path
```

To use regular expressions, also specify the `-regex` option.

## **Example 1**

To skip all HTML documents that contain the word "personnel" in the Title element, use the following:

```
-skip title "personnel"
```

## Example 2

To skip all HTML documents that contain both the word "private" and the phrase "internal user" in any paragraph element, use the following:

```
-skip title "personnel"  
-skip p "*internal use*"
```

See also [-regexp](#).

## Locale options

The following sections describe the Verity Spider locale options.

### -charmap

**Syntax:** `-charmap name`

Specifies the character map to use. Valid values are 8859 or 850. The default value is 8859.

### -common

Specifies the path to the Verity home directory, *cf\_root/lib/common*.

**Note:** This option is typically not needed, as long as the PATH environment variable is set correctly.

### -datefmt

**Syntax:** `-datefmt format`

Specifies the Verity import date format to use. Valid values are MDY (the default), DMY, YMD, USA, and EUR. (For descriptions of these values, see [“Date format options” on page 95](#).)

### -language

**Syntax:** `-language name`

Specifies the Verity locale to use in indexing. This option is being replaced by the semantically consistent the `-locale` option, and is still supported for backwards compatibility.

### -locale

**Syntax:** `-locale name`

Specifies the Verity locale to use in indexing, such as German (deutsch) or French (français). The default is English (english). This option is identical to the `-language` option.

### -msgdb

**Syntax:** `-msgdb path`

Specifies the path to the ind.msg message database file.

If Verity Spider was installed properly, this option should be unnecessary. By default, the ind.msg message database file is read from the following directory:

```
cf_root/lib/platform/bin
```

Where *platform* represents the platform directory.

## Logging options

The following sections describe the Verity Spider logging options.

### -loglevel

**Syntax:** `-loglevel [nostdout] argument`

Specifies the types of messages to log. By default, messages are written to standard output and to various log files in the subdirectory named `/log` beneath the Verity Spider job directory. If you add `nostdout` to the `-loglevel` option, messages are not written to standard output. Log files, however, are still created.

The following table describes valid message types:

Message type	Description
information	Licensing information written to <code>info.log</code> . Included with all arguments.
warning	Warning messages written to <code>warning.log</code> . Included with all arguments.
error	Error messages written to <code>error.log</code> . Included with all arguments.
badkey	Messages regarding keys that could not be indexed due to invalid documents, written to <code>badkey.log</code> . Included with all arguments.
progress	Current state of a document key written to <code>progress.log</code> . Note that a key with a progress of "inserting" might be a badkey and therefore skipped, rather than an indexed key. Included with all arguments.
summary	Inserted, indexed, and ignored messages written to <code>summary.log</code> . Included with all arguments except <code>skip</code> .
skip	Skipped documents, with explanation, written to <code>skip.log</code> . Included with all arguments, except <code>summary</code> .
debug	Internal Verity Spider processing messages, such as <code>enqueued</code> , written to <code>debug.log</code> . Included with both <code>debug</code> and <code>trace</code> arguments.
trace	Internal Verity Spider processing messages written to <code>debug.log</code> . Included only with the <code>trace</code> argument.

Choose one of the following arguments to determine which message types are logged:

Loglevel arguments	Description
summary	Includes the following message types: information, warning, error, badkey, progress, summary Use this option only if you do not want skip type messages.
skip	Includes the following message types: information, warning, error, badkey, progress, skip Use this option only if you do not want summary type messages.
verbose	Includes the following message types: information, warning, error, badkey, progress, summary, skip

---

Loglevel arguments	Description
debug	Includes the following message types: information, warning, error, badkey, progress, summary, skip, debug Note: Only use this argument at the direction of Verity technical support or for troubleshooting indexing problems.
trace	Includes the following message types: information, warning, error, badkey, progress, summary, skip, debug, trace Note: Only use this argument at the direction of Verity technical support or for troubleshooting indexing problems.

---

## Maintenance options

The following sections describe the Verity Spider maintenance options.

### -nooptimize

Prevents Verity Spider from optimizing the collection, thus reducing processing overhead during indexing. Use this option sparingly, as it leaves the collection in less than optimum shape. The following are some examples of when you might want to use this option:

- You want to manually perform custom optimization of the collection, using the `mkvdk` utility. By default, the Verity Spider optimization mimics the `mkvdk` actions of `maxmerge` and `vdbopt`. For more information on the `mkvdk` utility, see *Verity Collection Building Guide* and [Chapter 8, “Managing Collections with the `mkvdk` Utility,” on page 89](#).
- You are running multiple indexing jobs against a collection, and want to wait until they are all finished to optimize.

Generally, you should not leave a collection unoptimized for too long, as search times can slow significantly.

In brief, optimizing a collection means creating a small number of large partitions, which can greatly reduce search times.

### -purge

Deletes document tables and index files in the collection, and cleans up the collection's persistent store. The collection is then fresh with its original style files, and is not deleted from the file system.

### -repair

Specifies a failure-recovery mode for the collection, where the goal is to determine the causes of any errors, repair the errors (if possible), and restart a collection.

Although the Verity indexing engine always leaves the collection in a consistent, usable state, and no data can be lost or corrupted due to machine failures, it is possible for a process or event external to the Verity engine to corrupt one or more collections.

You can use the `-repair` option for constant failure-recovery operation, or you can run it selectively on collections that failed.



## Setting MIME types

You can use the MIME type criteria options, `-mimeinclude`, `-indmimeinclude`, `-mimeexclude`, and `-indmimeexclude`, to include or exclude MIME types.

### Syntax restrictions

When you specify MIME type criteria, keep in mind the restrictions described in the following sections.

#### Using the wildcard character (\*)

The asterisk (\*) wildcard character does not operate as a regular expression for the value of the MIME type criteria. Instead, you can only use it to replace the entire MIME type or MIME subtype.

For example, the following value is a valid substitute for `text/html`:

```
text/*
```

The following value is NOT a valid substitute for `text/html`:

```
text/h*
```

#### Multiple parameter values

When you specify a series of parameter values for a single instance of one of the MIME type criteria, and you use quotation marks, you must enclose each separate parameter value in single-quotation marks. For example:

```
-mimeinclude 'text/plain' 'application/*'
```

If you enclose the entire sequence of parameter values, as follows:

```
-mimeinclude 'text/plain application/*'
```

Verity Spider considers the entire expression a single value.

You can also use multiple instances of the MIME type criteria, each with a single parameter value, where quotation marks are necessary only if you use the wildcard character (\*). For example:

```
-mimeinclude text/plain  
-mimeinclude 'application/*'.Setting MIME Types
```

### MIME types and web crawling

When you index a website, Verity Spider evaluates your MIME type criteria against the "Content-Type" HTTP headers sent by the web server hosting that website. That web server passes along MIME type information based on its own internal tables.

When you encounter MIME types being dropped, make sure that the web server you are indexing has the necessary MIME type information. For information about specifying MIME types, see the documentation for your web server.

You can examine the indexing job's log files for indications that files are being skipped due to MIME types. For example, a typical ASCII file you might want indexed is a log file (filename.log). Unless the web server understands that files with .LOG extensions are ASCII text, of MIME type text/plain, you will see in the indexing job log file that .LOG files are skipped because of MIME type, even if you use the following:

```
-mimeinclude 'text/*'
```

## MIME types and file system indexing

When you index a file system, Verity Spider reads filenames and evaluates your MIME type criteria against an internal, compiled list of known MIME types and associated file extensions. You cannot edit this list. However, you can use the `-mimemap` option to create a custom MIME type mapping.

When you encounter MIME types being dropped, check whether Verity Spider recognizes that particular MIME type. For more information, see the table, [“Known MIME types for file system indexing” on page 131](#).

You can examine the indexing job's log files for indications that files are being skipped due to MIME types. For example, a typical ASCII file you might want indexed is a log file (filename.log). Since Verity Spider does not understand that files with .LOG extensions are ASCII text, of MIME type text/plain, you will see in the indexing job log file that .LOG files are skipped because of MIME type, even if you use the following:

```
-mimeinclude 'text/*'.Setting MIME Types
```

## Indexing unknown MIME types

Whenever you find MIME types being dropped, or you know you will be indexing files whose extensions are not known to Verity Spider by default, use the `-mimemap` option to point to a file that contains your own custom mappings for file extensions and MIME types.

You can also use the regular expression `'*/*'` for your MIME type criteria; for example:

```
-mimeinclude '*/*'
```

On either platform, you must include single-quotation marks for values that include wildcard characters.

Also use inclusion and exclusion criteria to finely control what is indexed, as follows:

- If your list of file types to index is rather long, use exclusion criteria (`-exclude`, `-indexclude`, `-mimeexclude`, or `-indmimeexclude`) to exclude extensions you know you do not want to index; for example:  

```
-exclude '*.exe' '*.com'
```
- If the list of file types you want to index is relatively small, use inclusion criteria (`-include`, `-indinclude`, `-mimeinclude`, or `-indmimeinclude`) to specify them; for example:  

```
-include '*.txt' '*.1st' '*.log'.Setting MIME Types
```

## Known MIME types for file system indexing

The following table lists the MIME types that Verity Spider recognizes when indexing file systems:

<b>Format</b>	<b>MIME type</b>	<b>Extension</b>
HTML	text/html	htm, html
ASCII	text/plain	txt, text
ASCII, source files	text/plain	c, h, cpp, cxx
PDF	application/pdf	pdf
MS Word	application/msword	doc
MS Excel	application/excel	xls
MS PowerPoint	application/vnd.ms-powerpoint	ppt
WordPerfect 5.1	application/wordperfect5.1	wpd
RTF	application/rtf	rtf
FrameMaker MIF	application/vnd.mif	mif



# CHAPTER 10

## Searching Collections with K2 Server

This chapter provides information about how to configure the Verity K2 Server, which is installed with ColdFusion MX.

### Contents

Using K2 Server . . . . .	133
Stopping K2 Server . . . . .	135
The k2server.ini parameter reference . . . . .	136
Using the rck2 utility to search K2 Server documents . . . . .	140

### Using K2 Server

You configure K2 Server to work with ColdFusion MX with the following steps:

- 1 Edit the k2server.ini file to specify the alias collection names you want to expose to K2 Server. (See “Editing the k2server.ini file” on page 133.)
- 2 Start K2 Server by running the k2server executable. (See “Starting K2 Server” on page 135.)
- 3 Specify hostname and port information for K2 Server. (See “Specifying K2 Server parameters in the ColdFusion MX Administrator” on page 135.)

### Editing the k2server.ini file

To enable a collection for searching using K2 Server, you must first configure the k2server.ini file. This file is located in the *cf\_root\lib\* directory in Windows, and in the *cf\_root/lib/* directory on UNIX.

In these pathnames, *cf\_root* refers to the ColdFusion MX root directory. In Windows, this is typically C:\CFusionMX; on UNIX, this is typically /opt/coldfusionmx.

The k2server.ini file consists of several parameters that typically remain unchanged. You must verify or make minor edits to settings in the portNo, vdkHome, and Coll-n sections.

## To edit the k2server.ini file:

- 1 Open the k2server.ini file in your text editor.

**Tip:** Use your text editor's search function to locate the appropriate code. For example, to locate the settings for the port number, as described in the next step of this procedure, search for portNo=.

**Note:** If you did not install ColdFusion MX into the default directory, edit the paths in this procedure to reflect the appropriate directories.

- 2 In the code for portNo=, verify that the value matches the value for the K2 Server port. The default value is 9901:

```
##portNo: TCP port number for client connections.  
portNo=9901
```

- 3 (Required only if you run K2 Server as a Windows service) In the code for vdkHome=, verify that the value matches the location of the Verity common directory. This is the cf\_root\lib\common directory in Windows, and is the cf\_root/lib/common directory on UNIX.

```
## vdkHome: directory containing Verity resources (common directory)  
- need it running as an NT service  
## vdkHome=c:/cfusionmx/lib/common
```

**Note:** If you run K2 Server as a Windows NT service, you must remove the pound signs (##) in the highlighted line above to uncomment the code. If the line remains as a comment, K2 Server will not execute correctly.

- 4 In the code for [Coll-0], specify in the collPath parameter the directory of a collection that K2 Server will search:

```
[Coll-0]  
collPath=c:\cfusionmx\verity\collections\test_01\file  
collAlias=test_01_file
```

The collPath value must point to an existing Verity collection; the k2server executable cannot be used to create a collection.

**Note:** The final subdirectory for your collPath might differ, based on whether it is an external collection (that is, a native Verity tool created it) or ColdFusion MX created it. If ColdFusion MX created the collection, there are file and custom subdirectories; these subdirectories are not present in external collections. For more information, see ["Collection structure and ColdFusion MX" on page 86](#).

- 5 In the next line, specify a collection alias in the collAlias parameter:

```
[Coll-0]  
collPath=c:\cfusionmx\verity\collections\test_01\file  
collAlias=test_01_file
```

You use this value to reference the collection in CFML.

**Note:** Collection alias values must be unique. They must be different from any collection names managed by ColdFusion MX.

The following CFML code performs a K2 mode search on the test\_01\_file collection:

```
<cfsearch  
  collection="test_01_file"  
  name="getData"  
  criteria="#form.criteria#">
```

**Note:** To search multiple collections, use a comma-delimited list. For example, use `collection="test_01_file,test_02_file"` in your `cfsearch` tag. Within a single `cfsearch` tag, the collections must be either all K2 Server-registered or all ColdFusion-registered; you cannot use one `cfsearch` tag to search a K2 Server-registered collection and a ColdFusion-registered collection.

In the following example, the `collPath` value points to a collection for the ColdFusion MX online documentation:

```
[Coll-1]
collPath=c:\cfusionmx\verity\collections\cfdocumentation\custom
collAlias=cfdoc_custom
topicSet=
knowledgeBase=
onLine=2
```

- 6 (Optional) Create a `Coll-n` section for other collections that you want to search with K2 Server. For each entry, increment the value `n` by one. The first collection is number 0, not number 1, as in the following example:

```
[Coll-2]
collPath=C:\cfusionmx\Verity\Collections\bbb\file
collAlias=bbb_file
```

- 7 Stop and restart K2 Server for changes in the `k2server.ini` file to take effect. For more information, see [“Stopping K2 Server” on page 135](#).

For more information about `k2server.ini` parameters, see [“The `k2server.ini` parameter reference” on page 136](#).

## Starting K2 Server

You start K2 Server from the command line on UNIX or in Windows. On UNIX, you run the `startk2server` script; in Windows, you run the `startk2server.bat` file. These files are located in the `cf_root\lib\` directory in Windows, and in the `cf_root/lib/` directory on UNIX.

In Windows, you can start K2 Server as a service by entering the following command in the `cf_root\lib\` directory:

```
k2server -ntservice 1 -inifile k2server.ini
```

**Note:** Macromedia does not recommend running K2 Server as a Windows service. You must stop the service before you modify or delete collections registered with K2 Server. You must then remember to restart the service. You must also verify that the `vdkHome` information in your `k2server.ini` file is uncommented—that is, it has no leading pound (#) signs—and points to the correct location of the common directory.

## Specifying K2 Server parameters in the ColdFusion MX Administrator

You use the Verity K2 Server page in the ColdFusion MX Administrator to specify the hostname and port number for the K2 Server you want to use. Make sure that you started K2 Server on the host you specify in the Verity Server hostname field. Also, the port number you enter must match the port number you specify in the `k2server.ini` file.

## Stopping K2 Server

You can run K2 Server as a Windows service or in a command window, as an ordinary application. Unless you use the `-ntService 1` option when starting K2 Server, K2 Server runs in the command window. There are several ways to stop K2 Server, depending on how it runs.

## Stopping K2 Server when run as a service

To halt K2 Server when it is running as a Windows service, do either of the following:

- Open the Services Control Panel and stop the K2 Server service.
- Open a command window and enter the following command:

```
k2server -ntService 0
```

## Stopping K2 Server when run as an application

When K2 Server is running as an application in a command window, you stop it by pressing Ctrl+C to kill the process in the window where it is running.

## Stopping K2 Server on UNIX

The ColdFusion MX installation includes a script that you run to halt K2 Server. By default, the stopk2server script is located in the `/cf_root/lib` directory.

## The k2server.ini parameter reference

The K2 Server configuration file, `k2server.ini`, contains many sections. The first section, `[server]`, provides parameters that control the behavior of the entire server. Each subsequent collection section (in the form `[Collection-1]`, `[Collection-2]`, and so on) controls each collection and search service configured for the server.

### Server section

The following table describes the parameters that you can use in the server section of the server configuration file. The K2 Server executable includes a sample configuration file (`k2server.ini`).

Parameter	Description
<code>serverAlias</code>	An arbitrary name used to identify the server.
<code>numThreads</code>	The default number of search threads to be started in the server process. If too many threads exist, the system can run out of memory; if too few threads exist, searches will be blocked and forced to wait for a Verity engine thread to become available. The value of <code>numThreads</code> is based on hardware resources and system needs.
<code>maxFiles</code>	<p>The maximum number of file handles that can be opened by a specific search thread. The default value for <code>maxFiles</code> is dependent on the limits of the OS used. The <code>maxFiles</code> value affects how file handles are shared between the operating system and the search engine. The <code>maxFiles</code> and <code>numThreads</code> values together can be used to tune system performance.</p> <p>The following values can be set for a server:</p> <pre>[server] numThreads=4 maxFiles=100</pre> <p>These entries for a K2 Server cause the system to support a maximum of 4 concurrent searches, with 100 file handles allocated for each search thread. The search engine determines default values per operating system. For large or fragmented collections, Macromedia recommends that you explicitly set a value for <code>maxFiles</code>.</p>



Parameter	Description
portNo	The TCP port number for client connections. The value of portNo is the same value assigned to portNo in the k2broker.ini file that identifies the broker referring to this server.
numListeners	The maximum number of clients that can connect to the server at one time. The numListeners value must be equal to or greater than the sum of all numThreads values specified by all K2 Brokers in the K2 search system. The numThreads value is set for a K2 Broker in the k2broker.ini file.
broker(n)	The brokers to ping on startup. Multiple brokers can be specified. For example: broker(1)=machinea:9900 broker(2)=machineb:9901
maxColSize	The maximum width of the fields to return to the results list, in bytes. The default is 2048 bytes.

## Search thread keywords

The following table describes keywords that you can use in your search threads:

Keyword	Description
vdkHome	The directory containing Verity resources.
vdkSortingFlag	A flag indicating whether the Verity engine sorts at the collection level. Valid values include: <ul style="list-style-type: none"> <li>• <b>NO, False, or 0</b> Do not perform sorting at the collection level (default)</li> <li>• <b>YES, True, or 1</b> Perform sorting at the collection level.</li> </ul> To implement sorting at the collection level, you must set vdkSortingFlag to YES in the k2server.ini file (in the server section) and the k2broker.ini file (in the broker section).
sortTruncDocs	The maximum number of documents to consider when sorting.
accessProfile	The Security Access Profile specified in the form of a query expression. The security access profile represents the access question that a document must pass in order for users to have access to it.
topicSet	The default pathname to a directory for the default topic set, which is an indexed set of topics. The value of the topicSet parameter identifies the default topic set to make available by every search service to clients at startup.
knowledgeBase	The default pathname to a knowledgeBase map file, which identifies numerous topic sets (indexed topics). The value of the knowledgeBase parameter identifies the topic sets (multiple) to make available to clients for every search service at startup).
charMap	A string that names the character set to use for strings that are sent to the server and generated by the server. This string must match the name of a .cs file in the root of the common directory that configures a character set and its mappings. For example, if your application uses character set 8859 for all of its interactions with the server, then set this charMap parameter to the string 8859. Valid values include, but are not limited to, the character sets supplied by Verity: 850 (default) for code page 850; 8859 for code page 8859.

Keyword	Description
locale	The name of the locale (combination of language, dialect, and character set) to use for all internal Verity engine operations. This name must correspond to a subdirectory in the common directory where the configuration file for the locale is found and where the message database and other locale-specific files are located. Leaving this parameter null means that the server uses the default internal locale, which is “english” written in the “850” character set.
resultCacheTimeout	The timeout in milliseconds for the result cache. The timeout occurs after 60 seconds or when the cache overflows based on resultCacheQuota.
resultCacheQuota	The number of slots per segment for the result cache. The result cache is composed of 16 segments, each of which has a number of slots for caching items: K2SearchNew, K2SearchRecv, K2DocReadBatch. The timeout occurs after <code>resultCacheQuota value * 16</code> . If <code>resultCacheQuota=10</code> , each of the segments has 10 slots. Since a search operation involves a call to K2SearchNew and a call to K2SearchRecv, an additional slot is used.
resultCacheEnabled	A flag indicating whether the result cache is enabled. Valid values include: <ul style="list-style-type: none"> <li>• <b>Yes, True, or 1</b> Enables the result cache.</li> <li>• <b>No, False, or 0</b> Disables the result cache (default).</li> </ul> By default, the cache is not enabled.
resultCacheMaxInBytes	Amount of memory, in bytes, to use for the cache.

## Collection sections

The K2 Server initializes a separate search service for each collection that you identify in the server configuration file. To add one or more collections to the configuration file, enter a separate block of keywords for each collection, in the following format:

```
[Coll-n]
collPath=<pathname>
collAlias=<value>
topicSet=<topicset>
knowledgeBase=<knowledgeBase>
numThreads=<value>
maxFiles=<value>
onLine=<value>
maxColSize=<value>
locale=<language>
charMap=<charmap>
inputDateFormat=<format>
```

Increment the block label for each collection that you configure, starting with Coll-0. The following table describes the keywords used to configure each collection and search service:

<b>Keyword</b>	<b>Description</b>
collPath	The pathname identifying the collection home directory.
collAlias	An arbitrary name used to identify the collection.
topicSet	The pathname to a directory for the default topic set, which is an indexed set of topics. The value of the topicSet parameter identifies the default topic set to make available to clients by every search service at startup. If not specified, the value of topicSet from the server section is used.
knowledgeBase	The pathname to a knowledgeBase map file, which identifies numerous topic sets (indexed topics). The value of the knowledgeBase parameter identifies the topic sets (multiple) to make available to clients for every search service at startup. If not specified, the value of the knowledgeBase parameter from the [server] section is used.
numThreads	The number of concurrent searches for the collection. If not specified, the value of numThreads from the [server] section is used.
maxFiles	The maximum number of files that can be opened by a specific search thread for a collection. If not specified, the value of the maxFiles parameter from the server section is used. The maxfiles and numThreads values together can be used to tune system performance. The following values can be set for a collection: <pre>[Coll-0] numThreads=4 maxFiles=100</pre> These entries for collection 0 cause K2 Server to support a maximum of 4 concurrent searches, with 100 file handles allocated for each search thread.
onLine	A flag indicating whether the server starts up with the collection on-line. Valid values include: <ul style="list-style-type: none"> <li>• <b>0</b> Start the server with the collection offline</li> <li>• <b>1</b> Start the server with the collection in a hidden state</li> <li>• <b>2</b> Start the server with the collection online (default).</li> </ul> In the hidden state, collections can be primed and tested, but are not yet available for searching by users. When collections are set offline, any queries currently running complete using these resources; subsequent queries do not see the resource.
maxColSize	The maximum width of the fields to return to the results list, in bytes. If not specified, the value of maxColSize from the server section is used.
locale	The name of the locale (combination of language, dialect, and character set) to use for all internal Verity engine operations. This name must correspond to a subdirectory in the common directory where the configuration file for the locale is found and where the message database and other locale-specific files are located. If not specified, the value of the locale parameter from the server section is used.

Keyword	Description
charMap	<p>A string that names the character set to use for strings that are sent into the server and generated by the server. This string must match the name of a .cs file in the root of the common directory that configures a character set and its mappings. If not specified, the value of the charMap parameter from the server section is used.</p> <p>For example, if your application uses character set 8859 for all of its interactions with the server, set this charMap parameter to the string 8859. Valid values include, but are not limited to, the character sets supplied by Verity: 850 (default) for code page 850; 8859 for code page 8859</p>
inputDateFormat	The input date format to be used. If there is no specified value for the inputDateFormat parameter, the default is MDY (Month-Day-Year), a numeric format.

## Using the rck2 utility to search K2 Server documents

The rck2 command-line utility lets you search collections associated with a K2 Server in a K2 Search System. The rck2.exe file, which starts the rck2 utility, is located in the *cf\_root\lib\\_nti40\bin* directory in Windows, and in the *cf\_root/lib/platform/bin* directory on UNIX.

In these pathnames, *cf\_root* refers to the ColdFusion MX root directory. In Windows, this is typically C:\CFusionMX; on UNIX, this is typically /opt/coldfusionmx. On UNIX, *platform* refers to the UNIX version of the server that runs ColdFusion: *\_solaris*, *\_hpux11*, or *\_ilnx21*.

### rck2 syntax

Use the following syntax to start rck2 from the command line:

```
rck2 -server <servername> -port <portno>
```

For example: `c:\cfusionmx\lib\_nti40\bin\rck2 -server localhost -port 9901`.

The following table describes rck2 syntax elements:

Syntax element	Description
-server <servername>	The server name for K2 Server to which to attach. The server name is defined in the k2server.ini file. The rck2 utility searches the collections attached to this server.
-port <portno>	The port number where K2 Server (specified by -server) is running.

### rck2 command options

The following table describes rck2 command options:

rck2 command	Description
p <sortspec>	The sort specification for the search results. By default, results are sorted by Score. Multiple fields must be specified in a space-separated list using asc or desc to indicate ascending or descending order. For example: <code>p score desc title asc</code>
m <maxdocs>	The maximum number of documents to return in the results list.

<b>rck2 command</b>	<b>Description</b>
c <collections>	The list of collections to search. Multiple collections must be specified in a space-separated list. For example: c coll1 coll2 coll3
f <fields>	The list of fields to retrieve. For example: f k2dockey title date
s <query text>	The query (or question) to be used to process the search. The query can be expressed as words and phrases separated by commas. Additionally, the query can include Verity query language, operators and modifiers.
g <collection>	Display collection information.
d <k2dockey>	Display fields for the K2 document key specified.
v <k2dockey>	Stream the document and display it with highlights.
r <docstart>	Display results starting with the first result in the results list. Fields specified using the f command are displayed. Docstart indicates the first result to be displayed. For example, r 10 displays results starting with the 10th document in the results list.
b <docstart>	Display results based on the last field selection.
i	Display information about K2 Server, including nodes and collections.
x <score precision>	Set score precision to 8- or 16-bit. By default, 16-bit precision is used.
h or ?	Display online Help for the rck2 command options.



# CHAPTER 11

## Searching Collections with the rcvdk Utility

This chapter provides information about using the rcvdk utility to search Verity collections.

### Contents

Using the Verity rcvdk utility . . . . .	143
Attaching to a collection using the rcvdk utility . . . . .	144
Viewing results of the rcvdk utility . . . . .	145

### Using the Verity rcvdk utility

Using the Verity rcvdk utility, you can check the contents of a collection from the command line. The rcvdk utility lets you write a variety of queries, using words and phrases separated by commas and Verity query language. A viewing option lets you see document contents and highlights in a simple text display.

The rcvdk executable is located in the *cf\_root*\lib\nti40\bin directory in Windows, and in the *cf\_root*/lib/*platform*/bin directory on UNIX.

In these pathnames, *cf\_root* refers to the ColdFusion MX root directory. In Windows, this is typically C:\CFusionMX; on UNIX, this is typically /opt/coldfusionmx. On UNIX, *platform* refers to the UNIX version of the server that runs ColdFusion: *\_solaris*, *\_hpux11*, or *\_ilnx21*.

To start the rcvdk utility on most systems, type the path and executable name at a command prompt. The following examples assume you have set your PATH variable, so you just have to enter rcvdk at a command prompt to run it.

For example:

```
c:\cfusionmx\lib\platform\bin\rcvdk /common = c:\cfusionmx\lib\common
```

When you start the rcvdk utility with no arguments, you get the following message, followed by the rcvdk prompt:

```
Type 'help' for a list of commands.  
RC>
```

The help command produces the following list of available commands:

```
RC> help
Available commands:
search      s Search documents.
results     r Display search results.
clusters    c Display clustered search results.
view        v View document.
summarize   z Summarize documents.
attach      a Attach to one or more collections.
detach      d Detach from one or more collections.
quit        q Leave application.
about       Display VDK 'About' info
help        ? Display help text; 'help help' for details.
expert      x Toggle expert mode on/off.
RC>
```

You can enter the letter q at the RC prompt at any time to quit the application.

## Attaching to a collection using the rcvdk utility

To search a collection, you first must attach to it using the `attach (a)` command. This command must include the pathname to a collection directory as an argument. After you press Return, the rcvdk utility reports whether the `attach` command was successful; for example:

```
RC>a /z/doc1/c/public/Collection/file_walking/collbldg/html
Attaching to collection:
/z/doc1/c/public/Collection/file_walking/collbldg/html
Successfully attached to 1 collection.
RC>
```

The rcvdk utility lets you attach to one or more collections. The specified collections remain attached until you detach from one or more collections using the `detach (d)` command.

## Basic searching

To retrieve all documents, use the `search (s)` command without arguments. After you press Return, a search update message is produced, as follows:

```
RC>s
Search update: finished (100%). Retrieved: 85(85)/85.
RC>
```

The search results indicate that 85 of the total 85 documents in the collection were retrieved. If you specify a query argument, such as “universal filter,” a subset of the total documents in the collection that contain the specified string is retrieved; for example:

```
RC>s universal filter
Search update: finished (100%). Retrieved: 18(18)/85.
RC>
```

In the message returned for the preceding search, the rcvdk utility indicates that 18 documents matched the query. You can perform more elaborate queries using the Verity query language, as shown in the following example:

```
RC>s universal filter <OR> filter.Troubleshooting and Maintenance Tools
```



## Viewing results of the rcvdk utility

After you have attached to a collection and issued a search command successfully, you can view the results list and look at the retrieved documents. You can use the options in the following table:

---

Option	Description
r	Displays the results list, starting with the first document. A maximum of 24 documents are displayed.
r n	Displays the results list, starting with the nth document. A maximum of 24 documents are displayed.
v	Displays the first or next document in the results list. Highlights are indicated using reverse video, if possible. If not, double angle brackets are used, as in: »universalk« »filter«  To exit the document display, enter the letter q.
v n	Displays the nth document in the results list. To exit the document display, enter the letter q.

---

The following is the results list for the “universal filter” search. For each document, these fields are displayed by default: Number, Score, and VdkVgwKey.

```
RC> r
Retrieved: 18(18)/85
Number SCORE VdkVgwKey
1: 1.00 d:\search97\s97is\locale\english\doc\collbldg\08_cbg3.htm
2: 0.97 d:\search97\s97is\locale\english\doc\collbldg\11_cbg2.htm
3: 0.97 d:\search97\s97is\locale\english\doc\collbldg\08_cbg7.htm
4: 0.97 d:\search97\s97is\locale\english\doc\collbldg\08_cbg1.htm
5: 0.95 d:\search97\s97is\locale\english\doc\collbldg\cbgtoc.htm
6: 0.95 d:\search97\s97is\locale\english\doc\collbldg\08_cbg4.htm
7: 0.93 d:\search97\s97is\locale\english\doc\collbldg\cbgix.htm
8: 0.92 d:\search97\s97is\locale\english\doc\collbldg\08_cbg6.htm
9: 0.90 d:\search97\s97is\locale\english\doc\collbldg\08_cbg.htm
10: 0.90 d:\search97\s97is\locale\english\doc\collbldg\04_cbg1.htm
11: 0.90 d:\search97\s97is\locale\english\doc\collbldg\01_cbg1.htm
12: 0.87 d:\search97\s97is\locale\english\doc\collbldg\f_cbg.htm
13: 0.87 d:\search97\s97is\locale\english\doc\collbldg\08_cbg2.htm
14: 0.84 d:\search97\s97is\locale\english\doc\collbldg\06_cbg1.htm
15: 0.80 d:\search97\s97is\locale\english\doc\collbldg\part4.htm
16: 0.80 d:\search97\s97is\locale\english\doc\collbldg\f_cbg1.htm
17: 0.80 d:\search97\s97is\locale\english\doc\collbldg\11_cbg5.htm
18: 0.80 d:\search97\s97is\locale\english\doc\collbldg\08_cbg5.htm
RC>
```

The following table describes each of the default fields:

---

Field name	Description
Number	The rank of the document in the results list. The document with the highest score is ranked number 1.
Score	The score assigned to each retrieved document, based on its relevance to the query. For a NULL query, no scores are assigned, so the Score column in the results list is blank.
VdkVgwKey	The document key used by the Verity engine to manage the document. If the document is accessed through the file system, the primary key is a pathname. If the document is accessed through a web server, using HTTP, the primary key is a URL.

---

## Displaying more fields

You can tell the `rcvdk` utility to display certain fields in the results list using the `fields` command, which is available in the expert mode. To go to the expert mode, enter `x` or `expert` at the RC prompt, then press Return.

All fields in a column are blank if the field is not defined for the collection's schema in the documents table (in `style.ddd`, `style.sfl`, or `style.ufl`). A field in a document's row is blank if the field was not populated by a gateway, bulk submit action, or filter.

## Displaying a field

The `fields` command includes the field name and length to be displayed. When used, the `fields` command overrides the default Score and VdkVgwKey fields for the results list.

The search engine returns fields for the results list, so if you do a search, then go to expert mode to use the `fields` command, you must run the search again in order to see the results list with the fields you requested. For example:

```
RC> expert
Expert mode enabled
RC> fields title 20
RC> s universal filter
Search update: finished (100%). Retrieved: 18(18)/85.
RC> r
Retrieved: 18(18)/85
Number title
1: Using the Universal Filter
2: Using the Zone Filter
3: The Zone Filter
4: Overview
5: Table of Contents
6: Universal Filter Configuration Using the
7: Index
8: The PDF Filter
9: Document Filters and Formatting
10: Collection Style Summary
11: Collection Basics
12: Universal Filter Document Types
13: Using the style.dft File
14: Supported Field Types
15:
16: Recognized Document Types
```

```
17: Custom Zone Definitions
18: The KeyView Filter Kit
RC>
```

## Displaying multiple fields

You can specify multiple fields with the `fields` command, as shown in the following example. The field order corresponds to the order of the columns, with the first field specified appearing in the second column. The first column is reserved for the rank order.

Rerun the search before you display the results list with the fields specified.

For example:

```
RC> fields score 5 title 40
RC> s universal filter
Search update: finished (100%). Retrieved: 18(18)/85.
RC>
```



# CHAPTER 12

## Troubleshooting Collections with Verity Utilities

This chapter provides information about using Verity utilities to configure, maintain, and troubleshoot Verity collections.

### Contents

<a href="#">Overview of Verity utilities</a> . . . . .	149
<a href="#">Using the Verity didump utility</a> . . . . .	150
<a href="#">Using the Verity browse utility</a> . . . . .	152
<a href="#">Using the Verity merge utility</a> . . . . .	153

### Overview of Verity utilities

The following command-line utilities are included with ColdFusion MX for performing a variety of operations on Verity collections:

Verity utility	Description	For more information
rcvdk	Search collections and display documents.	See <a href="#">Chapter 11, “Searching Collections with the rcvdk Utility,”</a> on page 143.
rck2	Search K2 Server collections.	See <a href="#">“Using the rck2 utility to search K2 Server documents”</a> on page 140.
mkvdk	Create and maintain collections.	See <a href="#">Chapter 8, “Managing Collections with the mkvdk Utility,”</a> on page 89.
didump	View collection word lists.	See <a href="#">“Using the Verity didump utility”</a> on page 150.
browse	Browse documents table and search results.	See <a href="#">“Using the Verity browse utility”</a> on page 152.
merge	Combine collections.	See <a href="#">“Using the Verity merge utility”</a> on page 153.

## Using the Verity didump utility

Using the didump utility, you can view key components of the word index per partition. The word list is a list of all words indexed by the Verity engine. The zone list is a list of all zones and the zone attribute list is a list of the zone attributes found by the Verity engine.

The didump executable, which starts the didump application, is located in the *cf\_root\lib\\_nti40\bin* directory in Windows, and the *cf\_root/lib/platform/bin* directory on UNIX.

In these pathnames, *cf\_root* refers to the ColdFusion MX root directory. In Windows, this is typically C:\CFusionMX; on UNIX, this is typically /opt/coldfusionmx. On UNIX, *platform* refers to the UNIX version of the server that runs ColdFusion: *\_solaris*, *\_hpux11*, or *\_ilnx21*.

For example:

```
c:\cfusionmx\lib\platform\bin\didump /common = c:\cfusionmx\verity\common
-pattern llama
c:\new\parts\00000001.did
```

## Viewing the word list with the didump utility

You can view the contents of the word list for a partition by using the didump utility with the *-words* flag. The command-line syntax must include the *-words* flag and a pathname to a partition file, like the following:

```
didump -words /z/collbldg/html/parts/00000003.did
```

An alphabetical listing of the words in the word index displays, as follows:

```
didump - Verity, Inc. Version 2.5.0 (_nti31, Jul 7 1999)
```

Text	Size	Doc	Word
A	10	3	4
a	34	5	24
abbreviations	4	1	1
about	4	1	1
acronym	5	1	2
acronyms	4	1	1
actual	4	1	1
administrator	3	1	1
advance	3	1	1
all	8	2	3
also	9	2	4
Always	4	1	1
always	9	2	3
ampersand	4	1	1

The columns in the display indicate the following:

- **Size** The number of bytes used by the Verity engine to store information about the word
- **Doc** The number of unique documents in which the word appears
- **Word** The total number of occurrences of a word for the partition

To view the occurrences of a specific word or pattern, enter a command using the *-pattern* option, as in the following example:

```
didump -pattern acronym 00000003.did
```

In this example, the didump utility displays information about the number of occurrences of the word *acronym*. You can display the individual occurrences of a word using the *-verbose* option.

## Viewing the zone list with the didump utility

The zone list contains a list of the zones identified by the zone filter. You can search the zones listed using the Verity IN operator in a query. To view the contents of the zone list, use the didump utility with the `-zones` flag plus the pathname to a partition, like the following:

```
didump -zones /z/collbldg/html/parts/00000003.did
```

This partition is for a collection containing the Verity Collection Building Guide in HTML format. The Verity universal filter invoked the HTML filter by default, and indexed the documents using these zones.

```
didump - Verity, Inc. Version 2.5.0 (_solaris, Jul 07 1999)
```

ZoneName	Fmt	Size	Doc	Regions
A	Wct	10239	85	5016
ADDRESS	Array	34	1	1
BODY	Array	197	85	85
CAPTION	Wct	298	31	85
CODE	Wct	3868	66	1829
H1	Array	80	83	83
H2	Wct	646	53	212
H3	Wct	517	49	171
H4	Wct	128	8	47
HEAD	Array	70	85	85
HTML	Array	165	85	85
TITLE	Array	70	85	85

The columns in the display indicate the following:

- **Fmt** The internal data format used to store the zone information.
- **Size** The number of bytes used by the Verity engine to store information about the zone.
- **Doc** The number of unique documents in which the zone appears
- **Region** The total number of instances of a zone for the partition

## Viewing the zone attribute list with the didump utility

The zone attribute list contains a list of the HTML attributes for the zones identified by the HTML zone filter. You can search the zone attributes listed using the Verity IN operator together with the WHEN operator in a query. To view the contents of the zone attributes list, use the didump utility with the `-attributes` flag plus the pathname to a partition, like the following:

```
didump -attributes /z/collbldg/html/parts/00000003.did
```

This partition is for a collection containing the *Verity Collection Building Guide* in HTML format.

```
didump - Verity, Inc. Version 2.5.0 (_solaris, Jul 9 1999)
```

Text	Size	Doc	Word
href 01_cbg.htm	10	2	4
href 01_cbg.htm#282870	3	1	1
href 01_cbg.htm#282872	6	2	2
href 01_cbg1.htm	8	2	3
href 01_cbg1.htm#286513	7	2	2
href 01_cbg1.htm#286520	3	1	1
...			

The columns in the display indicate the following:

- **Size** The number of bytes used by the Verity engine to store information about the zone attribute
- **Doc** The number of unique documents in which the zone attribute appears
- **Word** The total number of occurrences of a zone attribute for the partition

## Using the Verity browse utility

A documents table is built for each partition in a collection. The documents table is used for field searching and for sorting search results. The fields within the documents table are defined by the following collection style files:

- **style.ddd** Defines fields used internally by the Verity engine, identified by an initial underscore character (\_).
- **style.sfl** Defines standard fields (many of which are commented out to limit the size of the documents table).
- **style.ufl** Defines custom fields that are not included in the style.sfl file.

The value of each field can be filled in from source documents or can be provided explicitly. If a field is blank, it has not been populated.

The browse utility executable, which starts the browse utility application, is located in the *cf\_root\lib\nti40\bin* directory in Windows, and in the *cf\_root/lib/platform/bin* directory on UNIX.

In these pathnames, *cf\_root* refers to the ColdFusion MX root directory. In Windows, this is typically C:\CFusionMX; on UNIX, this is typically /opt/coldfusionmx. On UNIX, *platform* refers to the UNIX version of the server that runs ColdFusion: *\_solaris*, *\_hpux11*, or *\_ilnx21*.

For example:

```
c:\cfusionmx\lib\nti40\bin\browse /common = c:\cfusionmx\lib\common
c:\my_collection\parts\0000001.ddd
```

## Using menu options with the browse utility

Use the following browse command to start the utility and display a set of menu options:

```
browse 00000003.ddd
```

The system displays the following menu of options available for the browse utility:

```
D:\VERITY\colltest\parts>browse 00000003.ddd
BROWSE OPTIONS
?) help
q) quit
c) Number of entries in field
_) Toggle viewing fields beginning with '_'
v) Toggle viewing selected fields
##) Display all fields in specified record number
Dispatch/Compound field options:
n) No dispatch
d) Dispatch
s) Dispatch as stream
Action (? for help):
```



## Displaying fields

You can use several options to control the display of field information.

### To display all the document fields:

- 1 At the Action prompt, enter ##
- 2 Press Return twice to display the fields for the first document record.
- 3 Press Return to view the document fields for the next sequential record.

The following partial display of the results of the browse command includes internal fields, used by the Verity search engine. An internal field name starts with an underscore character (\_).

```
50 Created                FIX-date ( 4) = 12-Jan-1998 01:52:27 pm
51 Modified              FIX-date ( 4) = 24-Sep-1997 02:40:26 pm
52 Size                  FIX-unsq ( 4) = 5381
53 DOC_OF                FIX-unsq ( 4) = 0
54 DOC_SZ                FIX-unsq ( 4) = 4294967295
55 DOC_FN_OF             FIX-unsq ( 4) = 436
56 DOC_FN_SZ             FIX-unsq ( 2) = 58
57 _CACHE_FN_OF         FIX-unsq ( 4) = 2922
58 _CACHE_FN_SZ         FIX-unsq ( 2) = 0
59 _ParentID_OF         FIX-unsq ( 4) = 354
60 _ParentID_SZ         FIX-unsq ( 2) = 46
61 Title_OF              FIX-unsq ( 4) = 2481
62 Title_SZ              FIX-unsq ( 2) = 15
```

You can eliminate the internal fields. To do this, type the underscore character, then press Return. If you enter an underscore character again, then press return, the internal fields are displayed.

## Using the Verity merge utility

The merge utility lets you combine multiple collections with identical schemas. This is useful for merging smaller collections built from different sources into one, large collection. Also, you can use the merge utility to break up the collection into smaller collections of a roughly uniform size.

**Note:** The Verity merge utility is available only in Windows.

Collections can be merged only if they have identical schemas. Collections can be merged if they have exactly the same set of style files (and style file entries).

Breaking up a large collection helps to optimize search performance, because it allows many applications to perform multiple concurrent search requests over the different collections. After breaking up a large collection, you can also discard older collections to reclaim limited disk storage space.

The merge executable, which starts the merge application, is located in the *cf\_root\lib\\_nti40\bin* directory.

In the above location, *cf\_root* refers to the ColdFusion MX root directory.

For example:

```
c:\cfusionmx\lib\_nti40\bin\merge /common = c:\cfusionmx\lib\common
```

To obtain help for the merge utility, enter the following command:

```
merge -help
```

**Note:** After running the merge utility, you must optimize the collection, using the `mkvdk -optimize` option.

## Merging collections using the merge utility

The following is the syntax for using the merge utility to merge multiple collections into a single collection:

```
merge <newCollection> <srcCollection1> <srcCollection2> [srcCollectionN]
```

The utility reads `srcCollection1`, `srcCollection2` and so on and merges them into a single collection with the directory name given for `newCollection`. If the directory name given for `newCollection` does not exist, it is created.

## Splitting collections using the merge utility

The following is the syntax for using the merge utility to split a single large collection into smaller collections:

```
merge -split <srcCollection> <newCollection1> <newCollection2> [-number]
```

The merge utility reads `srcCollection` and splits it into roughly equal pieces, using the filenames given for `newCollection1` and so on.

If you want to split a very large collection into a large number of new collections, you can use the following command, instead of explicitly naming each new collection:

```
merge -split -number newCollection srcCollection
```

The merge utility reads the collection identified by `srcCollection` and splits it into the number of segments specified by the `-number` option. The name of the first new collection is generated by appending the first two letters in the alphabet (aa) to the directory name given for `newCollection`. Each subsequent filename is generated by incrementing one of the appended letters (up to zz) for a maximum of 676 partitions. For example, if the value of `-number` is 3, and the value of `newCollection` is `Collection1`, the collections are named, `Collection1aa`, `Collection1ab`, and `Collection1ac`.

**Note:** The maximum length of the directory name given for `newCollection` is two characters less than the length allowed by the file system.

# CHAPTER 13

## Verity Error Messages

This chapter provides information about error messages that might occur when using Verity in either VDK mode or K2 mode.

### Contents

VDK mode error codes .....	155
K2 mode error codes .....	159

### VDK mode error codes

All Verity Developer's Kit (VDK) API functions return an error code, and `VdkSuccess` is the successful return value. The following sections list the API error codes. These reflect actions of the `cfcollection`, `cfindex`, or `cfsearch` tags.

#### Generic error codes

Error code	No.	Description
<code>VdkSuccess</code>	(0)	Operation completed successfully.
<code>VdkFail</code>	(-2)	A general failure not covered by another API error code.
<code>VdkWarn</code>	(1)	A general warning.

#### Usage error codes

Error code	No.	Description
<code>VdkError_BadArgStruct</code>	(-10)	Invalid argument structure.
<code>VdkError_BadHandleType</code>	(-11)	Improper object type.
<code>VdkError_HandleNotFound</code>	(-12)	Object not found.
<code>VdkError_MissingArgs</code>	(-13)	Missing required arguments.
<code>VdkError_InvalidArgs</code>	(-14)	Invalid arguments.
<code>VdkError_MultipleSesNew</code>	(-16)	<code>VdkSessionNew</code> called twice.
<code>VdkError_NestedService</code>	(-17)	<code>VdkService</code> called reentrantly.

<b>Error code</b>	<b>No.</b>	<b>Description</b>
VdkError_NestedFree	(-18)	VdkSessionFree called reentrantly.
VdkError_Unsupported	(-19)	Using an unsupported feature.

## Runtime error codes

<b>Error code</b>	<b>No.</b>	<b>Description</b>
VdkError_NoMsgDb	(-20)	Cannot find the message database.
VdkError_FatalError	(-21)	Fatal error.
VdkError_OutOfMemory	(-22)	Out of memory.
VdkError_DiskFull	(-23)	Out of disk space.
VdkError_NoFileHandles	(-24)	Out of file handles.
VdkError_InvalidDoc	(-25)	Bad document ID or key (internal or external).
VdkError_FileNotFound	(-26)	File not found.
VdkError_ArgTooLarge	(-27)	Argument too large.
VdkError_InvalidSortSpec	(-28)	Invalid sort specification.
VdkError_GatewayNotAvail	(-29)	Gateway driver not available.
VdkError_VersionMismatch	(-30)	Argument or object mismatch.
VdkError_NoInstallDir	(-100)	Cannot find installation directory.

## Data error codes

<b>Error code</b>	<b>No.</b>	<b>Description</b>
VdkError_StyleFiles	(-31)	Invalid style files.
VdkError_Permissions	(-32)	Bad file or directory permission.
VdkError_CollNotAvail	(-33)	The collection is not available because it is down or under repair. This error occurs only when the Verity engine is attempting a submit action (for example, insert, update, or delete), to a collection. If this error is returned, the submit action does not occur.
VdkError_CollIlll	(-34)	The collection is very sick.
VdkError_CollRepair	(-36)	The collection has been repaired.
VdkError_CollReadOnly	(-37)	This collection is read-only. No submits are allowed.
VdkError_CollPurge	(-38)	Purge failed due to problems deleting from any of the following directories: pdd, work, trans
VdkError_CollPathTooBig	(-39)	Collection path supplied for the path member in VdkCollectionOpenArgRec is too long. For more information, refer to the description of the VdkPath_MaxSize macro in your Verity documentation.

<b>Error code</b>	<b>No.</b>	<b>Description</b>
VdkError_V3Legacy	(-35)	Unsupported legacy collection(s).
VdkError_LocaleIncompat	(-101)	Collection and session locales are incompatible.
VdkError_KBNotOpened	(-102)	Knowledge base is incompatible and cannot be opened.

## Query error codes

<b>Error code</b>	<b>No.</b>	<b>Description</b>
VdkError_QueryParse	(-40)	Query has a parsing error.

## Licensing error codes

<b>Error code</b>	<b>No.</b>	<b>Description</b>
VdkError_Signature	(-50)	Invalid/missing signature.
VdkError_LicenseFile	(-51)	Invalid license file.
VdkError_LicenseColl	(-52)	Too many collections open.
VdkError_LicenseVolume	(-53)	Too many documents in collection.
VdkError_LicenseAdvQuery	(-54)	No advanced query capability.
VdkError_LicenseHetero	(-56)	No heterogeneous collections.
VdkError_LicenseDataPrep	(-57)	Not licensed to index documents.
VdkError_LicenseStreams	(-58)	Not licensed for streams.
VdkError_LicenseTopics	(-59)	Not licensed for topics.
VdkError_LicenseThes	(-60)	Not licensed for thesaurus.
VdkError_LicenseAdvFeat	(-64)	Not licensed for advanced features.
VdkError_LicenseSesSpawn	(-65)	No spawning sessions.
VdkError_LicenseWatchers	(-66)	No watchers.
VdkError_LicenseAcrocoll	(-67)	No access to Acrobat.
VdkError_LicenseProfile	(-68)	No profilers.
VdkError_LicenseProfileLatency	(-69)	Low-speed profiler.
VdkError_LicensePrfCount	(-110)	Too many profiles.
VdkError_LicenseClustering	(-111)	No clustering.
VdkError_LicenseSummarization	(-112)	No summarization.
VdkError_LicenseNLQP	(-113)	No natural language queries.
VdkError_LicenseQBE	(-114)	No query-by-example.
VdkError_LicenseAdvSGML	(-115)	No support for advanced SGML search.
VdkError_LicenseZone	(-116)	No support for zone search.

<b>Error code</b>	<b>No.</b>	<b>Description</b>
VdkError_LicenseField	(-117)	No support for field search.
VdkError_LicenseAccrue	(-118)	No support for the ACCRUE operator.
VdkError_LicenseProximity	(-119)	No support for the proximity operators.
VdkError_LicenseStem	(-120)	No stemming.
VdkError_LicenseWildcard	(-121)	No support for wildcard queries.
VdkError_LicenseTypo	(-122)	No support for typo assist.
VdkError_LicenseOperator	(-123)	Unlicensed operator.
VdkError_LicenseInso	(-124)	Not licensed for INSO software.
VdkError_LicenseInvalid	(-125)	Invalid license.
VdkError_LicenseVgw	(-126)	No collection gateways.
VdkError_LicenseSoundex	(-127)	No support for Soundex queries.
VdkError_LicenseSentpara	(-128)	No support for SENTENCE or PARAGRAPH operators.
VdkError_Scoreop	(-129)	No support for Score operators.
VdkError_Opmod	(-130)	No support for query language modifiers.
VdkError_LicenseSession	(-131)	Too many top-level sessions.

## Security error codes

<b>Error code</b>	<b>No.</b>	<b>Description</b>
VdkError_InvalidUser	(-80)	Invalid user/password combination.

## Remote connection error codes

<b>Error code</b>	<b>No.</b>	<b>Description</b>
VdkError_HostNotAvail	(-90)	Cannot contact remote host.
VdkError_NotReEntrant	(-91)	Not reentrant.
VdkError_CallDenied	(-92)	Call cannot be executed.

## Filtering error codes

<b>Error code</b>	<b>No.</b>	<b>Description</b>
VdkError_BadFile	(-140)	Corrupt or unreadable file.
VdkError_EmptyFile	(-141)	Empty file.
VdkError_ProtectedFile	(-142)	Password protected or encrypted file.
VdkError_FilterNotAvail	(-143)	No appropriate filter for a file format.

Error code	No.	Description
VdkError_FilterLoadFailed	(-144)	Error occurred during filter initialization.
VdkError_FileOpenFailed	(-145)	File could not be opened.

## Dispatch error codes

Error code	No.	Description
VdkError_CouldntLoadDLL	(-200)	Cannot load DLL.
VdkError_NoSuchFunction	(-201)	Function not available.

## Warning error codes

Error code	No.	Description
VdkWarning_CollectionDown	(10)	The collection was down when it was opened.
VdkWarning_QueryComplex	(11)	Too many matching words.
VdkWarning_LowMemory	(12)	Memory is low for indexing.
VdkWarning_CollectionReadOnly	(13)	The collection is read-only.
VdkWarning_DriverNotFound	(14)	Couldn't locate specified driver.
VdkWarning_LargeToken	(15)	Returned a token greater than maxSize.
VdkWarning_ArgTooLarge	(16)	Argument too large.
VdkWarning_DataSrcNotAvail	(17)	Cannot locate collection data.
VdkWarning_SearchRestricted	(18)	Search restricted to a subset of the collection.

## K2 mode error codes

All K2 Client API functions return an error code, and K2Success is the successful return value. The following sections list the API error codes. These reflect actions of the `cfsearch` tag.

### Generic error codes

Error code	No.	Description
K2Success	(0)	Operation completed successfully.
K2Fail	(-2)	A general failure not covered by another API error code.
K2Warn	(1)	A general warning.

### Usage error codes

Error code	No.	Description
K2Error_NoConnectAvail	(-9)	A K2 connection is not available.
K2Error_BadArgStruct	(-10)	Invalid argument structure.

<b>Error code</b>	<b>No.</b>	<b>Description</b>
K2Error_BadHandleType	(-11)	Improper object type.
K2Error_HandleNotFound	(-12)	Object not found.
K2Error_MissingArgs	(-13)	Missing required arguments.
K2Error_InvalidArgs	(-14)	Invalid arguments.
K2Error_Unsupported	(-19)	Using an unsupported feature.

## Runtime error codes

<b>Error code</b>	<b>No.</b>	<b>Description</b>
K2Error_NoMsgDb	(-20)	Cannot find the message database.
K2Error_FatalError	(-21)	Fatal error.
K2Error_OutOfMemory	(-22)	Out of memory.
K2Error_DiskFull	(-23)	Out of disk space.
K2Error_NoFileHandles	(-24)	Out of file handles.
K2Error_InvalidDoc	(-25)	Bad document ID or key (internal or external).
K2Error_FileNotFound	(-26)	File not found.
K2Error_ArgTooLarge	(-27)	Argument too large.
K2Error_InvalidSortSpec	(-28)	Invalid sort specification.
K2Error_GatewayNotAvail	(-29)	Gateway driver not available.
K2Error_VersionMismatch	(-30)	arg or Vdk Object mismatch.
K2Error_NoInstallDir	(-100)	Cannot find installation directory.

## Data error codes

<b>Error code</b>	<b>No.</b>	<b>Description</b>
K2Error_StyleFiles	(-31)	Invalid style files.
K2Error_Permissions	(-32)	Bad file or directory permission.
K2Error_CollNotAvail	(-33)	The collection is not available because it is down or under repair. This error occurs only when the Verity search engine is attempting a submit action (for example, insert, update, or delete), to a collection. If this error is returned, the submit action does not occur.
K2Error_CollIll	(-34)	The collection is corrupt and needs repair.
K2Error_v3Legacy	(-35)	Unsupported on Legacy V3 database.
K2Error_CollRepair	(-36)	The collection has been repaired.
K2Error_CollReadOnly	(-37)	This collection is read-only. No submits are allowed.



<b>Error code</b>	<b>No.</b>	<b>Description</b>
K2Error_CollPurge	(-38)	Purge failed due to problems deleting from any of the following directories: pdd, work, trans
K2Error_CollPathTooBig	(-39)	Collection path supplied for the path member in K2CollectionOpenArgRec is too long.
K2Error_LocaleIncompat	(-101)	Collection and session locales are incompatible.
K2Error_KBNotOpened	(-102)	Knowledge base cannot be opened.

### Query error codes

<b>Error code</b>	<b>No.</b>	<b>Description</b>
K2Error_QueryParse	(-40)	Query has a parsing error.

### Security error codes

<b>ErrorCode</b>	<b>No.</b>	<b>Description</b>
K2Error_InvalidUse	(-80)	Invalid user/password combination.

### Remote connection error codes

<b>Error code</b>	<b>No.</b>	<b>Description</b>
K2Error_HostNotAvail	(-90)	Cannot contact remote host.
K2Error_NotReEntrant	(-91)	Not reentrant.
K2Error_CallDenied	(-92)	Call cannot be executed.

### File handling error codes

<b>Error code</b>	<b>No.</b>	<b>Description</b>
K2Error_BadFile	(-140)	Corrupt or unreadable file.
K2Error_EmptyFile	(-141)	Empty file.
K2Error_ProtectedFile	(-142)	Password protected or encrypted.
K2Error_FilterNotAvail	(-143)	No appropriate filter.
K2Error_FilterLoadFailed	(-144)	Error during filter initialization.
K2Error_FileOpenFailed	(-145)	File could not be opened.

### Dispatch error codes

<b>Error code</b>	<b>No.</b>	<b>Description</b>
K2Error_CouldntLoadDLL	(-200)	Cannot load DLL.
K2Error_NoSuchFunction	(-201)	Function not available.

## Warning error codes

Error code	No.	Description
K2Warning_CollectionDown	(10)	The collection was down when it was opened.
K2Warning_QueryComplex	(11)	Too many matching words.
K2Warning_LowMemory	(12)	Memory is low for indexing.
K2Warning_CollectionReadOnly	(13)	The collection is read-only.
K2Warning_DriverNotFound	(14)	Couldn't locate specified driver.
K2Warning_LargeToken	(15)	Returned a token greater than maxSize.
K2Warning_ArgTooLarge	(16)	Argument too large.
K2Warning_DataSrcNotAvail	(17)	Cannot locate collection data.
K2Warning_SearchRestricted	(18)	Searching subset of collection.

## TCP/IP error codes

Error code	No.	Description
K2TcpError_Memory	c100	Out of memory.
K2TcpError_ConnDrop	c200	Connection closed by remote host.
K2TcpError_WillBlock	c300	Will block on this call.
K2TcpError_Call_DNS	c600	DNS lookup failed (use IP address).
K2TcpError_Call_Send	c700	Send failed (maybe connection damaged).
K2TcpError_Call_Recv	c800	Recv failed (maybe connection damaged).
K2TcpError_Call_loctl	c900	ioctl failed (Internal error).
K2TcpError_Call_Socket	ca00	Socket failed (maybe out of file handles).
K2TcpError_Call_Bind	cb00	Bind failed (local address already in use).
K2TcpError_Call_Listen	cc00	Listen failed (maybe out of resources).
K2TcpError_Call_Accept	cd00	Accept failed (maybe out of resources).
K2TcpError_Call_Select	ce00	Select failed (maybe connection damaged).
K2TcpError_Call_Connect	cf00	Connect failed (connection not accepted).

# INDEX

## A

- AddHandler directive 59
- administration, initial tasks 17
- Apache
  - application isolation configuration 78
  - configuration overview 59
  - multihoming 66, 78
  - sample configuration files 63
- apialloc property 63
- application isolation
  - enabling 77
  - overview 77
  - web server configuration 78
- application variables 22
- Archives and Deployment page, ColdFusion MX Administrator 25

## B

- batch files 62
- bootstrap property 63
- browse utility 152
- built-in web server
  - overview 58
  - web root 58

## C

- Cache Real Paths, disabling for multihoming 65, 66
- Caching Settings page, ColdFusion MX Administrator 19
- certificate authority 68
- CF Admin Password page, ColdFusion MX Administrator 34
- cform tag
  - multihoming 65
  - scriptsrc attribute 65
- cform.js file 65
- CFIDE/scripts directory 65

- cfstat utility, using 29
- CFX Tags page, ColdFusion MX Administrator 33
- Charting page, ColdFusion MX Administrator 24
- client variables
  - about 20
  - creating tables for 21
  - migrating data 21
- Client Variables page, ColdFusion MX Administrator 20
- clustering
  - creating a cluster of JRun servers 81
  - licensing for multiple computers 81
  - overview 81
- Code Compatibility Analyzer page, ColdFusion MX Administrator 33
- ColdFusion MX Administrator
  - about 13
  - Archives and Deployment page 25
  - Caching Settings page 19
  - CF Admin Password page 34
  - CFX Tags page 33
  - Charting page 24
  - Client Variables page 20
  - Code Compatibility Analyzer page 33
  - CORBA Connectors page 33
  - Custom Extensions section 35
  - Custom Tag Paths page 33
  - Data & Services section 26
  - Data Sources page 26
  - Debugging & Logging section 27
  - Debugging IP Addresses page 30
  - Debugging Settings page 28
  - default location 13
  - Extensions section 33
  - Java and JVM Settings page 25
  - Java Applets page 33
  - K2 Server parameters, specifying 135

- layout 14
- Mail Server page 23
- Mappings page 22
- Memory Variables page 22
- password 70
- RDS Password page 34
- Sandbox Security page 35
- Security section 34, 69
- Server Settings section 18
- Settings Summary page 26
- user assistance, types of 14
- Verity Collections page 26
- Verity K2 Server page 27
- Web Services page 27
- ColdFusion MX, deploying 76
- ColdFusion security 72
  - collections
    - attaching to with the rcvdk utility 144
    - backing up with the mkvdk utility 98
    - creating with the mkvdk utility 91
    - deleting with the mkvdk utility 98, 99
    - indexing with Verity Spider 103
    - maintaining with the mkvdk utility 98
    - merging with the merge utility 154
    - repairing with the mkvdk utility 98
    - search modes 87
    - searching K2 Server documents with the rck2 utility 140
    - searching with the rcvdk utility 144
    - setup options for the mkvdk utility 92
    - splitting with the merge utility 154
    - structure of 86
  - collections, Verity
    - defined 26
    - managing 27
- Configure web server for ColdFusion MX applications
  - checkbox 60
- connection string, specifying arguments 40
- context root, multiple server instances 77
- CORBA Connectors page, ColdFusion MX Administrator 33
- Custom Extensions section, ColdFusion MX Administrator 35
- Custom Tag Paths page, ColdFusion MX Administrator 33

## D

- Data & Services section, ColdFusion MX Administrator 26
- data sources
  - adding to ColdFusion MX Administrator 39
  - adding to ColdFusion MX Administrator, considerations 40
  - security 72
- Data Sources page, ColdFusion MX Administrator 26
- data sources, connecting to
  - DB2 Universal Database 41
  - Informix 43
  - Microsoft Access 44
  - Microsoft Access with Unicode 46
  - Microsoft SQL Server 47
  - MySQL 48
  - ODBC Socket 50
  - other data sources 52
  - Sybase 54
- DB2 Universal Database, connecting to 41
- Debugging & Logging section, ColdFusion MX Administrator 27
- Debugging IP Addresses page, ColdFusion MX Administrator 30
- Debugging Settings page, ColdFusion MX Administrator 28
- deploying ColdFusion MX 76
- didump utility
  - executable 150
  - using 150
  - word list, viewing 150
  - zone attribute list, viewing 151
  - zone list, viewing 151

## E

- error codes
  - K2 Server 159
  - VDK 155
- errorurl property 63
- extension mappings 59
- Extensions section, ColdFusion MX Administrator 33
- external web servers
  - configuration 59
  - configuring for application isolation 78
  - overview 59
  - SSL 68

## F

failover 81  
files and directories, security 72

## H

hosting  
    application isolation 77  
    multihoming 65  
httpd.conf file  
    application isolation 78  
    elements added to 59  
    multihoming 66  
    properties stored in 62

## I

IIS  
    application isolation configuration 78  
    configuration overview 59  
    multihoming 65  
    sample configuration file 64  
Informix, connecting to 43  
IP/Port, security 73  
iPlanet  
    application isolation configuration 81  
    configuration overview 59  
    multihoming 67  
    sample configuration file 64  
ISAPI filter 59

## J

J2EE configuration Web Server Configuration  
    Tool 60  
J2EE sessions, failover 81  
Java and JVM Settings page, ColdFusion MX  
    Administrator 25  
Java Applets page, ColdFusion MX Administrator 33  
JavaScript, cform considerations 65  
JDBC  
    about 37  
    driver types 38  
JMC, defining a JRun server 76  
JRun servers  
    creating 76  
    custom jvm.config 77  
JRun web server, *see* built-in web server  
jrun.dll 59  
jrun.ini file 62  
jrun.trusted.hosts 82  
jrun\_iis6.dll 59

jrun\_iis6\_wildcard.dll 59  
jrun\_nsapi35.dll 59  
JRunScripts directory 63  
JVM  
    custom JVM for a JRun server 77  
    Java and JVM Settings page 25  
JWS port number 76

## K

K2 Server  
    about 88  
    configuration overview 133  
    document search limits 86  
    error codes 159  
    hostname, specifying 135  
    parameters, specifying in ColdFusion MX  
        Administrator 135  
    port number, specifying 135  
    search mode 87  
    starting 135  
    stopping 135  
k2server.ini file  
    collection sections 138  
    editing 133  
    location 133  
    parameter reference 136  
    search thread keywords 137  
    server section 136  
keystore 68  
keytool command 68

## L

libjrun\_nsapi35.so 59  
libjrun\_nsapi40.so 59  
load balancing 81  
LoadModule directive 59  
Log files, created by ColdFusion MX 31

## M

magnus.conf 59, 62  
Mail Server page, ColdFusion MX Administrator 23  
Mappings page, ColdFusion MX Administrator 22  
Memory Variables page, ColdFusion MX  
    Administrator 22  
merge utility  
    collections, merging 154  
    collections, splitting 154  
    executable 153  
    using 153

- Microsoft Access with Unicode, connecting to 46
  - Microsoft Access, connecting to 44
  - Microsoft SQL Server, connecting to 47
  - migrating client variable data 21
  - mkvdk utility
    - bulk insert and delete, using 97
    - getting started 91
    - inserting documents into collections 91
    - log file 90
    - mkvdk.exe 89
    - online Help 91
    - optimization keywords 99
    - optimized databases (VDBs) 100
    - overview 89
    - processing documents with 94
    - service-level keywords 95
    - squeezing deleted documents 99
    - syntax 89
  - mkvdk utility collections
    - backing up 98
    - creating 91, 92
    - deleting 98, 99
    - maintaining 98
    - maintenance options 97
    - repairing 98
    - setup options 92
  - mkvdk utility options
    - bulk submit 97
    - collection maintenance 97
    - collection setup 92
    - date format 95
    - document processing 96
    - general processing 93
    - messaging 96
    - performance tuning 100
  - mod\_jrun.so 59
  - mod\_jrun20.so 59
  - multihoming
    - Apache 66
    - cacheRealPath attribute, disabling 65, 66
    - CFIDE/scripts, copying 65
    - IIS 65
    - iPlanet 67
    - multiple server instances 77
    - overview 65
    - Sun ONE Web Server 67
  - multiple server instances
    - application isolation 77
    - creating 76
    - custom JVM for a JRun server 77
    - defining a JRun server 76
    - failover 81
    - load balancing 81
    - overview 75
    - web server configuration (application isolation) 78
  - MySQL, connecting to 48
- N**
- NameTrans directive 59
  - Netscape Enterprise Server (NES)
    - configuration overview 59
    - see also* Sun ONE Web Server
- O**
- obj.conf 59, 62
  - ObjectType directives 59
  - ODBC Socket, connecting to 50
  - OpenSSL 68
  - Oracle, connecting to 51
  - OS/390, connecting to 41
- P**
- password
    - ColdFusion MX Administrator 70
    - RDS 70
  - port, JWS 76
  - Post Office Protocol (POP) mail server 23
- R**
- rck2 utility
    - command options 140
    - K2 Server documents, searching 140
    - rck2.exe, location 140
    - syntax 140
  - rcvdk utility
    - collections, attaching to 144
    - collections, searching 144
    - fields, displaying multiple 147
    - results, viewing 145
    - using 143
  - RDS Password page, ColdFusion MX
    - Administrator 34
  - RDS password, security 70
  - replication buddy 82
  - root security context 72

## S

- sandbox
  - adding 72
  - configuring 72
  - security, using 70
- Sandbox Security page, ColdFusion MX Administrator 35
- scriptpath property 63
- scriptsrc attribute 65
- security
  - about 69
  - ColdFusion 72
  - data sources 72
  - directories and permissions, about 71
  - files and directories 72
  - IP/Port 73
  - RDS password 70
  - resources, restricting 71
  - root security context 72
  - sandbox, adding 72
  - sandbox, configuring 72
  - sandbox, using 70
- Security section, ColdFusion MX Administrator 34, 69
- Server Settings section, ColdFusion MX Administrator 18
- serverstore property 63
- session replication
  - enabling in JMC 82
  - overview 81
- session variables
  - failover 81
  - Memory Variables page 22
- Settings Summary page, ColdFusion MX Administrator 26
- shell scripts 62
- signed certificate 68
- Simple Mail Transfer Protocol (SMTP) mail server 23
- SSL 68
- ssl property 63
- Sun ONE Web Server
  - application isolation configuration 81
  - configuration overview 59
  - multihoming 67
- Sybase, connecting to 54

## U

- Unicode, Microsoft Access 46

## V

- VDK error codes 155
- verbose property 63
- Verity Collections page, ColdFusion MX Administrator 26
- Verity K2 Server page, ColdFusion MX Administrator 27
- Verity Spider
  - DNS lookups 103
  - flow control 102
  - multithreading 103
  - overview 101
  - performance 102
  - proxy handling 103
  - restart capability 102
  - state maintenance 102
  - syntax 104
  - vspider executable 103
  - web standards support 102
- Verity Spider MIME types
  - file system indexing and 130
  - known types for file system indexing 131
  - multiple parameter values 129
  - syntax restrictions 129
  - unknown types, indexing 130
  - web crawling and 129
  - wildcards, using 129
- Verity Spider options
  - content 119
  - core 106
  - locale 126
  - logging 127
  - maintenance 128
  - networking 112
  - path and URL 115
  - processing 107
- Verity utilities
  - overview 85
  - relationships with CFML 85
- virtual hosts
  - application isolation 79
  - multihoming 65
- virtual mappings, built-in web server 58
- virtual servers, Sun ONE Web Server 67

## **W**

- web root, built-in web server 58
- Web Server Configuration Tool
  - advanced configurations 65
  - batch files 62
  - cluster 82
  - command-line interface 60
  - configuration files 62
  - GUI mode 60
  - shell scripts 62
  - SSL 68
  - using 59
- web servers
  - built-in web server 58
  - configuring 59
  - configuring for load balancing and failover 81
  - external 59
  - overview 57
- Web Services page, ColdFusion MX Administrator 27
- wildcard, IIS 6 59