Mercury IT Governance Center™

Document Management Guide and Reference

Version: 7.0

This manual, and the accompanying software and other documentation, is protected by U.S. and international copyright laws, and may be used only in accordance with the accompanying license agreement. Features of the software, and of other products and services of Mercury Interactive Corporation, may be covered by one or more of the following patents: United States: 5,511,185; 5,657,438; 5,701,139; 5,870,559; 5,958,008; 5,974,572; 6,137,782; 6,138,157; 6,144,962; 6,205,122; 6,237,006; 6,341,310; 6,360,332, 6,449,739; 6,470,383; 6,477,483; 6,549,944; 6,560,564; 6,564,342; 6,587,969; 6,631,408; 6,631,411; 6,633,912; 6,694,288; 6,738,813; 6,738,933; 6,754,701; 6,792,460 and 6,810,494. Australia: 763468 and 762554. Other patents pending. All rights reserved.

U.S. GOVERNMENT RESTRICTED RIGHTS. This Software Documentation is a "commercial item" as defined at 48 C.F.R. 2.101 (October 1995). In accordance with 48 C.F.R. 12.212 (October 1995), 48 C.F.R. 27.401 through 27.404 and 52.227-14 (June 1987, as amended) and 48 C.F.R. 227.7201 through 227.7204 (June 1995), and any similar provisions in the supplements to Title 48 of the C.F.R. (the "Federal Acquisition Regulation") of other entities of the U.S. Government, as applicable, all U.S. Government users acquire and may use this Documentation only in accordance with the restricted rights set forth in the license agreement applicable to the Computer Software to which this Documentation relates.

Mercury, Mercury Interactive, the Mercury logo, the Mercury Interactive logo, LoadRunner, WinRunner, SiteScope and TestDirector are trademarks of Mercury Interactive Corporation and may be registered in certain jurisdictions. The absence of a trademark from this list does not constitute a waiver of Mercury's intellectual property rights concerning that trademark.

All other company, brand and product names may be trademarks or registered trademarks of their respective holders. Mercury disclaims any responsibility for specifying which marks are owned by which companies or which organizations.

Mercury provides links to external third-party Web sites to help you find supplemental information. The content and availability may change without notice. Mercury makes no representations or warranties whatsoever as to the content or availability.

Mercury
379 North Whisman Road
Mountain View, CA 94043
http://www.mercury.com

© 1997–2006 Mercury Interactive Corporation. All rights reserved.

If you have any comments or suggestions regarding this document, please send email to documentation@mercury.com.

Publication Number: ITGDocMgmt7.0/02

Table of Contents

List of Figures

List of Tables

Chapter

Getting Started with Mercury Document Management

In This Chapter:

Introduction to Mercury Document Management

The optional document management module in Mercury IT Governance Center™ gives you more control over document search and storage than does the standard Mercury IT Governance Center application. The Mercury document management technology is a modified version of the EMC Documentum product. With it, you can track, index, and search multiple versions of supporting documents attached to Mercury IT Governance Center entities in Mercury Demand Management™, Mercury Portfolio Management™, Mercury Program Management™, and Mercury Project Management™.


Note

If you have purchased the full Documentum product from EMC Documentum and installed it on your system, you cannot use it with Mercury IT Governance Center. Only the modified version that Mercury provides works for document management in Mercury IT Governance Center.

Documents managed in these environments are always directly related to a Mercury IT Governance Center entity—for example, a request (including portfolios, proposals, projects, and assets), project plan, or program—through the standard attachment field and document references functionality in Mercury IT Governance Center.

The document management module includes the following components, all of which are available only from Mercury for exclusive use with Mercury IT Governance Center:

- Integrated Content Server
- Documentum Foundation Classes (DFC)
- Full-text indexing software (index agent and index server)


Note

The Mercury document management module is supported in the Oracle database environment only.

For more information about the system requirements for document management, see the Mercury document *System Requirements and Compatibility Matrix.*

Functional Capabilities

Key functional capabilities of the Mercury document management module include the following:

- Searching the contents of documents associated with groups of entities to locate a particular entity in the group
- Check-in and check-out functionality (including check-out override capabilities)
- Version control of attached documents
- History of document versions is maintained
- Users can retrieve older document versions
- Full-text indexing

For information about how to use the Mercury document management module, see Chapter 7, *What Document Management Users Need to Know,* on page 155.

Availability and Licensing

Mercury document management is free to customers who purchase a Mercury IT Governance Center application such as Mercury Demand Management or Mercury Project Management. The license key supplied at the time of purchase gives you access to both the application and the document management code from the Mercury IT Governance Download Center (itg.merc-int.com/support/download/login.jsp).

Use-Case Scenario

Suppose that a large national insurance company, XYZ Corporation, has just installed Mercury IT Governance Center and the Mercury document management module. A business analyst working with the IT organization at XYZ is preparing a proposal for new software to be used by insurance investigators across the corporation. Before he can submit the proposal for review, the analyst must complete a business case document.

The Mercury IT Governance Center workflow associated with the proposal enforces this requirement. If the business case document is not attached to the proposal, the analyst cannot move to the next workflow step.

As the analyst checks out the business case document, and later checks in new drafts, document versions are created and stored. If necessary, users can access earlier versions of the business case document.

Documents managed using the Mercury document management module follow the same security rules (including field-level security rules) that apply to all Mercury IT Governance Center entities. This means that an application user sees only information that applies to his current roles and tasks.

The business analyst can use the document management module to search for Mercury entities based on the contents and metadata of documents attached to the entities. The analyst can use key words to locate relevant proposals, assets, requests, and other entities related to a business case, regardless of where the details about the entities reside.

Document Management Enhancements to the Standard Interface

The following figures illustrate the difference between standard attachment functionality and document management capabilities.

Figure 1-1. Search Requests page in Mercury IT Governance Center

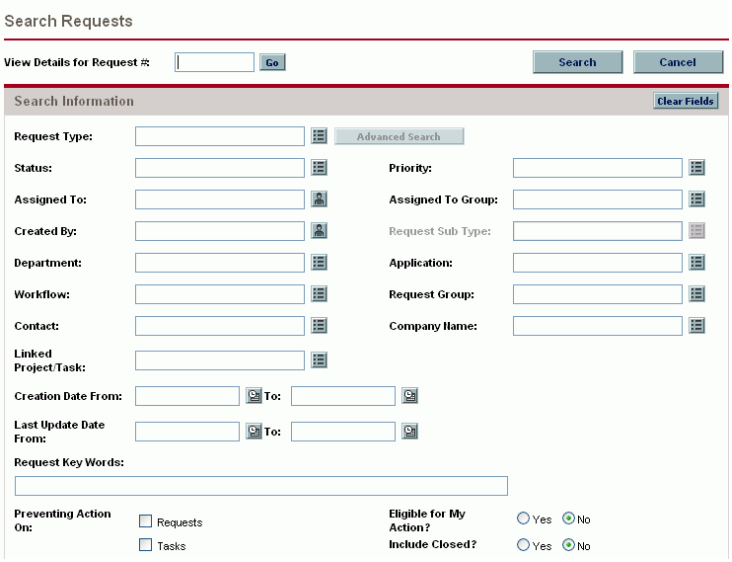


Figure 1-1 shows the Search Requests page in a system without document management. You can enter request key words to use to search the contents of

request **Notes** and **Description** fields, but this search will not include the contents of documents attached to requests.

Figure 1-2. Searching a request with document management installed

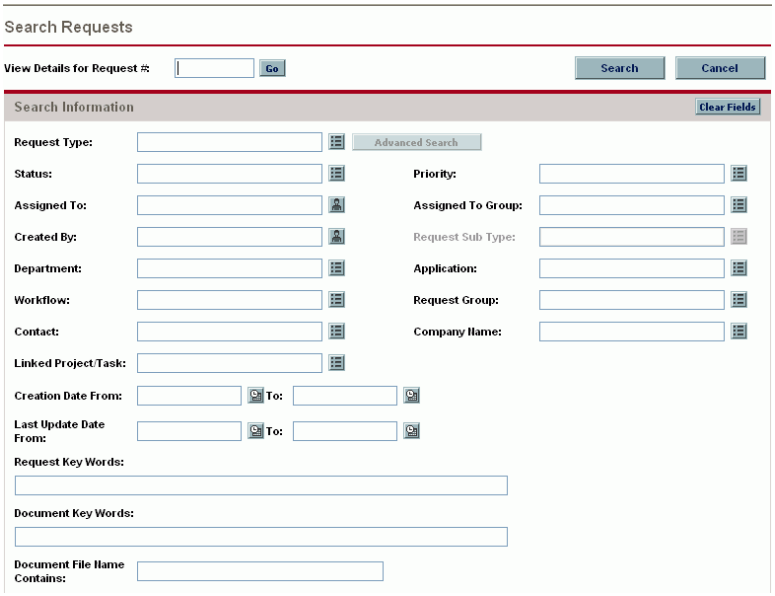


Figure 1-2 shows the Search Requests page in a system with document management. You can still use the **Request Key Words** field to search request notes and descriptions.

To include the contents of attached documents in your search, type the text to search for in the **Document Key Words** field. (Words that you type in the **Document Key Words** field are used to search the contents of documents attached to requests that meet the other filter criteria.) To search for documents with names that match known text, use the **Document File Name Contains** field to search the filenames of attached documents.

The Mercury document management module affects the following pages (and entities) in the Mercury IT Governance Center standard (HTML) interface.

Pages accessed through the following **Search** menu items contain the additional search fields:

- Initiative Requests
- Packages
- Programs
- Project Issues
- Project Resource Request

- Project Risks
- Project Scope Changes
- Projects
- Requests
- Tasks

With document management, the Printable Request/Request Detail report and Project Detail report display additional information about attached documents.

If You Already Use Document Management in Mercury IT Governance Center

The document management module in this release of Mercury IT Governance Center uses Content Server 5.3. If you enabled the document management module in Mercury IT Governance Center, and you want to continue to use document management, you can do one of the following:

- Install Documentum Foundation Classes version 5.3 on top of all DFC 5.2.5 installations, and continue to use Content Server 5.2.5.
- Install Documentum Foundation Classes version 5.3 on Mercury IT Governance Server machines that are not Content Server hosts, upgrade to Content Server 5.3 SP2 on your Content Server hosts, and install the full-text indexing software.


Note

For both of these upgrade scenarios, Mercury recommends that you install DFC 5.3 (on Mercury IT Governance Server machines that are not Content Server hosts) *before* you install Mercury IT Governance Center 7.0.

For detailed instructions on how to install Documentum Foundation Classes as a separate installation independent of Content Server installation, see Chapter 4, *Installing Documentum Foundation Classes,* on page 91.

For information how to upgrade from document management based on EMC Documentum 5.2.5 SP1 to document management based on EMC Documentum 5.3 SP2, see Chapter 6, *Upgrading the Document Management Module,* on page 119.

Terminology Changes

With the introduction of Content Server 5.3, the following two terms have changed:

- Docbases are now called *repositories,* except where the term *docbase* is used in the name of an object or attribute (for example, *docbase config object*).
- DocBrokers are now called *connection brokers.*

How Document Management Affects Performance

This section addresses the issue of how document management affects Mercury IT Governance Center performance.

Implementing document management as part of Mercury IT Governance Center affects the following functional areas:

- Attaching a document to a Mercury IT Governance Center entity (such as a request or project), either through attachment fields, or through the **References** section available for some entities
- Viewing a document that is attached to a Mercury IT Governance Center entity

Without the document management module, documents attached to Mercury entities are uploaded and stored on the Mercury IT Governance Server file system. With document management, attached documents are uploaded to the Mercury IT Governance Server, and then stored in a Documentum repository.

In the typical configuration, the Mercury IT Governance Server and the Content Server are located on the same local network. This ensures that any communication between the two servers enjoys fast, uninterrupted network access.

The overhead of storing and retrieving attached documents to and from the Content Server adds minimal overhead to client response time. With or without the document management module, the key factor that determines user response time is the quality of the wide-area network (WAN) between the client machine and the Mercury IT Governance Server. If it takes five minutes to attach a 2 MB document to a Mercury IT Governance Center request without document management, then the process will still take five minutes

with document management in place. (This statement assumes that the Mercury IT Governance Server and Content Server are on the same local network.)

Attaching Documents

With document management enabled, attaching a document to a Mercury IT Governance Center entity is a two-step process. First, the attached document is uploaded to the Mercury IT Governance Server for temporary storage. Next, the user saves the entity (for example, a request) to which the document is attached, and the document is copied to the content server (and the temporary copy removed).

During the first step, as the user attaches a document, the time required to upload that document with document management in place is the same as the time required if document management is not enabled. This is the key performance consideration for client users. The network quality between the client and the Mercury IT Governance Server directly affects the time it takes to upload documents, independent of whether document management is enabled.

For the second step, when the user saves the entity, the save time is increased by 50 to 100 percent over the save time for the same entity without an attached document. The time it takes to save an entity increases for each additional document attached (or for each new version of an existing document uploaded.)

Example:

Suppose that you have Mercury IT Governance Center running on a basic single 2.4 GHz CPU, 1 GB RAM server, with Documentum running on a single 3 GHz CPU, 2 GB RAM server. A Mercury IT Governance Center client, a Mercury IT Governance Server, and Content Server are all on the same LAN.

Attaching a 2 MB document to a request takes approximately five (5) seconds with or without document management in place. The time required to attach a document increases as document size increases. A 10 MB document takes closer to 15 seconds to attach. Again, the time required is independent of whether you use document management.

The estimated time required to save the request after attaching either the 2 MB or the 10 MB document to it is:

- Three seconds without document management in place
- Seven seconds with document management in place

Note

If you open a request that has documents attached, and save that request without attaching additional documents (or new versions of existing documents), then the request save time reverts to the original three seconds.

Overview of Document Management Deployment

A Content Server environment consists of a specific combination of operating system and database, and, optimally, an index server host machine for the full-text index server.

To deploy the document management module for the first time, you perform the following tasks either before or after you install or upgrade Mercury IT Governance Center:

- Install Content Server and configure it to work with Mercury IT Governance Center.

  Content Server installation and setup for use with Mercury IT Governance Center can take more than half a day. The time required for setup depends on server performance, the quality of the network connecting servers, and, if you are upgrading from earlier document management functionality, the number of attachments you plan to migrate.

  For information about how to install and configure Content Server, see Chapter 2, *Installing and Configuring Content Server,* on page 25.

- Install full-text indexing, and configure the index agent.

  For information on full-text indexing and how to install it, see *Installing Content Server Full-Text Indexing Software* on page 65.

- Install Documentum Foundation Classes (DFC) on any Mercury IT Governance Server machine that is not a Content Server host.

  For detailed instructions on how to install Documentum Foundation Classes as a separate installation independent of Content Server

installation, see Chapter 4, *Installing Documentum Foundation Classes,* on page 91.

Although the Mercury IT Governance Server uses only a subset of the features that the standard Content Server application provides, Content Server installation includes more than what is required for Mercury document management. For example, the installation procedure installs Apache Tomcat and an older version of the SDK that Mercury IT Governance Server does not support. Although this does not affect the Mercury IT Governance Server, you may notice some information displayed during installation that does not apply to Mercury IT Governance Center document management.

Installation Sequence

If you are installing the document management components for the first time, you can perform the installation either before or after you install or upgrade Mercury IT Governance Center.

Install Documentum 5.3 SP2 products in the following order:

1. On the Content Server host machine, install Content Server and configure a repository.

   For information about how to install and configure Content Server, see Chapter 2, *Installing and Configuring Content Server,* on page 25.

   Documentum Foundation Classes is automatically installed with Content Server.

2. On Mercury IT Governance Servers on which Content Server is not installed, install DFC.

   For detailed instructions on how to install Documentum Foundation Classes as a separate installation independent of Content Server installation, see Chapter 4, *Installing Documentum Foundation Classes,* on page 91.

3. Install the index server and index agent.

   For information about full-text indexing and the steps you perform to install it on Windows, UNIX, or Linux systems, see Chapter 3, *Installing Content Server Full-Text Indexing Software,* on page 65.

   For instructions on how to enable the Mercury Document Management Module after you install the required components, see Chapter 5, *Enabling Document Management,* on page 102.

For information about how to use the Mercury document management module, see Chapter 7, *What Document Management Users Need to Know,* on page 155.

Prerequisite Documents

This section describes the Mercury and Documentum guides and reference documents required to install and configure the Mercury document management module. Mercury provides relevant guides and manuals from EMC Documentum when you purchase Mercury IT Governance Center.

Mercury Documents

The following Mercury documents are required for the document management module installation:

- *Configuring the Standard Interface*
- *System Administration Guide and Reference*
- *System Requirements and Compatibility Matrix*

EMC Documentum Guides

To supplement the information provided in this document, Mercury provides a set of guides from EMC Documentum that contains complete details on how to install and administer the Documentum components of the document management module.

You can access these guides from the Mercury IT Governance Download Center (itg.merc-int.com/support/download/login.jsp). For information on how, see *Getting Documentation from the Mercury IT Governance Download Center* on page 23.


Note

The EMC Documentum documents listed here provide information about the full Documentum system, of which, only a subset is used for document management in Mercury IT Governance Center.

The following EMC Documentum guides are available:

- *Content Server Installation Guide* provides information about installing Content Server, which is the cornerstone of the Mercury document management module. This manual contains the information you need to

install or upgrade Content Server on Windows, Solaris, HP-UX, AIX, or Linux systems with the Oracle relational database management systems. It describes decisions you must make and requirements that must be met before you install the server. It also provides step-by-step instructions on how to install and upgrade Content Server.

- *Documentum Foundation Classes Installation Guide* contains information and instructions on how to install Documentum Foundation Classes (DFC) 5.3 or upgrade from DFC 5.2.5 to 5.3.
- *Content Server Full-Text Indexing Installation Guide* contains information and instructions you need to install or upgrade the full-text indexing system used with Content Server on Windows, UNIX, or Linux. This document addresses the decisions you must make and requirements that your system must meet before you install the full-text indexing software. It also provides instructions on how to install and upgrade the software in several different configurations. Use this guide in conjunction with the *Content Server Installation Guide.*
- *Documentum 5.3.x System Migration Guide* presents a "whole-system" overview of how to migrate your 5.2.5 Documentum system to Documentum 5.3. This book is designed to help you plan your migration. It provides recommended best practices and addresses cross-product and cross-platform migration issues.
- *Content Server DQL Reference Manual* is the reference manual for Documentum's Document Query Language, supported by Content Server.
- *Content Server Administrator's Guide* contains the following:
  - Information, instructions, and procedures for the basic system administration tasks for a Content Server installation.
  - An overview of the system configuration and guidelines for making configuration decisions.
  - Information on how to configure repositories, Content Servers, clients, and sessions.
  - Information about how to perform routine maintenance.
  - Information about connection brokers, full-text indexing administration, managing the content storage area, and repository security.

Related Documents

This section lists supporting documentation that is useful to for understanding document management in Mercury IT Governance Center.

Mercury Documents

Supplementary documents from Mercury include:

- *Security Model Guide and Reference*

Documentum Documents

The following EMC Documentum document contains information that is helpful to understanding and using the Mercury document management module:

- *Content Server DQL Reference Manual* contains reference information about Document Query Language, or DQL. DQL is a superset of the ANSI SQL that provides a single, unified query language for all the objects that Content Server manages. You can use the IDQL utility to enter ad hoc DQL queries against a repository.

For more information about Documentum, go to the dm_developer Web site (dmdeveloper.com).

Getting Documentation from the Mercury IT Governance Download Center

At the Mercury IT Governance Download Center, you have access to the same PDF files that are available through the standard interface after Mercury IT Governance Center installation, and to documents that are only available at that location. The EMC Documentum guides are available only through the download center.

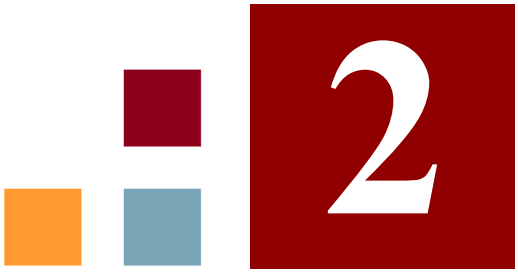
To log in to the Mercury IT Governance Download Center, you must have both Mercury support and the download center accounts.

To access EMC Documentum documentation from the Mercury IT Governance Download Center:

1. Log on to the Mercury IT Governance Download Center (itg.merc-int.com/support/download/login.jsp).
2. Read the software end user license agreement, click **I accept these terms,** and then click **Submit.**
3. In the left pane, under **Mercury IT Governance Center,** click **Tools.**

   In the right pane, the EMC Documentum products that are available for download are listed.
4. To the right of the product listing, in the **Documentation** column, click **Document Set.**

Chapter



Installing and Configuring Content Server

In This Chapter:

Overview of Installation and Configuration

A Content Server environment consists of a specific combination of operating system and database, and, optimally, an index server host machine for the full-text index server.

Deploying the Mercury IT Governance Center document management module for the first time involves installing Content Server, and then configuring the server with Mercury IT Governance Center.

Content Server installation and setup for use with Mercury IT Governance Center can take more than half a day. The time required for setup depends on server performance, the quality of the network connecting servers, and, if you are upgrading from earlier document management functionality, the number of attachments you plan to migrate.


Note

Keep in mind that you can set up document management to work with Mercury IT Governance Center *after* you deploy Mercury IT Governance Center. Mercury IT Governance Center runs normally without the document management functionality, so you can perform the integration with the Content Server at a later, more convenient time, if necessary.

Although the Mercury IT Governance Server uses only a subset of the features that the standard Content Server application provides, Content Server installation includes more than what is required for Mercury document management. For example, the installation procedure installs Apache Tomcat and an older version of the SDK that Mercury IT Governance Server does not support. Although this does not affect the Mercury IT Governance Server, you may notice some information displayed during installation that does not apply to Mercury document management.

This section provides information about Content Server concepts and the high-level steps required to install and configure the document management module.

Content Server Components

To understand Content Server functionality, it is useful to know something about its components and structure. This section contains information on the connection broker application, the document repositories, and the Documentum Foundation Classes, and how these components interact with one another and with Mercury IT Governance Center.

Connection Broker

The connection broker is part of Content Server, and is created during Content Server installation. It runs in the background and provides connection information to client applications. Mercury IT Governance Server is a client application to the connection broker.

The connection broker listens for incoming requests on a port you specify. During Content Server installation, you provide the machine hostname and port number. (The default port number is 1489. Although you can specify any unused port on the machine, Mercury recommends that you keep the default setting.) The Mercury IT Governance Server requires this information to communicate with Content Server.

Repository

Managed documents are stored on Content Server in a repository. This virtual storehouse consists of content and index files and object metadata (properties that describe file characteristics such as creation date, author, version number, and so on).

The repository has a file system component and a database component. For this reason, you must ensure that the Content Server host has access to an Oracle database.

File content and indexes are stored on the file system, while the object metadata are stored in an Oracle database. Mercury IT Governance Center stores documents in the repository that you specify during Content Server installation.

A single connection broker can route requests to multiple repositories. You can create a separate repository for each IT Governance Server instance, and then point each IT Governance Server instance to its own repository. This separates the storage areas (physical disk and database schema) and gives you more control over your hardware. You can create multiple repositories on the same Content Server.

Product Communication

Mercury IT Governance Center communicates with the document management module through Documentum Foundation Classes (DFC). DFC is an Application Programming Interface (API) library that is part of the Mercury document management module. For information about DFC, including how to install it, see Chapter 4, *Installing Documentum Foundation Classes,* on page 91.

About Creating a Repository

The repository is where the documents and their metadata are stored. Every repository you create requires:

- A service listed in the `/etc/services` directory.
- A unique database schema created in an Oracle database. (Multiple repositories cannot share a schema.)

  Mercury recommends that you use the same string for the user name of the Documentum installation owner and for the name of the Oracle database schema created for the repository.
- As you create a repository, you are prompted to specify an ID for it.

  Enter a number between 0 and 16777215 that is unique to the repository.
- The server configuration utility parses your `tnsnames.ora` file to list database SID information, and prompts you to select the database to use. If the database that contains your schema is not listed, the problem might be in the `tnsnames.ora` file.

  After the information is collected, the installation program creates and configures the repository.

  To find log information that can help you diagnose any errors that occur, look in the following directory:

  ```
  $DM_HOME/install/setup/*.log
  ```
- If you add a repository to the existing Content Server (or modify an existing repository associated with Content Server), then you must stop the connection broker, and restart it before you can access the new (or modified) repository. For information about how to stop and start the connection broker, see *Starting and Stopping the Connection Broker and Repository* on page 61.

- If, after you create a repository, you cannot start the connection broker or start and connect to the repository, try to start and connect manually.

  For information about the source of the problem, see the error information reported to the console.

  Before the server configuration utility can create a repository, it tries to start the connection broker. If it cannot start the connection broker, it cannot continue.

Documentum Foundation Classes

Mercury IT Governance Center communicates with Content Server through an application programming interface (API) library called *Documentum Foundation Classes,* or *DFC.*

To function correctly, a Mercury IT Governance Server must be able to locate DFC. When you run the `kConfig.sh` script to configure document management (see *Configuring Document Management in Mercury IT Governance Center* on page 102), a properties file named `dfc.properties` is added to the `<ITG_Home>/server/<server_name>/conf` directory. This file includes DFC installation directory information.

DFC uses `*.dll` native library files on Windows, and UNIX native library files such as `*.so` on Linux. To communicate with Content Server, the Mercury IT Governance Server startup script `(kStart.sh)` and configuration script `(kConfig.sh)` must have access to these library files.

DFC is automatically installed with Content Server. If you install Content Server on the same machine as the Mercury IT Governance Server, there is no need to perform a separate DFC installation. However, if Content Server and the Mercury IT Governance Server are on separate machines, you must install DFC separately on the Mercury IT Governance Server.

For information about how to install DFC, see Chapter 4, *Installing Documentum Foundation Classes,* on page 91.

Installation Sequence

Install Documentum 5.3 SP 2 products in the following order:

1. On the Content Server host machine, install Content Server and configure a repository.

   Dependent products such as DFC are automatically installed with Content Server.

2. On Mercury IT Governance Servers on which Content Server is not installed, install DFC.

3. Install the index server and index agent.

   For information about how to install the full-text indexing software, see *Installing Content Server Full-Text Indexing Software* on page 65.

High-Level Steps for Installing and Configuring the Document Management Module

The high-level steps used to install and configure the Mercury document management module are as follows:

1. Check the Mercury document *System Requirements and Compatibility Matrix* to make sure that your system meets the minimum requirements for document management installation and setup.

2. Install Mercury IT Governance Center, as described in the *System Administration Guide and Reference.*

3. Install the Oracle client software on the machine that is to host Content Server.

4. Read all information related to Content Server installation.

   For information on what to read before you install and configure Content Server, see *Prerequisite Documents* on page 21 and *Related Documents* on page 23.

5. Install and configure the document management module (Content Server and full-text indexing software).

   You can install the module on the machine running the Mercury IT Governance Server, or on a different machine. For information about how to set up Content Server on Windows, see *Installing Content Server on Windows Systems* on page 35. For information about how to set up the

server on UNIX or Linux, see *Installing Content Server on a UNIX or Linux System* on page 45. For more information, and to answer any advanced questions about installation, see the EMC Documentum documents *Content Server Installation Guide* and *Content Server Administrator's Guide,* which are described in *Prerequisite Documents* on page 21.

6. If you installed Content Server and the Mercury IT Governance Server on separate machines, install DFC on the Mercury IT Governance Server.

   For information about how to install DFC, see Chapter 4, *Installing Documentum Foundation Classes,* on page 91.

7. Although Mercury IT Governance Center does not use the email notification feature in Documentum, you must specify a value in the **SMTP** field during Content Server installation.

You can specify an invalid SMTP server, but you must complete the field.

8. Configure the document management components to work with Mercury IT Governance Center.

   This step is described in *Configuring Document Management in Mercury IT Governance Center* on page 102. (Briefly, browse to the `dfc.properties` file to the `<ITG_Home>/server/<server_name>/conf` directory on the Mercury IT Governance Server, and then run the `<ITG_Home>/bin/kConfig.sh` script. This establishes the communication between the Mercury IT Governance Server and Content Server.)

Perform this configuration separately for every server in a Mercury IT Governance Server cluster.

9. Test the connection between Mercury IT Governance Server and Content Server.

10. Test the document management functionality in Mercury IT Governance Center.

    For example, add a document attachment to a request, modify the document, and then check to make sure that two versions of the document exist in the system. Also, check to make sure that key words added to the document produce the search results you expect.

For more information about using document management in Mercury IT Governance Center, see Chapter 7, *What Document Management Users Need to Know,* on page 155.

About Content Server Configuration

Content Server configuration occurs in two primary locations. The first location is in the flat file named `server.ini`, which contains information that Content Server uses at startup (repository name, connectivity, password, owner; connection broker connectivity; other parameters such as the number of maximum concurrent sessions, and so on).

The `server.ini` file is similar to the `server.conf` file in Mercury IT Governance Center. On Windows systems, the `server.ini` file resides in the `%DOCUMENTUM%/dba/config/<repository>` directory. On UNIX and Linux machines, the file is located in `$DOCUMENTUM/dba/config/<repository>`. After you update the `server.ini` file, restart Content Server to apply your changes.

The second primary Content Server configuration location is in the repository in the server configuration object named `dm_server_config`. Each repository is associated with a corresponding server config object. The following example DQL query returns one row for each repository on Content Server.


Note

The IDQL utility is an interactive tool that lets you enter ad hoc DQL queries against a repository. IDQL is also a useful as a tool for testing and other tasks that support an application or installation because it allows you to run scripts and batch files.

IDQL is included and installed with Content Server. It is found in `$DM_HOME/bin` on Linux and in `%DM_HOME%\bin` on Windows.

For more information about the IDQL utility, see the EMC Documentum document *Content Server Administrator's Guide.*

To run a DQL query against a repository, you must first start an IDQL session, as follows:

1. At the command prompt, go to the `$DM_HOME/bin` directory.
2. Run the IDQL utility executable, `idql32.exe`.
3. At the prompt, type the repository name.
4. Press **`Enter`**.
5. At the prompt, type the account name for a user with at least System Administrator privileges in the repository.
6. Press **`Enter`**.

7. At the prompt, type the password for the user account.
8. Press **Enter.**

   An interactive document query interface session starts. You can type your query at the prompt.

   Example query:

```
1> select "object_name"
2> from "dm_server_config"
3> go
```

After you run this statement, you can interact with a particular repository server configuration by updating that repository server config object. For example, to see the configuration parameters for a repository named "ITGdocs," run the following DQL statement:

```
1> select * from "dm_server_config"
2> where "object_name" = 'ITGdocs'
3> go
```

Note

To close an IDQL session, enter the `quit` command at the IDQL prompt.

You can update configuration information in the repository server configuration object while Content Server is running, without having to restart it.

Preparing the Database for Content Server Installation

Every repository must have a correctly configured Oracle database. The general database requirements are as follows:

- If you install the database on the Content Server host with a Linux system, verify that the system path includes the directory for the database.
- If you install the database on the Content Server host with a Windows operating system, ensure that the database service is set to start automatically. Server installation sometimes requires a restart of the computer. After the restart, installation does not proceed correctly unless the database starts automatically.

- If you install your Oracle database on a machine other than the Content Server host, verify the following:
  - The remote machine has an operating system that the Oracle version supports.
  - You can connect to the database client from the system on which you plan to install Content Server.
- If you are creating a new database or tablespace for Content Server, designate UTF-8 as the code page.
- Install only the English version of a database. Content Server does not support localized versions.
- If you create a remote Content Server for a distributed content environment, the `server.ini` file from the primary Content Server host is copied from the primary host to the remote host. To ensure that the `database_conn` key on the primary Content Server host is valid on the remote hosts, ensure that the values used on the primary and remote hosts for database connectivity are identical.
- You must install the database client on remote Content Server hosts. The remote Content Server configuration program must connect to the database to create the server config object, acs config object, file store storage object, and location objects for the remote server.
- Content Server uses the repository owner account to connect to the database. The sections on the repository owner in the *Content Server Installation Guide* provide more information.The server runs as the installation owner, but a separate account must exist to give the server access to the database tables underlying the repository. Each repository must have a unique repository owner and each repository owner must have a unique database account.

  You can create the repository owner account and the database or tablespace that the repository uses before you install Content Server, or the server installation software can create the account and database or tablespace. Before you begin installation, decide whether to create the account yourself or allow the installation program to create the account. The account must have the CONNECT and RESOURCE privileges to do the following:
  - Connect to the database
  - Create tables, views, and indexes in the database

- Insert records (rows) into the tables
- Drop tables, views, and indexes

- If you choose to have the Content Server installation software create the repository owner account in the database for you, you must have the database administrator user name and password.

The Oracle database must meet the following requirements:

- Create the repository database with the UTF-8 code page. On Oracle 9i, when you create the database and choose the database character set (code page), select Unicode (AL32UTF8). If you choose to migrate an existing database to UTF-8, use AL32UTF8.

Installing Content Server on Windows Systems

This section provides information about how to prepare to install Content Server in a Windows environment, and how to perform the installation.

You do not need a special license to install the Mercury version of Content Server.

For more installation information, see the Documentum documents described in *Prerequisite Documents* on page 21 and *Related Documents* on page 23.

Oracle Requirements

The Oracle database must meet the following requirements:

- Create the repository database with the UTF-8 code page.

  On Oracle 9i, when you create the database and choose the database character set (code page), select Unicode (AL32UTF8).

- Ensure that SQL*Plus is installed on the Content Server host. SQL*Plus is needed for creating tablespaces and the database user (repository owner) account.

- Ensure that client side of SQL*Net is set up correctly.

  The database aliases must be in the `tnsnames.ora` file, and the `tnsnames.ora` file must be configured on the Content Server host. Use the

SQL*Net configure tool to alter values in the `tnsnames.ora` file. You can edit the `tsnames.ora` file using the SQL*Net icon in the Oracle group.

- Install the Oracle client software on the Content Server host. Use the Oracle 9 or 10 client software.
- The Oracle Listener process must be running on the machine where the Oracle database resides.
- Verify that you can connect to the Oracle database by using SQL*Plus from the system where you intend to install Content Server.

Preparing to Install Content Server on Windows Systems

Before you install Content Server on a Windows host, you must perform some configuration steps and set up the required user accounts on the system.

The configuration tasks are as follows:

1. Manually add the following value (path to Java) to the PATH system environment variable:

   <DFC_install_location>\<java_version>

   where *<java_version>* is the Java Runtime Environment (JRE) version supported for the Content Server version you plan to install.

Note

The DFC installation location is set during server installation. It is typically `C:\Program Files\Documentum`.

2. Install the Oracle client software on the machine on which you plan to install Content Server, and configure the Oracle client for connectivity to the Oracle database to be used to store the repository schema.
3. Use the `tnsping` command and SQL*Plus to verify that the Oracle client software is correctly installed.

Note

For information on how to run the `tsnping` command, see oracleutilities.com/OSUtil/ping.html. For information on how to use SQL*Plus, go to comp.nus.edu.sg/~ooibc/courses/sql/sqlplus.htm.

4. Ensure that `sqlnet.ora` and `tnsnames`.ora are configured correctly.

Note

To specify connections, use fully-qualified names in `tnsnames.ora` in the following syntax:

Example: sqlplus system/*<password>*@*<SID>*

5. Check to make sure that the machine on which you plan to install Content Server has access to a valid SMTP server for email notifications.

Note

Although Mercury IT Governance Center does not use this notification mechanism, you cannot install Content Server without it. If a valid SMTP server host name is not available during installation, supply an invalid host name so that the installation can finish. Do not leave the field blank.

6. Stop all nonessential services and quit all nonessential programs.
7. To download the Content Server bundle:
   a. Log on to the Mercury IT Governance Download Center (itg.merc-int.com/support/download/login.jsp).
   b. In the left pane, select **Mercury IT Governance Center > Tools.**

      Wait for the page to load.
   c. In the right pane, in the **Download** column, click **DCTM_ITG-5.3_SP2-Windows.zip.**
   d. Save the `DCTM_5.3_SP2_Win.zip` file to a temporary directory from which you will run the installation.

Note

In this manual, this temporary directory is referred to as `%DM_INSTALL%`.

8. Use an unzip utility such as Unzip or use the Java `jar xvf` command to unzip the installation bundle and extract the installation files.
9. Select the installation location (referred to in this document as the `C:\Documentum\` directory or `%DM_HOME%`).

Warning

The directory string cannot contain spaces or special characters.

10. Check the regional and language options settings on Windows to make sure that the date format selected displays a four-digit year.
11. If you install a distributed configuration, ensure that all host computers in the configuration are set to the same Universal Time Coordinated (UTC) time.

12. Select or create a Windows user to function as the installation owner, and install Content Server using this owner account.

    Content Server runs under the account of the installation owner. Use this account to perform all Content Server administration.

    The Windows user that you log on as for installation must meet the following requirements:

    - The installation owner account can be a local or domain account, but if it is a domain account, then it must be a member of the local administrator group.
    - The installation owner account must have Full Control permission in the `%DM_HOME%` directory.
    - The installation owner account must not be the same account as the Windows administrator.
    - The installation owner account must have the following rights, which are granted during installation:
        - Act as part of the operating system
        - Create a token object
        - Increase quotas
        - Log on as a service
        - Log on locally
        - Replace a process-level token
    - The user name is restricted to alphanumeric, hyphen (-), and underscore (_) characters.

      The user name you specify when you install Content Server must match this Windows user name including case, even though Windows user accounts are not case-sensitive.
    - The password is restricted to alphanumeric characters, hyphens (-), underscores (_), and periods (.).

13. Choose an Oracle database in which to install the repository. Verify that you can connect to this database using SQL*Plus from the system on which you plan to install Content Server.

    Content Server installer automatically creates the repository schema in this database.

14. Enter the Oracle SYSTEM username and password when prompted.

Installing Content Server on Windows Systems

To install Content Server on a Windows system:

1. Go to the `%DM_INSTALL%\win\5.3` directory, and then double-click `DCTM_5.3_SP2_Win.zip`.
2. Double-click `Content_Server_5.3_SP2_windows_oracle.exe`.

   The Content Server installer program starts and displays the Welcome page.
3. Click **Next.**

   The installer verifies your system requirements and prompts you to enter the installation directory.
4. Type the full path of the directory in which you want to install Content Server (for example, `C:\Documentum`).

   The directory path name cannot include spaces.
5. Click **Next.**

   The installer prompts you to indicate whether you want to install optional components for DFC.
6. Leave both checkboxes unselected and click **Next.**
7. Accept the default DFC installation directory `(C:\Program Files\Documentum)` or type the full path of the directory in which you want to install DFC.

   This directory name can contain spaces.
8. Click **Next.**
9. Accept the default DFC user directory (the `%DM_HOME%` directory) or type the full path of a different directory to use as the DFC user directory.

10. Click **Next.**
11. In the **Primary Connection Broker Host Name** field, type the name of your primary connection broker host computer.

Note

This must be the same machine on which you are installing Content Server.

12. In the **Port Number** field, type the port number for your primary connection broker host computer.

Note

The port that you specify must not be used by any other process. (The default port is 1489.)

Make note of the machine name and port number you enter so that you can provide these later as you integrate the document management module with the Mercury IT Governance Server.

13. Click **Next.**
14. Leave the **Enable Trusted Content Services** checkbox unselected and click **Next.**
15. Leave the **Enable Content Services for EMC Centera** checkbox unselected and click **Next.**

    The installer prompts you to specify port numbers for Apache Tomcat. Content Server uses Tomcat internally to run required Java programs.
16. Accept the default port, or type the number of any unused port above 1024 on which Apache Tomcat can listen for requests.

Note

Mercury IT Governance Center does not make use of this Content Server feature.

17. Click **Next.**
18. Accept the default port, or type the number of any unused port above 1024 to use to stop the Tomcat server.

Note

Mercury IT Governance Center does not make use of this Content Server feature.

19. Click **Next.**

20. On the confirmation page, verify your installation parameters, and then click **Next.**

    The installer prompts you to indicate whether you want to configure the server now or later.

21. Select **Configure server now,** and then click **Next.**

    The installer prompts you to indicate whether you want to restart your computer.

22. Select **Yes.**

    After you restart your machine, the configuration process starts automatically.

23. Proceed to *Configuring Content Server Components on a Windows System* on page 41.

Configuring Content Server Components on a Windows System

To configure Content Server components on a Windows system:

1. Restart your computer and log on as the Content Server installation owner.

   The configuration wizard starts and displays the Welcome page for the Oracle Server Configuration Program.

Note

If the configuration wizard does not start, change to the `%DM_HOME%\product\5.3\install\` directory and run `Server_Configuration_Program.exe`.

2. Click **Next.**

   The configuration program performs a series of system checks, and then prompts you to enter the Content Server installation owner password.

3. Type the Content Server installation owner password.

4. Click **Next.**

5. Leave the **Enable Trusted Content Services** checkbox unselected and click **Next.**

6. Leave the **Enable Content Services for EMC Centera** checkbox unselected and click **Next.**

   The configuration prompts you to choose between the express and custom configuration methods.

If your organization has an external storage solution such as network attached storage (NAS) or storage area network (SAN) in place, do not use the Documentum express configuration procedure. You must use the custom configuration instead.

For information about custom and express methods for configuring Content Server components, see “Express or Custom Docbase Configuration” in the EMC Documentum document *Content Server Installation Guide*.

7. If you are not using an external storage solution such as NAS or SAN, select **Express Configuration.** If you are using an external storage solution, select **Custom Configuration,** and then follow the instructions provided.

The custom configuration wizard may prompt you for additional information that is beyond the scope of this guide. For advanced information on how to perform a custom configuration, see the *Content Server Installation Guide* from EMC Documentum.

   Later, if you want to update or delete a repository or perform another configuration task, you can run the wizard again and select the custom configuration option.

8. Click **Next.**

   The configuration wizard prompts you for repository information.

9. In the **Repository Name** field, type a name for the repository to create.

The repository name can contain up to 32 characters. Make a note of the name so that you can provide it later when you configure Mercury IT Governance Server.

10. In the **Repository ID** field, type a unique ID number between zero (0) and 16777215 for the repository.

11. In the **Repository description** field, you can type a description of the repository (optional).

12. In the **Repository size** list, select one of the following:

    Small: Single table space with an initial data file size of 100 MB.

    Medium: Separate table spaces for data and indexes, with initial data file size of 180 MB and an initial index file size of 180 MB.

    Large: Separate table spaces for data and indexes, with initial data file size of 250 MB and initial index file size of 250 MB.

Note

As it creates a schema, the configuration wizard creates a new tablespace in the database.

13. In the **Authentication Domain** list, select the authentication domain.
14. Next to **Service Startup Type,** click one of the following:
    - To start the repository service automatically at server restart, select **Automatic.**
    - To start the repository service manually at server restart, select **Manual.**

      If other services must start before Content Server can start (for example, the Oracle database that contains the repository, if it resides on the same machine), then specify a manual startup.
15. Click **Next.**

    The configuration wizard prompts you to indicate whether you want to create an Oracle database user account or use an existing account.
16. Select **Create new Oracle user account and tablespaces.**
17. Click **Next.**
18. Supply the following database connection information:
    a. In the **Database Connection String** list, select the database in which you want to install the repository.

       The **Database Connection String** list only displays the names of databases that are correctly configured in the Oracle client `tnsnames.ora` file and accessible from this machine.

    b. In the **Database User Name** field, type the user name for the Oracle schema.

       The default name is the same as the repository name you provided (see step 9 on page 42).

c. In the **Database User Password** field, type the password for the Oracle schema.

d. In the **Confirm User Password** field, retype the password for the Oracle schema.

e. In the **Database Administrator Name** field, type the user name for the SYSTEM account on the Oracle database.

f. In the **Database Administrator Password** field, type the password for the SYSTEM account on the Oracle database.

19. Click **Next.**

    To send email notifications for some system events, Content Server requires an SMTP server.

20. Supply the following SMTP information:

    a. In the **SMTP Server Name** field, type the name of an SMTP server on your network.

    b. In the **Installation Owner's Email Address** field, type the email address of the person you want to receive Content Server email notifications.

Email traffic is minimal. Under normal operating conditions, this account receives no email messages.

21. Click **Next.**

    The server configuration wizard displays a progress bar. Configuration takes several minutes.

22. After configuration finishes, review the information displayed on the summary page, and then click **Next.**

23. On the confirmation page, click **Finish.**

Post-Installation Tasks

After you complete Content Server installation and configuration, do the following:

1. Use the idql32 tool to test the installation.

Note

For information on how to submit a query, see the EMC Documentum document.

2. Restart the machine.
3. Check to make sure that the Mercury IT Governance Server is not running.
4. Run the `kConfig.sh` script.

Installing Content Server on a UNIX or Linux System

This section contains information about installing Content Server in a UNIX or Linux environment. It addresses issues to consider before you install, what tasks to perform before you install and, finally, how to install.

Note

You do not need a special license to install the Mercury version of Content Server.

For supplemental information on Content Server installation, see the EMC Documentum guides described in *Prerequisite Documents* on page 21 and *Related Documents* on page 23.

Installation Considerations

Before you install Content Server on Linux, consider the following:

- Because Content Server is not a Java application, there are version-dependent installers.
- Content Server installation requires root access to the host machine.

Warning

Unlike the Mercury IT Governance Server installation, the Content Server installation and the stand-alone DFC installation programs run only in graphical mode. On Linux hosts, this may require that you have the X Window System emulation software installed.

Preparing to Install Content Server on a UNIX or Linux System

Before you install Content Server on a Linux or UNIX system, use the checklist provided in *Table 2-1* to ensure that your system is correctly set up:

Table 2-1. Preinstallation task checklist (page 1 of 2)

Done	Preinstallation Task
___	Create the installation directories (see *Content Server Installation Directories* on page 47). You can create these directories before you install or let the installer create them later.
___	If you plan to use the graphical installer: ■ Install the X Window System on the UNIX host. ■ Add the `xterm` program directory to the PATH variable of the Documentum installation owner. You can install the xterm program in any of several locations, depending on your operating system and software packages installed. Typical locations include `/usr/openwin/bin` on Solaris and `/usr/bin/X11` on HP-UX and AIX.
___	Set up the services file (see *Setting Up the Services File* on page 50).
___	Set up a group account, an installation owner account, and a repository owner account (see *Required Accounts* on page 48).
___	Content Server uses a semaphore. Check to make sure that semaphores are enabled on the host machine.

Table 2-1. Preinstallation task checklist (page 2 of 2)

Done	Preinstallation Task
___	Check to make sure that the installation owner account has read, write, and execute permission on the `/var/tmp` directory.
___	To support external password validation, set up a group account whose members are the installation owner, any other Content Server administrators and repository owners. This group owns the external password validation program.
___	If you plan to install on an AIX host, check to make sure that AIX is running in 32-bit mode.

Content Server Installation Directories

Determine the directories in which you plan to install Content Server, and then set the `$DM_INSTALL` and `$DM_HOME` environment variables in the installation owner environment. You can create the installation directories before you install the server or you can let the Content Server installer create the directories from your input. If you allow the server installer to create the directories, make sure that the directories you specify during installation match those in the environment variables.

The `$DM_INSTALL` environment variable corresponds to the directory where you plan to install Content Server. (The installation owner must have read, write, and execute permission on the `$DM_INSTALL` directory and its subdirectories.) The `$DM_HOME` environment variable corresponds to the `$DM_INSTALL/product/5.3` directory.

Note

The environment variables and installation directories must contain only ASCII characters. The directory in which you install Content Server cannot contain spaces or the following characters:

```
! \ / : * ? " < > |
```

Default Operating System Permissions on Directories and Files

As Content Server creates directories and files in the server installation, it assigns default operating system permissions to them. The default permissions assigned to directories are 777 and the default permissions assigned to files are 666. To change the defaults assigned to public directories and files, set the `umask` key in the `server.ini` file. Setting `umask` affects all public directories and files created after you set the key.

The `umask` key works similarly to the UNIX umask functionality. The value is subtracted from the default permissions to determine the actual permissions assigned to a file or directory. For example, if you set `umask=2`, then the default permissions assigned to directories becomes 775 and the default permissions for files becomes 662. Or, if you set `umask=20`, then the permissions become 757 for directories and 626 for files.

Required Accounts

On Linux, every Content Server installation must have group and user accounts for the installation owner, repository owner, and repository users. Some of these accounts must be in place before you begin to install. You can set up other accounts during or after installation.

Installation Owner Group

To support external password validation, set up a group account whose members are the installation owner, any other Content Server administrators, and repository owners. This group will own the external password validation program.

Required Individual Accounts

Every Content Server installation must have an owner and each repository must have an owner. Each repository has users. The individual responsibilities of the installation owner, repository owner, and repository users are described in the following sections.

Installation Owner Account

The installation owner is the user whose account is used to install Content Server and create a repository. The server runs under the installation owner account.

The installation owner must have an operating system account. The installation owner user name must consist of ASCII alphanumeric characters, dashes (-) and underscores (_). The first character must be a letter. The installation owner password must consist of letters, numbers, dashes, underscores, and periods.

The installation owner account must have read, write, and execute permission on the `/var/tmp` directory and on the installation directory (`$DM_Install` and its subdirectories).

Warning

Do not use the root account as the installation owner account.

As installation owner, you can perform all administrative or maintenance tasks associated with repository installation. After you create a repository, you can create additional repository accounts with Superuser or System Administrator privileges. You can also use those accounts for repository administration.

Note

After you install Content Server, you must enable the automatic deletion of old audit trail objects from the repository. For information on how to do this, see Chapter 8, "Postinstallation Tasks," in the EMC Documentum document *Content Server Installation Guide*.

You can create an operating system account to use exclusively for server installation and repository maintenance. You can use a single user account as installation owner for multiple Documentum installations on your network.

On Linux, you can create multiple server installations on a single host computer. You can have separate installation owners for each installation or you can use separate environment files to enable a single installation owner to own all of the installations.

Repository Owner Account

The repository owner is the user whose account is used to connect to the database. The repository owner owns all objects in the database. Each repository must have a unique repository owner.

The repository owner user name and password must consist of letters, numbers, dashes (-) and underscores (_). The first character in the name must be a letter, and all characters must be ASCII characters. The corresponding password must consist of a combination of letters, numbers, dashes, underscores, and periods.

During server installation, you can specify an existing database account for database access. If you designate an existing account to use for database access, that user becomes the repository owner. Alternatively, the installer can create a database user during installation. The new user then becomes the repository owner. If the installer creates the database user, the database user name defaults to the repository name.

Assign the following privileges to the Relational Database Management System user account of the repository owner:

- Connect to the database
- Create tables, views, and indexes in the database
- Insert records (rows) into tables
- Drop tables, views, and indexes
- Unlimited tablespace

If you allow the Content Server installer to create a database account for the repository owner, the required privileges are automatically granted to the repository owner. If you create the account before you run the installer, assign the `CONNECT` and `RESOURCE` privileges to the account.

Setting Up the Services File

The `services` file contains information on the port numbers used by the services or processes that run on a host. The `services` file must contain an entry for each repository running on a host.

On UNIX and Linux, you must manually create the service name entry in the services file before you install the server. The service name entries are made in the `/etc/services` file or NIS services map for each repository running on the host. You must have root privileges to edit the `/etc/services` file.

The repository does not have a default service name or default port number. The service name you place in the `services` file must be the same name you provide during repository configuration, which is then used to create the `server.ini` file. The service name for the repository can be the same as the repository name, but this is not a requirement.

Create the service name entries using the following format:

```
<service_name> <port_number>/tcp #Comment here, if needed
```

If Network Information Service (NIS) is running, the local services file `(/etc/services)` is ignored. Place the entries in the NIS services map. Use the `ypwhich` command to identify the host name of the NIS master server, if one exists.

The port number can be any unused port number greater than 1024. (Linux reserves port numbers up to 1024 for system use.) For example, if the repository service were named mugwort, the services file entry might be:

```
mugwort 1497/tcp # repository
```

If you have multiple repositories on a single machine, create a services file entry for each repository. Make sure that each has a different name and port number.

Before you start to install Content Server, perform the following steps:

1. Install the Oracle client software on the machine on which you plan to install Content Server, and then configure the Oracle client for connectivity to the Oracle database that you plan to use to store the repository schema.
2. Use the `tnsping` command and SQL*Plus to verify that the Oracle client software is correctly installed.

For information on how to run the `tsnping` command, see oracleutilities.com/OSUtil/ping.html. For information on how to use SQL*Plus, go to comp.nus.edu.sg/~ooibc/courses/sql/sqlplus.htm.

3. Make sure that `sqlnet.ora` and `tnsnames.ora` are configured correctly.

Use fully-qualified names in `tnsnames.ora` for connections.
Example: Use `MITGDEV_TEITGDS1` instead of `MITGDEV`.

4. After you verify that the Oracle client software is correctly installed, restart the machine.
5. Create a UNIX or Linux user account (referred to in this document as dctm) that you can use to install, execute, and administer Content Server.

The user account password cannot contain special characters such as $ or @.

6. Ensure that the machine on which you plan to install Content Server has access to a valid SMTP server for email notifications.
7. Stop all nonessential services and quit all nonessential programs.
8. To configure the runtime environment of the dctm user, in one of the *.rc files, set the following environment variables:

```
setenv DOCUMENTUM <install_directory>
```

Note

Before you begin to install then Content Server, check to make sure that this install directory exists.

Set these variables in the installation owner's `.cshrc` file (C shell) or `.profile` file (Bourne or Korn shells). Alternatively, set the variables in a file called by the `.cshrc` file or `.profile` file.

```
setenv DM_HOME $DOCUMENTUM/product/5.3
setenv DOCUMENTUM_SHARED $DOCUMENTUM/shared
setenv LD_LIBRARY_PATH $DM_HOME/bin:$JAVA_HOME/lib
```

Note

LD_LIBRARY_PATH is a Solaris-specific shared library environment variable. This variable name and value varies, depending on the UNIX operating system and version. For descriptions of all required environment variables, see the *Content Server Installation Guide.*

9. Download the Content Server bundle `(DCTM_ITG-5.3_SP2-<operatingsystem>.zip)` from the Mercury IT Governance Download Center (itg.merc-int.com/support/download/login.jsp) and copy it to a temporary directory from which to run the installation.

Note

This temporary directory is referred to in this manual as `$DM_INSTALL`.

10. Use an unzip utility such as Unzip or use the Java `jar xvf` command to unzip the installation bundle and extract the following installation files:

```
ContentServer_<operatingsystem>.bin
consistency_checker.ebs
dfcSetup.jar
jdkSetup.jar
server.jar
suite.jar
tomcat 4127Setup.jar
setupError.log
```

Installing Content Server on a UNIX or Linux System

To install Content Server on a UNIX or Linux system:

1. Go to the `$DM_INSTALL/<operatingsystem>/5.3 directory`.
2. To untar the file, run the following:

   ```
   tar xvf filename

   ContentServer_<operatingsystem>.zip
   ```

   The following files are created:

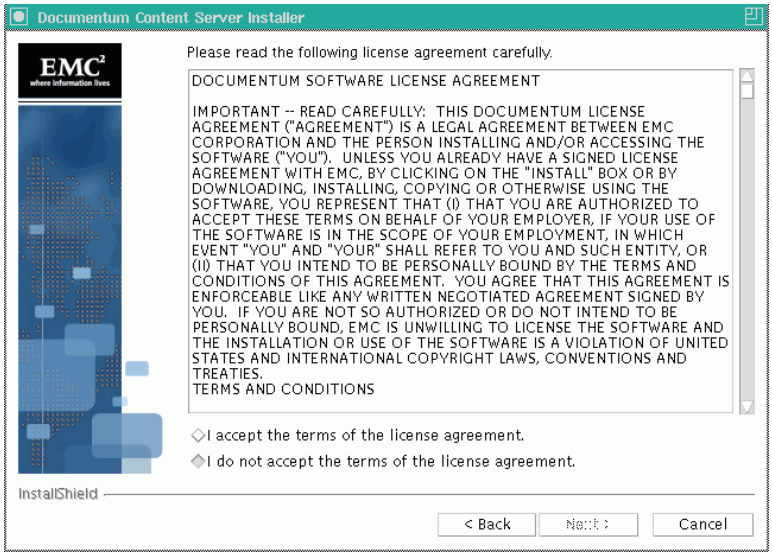
   ```
   consistency_checker.ebs
   ContentServer.exe
   dfcSetup.jar
   jdkSetup.jar
   server.jar
   suite.jar
   tomcat4127Setup.jar
   ```

3. Run `ContentServer.bin`.

   The installer displays a flash screen, and then the welcome page.

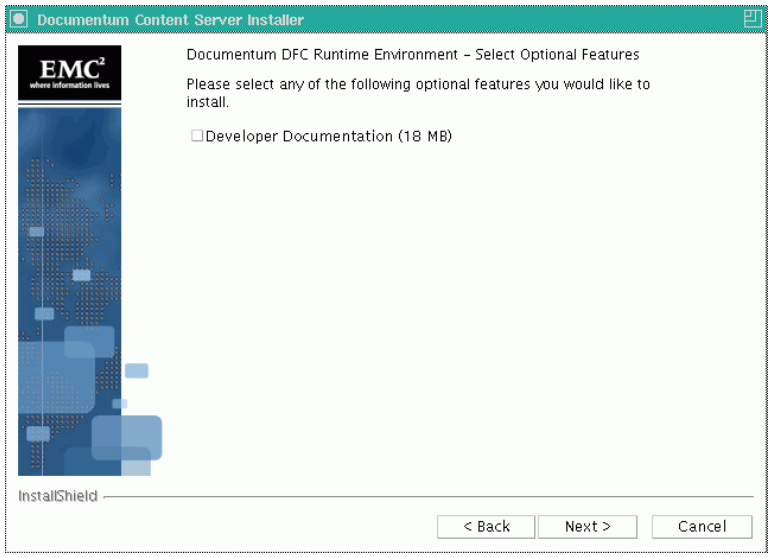
4. Click **Next.**

   The Documentum Software License Agreement page opens.



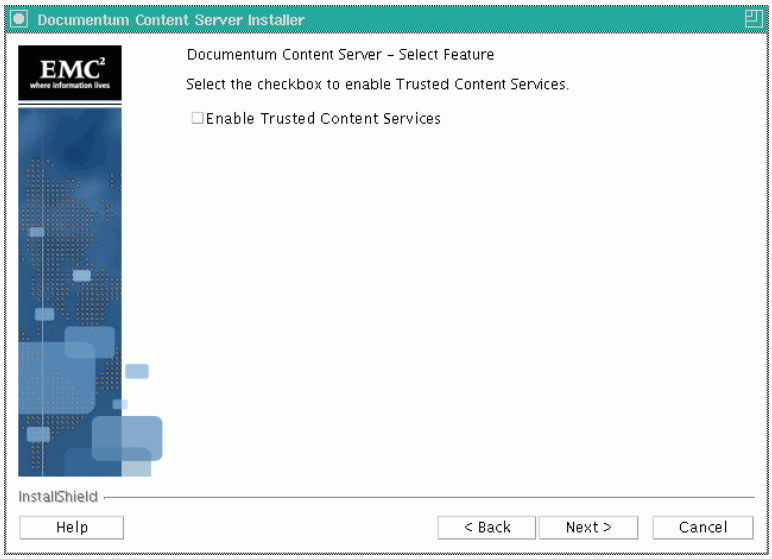
5. Select **I accept the terms of the license agreement,** and then click **Next.**

   The Optional Features page lets you select developer documentation. Mercury provides you with this documentation.

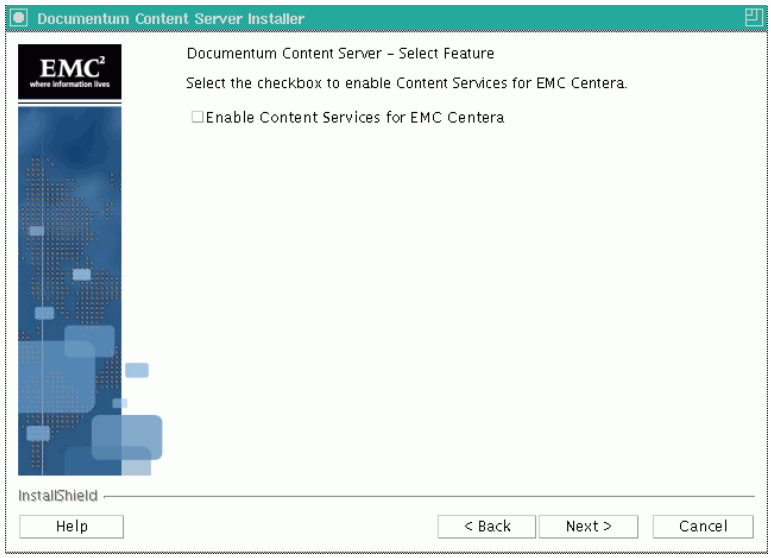


6. Leave **Developer Documentation (18 MB)** unselected and click **Next.**

   The installer prompts you to indicate whether you want to enable Trusted Content Services.

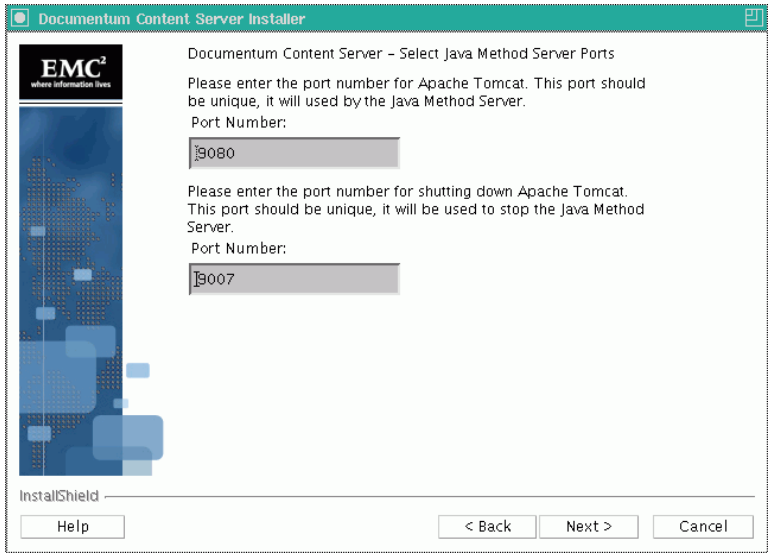


7. Leave **Enable Trusted Content Services** unselected and click **Next.**



a. Leave **Enable Content Services for EMC Centera** unselected and click **Next.**

The installer program next prompts you to specify the Java method server ports.



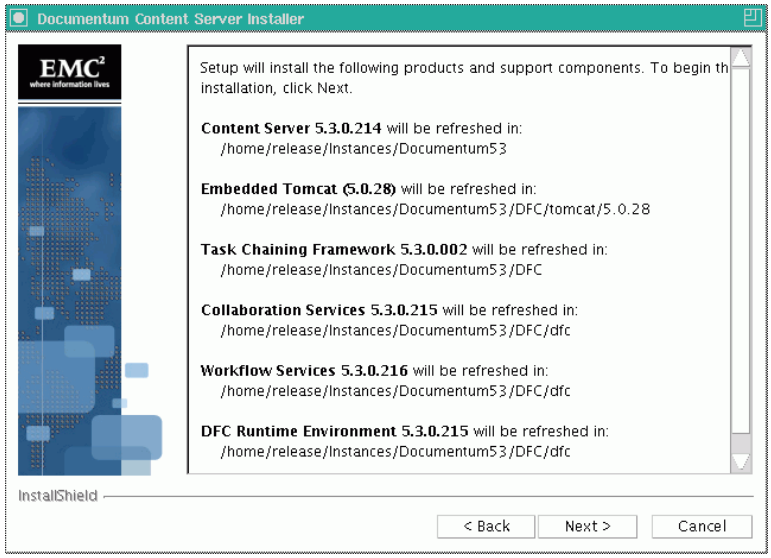
8. Specify the Java method server ports, as follows:

   a. Accept the default port number (9080) displayed in the top **Port Number** field, or type a different (and unique) port number for Apache Tomcat.

   b. Accept the default port number (9007) displayed in the bottom **Port Number** field, or type a different (and unique) port number to be used to shut down the Java method server.

9. Click **Next.**

   The installer program lists the product and support components to be installed.



10. Verify your installation parameters, and then click **Next.**

    The installer prompts you to indicate whether you want to run the `dm_root_task script.`

11. If you have the root user password, select the **Run dm_root_task now** checkbox, and then click **Next** to run the script. Otherwise, leave the checkbox unselected and click **Next.**

Warning

If you choose not to supply the root password during the procedure, you must run the `$DOCUMENTUM/dba/dm_root_task` script or the `sudo` command after installation and before you continue as root. Otherwise, the connection broker cannot start, and you cannot create a repository.

The installer installs the products and components, and then displays its final page.

12. Click **Finish.**

    The installer prompts you to indicate whether you want to configure the server now or later.

13. If you did not run the `dm_root_task` script during the install process, go to the `$DM_Install` directory (as root), and run the script as follows:

    ```
    ./dm_root_task
    ```

    The script prompts you for the group ID of the Documentum user.

14. Enter the group ID.

    The script sets the permissions for some files.

15. Select **Configure server now,** and then click **Next.**

    The installer prompts you to indicate whether you want to restart your computer.

16. Click **Yes.**

17. Proceed to *Configuring Content Server on a UNIX or Linux System*.

Configuring Content Server on a UNIX or Linux System

To configure Content Server on a Unix or Linux system:

1. After you restart your machine, log on as the Content Server installation owner.

   The Documentum Server Configuration Program for Oracle starts and displays a welcome page.

2. Click **Next.**

3. Leave **Enable Trusted Content Services** unselected, leave the **License Key** field empty, and click **Next.**

4. Leave **Enable Content Services for EMC Centera** unselected and click **Next.**

   The Select Configuration Type page prompts you to select either express or custom configuration.

5. If you are not using an external storage solution such as NAS or SAN, select **Express Configuration.** If you are using an external storage solution, select **Custom Configuration,** and then follow the instructions provided.

Note

The custom configuration wizard may prompt you for additional information that is beyond the scope of this guide. For advanced information on how to perform a custom configuration, see the *Content Server Installation Guide* from EMC Documentum.

Later, if you want to update or delete a repository or perform another configuration task, you can run the wizard again and select the custom configuration option.

6. Click **Next.**

   The next installer page prompts you to enter repository information.

7. In the **Repository Name** field, type a name for the repository to create.

Note

The repository name can contain up to 32 characters. Make a note of the repository name so that you can provide it later when you configure Mercury IT Governance Server.

8. In the **Repository ID** field, type a repository ID number between zero (0) and 16777215.
9. In the **Description** field, you can type a description of the repository.
10. In the **Repository size** list, select one of the following:

    Small: Single table space with an initial data file size of 100 MB.

    Medium: Separate table spaces for data and indexes, with initial data file size of 180 MB and an initial index file size of 180 MB.

Note

Mercury recommends that you select a medium-size database to start, and then adjust the size later, if necessary.

Large: Separate table spaces for data and indexes, with initial data file size of 250 MB and initial index file size of 250 MB.

Note

As it creates a schema, the configuration wizard creates a new tablespace in the database.

11. In the **Service Name** field, type the services name for the repository.

Note

This is the service name you specified in the services file.

12. Click **Next.**

    The Select Database Account step prompts you to indicate whether you want the program to create a new Oracle account and tablespaces, or use an existing account and tablespaces.

13. Select **Create new Oracle user account and tablespaces.**

14. Click **Next.**

    The next step prompts you to enter information about the Oracle database connection.

15. Do the following:

    a. Under **Database Connection String,** click the button, and then select a connection string.

    b. In the **Database User Name** field, type the name for a new database user.

    c. In the **Database User Password** field, type a password for the new database user.

    d. In the **Confirm User Password** field, retype the password for the new database user.

    e. In the **Database Administrator Name** field, type the database administrator name.

    f. In the **Database Administrator Password** field, type the database administrator password.

16. Click **Next.**

17. If you are running the Mercury IT Governance Server on a machine other than the Content Server host, install DFC on the Mercury IT Governance Server.

    For information about how to install DFC, see Chapter 4, *Installing Documentum Foundation Classes,* on page 91.

18. Create a repository (one per server or server cluster), as follows:

As you create a repository, you are prompted to specify a repository ID. Type a number between 0 and 16777215 that is unique to the repository. The connection broker is started for you later.

a. As root, edit the `/etc/services` file to include a line with the following format:

   `<RepositoryName> <RepositoryPort>/tcp # RepositoryForITG`

   `<RepositoryName>` and `<RepositoryPort>` are user-specific. The port must be an unused port on the machine running Content Server, and the repository name must be an alphanumeric string unique to the repository. (The repository name can contain hyphens (-) and underscores (_), but no other special characters.)

Note

Make a note of the machine name and port number you enter so that you can provide these later as you set up the document management module to work with the Mercury IT Governance Server.

b. Create an empty Oracle database schema to use to create repositories.

   The repository configuration utility creates the schema and a tablespace for you.

c. Run the server configuration utility in express mode, as follows:

```
cd $DM_HOME/install
./Server_Configuration_Program.bin
```

   The Content Server configuration utility is a graphical wizard that prompts for information, validates it, and creates the repository.

Post-Installation Tasks

After you install Content Server, do the following:

1. Use the Interactive DQL editor (IDQL) tool to test the installation.

Note

For information about IDQL and how to use it, see Appendix B, “IAPI and IDQL” in the EMC Documentum document *Content Server Administrator’s Guide.* For information on how to execute an idql statement, see *Example query:* on page 33.

2. Restart the machine.
3. Check to make sure that the Mercury IT Governance Server is not running.
4. Run the `kConfig.sh` script.

Starting and Stopping the Connection Broker and Repository

This section provides steps you can use to stop and start the connection broker and repository on Windows, UNIX, or Linux systems. It also contains information about what to do if you cannot start the repository.

Because the connection broker and repository run as separate processes on Content Server, you must start and stop them independently. To start and stop a repository, the connection broker must be running. This means that you must start the connection broker before starting the repository, and you must stop the repository before you stop the connection broker.

To start the connection broker and repository:

1. Start the connection broker and then wait for a minute or so.
2. Only after the connection broker starts, start the repository.

To stop the connection broker and repository, use the reverse process, stopping the repository first, and then the connection broker.

Starting and Stopping the Connection Broker and Repository On Windows

For a Content Server running on a Windows system, you can start and stop the connection broker and repository using any of the following three methods:

- Because the connection broker and repository run as Windows services, you can start and stop them from the Services page of the Microsoft Management Console.
- Start the Documentum Server Manager, and then use the Start or Stop buttons on the **Repository** and **Connection Broker** tabs.

  To start the Documentum Server Manager:

  - On the Content Server host, select **Start > Programs > Documentum > Documentum Server Manager.**

- Open the Command Prompt window and start or stop the connection broker or repository from the command line.
  - To start the connection broker, run the following:

    ```
    > cd %Documentum%/dba
    > ./dm_launch_<connection broker>
    ```

  - To stop the connection broker, run the following:

    ```
    > cd %Documentum%/dba
    > ./dm_stop_<connection broker>
    ```

  - To start the repository, run the following:

    ```
    > cd %Documentum%/dba
    > ./dm_start_<repositoryname>
    ```

  - To stop the repository, run the following:

    ```
    > cd %Documentum%/dba
    > ./dm_shutdown_<repositoryname>
    ```

Starting and Stopping the Connection Broker and Repository on UNIX or Linux

You can find the following scripts in the `$DOCUMENTUM/dba` directory:

- To stop a repository on UNIX or Linux, run `$ dm_shutdown_<repository name>`.
- To start a repository on UNIX or Linux, run `$ dm_start_<repository name>`.
- To stop the connection broker on UNIX or Linux, run `$ dm_stop_<connection broker>`.
- To start the connection broker on UNIX or Linux, run `$ dm_launch_<connection broker>`.

If You Cannot Start the Repository

If you cannot start the repository, use the following command sequence to produce a log file for debugging purposes:

On Windows

```
> cd %DOCUMENTUM%\dba
> .\dm_start_<repositoryname>
> .\dm_shutdown_<repositoryname>
> cd log
> ls <repositoryname>.log*
```

On UNIX or Linux

```
> cd $Documentum/dba
> ./dm_start_<repositoryname>
> ./dm_shutdown_<repositoryname>
> cd log
> ls <repositoryname>.log*
```

The log file provides a message similar to the following:

```
Fri Jan 14 07:59:21 2005 996756 [DM_SERVER_E_REGISTER_IN_USE]
error: "The server failed to register itself as there is already
a server on port (4678). Error (515) Service name already in
use. errno: 125, message: Address already in use."
```

Content Server session logs are written to the `$DOCUMENTUM/dba/log` directory. Each repository running on a Content Server has a corresponding log file, which is created every time Content Server is restarted.

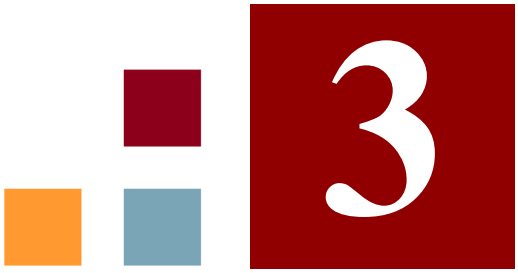
The current log file for a repository is named `<repository name>.log`. Previous log files names have the suffix `.save.<timestamp>`.

For example, if your repository is named "ITGdocs," then, in the `$DOCUMENTUM/dba/log` directory, you would see something similar to the following:

```
$ ls
ITGdocs.log
ITGdocs.log.save.09.12.2004.18.05.14
ITGdocs.log.save.11.11.2004.15.21.25
(and so on)
```

For more information about Content Server log files, see the EMC Documentum *Content Server Administrator's Guide.*

Chapter



Installing Content Server Full-Text Indexing Software

In This Chapter:

About Full-Text Indexing

Full-text indexing enables the rapid searching and retrieval of text strings within content files and content file attributes. If you are using distributed content, all content is copied to the primary content store for indexing. The drive on which the primary content store resides must have sufficient space for the primary content store plus the content copied from remote stores for indexing.

During Content Server installation, you are prompted to designate the languages for which grammatical normalization is enabled. Grammatical normalization ensures that all forms of a word are indexed and that a search for one form of a word also returns other forms.

Full-text indexing is enabled in the repository by default when the repository is created or upgraded to this Content Server version. However, Content Server itself does not create or maintain the full-text index. You must install the full-text indexing software components, which create and maintain the index.

This chapter contains information and instructions you need to install or upgrade the full-text indexing system used with the documentum management module provided with Mercury IT Governance Center. It also provides step-by-step instructions on how to install and upgrade the software.

Overview of Full-Text Indexing

Full-text indexes enable document management users to search for specific text in stored documents or document attributes. This section contains overview and conceptual information needed before making configuration decisions about full-text indexing.

The full-text indexing software consists of three components: Content Server, the index agent, and the index server. Content Server manages the objects in a repository, generates the events that trigger full-text indexing operations, queries the full-text indexes, and returns query results. For a complete description of the full-text indexing process, the chapter “Full-Text Indexing” in the EMC Documentum document *Content Server Administrator’s Guide.*

About the Indexing Process

The indexing process does not destroy existing content or attributes in a repository. Indexing is governed by queue items. During normal repository operations, queue items are generated by operations such as Save operations. When the index agent runs in migration mode, a single queue item, the *high-water mark,* governs indexing. (For information about index agent modes and the high-water mark, see *Index Agent Modes* on page 68).

During indexing, the content files and attributes are read, but not modified. For a complete description of the indexing process, see the chapter Full-Text Indexing" in the EMC Documentum document *Content Server Administrator's Guide.*

About the Indexing Software

Two software components, the *index agent* and the *index server,* underlie full-text indexing operations. This section provides information about these components.

Index Agent

The index agent exports documents from a repository and prepares them for indexing. It is a Web application than runs in an instance of the Apache Tomcat servlet container. Tomcat is automatically installed during index agent installation. Each index agent runs in its own Tomcat instance.

A given index agent runs against only one repository. Typically, you install the index agent on the Content Server host, but you can install it on a different machine.

If you install the index agent on a machine other than the Content Server host, that machine must be running a supported operating system. For a list of the supported operating systems, see the Mercury document *System Requirements and Compatibility Matrix.*

Index Agent Modes

The index agent runs in one of two modes: *migration mode* or *normal mode.* In migration mode, the index agent prepares all indexable objects for indexing in object ID order. A single queue item, the *high-water mark,* records the ID of the most recent object indexed. The index agent reads the value in the queue item, exports the next batch of indexable objects from the repository, and updates the queue item. The index agent can run in migration mode to create new indexes against a 5.2.5 repository.

Content Server generates a queue item if an event such as a check-in or save requires that a new or modifed object be indexed. In normal mode, the index agent reads the queue item, prepares the object for indexing, and updates the queue item.

After the index agent successfully submits the object for indexing, it deletes the queue item from the repository. If the index agent does not successfully submit the object, the queue item remains in the repository and the error generated by the unsuccessful attempt to index the object is stored in the queue item. The index agent can run in normal mode only against a 5.3 repository.


Note

An index agent running in normal mode and an index agent in migration mode cannot simultaneously update the same index.

Index Server

The index server creates full-text indexes and responds to full-text queries from Content Server. Depending on the configuration, a single index server instance can serve one or multiple repositories.


Note

Because the index server operations are processor- and memory-intensive, Mercury recommends that you install the index server on a machine other than the Content Server host. You must install the index server on the same operating system that is running on the Content Server host.

Full-Text Indexing Components Configuration Options

Documentum supports the following two configurations for the full-text indexing components:

- Content Server, repository, index agent, and index server on a single host
- Content Server and repository on one host with the index agent and index server on a separate host

Each repository requires its own index agent. For example, if you have multiple repositories in a single Content Server installation, you must install a separate index agent for each repository. However, regardless of where the indexing software resides, a single index server can serve multiple repositories.

Preparing to Install Full-Text Indexing

This section provides the steps you perform to prepare to install the full-text indexing software for the first time.

To prepare for full-text indexing software installation:

1. If you plan to install the indexing software on a machine other than the Content Server host, do the following to ensure that the DNS entries for the two machines are correct (so that they can locate each other on the network):

   a. On the index server machine, look up the Content Server host:

      `nslookup <FQDN_of_Content_Server_host>`

      where `<FQDN_of_Content_Server_host>` is the fully-qualified domain name of the Content Server host.

      This returns one or more IP addresses for the Content Server host.

   b. Use the first IP address returned in step 1 for a reverse lookup:

      `nslookup <IP_address_returned>`

      The correct return value is the FQDN you entered in step a on page 69.

   c. If the two `nslookup` commands do not return the correct values, update the DNS servers used by the two hosts to reflect the correct FQDNs.

d. If necessary, on a Windows system with more than one network card, update the host files to ensure that the correct IP address for each host is listed first.

e. If the `nslookup` commands succeeded and return the correct values, ping the index server host from the Content Server host to ensure it responds and to ensure that the IP address that responds to the ping is the IP address defined in the ftengine config object.

2. Disable any antivirus software running on the system.

   To install the index agent and index server, you must be logged on to the system as the same user who installed Content Server (the Content Server installation owner).

3. If you plan to install the index agent and index server on a machine other than the Content Server host, ensure that the Content Server installation owner user account exists on that machine.

4. If you plan to install the index agent and index server on a UNIX or Linux system, set the environment variables listed in *Table 3-1* in the installation owner environment.

Table 3-1. Required environment variables (page 1 of 2)

Environment Variable	Description	Required Values
DOCUMENTUM	The directory in which the indexing software is installed	Any directory in the installation owner's environment
DOCUMENTUM_ SHARED	The directory in which DFC is installed	Any directory in the installation owner's environment

LD_LIBRARY_ PATH, SHLIB_ PATH, or LIBPATH	Index server library location	`$DOCUMENTUM/fulltext/ IndexServer/lib`
		`$DOCUMENTUM/fulltext/ fast40`
		`$DOCUMENTUM_SHARED/dfc`
		`$DOCUMENTUM_SHARED/ IndexAgents/ftintegrity`

FASTSEARCH	Location of the index server	`$DOCUMENTUM/fulltext/ IndexServer`

Table 3-1. Required environment variables (page 2 of 2)

Environment Variable	Description	Required Values
DISPLAY	Controls the display	`localhost:0.0`
LC_ALL		`C`
JAVA_HOME	Home directory for the Java installation on the host	Any directory in the installation owner environment

The index server installation includes a script that sets required environment variables for running the index server. The script is `setupenv.sh` or `setupenv.csh`, depending on the shell from which you run, and it is located in the `<indexserver_install_dir>/bin` directory.

5. To ensure that the environment variables are set correctly, run the `setupenv.sh` or `setupenv.csh` script.
6. For performance reasons, Mercury recommends that you mount or share the drive or drives on which the repository file stores are located with the index server host (see *Sharing the Drives on which Content Files Reside*).

Sharing the Drives on which Content Files Reside

The index server requires access to the content files in a repository. If you install the index server on the Content Server host, then the index server has direct access to the file store storage areas and you need not do anything more.

Note

Because the index server operations are processor- and memory-intensive, Mercury recommends that you install the index server on a machine other than the Content Server host. You must install the index server on the same operating system that is running on the Content Server host.

If the index server is not installed on the Content Server host, the default behavior of the index agent is to use the `Getfile` method to retrieve a temporary copy of a file, store it in a temporary location, and pass that location to the index server. After indexing the file, the index server deletes the temporary copy.

For performance reasons, Mercury recommends, but does not require, that you mount or share the drive or drives where the repository file store storage areas

reside with the index server host. When the drives are shared or mounted, the index agent uses the `Getpath` method to pass to the index server the direct path to a file that must be indexed.

Mount or share the drives before you install the indexing software. After you install the software, edit the `indexagent.xml` file to map the file stores for the index agent and use the index agent administrative interface to indicate that the file stores are mapped. For instructions on how to edit the `indexagent.xml` file, see *Modifying the indexagent.xml File to Map File Stores* on page 86.

You can share or mount the drive or drives so that the content files are read-only. Mercury strongly recommended that you mount or share drives so that the paths are logically identical on the Content Server host and on the index server host.

On Windows hosts, use UNC paths. On UNIX or Linux, use NFS and, if necessary, symbolic links. If you must mount from a Windows platform to a UNIX or Linux platform, use third-party utilities to mount or share the drives. The changes to the `indexagent.xml` file depend on whether the paths are logically identical. For instructions on how to share or mount drives, see the documentation for your operating system.


Note

Even if the file store storage area drives are mounted, XML content is retrieved using the `Getfile` method rather than the `Getpath` method. In addition, content located in Centera stores, external stores, or encrypted file stores must be retrieved for indexing using the `Getfile` method.

Installing the Full-Text Indexing Components on a Windows System

This section provides the steps you perform to install the full-text indexing software on a Windows system, and then create a full-text index. Use these instructions to install the index agent or the index server software. The same installation program is used for both components. You can install either or both of the components on a given host.

Note

Note that the installer installs the index agent configuration program, which you use to configure an index agent instance. If you do not configure the index agent immediately after you install the configuration program, you can configure it later.

To install the index server and the index agent configuration program:

1. Ensure that the repository for which you are installing the index server and index agent is running.
2. Log in to the index server and index agent host as the Content Server installation owner.
3. Copy the installation files from the Mercury IT Governance Download Center (itg.merc-int.com/support/download/login.jsp) to a temporary location on the host:
4. Double-click `Full-text_Indexing_Components_5.3_SP2_windows.exe`.

   The Welcome page opens.
5. Click **Next.**

   The license agreement page opens.
6. Click **I accept the terms of the license agreement,** and then click **Next.**

   The installer program lists the programs you can install.
7. Leave **Documentum Index Agent Configuration Program** and **Documentum Index Server** selected and click **Next.**
8. Indicate whether to install the developer documentation and the primary interop assembly installer, and then click **Next.**

9. If required, install Documentum Foundation Classes (DFC), as follows:
    a. Accept the default installation directory (`C:\Program Files\Documentum`) or specify a different directory.
    b. Accept the default user directory (`C:\Documentum`) or specify a different directory.
10. Click **Next.**
11. If a `dmcl.ini` file does not exist on the machine, provide the following connection information:
    a. In the text field, type the host name of the computer on which a connection broker is running.
    b. In the text field, type the port number that the connection broker uses.
12. Click **Next.**
13. Install the index server, as follows:
    a. Accept the default index server installation directory or specify a new directory, and then click **Next.**
    b. Type the password for the account you used to log in, and then click **Next.**

       The installer verifies the password.
    c. Type the base port number for the index server, and then click **Next.**

       The index server requires 4,000 available ports in sequence; for example, if the base port you designate is 3000, the index server uses ports 3000 through 7000. The default base port is 13000.
    d. To enable support for grammatical normalization and parts of speech to be indexed, select the checkbox.

       Specifying the parts of speech to index can reduce the size of the indexes and the disk space required to maintain them. You can enable grammatical normalization only for the languages listed.

       If you enable grammatical normalization, it is enabled by default for Japanese and Korean and cannot be disabled. Content files in languages that you do not select or that are unavailable for normalization are still indexed. For more information about this setting, see the section

"Choosing languages for grammatical normalization" in the EMC Documentum document *Content Server Full-Text Indexing Installation Guide.*

e. Choose languages for grammatical normalization and the parts of speech to be indexed.

f. Accept the default directory `(%DOCUMENTUM%)` for the full-text indexes or specify a different directory, and then click **Next.**

If you specify a different directory, ensure that its name contains no spaces. The installer creates the `\data\fulltext` directory in the location you specify.

The installer program displays a list of the products to be installed.

14. Click **Next.**

The installation program displays a progress bar so that you can follow the progress of the software installation.

15. Click **Finish.**

16. After you install the full-text indexing software, ensure that the index server starts, as follows.

- On Windows, select **Yes, restart my computer,** and then click **Next.**

The index server starts.

Configuring the Index Agent on Windows Systems

The index agent configuration program configures the index agent to process documents for a particular repository and to pass the documents to the correct index server instance for indexing. Use these instructions to run the index agent configuration program.

To configure the index agent on a Windows system:

1. To start the configuration program, after the host reboots and you log in as the installation owner, select **Start > Programs > Documentum > Index Agent Configuration Program.**

The Welcome page opens.

2. Click **Next.**

3. Type the installation owner password, and then click **Next.**
4. Type the port numbers for the index agent to use to communicate with, and to stop, Tomcat.
5. Click **Next.**

   The default ports for the first index agent on the host are 9081 and 9008. If the index agent is on the Content Server host, ensure that the port numbers are not the port numbers that the Java method server or Site Caching Services use.
6. Select the repository for which the index agent is to prepare documents.
7. Click **Next.**

   The list displays the repositories that project to the connection brokers listed in the `dmcl.ini` file on the host. The `dmcl.ini` file was created during installation if a `dmcl.ini` file was not already on the host.
8. Type the user name and password for the Superuser account for the index agent to use to connect to the repository.

   Use this user name and password later to access the Index Agent Admin Tool.
9. If the index agent is running against a 5.3 SP1 or later repository, indicate whether to configure it in normal mode or migration mode.
10. Type the host where the index server for this index agent is running and the base port number for the index server.
11. Click **Next.**

    A summary dialog box opens.
12. Click **Next.**

    The index agent is created and Tomcat is started.
13. Click **Finish.**

Installing the Full-Text Indexing Components on UNIX and Linux Systems

This section provides the steps you perform to install the full-text indexing software and configure the index agent on UNIX and Linux systems. Use these instructions to install the index agent and the index server software. The same installation program is used for both components. You can install either or both of the components on a particular host.

Note

Note that the installer installs the index agent configuration program, which you use to configure an index agent instance. If you do not configure the index agent immediately after you install the configuration program, you can configure it later.

To install the index server and the index agent configuration program:

1. Ensure that the repository for which you are installing the index server and index agent is running.
2. Log in to the index server and index agent host as the Content Server installation owner.
3. Copy the installation files from the Mercury IT Governance Download Center (itg.merc-int.com/support/download/login.jsp) to a temporary location on the host:
4. At the prompt, type the name of the installation executable file for the operating system on which you are installing the full-text indexing components.

Operating System	Executable File Name
HP-UX	`fulltextHpuxSuiteSetup.bin`
AIX	`fulltextAixSuiteSetup.bin`
Solaris	`fulltextSolSuiteSetup.bin`
Linux	`fulltextLinuxSuiteSetup.bin`

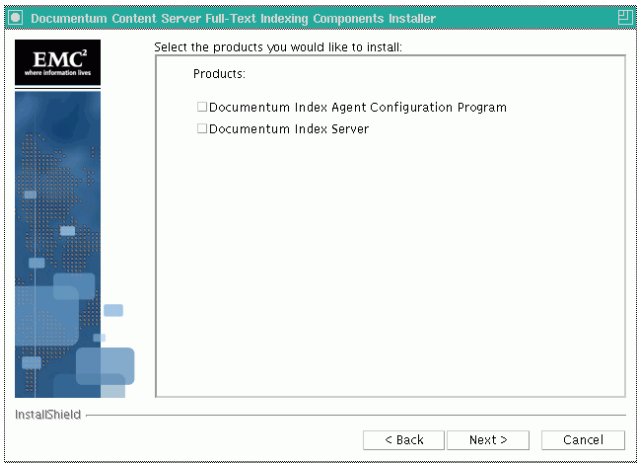
5. Press **`Enter.`**

   The Documentum Content Server Full-Text Indexing Components Installer starts and displays the Welcome page.

6. Click **Next.**

   The license agreement page opens.

7. Click **I accept the terms of the license agreement,** and then click **Next.**



   The installer program lists the programs you can install.

8. Leave **Documentum Index Agent Configuration Program** and **Documentum Index Server** selected (the default) and click **Next.**

9. On the Select Optional Features page leave the **Developer Documentation** checkbox unselected and click **Next.**

10. If required, install Documentum Foundation Classes (DFC).

    On UNIX and Linux, the DFC directories are determined by environment variables set before installation.

11. If a `dmcl.ini` file does not exist on the machine, provide the following connection information:

    a. In the text field, type the host name of the computer on which a connection broker is running.

    b. In the text field, type the port number that the connection broker uses.

12. Click **Next.**

13. Install the index server, as follows:

    a. Accept the default index server installation directory or specify a new directory, and then click **Next.**

    b. Type the base port number for the index server, and then click **Next.**

       The index server requires 4,000 available ports in sequence; for example, if the base port you designate is 3000, the index server uses ports 3000 through 7000. The default base port is 13000.

c. To enable support for grammatical normalization and parts of speech to be indexed, select the corresponding checkbox.

   Specifying the parts of speech to index can reduce the size of the indexes and the disk space required to maintain them. You can enable grammatical normalization only for the languages listed.

   If you enable grammatical normalization, it is enabled by default for Japanese and Korean and cannot be disabled. Content files in languages that you do not select or that are unavailable for normalization are still indexed. For more information about this setting, see the section "Choosing languages for grammatical normalization" in the EMC Documentum document *Content Server Full-Text Indexing Installation Guide.*

d. Choose languages for grammatical normalization and the parts of speech to be indexed.

e. Accept the default directory `($DOCUMENTUM)` for the full-text indexes or specify a different directory, and then click **Next.**

   If you specify a different directory, ensure that its name contains no spaces. The installer creates the `/data/fulltext` directory in the location you specify.

   The installer program displays a list of the products to be installed.

14. Click **Next.**

    The installation program displays a progress bar so that you can follow the progress of the software installation.

15. Click **Finish.**

16. After you install the full-text indexing software, ensure that the index server starts, as follows.

    Go to the `$DOCUMENTUM/fulltext/IndexServer/bin` directory, type `startup.sh,` and then press **`Enter.`**

The index server starts.

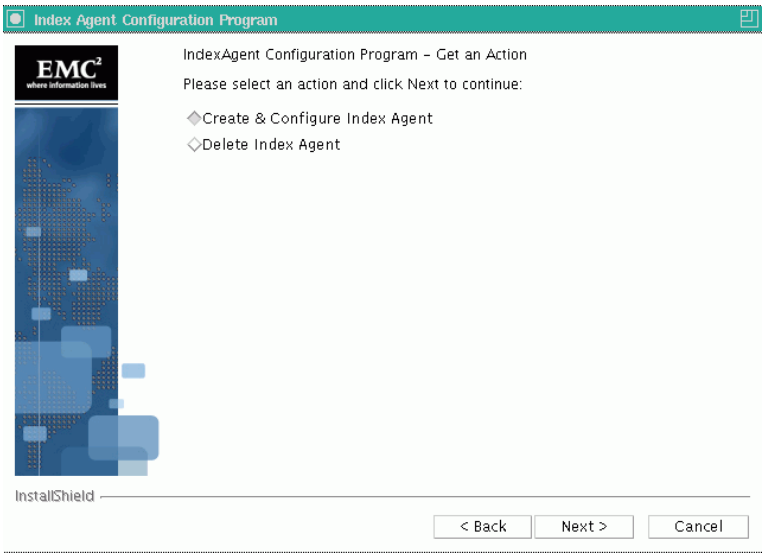
Configuring the Index Agent on UNIX and Linux Systems

The index agent configuration program configures the index agent to process documents for a specific repository and to pass the documents to the correct index server instance for indexing.

To run the index agent configuration program:

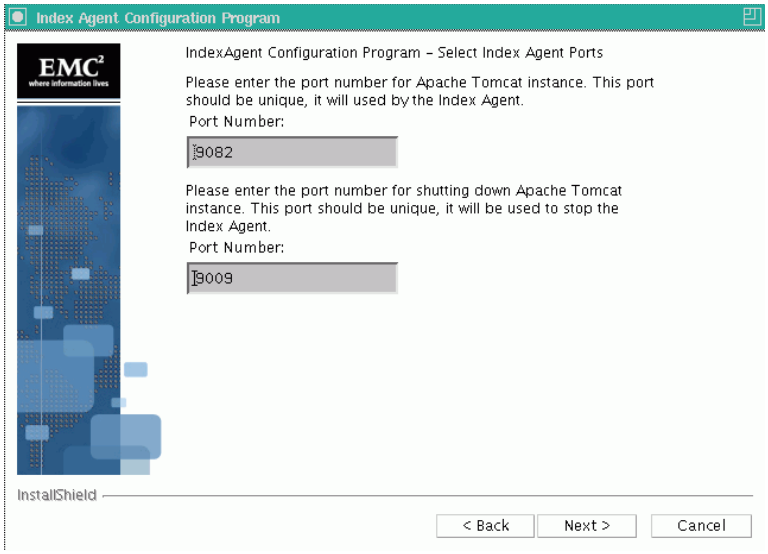
1. Navigate to `$DOCUMENTUM_SHARED/IndexAgents` and start the configuration program for your operating system, as follows:
   - For AIX, use `IndexAgent_Configuration_Program.aix`
   - For Solaris, use `IndexAgent_Configuration_Program.bin`
   - For HP-UX, use `IndexAgent_Configuration_Program.hp`
   - For Linux, use `IndexAgent_Configuration_Program.linux`

   The Welcome page opens.
2. Click **Next.**



3. Leave **Create & Configure Index Agent** selected and click **Next.**

   The Select Index Agent Ports page opens and prompts you for port numbers for Tomcat and for the index agent. The index agent runs in the Apache Tomcat servlet container. You must designate two ports for the index agent and Tomcat to use.



4. On the Select Index Agent Ports page:

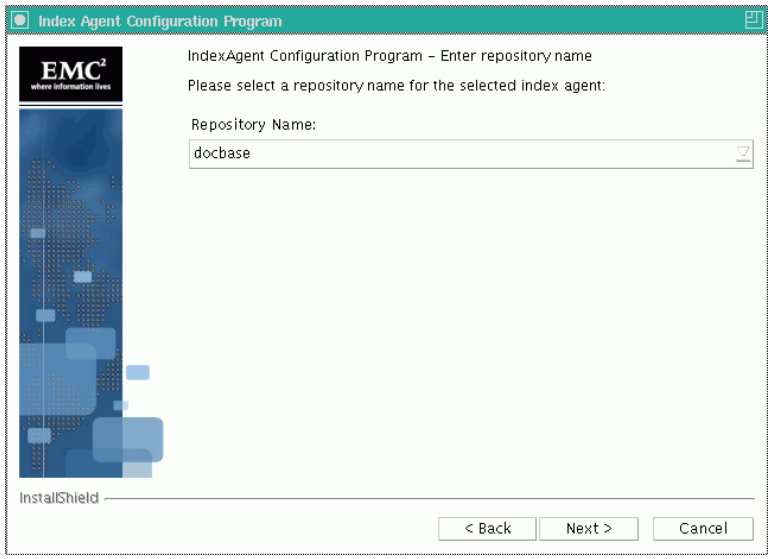
   a. In the top **Port Number** field, accept the default value specified for Apache Tomcat (9082), or type a different port number for the index agent to use to communicate with Tomcat.

   b. In the bottom **Port Number** field, accept the default value specified for the index agent (9009), or type a different port number for the index agent to use to stop Tomcat.

Note

If the index agent is on the Content Server host, do not specify port numbers used by the Java method server or Site Caching Services.

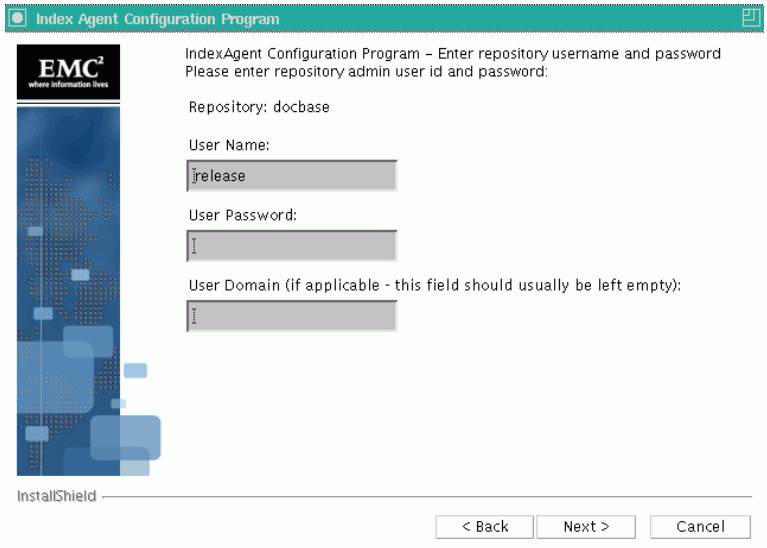
c. Click **Next.**



The **Repository Name** list displays the repositories that project to the connection brokers listed in the `dmcl.ini` file on the host. The `dmcl.ini` file was created during installation if there was not already a `dmcl.ini` file present on the host.

5. In the **Repository Name** list, select the repository for which the index agent is to prepare documents.
6. Click **Next.**

   The next page prompts you for the user name and password for the repository you selected.

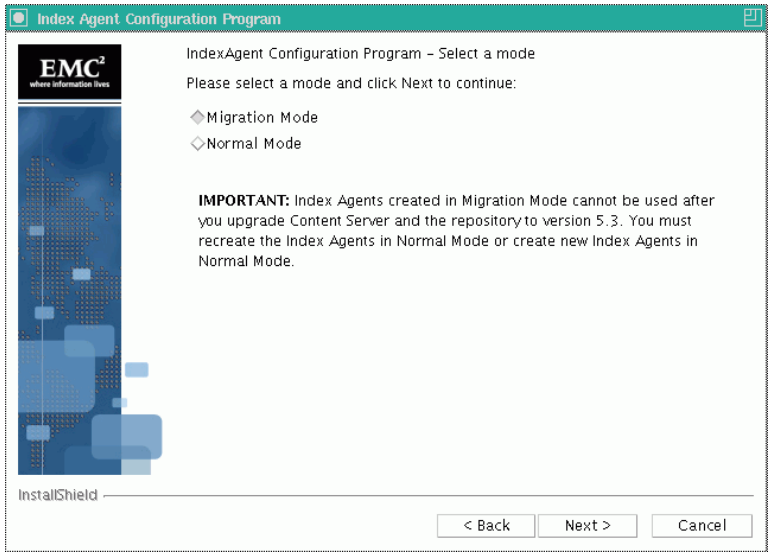


7. In the **User Name** and **User Password** fields, type the user name and password for the Superuser account that the index agent is to use to connect to the repository.
8. Click **Next.**

Note

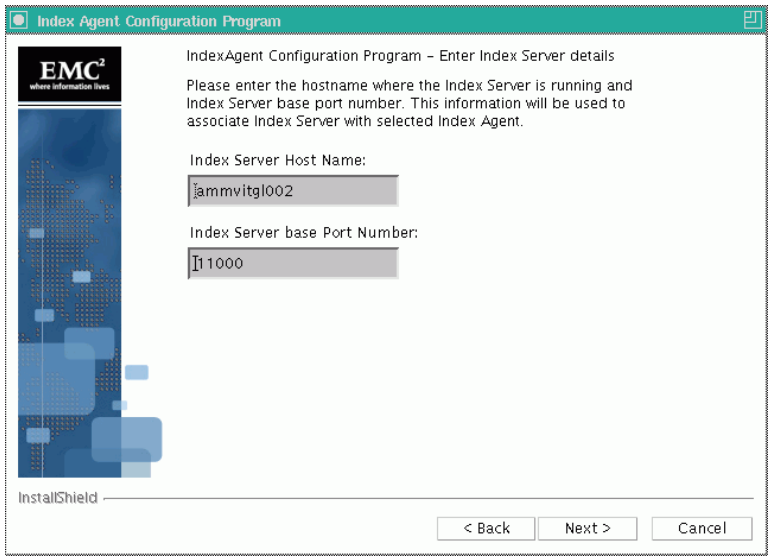
The machine on which the index server and index agent are installed must be identified using a fully-qualified domain name. For example, you could use a host name such as `isolde.documentum.com`, but not an IP address such as `172.04.8.275`.

The index agent configuration program validates the user name and password you entered.



9. Select the mode in which to run the index agent, as follows:
   - If you already have a full-text index that is based on Content Server 5.2.5, select **Migration Mode.**
   - If you are setting up document management for the first time, select **Normal Mode.**

10. Click **Next.**



11. On the Enter Index Server Details page: Type the name of the machine on which the index server for this index agent is running and the base port number for the index server.

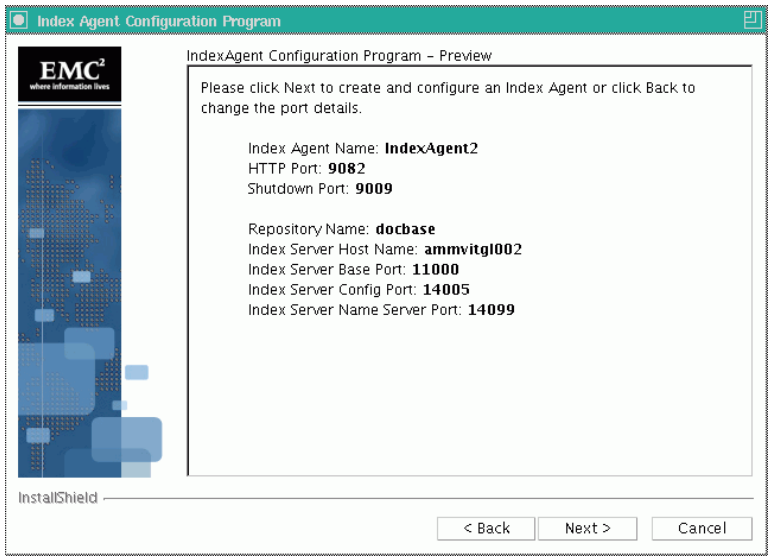
    a. In the **Index Server Host Name** field, type the name of the host on which the index server for this index agent is running.

    b. In the **Index Server Base Port Number** field, accept the default value or type a different base port number for the index server.

Note

The index server requires a contiguous range of four thousand free ports. The default range is from 13000 to 17000.

c. Click **Next.**



The Preview page lists the configuration settings you specified for the index agent and server.

12. Review the configuration settings, and then click **Next.**
13. To start the index agent and its Tomcat instance, navigate to `<$DOCUMENTUM_SHARED>/IndexAgents/<IndexAgentN>/`, where `<IndexAgentN>` is the number corresponding to the new index agent instance, and then type **`startupIndexAgent.sh`.**
14. Click **Finish.**

Modifying the indexagent.xml File to Map File Stores

If you have shared or mounted the drives that contain the repository file stores and installed the indexing software, you must manually edit the index agent configuration file to indicate that the drives are shared. The changes depend on whether the file system paths to the content are identical on the Content Server host and index server host.

To modify the `indexagent.xml` file and map the file stores:

1. On the index agent host, go to the following directory:

   ```
   $DOCUMENTUM_SHARED/IndexAgents/IndexAgent1/webapps/
   IndexAgent1/WEB-INF/classes/
   ```

2. Open the `indexagent.xml` file in a text editor.

3. If the paths to the content files are identical on the Content Server host and index server host, locate the `<exporter></exporter>` element and change the value of the `<all_filestores_local>` element to true, as follows:

   ```
   <all_filestores_local>true</all_filestores_local>
   ```

4. If the paths to the content files are different, create a file store map within the `<exporter>` element.

   Do not modify the value of `<all_filestores_local>`. For example, if Content Server is on a host called Dandelion where `filestore_01` is physically located in the directory `/Dandelion/Documentum/data/repository_name/content_storage_01` and the index agent and index server are on a host from which the drive on the Content Server host is shared as `/mappingtoDandelion/repository_name/content_storage_01`, create an alias as follows:

   ```
   <local_filestore_map>
      <local_filestore>
         <store_name>filestore_01</store_name>
         <local_mount>/mappingtoDandelion/<repository_name>
         /content_storage_01</local_mount>
      </local_filestore>
   <!-- and so on for each filestore --!>
   </local_filestore_map>
   ```

   If you are indexing content stored on an NAS device or a Windows 2003 Server host, you may see the following error message in the message attribute of the `dmi_queue_item`:

   ```
   DocumentRetriever :ERROR Retrieval error: Couldn't open file
   <file path/name> ERROR Processor error status:
   DataNotAvailable Not read permission
   ```

To resolve this error, edit the `<local_mount>` element or elements in the `IndexAgent.xml` file that reference the storage area or areas on the NAS device. Add two backslashes immediately after the opening `<local_mount>` element. For example, assume the following references a storage are on an NAS device:

```
<local_mount>\\100.2.4.32\share3\c\data_for_example\content_
storage_1</local_mount>
```

After editing, it is as follows:

```
<local_mount>\\\\100.2.4.32\share3\c\data_for_example
\content_storage_1</local_mount>
```

5. Save the `indexagent.xml` file.
6. Start a browser and open the Index Agent Admin Tool at the following URL:

   `http://<hostname>:<portno>/<IndexAgentN>/login.jsp`

   where *<hostname>* is the name of the host where the index agent is running, *<portno>* is the port where the index agent is listening, and *<IndexAgentN>* is the number assigned to the index agent instance. If the browser is on the index agent host, replace hostname with localhost.
7. Stop the index agent.
8. Indicate which filestores are mapped.
9. Restart the index agent.

   For information on how to stop and start the index agent, see *Starting and Stopping the Index Agent*.

Starting and Stopping the Index Agent

If the index agent is running in migration mode, use the Index Agent Admin Tool to start or stop it.


Note

Note that stopping the index agent does not stop or start the Tomcat process in which the index agent runs.

To start or stop the index agent running in migration mode:

1. Start a browser and go to the following URL:

   `http://<hostname>:<portno>/<IndexAgentN>/login.jsp`

   where `<hostname>` is the name of the host where the index agent is running, `<portno>` is the port where the index agent is listening, and `<IndexAgentN>` is the number assigned to the index agent instance. If the browser is on the index agent host, replace `<hostname>` with `<localhost>`.

2. Log in to the Index Agent Admin Tool.
3. To start the index agent, in the index agent status line, click **Start.**
4. Click **OK.**
5. To stop the index agent, in the index agent status line, click **Stop.**
6. Click **OK.**

Administering Full-Text Indexing

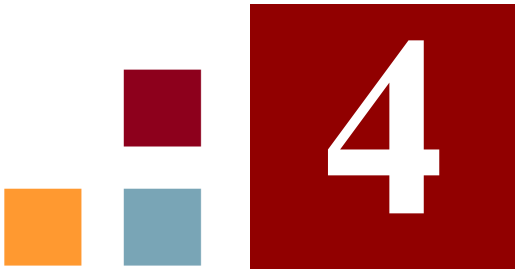
To administer full-text indexing, you use the Index Agent Admin Tool. You can use this tool to map file stores, monitor indexing, and stop or start the index agent and index server on a host.

The Index Agent Admin Tool is installed as part of the index agent and index server installation. It is a JSP page, which you can access by going to the following URL:

```
http://<hostname>:<portno>/<IndexAgentN>/login.jsp
```

where <*hostname*> is the name of the machine on which the index agent is running, <*portno*> is the port that the index agent uses to listen, and <*IndexAgentN*> is a number that designates an index agent instance.

Chapter



Installing Documentum Foundation Classes

In This Chapter:

About EMC Documentum Foundation Classes

Documentum Foundation Classes (DFC) is automatically installed with Content Server. If you install Content Server on the same machine as the Mercury IT Governance Server, there is no need to perform a separate DFC installation. However, if Content Server and the Mercury IT Governance Server are on separate machines, you must install DFC on the Mercury IT Governance Server.

Regardless of whether you plan to upgrade Content Server from 5.2.5 to 5.3, you must upgrade DFC on all the machines on which it is installed. There is no need to uninstall DFC 5.2.5 SP1 before you upgrade to DFC 5.3 SP2.

This chapter provides the information you need to prepare to install DFC, and then to install it on a Windows system or on a UNIX or Linux system. For supplementary information, see the EMC Documentum document *Documentum Foundation Classes Installation Guide.*

Before You Install Documentum Foundation Classes

This section describes the steps to take before you install DFC.

Setting the Environment Variables for DFC (UNIX and Linux)

DFC uses several environment variables to find its components. On Windows systems, the DFC installation program sets the environment variables. On UNIX and Linux systems, the DFC installation program does not set environment variables, so you must set them manually before you install. If the installation program does not find the required environment variables, installation fails.

The following sections address what to consider before you install DFC.

File System Locations for DFC Components

DFC maintains components at different file system locations. This section provides descriptions of these directories.

DFC Program Root Directory

DFC installs program files under the program root directory. On Windows systems, the installation program asks for a program root directory and uses the `C:\Program Files\Documentum` directory if you do not specify a location.

On UNIX and Linux, the installation program uses the environment variable `DOCUMENTUM_SHARED` to determine the program root directory. If this variable is undefined, the installation fails.

DFC User Root Directory

DFC creates client-oriented directories (for example, `checkout` and `export`) in the user root directory. On Windows systems, the installation program asks for a user directory root and uses `C:\Documentum` if you do not specify a location.

On UNIX and Linux systems, the installation program uses the environment variable `DOCUMENTUM` to determine the user directory root. If this variable is undefined, the installation fails.

Directory for Shared Libraries

The DFC installation program places shared libraries at specific locations relative to the program root directory. On Windows systems, the installation program uses the shared subdirectory of the program root directory. It prepends the full path of this directory (followed by a separator character) to the value of the `PATH` system environment variable.

On UNIX and Linux systems the installation program uses the `dfc` subdirectory of the program root directory. You must place the full path of this directory onto the library path. The library path environment variable has different names in different operating system versions, as follows:

- `LD_LIBRARY_PATH` in Solaris or Linux
- `SHLIB_PATH` in HP-UX
- `LIBPATH` in AIX

Directory for DFC Configuration Files

The installation program creates the config directory to store configuration files. The installation program creates the config directory under the program root directory on UNIX or Linux systems, and under the user root directory on Windows systems. For DFC to operate successfully, the classpath must contain the full path to the config directory.

On Windows systems, the installation program prepends the full path of the config directory (followed by a separator character) to the value of the `CLASSPATH` system environment variable.

On UNIX and Linux systems, you must place the full path of the `config` directory onto the classpath. For example, in the syntax of the `csh` shell, prepend `$DOCUMENTUM_SHARED\config:` to the value of the `CLASSPATH` environment variable. You can do this before or after running the installation program, because the installation program does not use this setting.

If you must specify a directory to contain the config directory, do not use `DFC_DATA`. (This environment variable has been deprecated.) If the DFC installation program finds `DFC_DATA` set to the full path of a directory, it uses the `config` subdirectory of that directory as the location of configuration files. To specify a directory, use the `dfc.data.dir` property in the `dfc.properties` file instead.

Locations of DFC Classes

The Java runtime environment uses the `CLASSPATH` environment variable to find DFC classes and the config directory. On a Windows system, the installation program places the full paths to `dctm.jar` and the config directory (with separators) at the front of the classpath. On a UNIX or Linux system, the installation program does not modify the classpath. You must place the full paths of `dctm.jar` file and the config directory onto the classpath.

Installing DFC on Windows Systems

This section provides the instructions on how to install DFC on a Windows system.

1. Log on to the machine that is hosting the IT Governance Server as a user with administrator privileges.
2. Access the Mercury IT Governance Download Center, as follows:
   a. Open a Web browser window and enter the following URL:

      `itg.merc-int.com/support/download/login.jsp`

   b. In the **User Name** field, type the user name for your download center account.
   c. In the **Password** field, type the password for your download center account.
   d. Click **Login.**
3. Read the software end user license agreement, click **I accept these terms,** and then click **Submit.**
4. In the left pane, under **Mercury IT Governance Center,** click **Tools.**

   In the right pane, the EMC Documentum products that are available for download are listed.
5. To the right of the listing for the product, in the **Download** column, click **DFC_5.3_SP2_windows.exe.**
6. Copy the installation file `DFC_5.3_SP2_windows.exe` into a temporary directory from which you will run the installation.
7. Double-click `DFC_5.3_SP2_windows.exe` to extract the following files:

   ```
   bofciSetup.jar
   bofsubscriptionSetup.jar
   bofworkflowSetup.jar
   dfcWinSetup.jar
   dfcWinSetup.exe
   dfcWinSetup.jar
   jdfwin131_04setup.jar
   ```
8. Double-click the `dfcWinSetup.exe` file.

   The Documentum DFC Runtime Environment Installer opens to the Welcome page.

9. Click **Next.**
10. Accept the default DFC installation directory displayed in the **Destination Directory** field (`C:\Program Files\Documentum`), or type the full path to a different directory.

    In this guide, this DFC installation directory is referred to as `$DFC_HOME`.
11. Click **Next.**

    The DFC installer prompts you to indicate whether you want to install optional features.
12. Leave the checkboxes unselected and click **Next.**

    The installer program next prompts you to specify the DFC user directory. The default directory is `C:\Documentum`.
13. Accept the default show in the **User Directory** field, or specify a different directory.
14. Click **Next.**

    The DFC installer prompts you to enter the location (host name) and port number of the connection broker to which you want to connect.
15. In the **Primary Connection Broker Host Name** field, type the name of the machine that hosts the connection broker.
16. Accept the default port number (1489) displayed in the **Port Number** field, or type a different port number for the machine that hosts the connection broker.

    Enter the information for the connection broker created during Content Server installation.

    You can use an IP address or a symbolic address such as `MyHost.MyCompany.com`. The installation program skips this step if it finds a `dmcl.ini` file that contains the required information.
17. Click **Next.**

    The installation program displays a summary of what is to be to installed and where. Make a note of any information you want to record.
18. Review the summary information, and then click **Next.**

    The installer program prompts you to indicate whether you want to specify a global registry for this DFC to use.

19. Leave the checkbox unselected and click **Next.**
20. After DFC is successfully installed, click **Finish.**
21. Ensure that the version of the DMCL shared library just installed is the one that DFC always uses.

    The shared library has filename `dmcl40` or `libdmcl40`. The filename extension varies with the operating system.

    The installation program replaces copies of the shared library that it finds, but other copies may exist on the machine. It is safe to replace all of them with the current version, but if you do not wish to do so, you must ensure that the old version does not precede the current version in any path environment variable that the current DFC might use.
22. If a machine hosts Content Server, manually replace the DMCL shared library that is in the server's `bin` directory.

Installing DFC on a UNIX or Linux System

This section provides instructions on how to install DFC on a UNIX or Linux system.

To install DFC on a UNIX or Linux system:

1. Ensure that you have set the environment variables (`DOCUMENTUM_SHARED`, `PATH`, and `Library path`).

   For information about the environment variables to set, see *Setting the Environment Variables for DFC (UNIX and Linux)* on page 92.
2. Depending on you operating system version, run one of the following installation programs:

   - `dfcSolSuiteSetup.bin` for Solaris
   - `dfcLinuxSuiteSetup.bin` for Linux
   - `dfcAixSuiteSetup.bin` for AIX
   - `dfcHpux11SuiteSetup.bin` for HP-UX (32 bit)

   –The installation program starts and the Welcome page opens.
3. Click **Next.**

4. Leave the **Developer Documentation** checkbox unselected and click **Next.**
5. In the **Primary Connection Broker Host Name** field, type the name of the machine that hosts the connection broker.

   You can use an IP address or a symbolic address such as `MyHost.MyCompany.com`. (The installation program skips this step if it finds a `dmcl.ini` file that contains the required information.)
6. Accept the default port number (1489) displayed in the **Port Number** field, or type a different port number for the machine that hosts the connection broker.
7. Click **Next.**

   The installation program displays a summary of what it will install and where in will install it. Make a note of any information you want to keep.
8. Review the summary, and then click **Next.**

   The installer program prompts you to indicate whether you want to identify a global registry for DFC to use.
9. Leave the checkbox unselected.
10. Click **Next.**
11. After DFC installation completes successfully, click **Finish.**
12. Ensure that the version of the DMCL shared library just installed is the one that DFC always uses.


Note

The shared library has filename `dmcl40` or `libdmcl40`. The filename extension varies with the operating system. To determine the location in which the installation program places the shared library, see *Directory for Shared Libraries* on page 93.

   The installation program replaces copies of the shared library that it finds, but other copies may exist on the machine. You can replace all of them with the current version. If you choose not to, you must ensure that the old version does not precede the current version in any path environment variable that DFC might use.
13. If the machine has a Content Server installation, manually replace the DMCL shared library that is in the server's `bin` directory.

Troubleshooting DFC Installation

The installation program maintains an error log, which it writes to a file named `setupError.log` located in the working directory. If it cannot write to the working directory, it writes to the home directory of the user who ran the installation. If installation fails, reading this file can help you detect what went wrong. If it does not, and you must call Mercury support, supply your support contact with the entire log file, unedited. The `setupError.log` file does not contain passwords or other secure information.

Chapter

Enabling Document Management

In This Chapter:

Enabling Document Management

This section provides the procedure you use to configure Mercury IT Governance Center to work with document management on Windows or on UNIX or Linux. If you are running multiple Mercury IT Governance Servers in separate file systems, be sure to perform this procedure on every server.

Enabling document management involves configuring the Mercury IT Governance Server so that it has access to the repository that you created for use with Mercury IT Governance Center during Content Server installation. If you are enabling document management on a Mercury IT Governance Server that already contains attached documents, the documents are migrated to the document management repository.

Before you enable document management, keep in mind that, once existing attachments are migrated to the Content Server repository, and new attachments are added through the Mercury IT Governance Center standard interface, you cannot disable document management and revert to storing attachments in the Mercury IT Governance Center file system. However, in a new Mercury IT Governance Center instance that does not yet have any attached documents, you can disable document management at any time and revert to storing attachments on the Mercury IT Governance Center file system.

For information about how to disable document management (after you have enabled it) in Mercury IT Governance Center, see *Disabling Document Management* on page 110.

Configuring Document Management in Mercury IT Governance Center

To configure Mercury IT Governance Center to work with the document management module:

1. Make the DFC native libraries available to the Mercury IT Governance Server by setting the shared library path environment variable on the Mercury IT Governance Center host.

   The shared library path varies with the operating environment, as follows:

   - On Microsoft Windows, use `%PATH`.
   - On Sun Solaris, use `$LD_LIBRARY_PATH`.
   - On HP-UX, use `$SHLIB_PATH`.

- On IBM AIX, use `$LIBPATH`.
- On Linux, use `$LD_LIBRARY_PATH`.

The shared library path must include the location of the native libraries (the `*.dll` files on Windows and the UNIX native files on UNIX) installed with DFC. The location of the native library depends on the operating environment, as follows:

- On Windows, the location is `%DOCUMENTUM%/shared`, where `%DOCUMENTUM%` is the Content Server installation directory.
- On UNIX, the location is `$Documentum/shared/dfc`, where `$Documentum` is the Content Server installation directory.

If, for example, Content Server is installed in the `/var/Documentum` directory on Solaris, and the Mercury IT Governance Center user uses the `~/.bashrc` logon script, you would add the following two lines to the `~/.bashrc` file:

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/var/Documentum/shared/dfc
export LD_LIBRARY_PATH
```

2. Copy the `dfc.properties` file, which is located in the shared directory under the Content Server installation directory, to `<ITG_Home>/<server_name>/conf`.
3. Run the `<ITG_Home>/bin/kStop.sh` script to stop the Mercury IT Governance Server, or, on Windows, stop the server from Windows Services.


Note

Before you perform the next step, check to make sure that the account under which you are to run the `kConfig.sh` script has read, write, and execute permissions to the directories for Mercury IT Governance Center and for the Content Server.

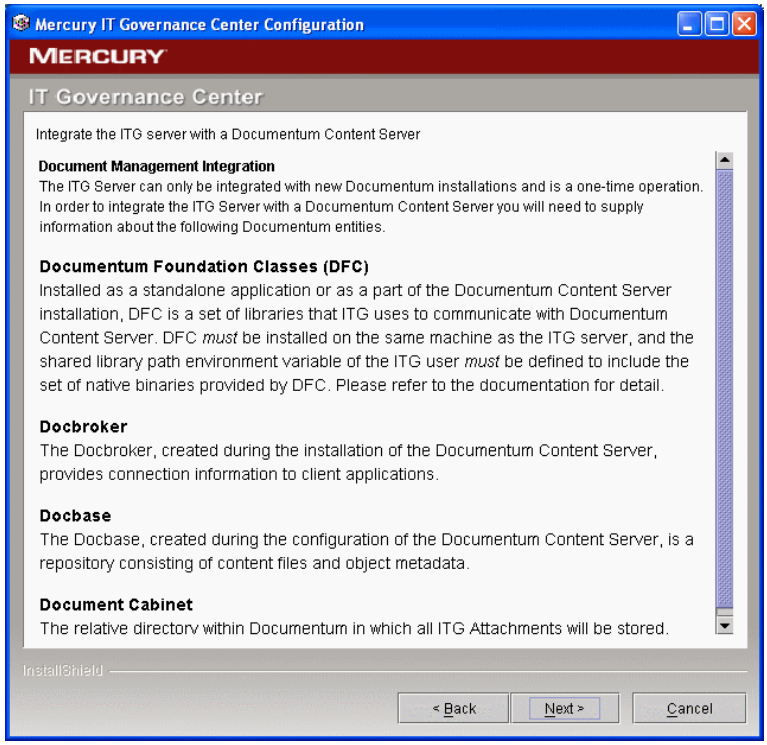
4. Run the `<ITG_Home>/bin/kConfig.sh` script.


Note

If the `kConfig.sh` script fails at any point, you can rerun it.

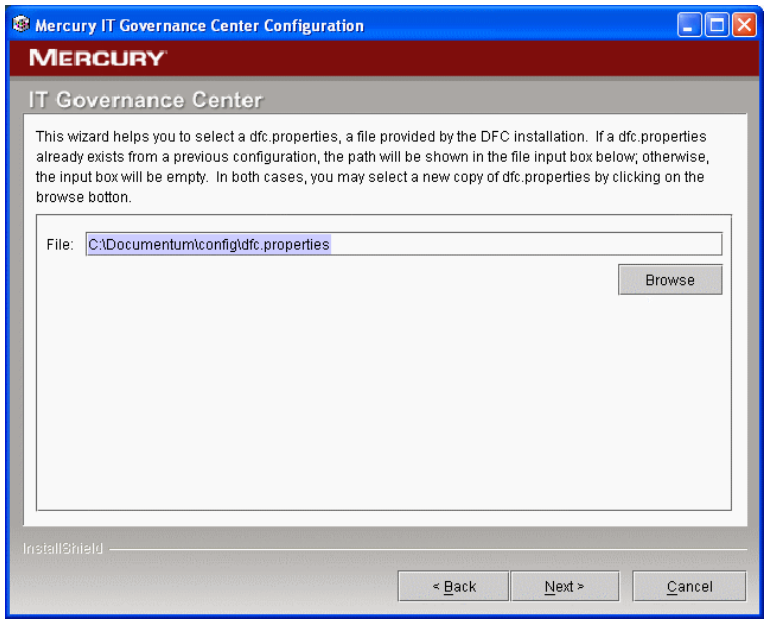
The Mercury IT Governance Center Configuration wizard starts.

5. On the first page of the wizard, select the **Integrate the ITG server with a Documentum Content Server version 5.3** checkbox, and then click **Next.**



The configuration wizard lists the Documentum components for which you must supply information.

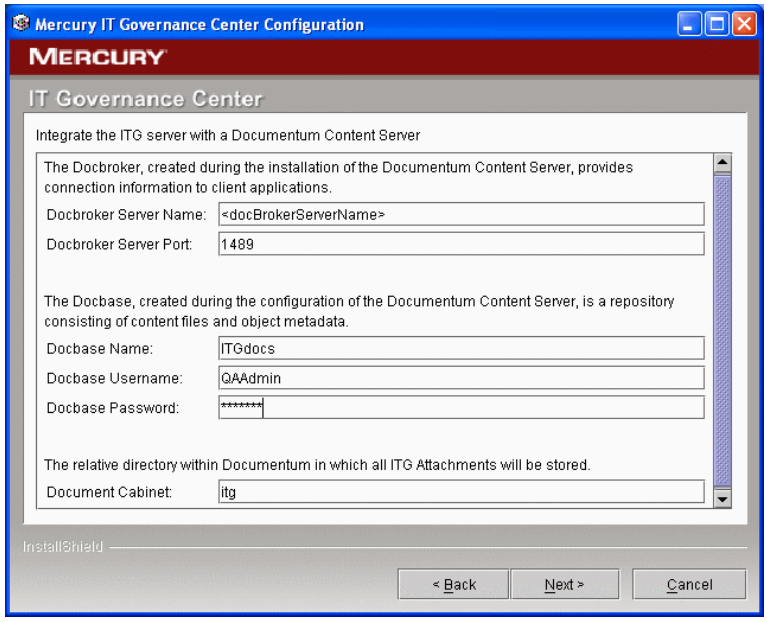
6. Click **Next.**



7. In the **File** field, enter the full path to the `dfc.properties` file.

   In typical installations this file is in the `DFC User` directory, under the `config` subdirectory.

8. Click **Next.**



9. Enter the following Content Server information:

   a. In the **Connection Broker Server Name** field, type the name of the connection broker server (the machine that hosts Content Server).

   b. In the **Connection Broker Server Port** field, type a port number for the connection broker.

   c. In the **Repository Name** field, type the name of the repository.

   d. In the **Repository Username** field, type the username for the account of the user who installed Content Server.

This is the “installation owner” operating system account on the computer hosting Content Server, and not the database user of the repository schema.

   e. In the **Repository Owner Password** field, type the password for the account of the user who installed Content Server.

This is the “installation owner” operating system account on the computer that hosts Content Server, and not the database user account of the repository schema.

   f. In the **Document Cabinet** field, type the relative directory in which to store Mercury IT Governance Center attachments.

Repository objects are organized into folders. The cabinet is a top-level folder in which Mercury IT Governance Server instance stores attached documents.

In `<ITG_Home>/conf/dms.conf`, the property `dms.documentum.cabinetName` specifies the cabinet name to use. The default is `/itg`, and there is little reason to use a different cabinet name.

10. Click **Next.**

    The configuration utility configures the server and copies any existing document attachments into Content Server.

11. After the setup is complete, on the last configuration page, click **Finish.**

12. Run the `<ITG_Home>/bin/kStart.sh` script to start the Mercury IT Governance Server. Alternatively, on Windows, use the Windows service to start the server.

    After you restart the Mercury IT Governance Server, document management is enabled.

Note

You must move attached documents into the document management system before you can access them through the Mercury IT Governance Center standard interface.

Mercury recommends that you allow the `kConfig.sh` script to copy existing attached documents into the document management system; verify that you can still access the documents through Mercury IT Governance Center after the server is running again, and then manually delete (or archive) the original documents in the attachments directory.

Troubleshooting Installation and Configuration

This section addresses some of the common problems that can occur during or after document management installation and configuration.

Problem

After you successfully install Content Server and integrate it with Mercury IT Governance Center, the Mercury IT Governance Center instance fails to come up during a `kStart.sh` script run and the error `DM_USER_E_NOT_DOCUMENTUM_USER` is displayed. The user referenced in the server log is CustomerKintana. However, the actual user is Customer\Kintana (Kintana under the Customer domain). The Mercury IT Governance Center instance was started successfully before you installed Content Server.

Solution

Test the repository user name and password by logging in to Content Server using the IDQL button in the Content Server Manager application. You are not required to specify the domain for user names. (User names are case-sensitive.)

Problem

The password for the connection between Mercury IT Governance Center and Content Server is set when you first configure the connection between Mercury IT Governance Center and Content Server. Later, you change the operating system name and password for Content Server. Running the `kConfig.sh` script triggers a warning message that indicates that you can only run the configuration for Content Server during the initial installation.

Solution

Make the following changes to <*ITG_Home*> configuration:

1. To obtain a new password, run the <*ITG_Home*>/bin/kEncrypt.sh script, as follows:

   ```
   sh kEncrypt.sh <your_password>
   ```

2. In the <*ITG_Home*>/conf directory, open the dms.conf file.
3. Do the following:
   a. Specify your new password as the value for the superUserPassword parameter.
   b. Replace #!# with \#\!\#.
4. If the Windows system account has changed, then reset the dms.documentum.superUserName parameter.
5. To update all of the Mercury IT Governance Center configuration files, run the <*ITG_Home*>/bin/kUpdateHtml.sh script.
6. Restart Mercury IT Governance Center.

Problem

During Content Server installation on UNIX or Linux, you see the following error message displayed:

```
Could not create the ITG Folder in the Documentum Content Server
Exception: FATAL ERROR: Permission denied Failed to instantiate
exclusively - lock.
```

Source

This can happen if the permissions for the DFC directories are not set up correctly.

Solution

Make sure the permissions on the DFC directories are set up correctly.

Check to make sure that the user has the required permission for the DFC directories. On Linux, the Mercury IT Governance Center owner must have read, write, and execute permissions. On Windows, the user must have administrator-level rights (or group rights) to the local machine.

Problem

Mercury IT Governance Center cannot connect to the repository. Content Sever was correctly installed, and the connection broker and repository were correctly configured. However, a Mercury IT Governance Center error message `(DM_SESSION_E_AUTH_FAIL)` indicates that the username and password provided for the repository are invalid. Verification through SQL*Plus and Toad shows that the repository user name and password supplied for Documentum are correct.

Source

Two user accounts are related to Content Server; one is the Documentum Oracle user name and schema owner. The second is the Content Server user account, to which Mercury IT Governance Center connects to access document management functionality. This user is listed in the Documentum `server.ini` file. To verify that the correct user name and password are being used, run the IDQL query tool from the Documentum Server Manager application.

Solution

Verify that the correct user name and password are being used, by running the IDQL query tool from the Documentum Server Manager application.

Problem

The Content Server configuration utility cannot locate the `dmcl.ini` file.

Source

The `dmcl.ini` file is a configuration file that Content Server clients use to locate a connection broker and repository. Content Server tries to create this file in the standard WINDOWS or WINNT directory (for example, `C:\WINDOWS`) on the host computer, but cannot if the home directory of the operating system user is different than the standard directory.

Solution

Create a new system environment variable named `DMCL_CONFIG`, and specify the full path to the `dmcl.ini` file as its value.

Disabling Document Management

After you migrate existing attachments to the Content Server repository *and* add attachments to it through the Mercury IT Governance Center standard interface, you cannot revert to storing attachments in the Mercury IT Governance Center file system. However, until you have performed these tasks, you can disable document management and revert to storing attachments on the Mercury IT Governance Center file system. This section describes how.

To disable document management functionality after you have enabled it, but before attachments are migrated to the repository and added to the repository through the Mercury IT Governance Center standard interface, do the following:

1. Update the `<ITG_Home>/conf/dms.conf` file content to include the following two lines:

   ```
   dms.filesys.attachmentDir=C\:\\ITG\\product\\attachments
   dms.driverName=com.kintana.dms.filesys.DMSFileSystemDriver
   ```

   Where `C:\ITG\product` is the `<ITG_Home>` directory. Note the required escaping of colon `(:)` and backslash `(\)` characters.


Note

This example is for a Mercury IT Governance Server running on Windows. On Linux, use single forward slashes `(/)` as directory delimiters.

2. To implement the configuration change, stop, and then restart the Mercury IT Governance Server.

For information about how to uninstall Content Server, see the *Documentum Content Server Installation Guide.*

Verifying Document Management Setup in Mercury IT Governance Center

Before you begin to use Mercury IT Governance Center with the document management module, verify the installation and configuration, as follows:

- If your Mercury IT Governance Center instance from an earlier release contained attachments, make sure that you can still access those attachments.
- Attach new documents, modify them, and then check to make sure that the documents are accessible and correctly versioned.

After your organization starts to use the document management module, you cannot easily revert the document management functionality and go back to storing attached documents on the Mercury IT Governance Center file system.

Migrating a Repository to a New Content Server Instance

If you want to move your Documentum repository to another machine, you must first install a new instance of Content Server on the target machine, and then copy the repository to the new Content Server instance. You can use any of several methods to move a repository from one Content Server to another. This section provides instructions for the following two methods, which you can perform without having to purchase an additional tool:

- Copy the repository file system, and export, and then import the database schema separately to a new Content Server instance.
- Use the Documentum Dump and Load utilities to export and import the repository as a single entity. This procedure is only recommended for use with small to moderate-sized repositories.

If you are moving the Mercury IT Governance Server to a different machine that is not hosting Content Server, you must install DFC on the new server. After you successfully clone the document management module, you can reconfigure the new Mercury IT Governance Server to point to the new repository.

Copying a Repository

You can clone a repository by copying all server-side files and the corresponding Oracle database schema. The following procedure provides the steps you use to clone a repository. The term source in these steps refers to the repository you are copying, and target refers to the Content Server that is to receive the copy.

To clone a repository, your system must meet the following requirements:

- The source Content Server and target Content Server must be running the same operating system.
- On the target Content Server, the operating system user to own the copied repository must have the same user name and password as the operating system user on the source Content Server.


Note

Mercury recommends that you use a clean, unused Content Server as the target. You must change the security keys to match the source Content Server, which affects any existing repositories on the target.

To clone a repository:

1. Stop the source repository.
2. Make a note of the name and ID of the source repository to copy.

   To determine the repository ID, use a text editor such as Notepad to open the `server.ini` file on the source `($DOCUMENTUM/dba/config/<repository>/server.ini)`, and then check the value of the `repository_id` parameter.
3. Copy the `aek.key` file `($DOCUMENTUM/dba/secure/aek.key)` from the source to the same directory on the target.
4. On the target, do the following:

   a. Log on as the same operating system user who owns the repository on the source Content Server.

   b. Create a new repository and give it the same name and ID as the source repository.

      If you are creating the repository database schema manually, Mercury recommends that you specify the same data and index tablespace names as the source repository database schema uses. This facilitates data importation later.

5. Stop the target repository.
6. On the source, use the zip or tar utility to unpack the entire repository directory (`$DOCUMENTUM/data/<repository>`).
7. Export the repository schema from the source database.
8. Connect as SYSTEM to the target database, and then drop the new target repository schema.
9. Recreate an empty target schema and specify the same name and password used for the source target schema, as follows:

```
SQL> create user <repository schema name>
2 identified by <repository schema password>
3 default tablespace <data tablespace>
4 temporary tablespace <temp tablespace>
5 quota unlimited on <data tablespace>
6 quota unlimited on <index tablespace>
```

Warning

If you are using Oracle 10g, do *not* include the next line (*7 quota unlimited on <temp tablespace>;*).

```
7 quota unlimited on <temp tablespace>;
SQL> grant connect to <repository schema name>;
SQL> grant create session to <repository schema name>;
SQL> grant create sequence to <repository schema name>;
SQL> grant create table to <repository schema name>;
SQL> grant create view to <repository schema name>;
```

10. Import the source repository schema export file into the target schema that you recreated in step 9.
11. Connect to the new target repository database schema, check the following, and make updates to refer to the new target repository location rather than the original source repository:

```
select r_install_domain, web_server_loc from dm_server_
config_s;
select host_name from dm_mount_point_s
select target_server from dm_job_s;
select projection_targets in dm_server_config_r;
```

12. To ensure that all views are rebuilt, run the following SQL statement in the target database schema:

```
update dm_type_s set views_valid=0;
```

13. Copy the zip or tar of the source repository file system to the target Content Server, and unzip or untar it into the corresponding target folder `$DOCUMENTUM/data/<repository>`.

14. Edit the target `server.ini` file (`$DOCUMENTUM/dba/config/<repository>/server.ini`) to ensure that the `preserve_existing_types` key in the `SERVER_STARTUP` section is set to `TRUE`, as follows:

    ```
    preserve_existing_types=T
    ```

15. Save the `server.ini` file.
16. Start the target repository.

Dumping and Loading a Repository

Mercury provides the Dump and Load tools that you can use to export and import a complete repository in a single bundled file. These tools are similar to the Oracle `imp` and `exp` commands.

This section provides information on how to use the Dump and Load tools to dump the contents of the source repository into a flat binary file, and then load it into a target repository. This procedure is not recommended for use with large repositories. However, for small- to moderate-sized implementations, it is simpler to use than the method provided in the section *Copying a Repository* on page 112.

Because the load procedure does not automatically create a new repository, you must create an empty target repository in which to load the dump from the source. After you create the target repository, then you can dump and load the target repository contents.

To dump and load the entire contents of a repository:

1. In the source repository, use DQL to run `dmclean`:

    ```
    1> execute "do_method"
    2>    with method = 'dmclean'
    3> go
    ```

2. In the source repository, use DQL to create a `dm_dump_record` object (dump file):

    ```
    1> create "dm_dump_record" object
    2>    set "file_name" = '<full path to output file>'
    3>    set "dump_operation" = 'full_docbase_dump'
    4>    set "include_content" = true
    5> go
    ```

    Creating the dump file may take a while. For a system with thousands of documents, expect this to take a couple of hours.

3. After the dump is completed, use a binary transfer protocol to move the output file to the target system.
4. On the target repository, use DQL to run the following:

```
1> create "dm_load_record" object
2>    set "file_name" = '<full path to dump file>'
3>    set "relocate" = true
4> go
```

   The dump is loaded into the target repository. This process takes about the same amount of time taken to create the dump file.

5. To update the full-text indexes, after the load is complete, in the target repository, run the following:

```
1> execute "update_ftindex"
2>    with "name" = 'filestore_01'
3> go
```

6. To update the IT Governance Server to reference the new target repository, do one of the following:
   - Run the <*ITG_Home*>/bin/kConfig.sh script, and choose the **Integrate the ITG server with a Documentum Content Server** option.
   - Update the <*ITG_Home*>/conf/dms.conf file. To update the password, you must run the <*ITG_Home*>/bin/kEncrypt.sh script to encrypt it. In addition, you must escape the pound (#), exclamation (!), colon (:), and equals (=) characters in the password.

     For example, change "#!#the:encrypted=string#!#" to "\#\!\#the\:encrypted\=string\#\!\#".

7. On the Mercury IT Governance Server machine, do the following:
   - Check to make sure that you can start the Mercury IT Governance Server.
   - If your Mercury IT Governance Center instance from an earlier release contained attachments, make sure that you can still access those attachments.
   - Attach new documents, modify them, and then check to make sure that the documents are accessible and correctly versioned.

Destroying the Load Object

Mercury recommends that, after you verify that you can access the expected objects, you destroy the load object. This frees up some space and ensures that the load object does not interfere with future load operations in the target repository. For more information about dumping and loading a repository, see the EMC Documentum document *Content Server Administrator's Guide.*

To destroy the load object:

1. In the target repository, use DQL to determine the load record ID, as follows:

```
1> select max("r_object_id")
2>   from "dm_load_record"
3> go
max(dm_load_record.r_object_id)
------------------------------
3100162e80000100
```

2. Run the API.
3. Issue the following `destroy` command using the load record ID you obtained in step 1:

```
API> destroy,c,3100162e80000100
```

Migrating Document Management to a New IT Governance Center Instance

Before you use the procedure described in this section to migrate document management to a new instance of Mercury IT Governance Center, see the chapter on migrating instances in *System Administration Guide and Reference.*

To migrate the document management module to the new instance:

1. Install Mercury IT Governance Center on the new instance, as described in the *System Administration Guide and Reference.*
2. Make sure that DFC is installed on the new machine.

   For information about how to install DFC, see Chapter 4, *Installing Documentum Foundation Classes,* on page 91.
3. Copy the `dfc.properties` file to the new machine.

Note

You can copy the dfc.properties file to the IT Governance Server without editing it.

4. Install the document management module on the new instance.
5. Enable document management in Mercury IT Governance Center, as described in *Enabling Document Management* on page 102.

Security Considerations

Security in the Mercury document management module follows the standard Mercury IT Governance Center model for document attachments. Using requests as an example:

- As is true for document attachments under the control of standard Mercury IT Government Center functionality (without document management), users must have the Demand Mgmt: Edit Requests access grant to add, check out, or check in documents related to requests.

  For requests in particular, attachment fields can be subject to additional field-level security if configured in the associated request type.

- Users who have permission to view the contents of an attachment field can view the attached document, the descriptive fields, and the version history.

The document management module provides a System: Override Document Check Out access grant that lets the user delegate document control to another person in the organization.

For more information, see the Mercury document *Security Model Guide and Reference* and Chapter 12, “Protecting Database Objects,” in the Documentum document *Content Server Administrator’s Guide.*

Reporting Meta Layer Entities Associated with Document Management

The Reporting Meta Layer (RML) of Mercury IT Governance Center allows you to use third-party reporting software such as Documentum Content Server (Content Server) and full-text indexing software to define custom reports. Any third-party reporting tool that can run SQL queries on an Oracle database can work with Mercury IT Governance Center reporting capabilities by:

- Using the RML schema in the Mercury IT Governance Center database as its data source
- Building reports using the standard capabilities of the Mercury IT Governance Center reporting system


Note

If you plan to work with the RML, you must be an experienced system or database administrator and have basic Mercury IT Governance Center system knowledge, which is documented in the *System Administration Guide and Reference.*

Table 5-1 lists the RML entities that are related to document management in Mercury IT Governance Center.

Table 5-1. RML entities associated with document management

Token	Description
MPKG_REFERENCES.RML	Used to relate transaction entities.
MPRJ_PROJECT_ REFERENCES.RML	Used to view the references of projects.
MPRJ_TASK_REFERENCES.RML	Used to view the references of tasks.
MREL_REFERENCES.RML	Used to view the references of releases in Deployment Management.
MREQ_REFERENCES.RML	Used to view the references of requests in Demand Management.

For more information about the RML entities listed in *Table 5-1*, see the Mercury document *Reports Guide and Reference.*

Chapter

Upgrading the Document Management Module

In This Chapter:

What's New in Document Management

If the Mercury IT Governance Center document management module is already enabled on your system, it is based on EMC Documentum 5.2.5 SP1. Document management in this release of Mercury IT Governance Center is based on EMC Documentum 5.3 SP2.

The main changes to the document management module are:

- Documentum 5.3 SP2 supports Oracle 10g.
- Documentum 5.3 SP2 supports Linux.
- More robust full-text indexing that requires that you install full-text indexing software.

  To handle full-text indexing, an additional host machine is required for Content Server 5.3 installation on Windows systems, and is recommended for Content Server 5.3 installations on UNIX and Linux systems.
- In 5.2.5-based document management, you could perform wild card searches on attached documents. For example, if you specified the string "anchored * fleet" as the search criterion, the results included all documents that contained a phrase that began with "anchored" and ended in "fleet." In 5.3-based document management, you can no longer perform wild card searches.

- In 5.2.5-based document management, you could search for synonyms in attached documents. In 5.3-based document management, you can no longer search for synonyms.

This chapter contains information on how to upgrade from a 5.2.5-based document management module to a 5.3-based document management module.

For details about the system requirements for upgrading the Mercury document management module, see the *System Requirements and Compatibility Matrix.*

Overview of Upgrading Document Management

This section addresses the high-level tasks to perform to upgrade from 5.2.5 SP1 to 5.3 SP2. If you already have the document management module enabled, and you are upgrading to Mercury IT Governance Center 7.0, you can do one of the following:

- Keep Content Server 5.2.5 SP1 on your system, but replace all instances of DFC 5.2.5 with DFC 5.3, which is compatible with Content Server 5.2.5. Regardless of whether you perform a full upgrade, you cannot continue to use DFC 5.2.5 with Mercury IT Governance Center 7.0.

  For information about how to install DFC, see Chapter 4, *Installing Documentum Foundation Classes,* on page 91.

- Upgrade the Document Management Module.

Architecture Changes for Content Server

The biggest change in Content Server 5.3 is a changed architecture for handing the new full-text indexing system. Full-text indexing is now handled by a separate index server and index agents.

For performance reasons, Mercury recommends that you set up and run the new full-text index agent and index server on a separate machine. If you are running Content Server on a Windows system, setting up a separate host machine for index agent and index server is required, due to the restrictions of running different DFC versions on the same host machine.

The index agent prepares documents in a repository for full-text indexing. (Full-text indexing allows users to easily locate documents containing specific

text.) There can be one or more index agents running against a particular repository, on the Content Server host or a remote host.

The index server creates and maintains full-text indexes and responds to full-text queries from Content Server. There can be one or more index servers on the network, for scalability and improved performance.

Upgrade Task Sequence

The sequence of tasks to perform to upgrade from 5.2.5 SP1 to 5.3 SP2 is as follows:

1. Before you upgrade Mercury IT Governance Center, install DFC 5.3 on any Mercury IT Governance Server machine that does not host Content Server. There is no need to first uninstall DFC 5.2.5.

   For information on how to install DFC, see *Installing Documentum Foundation Classes* on page 91

2. Back up all existing repositories.
3. Install Content Server 5.3 SP2.

   You can install Content Server with the 5.2.5 version in place. There is no need to uninstall it first.

4. Install the full-text indexing software.
5. Test the upgraded system.

This following sections provide the information you need to perform a full upgrade.

Migrating Full-Text Indexes

Content Server release 5.2.5 used the Verity full-text engine for full-text indexing. Content Server 5.3 uses the index agent and index server for full-text indexing. Before you migrate to the new implementation, you must make some configuration decisions.

In Documentum 5.3, full-text indexing involves three software components: Content Server, the index agent, and the index server. Content Server manages the objects in a repository, generates the events that trigger full-text indexing operations, queries the full-text indexes, and returns query results. Each

repository in which full-text indexing is enabled requires its own index server and index agent.

The index agent exports documents from a repository and prepares them for indexing. The index agent is a Web application running in an instance of the Apache Tomcat servlet container. Installing the index agent also installs Tomcat. Each index agent runs in its own Tomcat instance.

A given index agent runs against only one repository. You can install the index agent on the Content Server host or on a different machine. (For a list of the support operating systems, see the *System Requirements and Compatibility Matrix*.)

The index server creates full-text indexes and responds to full-text queries from Content Server. A particular index server runs against only one repository.

Because the index server operations are processor- and memory-intensive, Mercury recommends that you install the index server on a machine other than from the Content Server host. You must install the index server on a supported operating system. For a list of the supported operating systems, see the *System Requirements and Compatibility Matrix.*

Documentum supports the following two configurations for the full-text indexing components:

- Content Server, repository, index agent, and index server on a single host
- Content Server and repository on one host with the index agent and index server on a separate host

You must install the index agent and index server on the same operating system that is running on the Content Server host. Each repository requires its own index agent and index server. If you have multiple repositories in a single Content Server installation, you must install a separate index agent and index server for each repository.

Sharing the Drives on which Content Files are Stored

The index server must have access to the content files in a repository. If the index server is not installed on the Content Server host, for performance reasons, Mercury recommends that you mount or share the drive or drives where the repository file stores are located with the index server host.

Note

You can share or mount the drive or drives so that the content files are read-only. For information about how to share or mount drives, see the documentation for your operating system.

Designating Ports for the Index Agent and Index Server

The index agent runs in the Apache Tomcat servlet container. During index agent configuration, you designate two ports for the index agent and Tomcat to use. The default ports for the first index agent on a host are 9081 and 9008. If the index agent is on the Content Server host, ensure that the designated ports are not the ports used for the Java method server.

The index server requires a contiguous range of 4000 (four thousand) free ports. You designate which ports to use during installation. The default port range is 13000 to 17000.

Full-Text Index Migration Options

Because Content Server 5.3 uses a different index server than 5.2.5, you must migrate your repositories to the new full-text indexing implementation.

The following three migration models are supported:

- **Pre-upgrade migration of full-text indexes.** In this model, you perform upgrade tasks in the following sequence:

  a. Migrate the full-text index. To do this, you install and run the index agent and index server against your 5.2.5 repositories.

  b. Upgrade Content Server and repositories.

  c. Connect the upgraded repositories to the new full-text index.

  This strategy ensures that, even if it takes some time to create the full-text index, there is little or no down time of the full-text index subsystem, because it is in place before the upgrade.

- **Pre-upgrade migration of full-text indexes using a copy of your repository.** If you want to test your Content Server and repository upgrade against a test system before you upgrade your production system, you can run the index agent and index server against a copy of your repository.
- **Post-upgrade migration full-text indexes.** In this case, you upgrade the Content Server and repositories, and then re-index the content in the repository. This means that full-text sub-system will be in an inconsistent state from the time of the upgrade to the time when re-indexing operation completes.


Note

If your repositories are very large, Mercury recommends that you migrate the full-text indexes before you upgrade Content Server.

Preparing to Upgrade Content Server

This section provides information on what to consider before you upgrade Content Server, what pre-installation tasks to perform, and provides instructions on how to perform the upgrade.

This section addresses how to prepare to upgrade Content Server on Windows, UNIX, and Linux systems.

Before you begin to upgrade to Content Server 5.3 SP2, keep the following points in mind:

- Review *Full-Text Index Migration Options* on page 124, and decide whether you want to perform a pre-upgrade or post-upgrade migration to the new full-text indexing system.
- Using dump and load operations to upgrade a repository is not supported. Only in-place upgrade is supported.
- If you have Apache Tomcat or another application server on the Content Server host as a Java method server, stop the application server before you begin the upgrade or installation. On Windows, ensure that the application server does not start automatically after a host restart.

- If you are installing on UNIX or Linux, ensure that you have set up the required environment variables. For example, you must create a new installation directory and set the value of `$DM_HOME` to point to that directory.

  For more information about required environment variable settings, see *Content Server Installation Directories* on page 47.

- A Content Server upgrade involves upgrading the server and repository. After you upgrade, you cannot revert to previous versions of the server.
- The amount of time required to upgrade a repository depends on the size of the repository and can be substantial. Allow enough time for backing up the repository and performing the upgrade.

Supported Upgrade Path on Windows Systems

If you are on Windows with Oracle 9.2, you can upgrade Content Server directly, and then optionally upgrade to Oracle 10g. If you are on Oracle 8.1.7, you must upgrade Oracle to 9.2 before you upgrade the server.

Upgrading Full-Text indexes

Content Server 5.3 SP2 has a new implementation of full-text indexing. Because existing Verity indexes do not work with Content Server 5.3, you must replace them before or after you upgrade Content Server with new indexes. For information on how to plan for and create new full-text indexes, see *Installing Content Server Full-Text Indexing Software* on page 65.

Backing Up the Repository

Before you upgrade a repository, back it up. You can use any of several third-party tools to create repository backups.

Repository Formats

Repository formats (dm_format objects) are upgraded by the `dm_apply_formats.ebs` script, which reads values from the `formats.csv` file. If the attributes of a format in the repository do not match the format descriptions in the `formats.csv` file, the script overwrites the existing values with the values in the file.

Upgrading Content Server on a Windows System

A Windows host can have only one Content Server software installation, from which you can create multiple repositories. When you upgrade the installation, you must upgrade all of the repositories in that installation.

To upgrade Content Server on a Windows system:

1. Log in to the host system using the installation owner account.

   Use an account that is a member of the local Administrators group. User accounts on Windows are not case-sensitive, but Content Server installation fails if you connect to the host using the incorrect case in the user name. For example, if the account is set up as JPSmith and you connect as jpsmith, you can log in to the host, but server installation fails.

2. Save the compressed distribution file `(DCTM_5.3_SP2_Win.zip)` to the Content Server host's hard disk.

3. Navigate to the file location on the host's hard disk.

4. Double-click the file.

   Several files are extracted, including `Content_Server_5.3_SP2_windows_oracle.exe`.

5. Run the consistency checker and correct any errors:

   ```
   dmbasic -fconsistency_checker.ebs -eEntry_Point_Register --
   repository_name superuser password
   ```

6. Disable all jobs in all repositories on the host.

7. Shut down the repositories and connection brokers, as follows:

   a. Select **Start > Programs > Documentum > Server Manager.**

   b. Select the repository, and then click **Stop.**

   c. Click the **Connection Broker** tab.

   d. For each connection broker, select the connection broker, and then click **Stop.**

8. To shut down the Java method server's Tomcat instance, stop the service named Documentum Java method server.

9. To start the installation program, double-click the `Content_Server_5.3_SP2_windows_oracle.exe` file.

   The Welcome page opens.

10. Click **Next.**

    The license agreement page opens.

11. Click **I accept the terms of the license agreement,** and then click **Next.**

12. Leave the checkboxes unselected and click **Next.**

    The installation program indicates that an earlier version of the server exists on the host and prompts you to indicate whether you want to upgrade.

13. Click **OK.**

    The next page displays a checkbox that lets you enable Trusted Content Services.

14. Leave the checkbox unselected and click **Next.**

    The next page displays a checkbox that lets you enable Content Services for EMC Centera.

15. Leave the checkbox unselected and click **Next.**

    The next page displays a checkbox that lets you enable Content Storage Services.

16. Leave the checkbox unselected and click **Next.**

    The installer prompts you to specify port numbers for Apache Tomcat. Content Server uses Tomcat internally to run required Java programs.

    The first port is the primary port that the Java method server (Apache Tomcat) uses to communicate with Content Server.

17. Accept the default port, or type the number of any unused port above 1024 on which Apache Tomcat can listen for requests.

    The second port is the port used for Tomcat administration.

18. Accept the default port, or type the number of any unused port above 1024 to use to stop the Tomcat server.

19. Click **Next.**

    The installer program lists the components to be installed.

20. Click **Next.**

    The components are installed.

21. Indicate when you want to configure the server, as follows:

    - To configure the server immediately, select **Configure server now.**
    - To configure the server later, select **Configure server later.**

22. Click **Next.**

    The installer program displays a message to indicate that the software has been installed.

23. Click **Next.**

    The installation is not complete until the host is restarted.

24. Indicate whether you want to restart the host now or later:

    - To restart the host now, click **Yes, restart my system,** and then click **Finish.**

      The host restarts.

    - To restart the host later, click **No, restart my system at a later time,** and then click **Finish.**

Server installation is complete.

Configuring the Connection Broker, Server, and the Repository on Windows

This section provides instructions on how to configure the connection brokers, servers, and repositories on a Windows system after you upgrade Content Server. You perform these steps after the system restart following installation.

To upgrade the repository, server, and the connection broker:

1. Log in to the system as the Content Server installation owner.

   If the server configuration program does not start automatically, do the following:

   a. Restart the host manually.

   b. Log in as the Content Server installation owner.

   c. Select **Start > Programs > Documentum > Server Manager.**

   d. On the **Utilities** tab, click **Server Configuration.**

      The server configuration program starts and the Welcome window opens.

2. Type the password for the installation owner, and then click **Next.**

   The installer verifies the password.

3. Click **Next.**

4. Select **Custom Configuration** and click **Next.**

5. Choose whether to upgrade a connection broker, a repository, or both and click **Next.**

   If you select **Connection Broker,** the Connection Broker Configuration dialog box opens.

   If you select **Repository** but not **Connection Broker,** the repository Configuration dialog box opens. Skip to step 9 on page 131.

6. Leave the **Enable Trusted Content Services** checkbox unselected and click **Next.**

7. Leave the **Enable Content Services for EMC Centera** checkbox unselected and click **Next.**

8. To upgrade a connection broker on the Content Server host.
    a. Select **Upgrade an Existing Connection Broker.**
    b. Select the connection broker to upgrade.
    c. Click **Next.**

       The connection broker is upgraded.
    d. Choose whether to upgrade additional connection brokers on the local host or continue with server configuration.
    e. To upgrade an additional connection broker, repeat step a through step d.
    f. Click **Next.**

    The Repository Configuration dialog box opens.
9. Select **Upgrade Existing Repository,** select the repository, and then click **Next.**
10. Leave the **Enable Content Storage Services** checkbox unselected and click **Next.**
11. Leave the **Enable Collaborative Services or Collaborative Services with Rooms** checkbox unselected and click **Next.**
12. Leave the **Enable Retention Policy Services** checkbox unselected and click **Next.**
13. Provide connection information for the connection broker to which the repository projects, as follows:
    a. Accept the default connection broker port number (1489) or specify a different port.

       The port number is the port where the connection broker listens.
    b. Type the connection broker host name.
    c. To test the server's ability to connect to the connection broker, select **Test Connection Broker.**
    d. Click **Next.**

14. Choose the mode in which clients are to connect to the repository.

    For unsecure connections, select **Native.**

    For secure connections, select **Secure.**

    If you want clients to be able to use either connection mode, select **Native and Secure.**

    The configuration program prompts you to indicate whether you want to accept or modify the `server.ini` and `webcache.ini` files.

15. Leave the options unselected and click **Next.**

    The repository service starts.

16. Provide SMTP server information.

    The SMTP server is used to send email notifications to the installation owner and repository users.

    a. Type the name or IP address of a computer on your network that hosts an SMTP server.

       The computer can be a remote host or the computer hosting Content Server. All UNIX operating systems and Windows 2000 Server include an SMTP server.

    b. Type the email address of the installation owner.

    c. Click **Next.**

       If the configuration program cannot connect with the SMTP server, it displays a warning message. Provide a valid host name or address for the SMTP server or ignore the warning and proceed with the installation.

17. To use the current repository as the global registry repository, select **Use this Repository.**

    The global registry repository is used to store SBOs and network locations. For more information on the global registry, see *Preparing for the Global Registry* on page 141 and *Enabling a Repository as a Global Registry* on page 143.


Note

If you do not have the necessary information, you can leave the checkbox unchecked, and then set the information manually in the `dfc.properties` file later.

a. Accept the user login name of the global registry user or type a new user login name.

b. Type the password of the global registry user.

c. Confirm the global registry user password, and then click **Next.**

18. To use a different repository as the global registry repository, select **Specify a Different Repository** and click **Next.**

    The global registry repository is used for storing SBOs and network locations. For information on the global registry see *Preparing for the Global Registry* on page 141 and *Enabling a Repository as a Global Registry* on page 143.

    a. Type the repository name.

       The repository must be known to the connection broker you specified for DFC installation.

    b. Type the user login name of the global registry user for the repository.

    c. Type the user password for the global registry.

19. To designate the global registry repository later, select **Do Later,** and then click **Next.**

    The configuration program prompts you to indicate whether you want to accept or modify the repository configuration scripts.

20. Click **Next.**

21. Do one of the following:

    - To configure additional repositories, select the checkbox, click **Next,** and then return to step 9 on page 131.
    - To continue, click **Exit from server configuration,** and then click **Next.**

22. Click **Finish.**

    The upgrade is complete.

23. Start the Tomcat instance running the Java method server and ACS server

    For information on how to do this, see *Starting and Stopping the Java Method Server On Windows*.

    The Tomcat instance starts automatically only on Windows hosts that restart after installation.

24. If you created new full-text indexes before you upgraded the server, perform the following tasks:

    a. Shut down and delete the existing index agent, which is running in migration mode.

       For information on how to do this, see *Starting and Stopping the Index Agent* on page 88.

    b. Create a new index agent, which will be in migration mode.

    c. Provide the host name and base port values that were provided when you installed the index server during the migration process.

Upgrading Content Server on UNIX and Linux Systems

A UNIX or Linux host can have more than one Content Server installation and you can create multiple repositories from each installation. You can upgrade each installation at a different time. When you upgrade the installation, you must upgrade all of the repositories in that installation.

To upgrade Content Server on a UNIX or Linux system:

1. Log in to the host system using the installation owner account.

   This account must be a member of the Documentum and installation owner group.

2. To confirm the account group membership, as follows:

   For Solaris operating systems, type **`id -a`** at the command prompt.

   For AIX, HP-UX, or Linux operating systems, type **`id`** at the command prompt.

3. Save one of the following compressed distribution files to the Content Server host:

Operating System	Distribution File Name
HP-UX	`DCTM_5.3_SP2_HPUX.zip`
AIX	`DCTM_5.3_SP2_AIX.zip`
Solaris	`DCTM_5.3_SP2_Solaris.zip`
Linux	`DCTM_5.3_SP2_Linux.zip`

4. Navigate to the file directory on the host.
5. To extract the file, uncompress and untar the file, as follows:

   ```
   tar xvf filename
   ```

6. Run the consistency checker and correct any errors:

   ```
   dmbasic -fconsistency_checker.ebs -eEntry_Point_Register --
   repository_name superuser password
   ```

7. Disable all jobs in all repositories in the installation you are upgrading.
8. Shut down the repositories and connection brokers, as follows:
   a. For each repository, run the `dm_shutdown_repository` script, where repository is the name of the repository to be stopped.
   b. Stop each connection broker using the `dm_stop_connection broker` utility.
9. To shut down the Java method server's Tomcat instance, run the `$DM_HOME/tomcat/bin/shutdown.sh` script.
10. To start the installation program:
    a. Type **<*executable file name*>.**

Operating System	Executable File Name
HP-UX	`serverHpuxSuiteSetup.bin`
AIX	`serverAixSuiteSetup.bin`
Solaris	`serverSolSuiteSetup.bin`
Linux	`serverLinuxSuiteSetup.bin`

    b. Press **Enter.**

    The Welcome page opens.
11. Click **Next.**

    The license agreement page opens.
12. Click **I accept the terms of the license agreement,** and then click **Next.**

13. Leave the checkboxes unselected and click **Next.**

    The installation program indicates that an earlier version of the server exists on the host and prompts you to indicate whether you want to upgrade.

14. Click **OK.**

    The next page displays a checkbox that lets you enable Trusted Content Services.

15. Leave the checkbox unselected and click **Next.**

    The next page displays a checkbox that lets you enable Content Services for EMC Centera.

16. Leave the checkbox unselected and click **Next.**

    The next page displays a checkbox that lets you enable Content Storage Services.

17. Leave the checkbox unselected and click **Next.**

    The installer prompts you to specify port numbers for Apache Tomcat. Content Server uses Tomcat internally to run required Java programs.

    The first port is the primary port that the Java method server (Apache Tomcat) uses to communicate with Content Server.

18. Accept the default port, or type the number of any unused port above 1024 on which Apache Tomcat can listen for requests.

    The second port is the port used for Tomcat administration.

19. Accept the default port, or type the number of any unused port above 1024 to use to stop the Tomcat server.

20. Click **Next.**

    The installer program lists the components to be installed.

21. Click **Next.**

    The installer program prompts you to indicate whether you want to perform the root task, which sets file permissions for the password checking and password changing programs.

22. If you want to perform the root task:
    a. Click **Yes.**
    b. Type the root password, and then press **Enter.**

       A new window opens.
    c. Provide the name of the administrator group, and then press **Enter.**
    d. Click **O** and overwrite the files in `$DOCUMENTUM/dba`.

    The installer program returns to the main Xterm window to continue installation. It then lists the components to be installed.
23. Click **Next.**

    The components are installed.
24. Indicate when you want to configure the server, as follows:
    - To configure the server immediately, select **Configure server now.**
    - To configure the server later, select **Configure server later.**
25. Click **Next.**

    The installer program displays a message to indicate that the software has been installed.
26. Click **Next.**

    The installation is not complete until the host is restarted.
27. Indicate whether you want to restart the host now or later:
    - To restart the host now, click **Yes, restart my system,** and then click **Finish.**

      The host restarts.
    - To restart the host later, click **No, restart my system at a later time,** and then click **Finish.**

Configuring the Connection Broker, Server, and the Repository on UNIX and Linux

This section provides instructions on how to configure the connection brokers, servers, and repositories on a UNIX or Linux system after you upgrade Content Server.

To upgrade the repository, server, and the connection broker:

1. To start the configuration program, type the following commands:

   ```
   cd $DM_HOME/install
   dm_launch_server_config_program.sh
   ```

   The server configuration program starts and the Welcome window opens.

2. Click **Next.**

3. Select **Custom Configuration** and click **Next.**

4. Choose whether to upgrade a connection broker, a repository, or both and click **Next.**

   If you select **Connection Broker,** the Connection Broker Configuration dialog box opens.

   If you select **Repository** but not **Connection Broker,** the repository Configuration dialog box opens. Skip to step 8.

5. Leave the **Enable Trusted Content Services** checkbox unselected and click **Next.**

6. Leave the **Enable Content Services for EMC Centera** checkbox unselected and click **Next.**

7. Upgrade a connection broker on the Content Server host.

   a. Select **Upgrade an Existing Connection Broker.**

   b. Select the connection broker to upgrade.

   c. Click **Next.**

      The connection broker is upgraded.

   d. Choose whether to upgrade additional connection brokers on the local host or continue with server configuration.

e. To upgrade an additional connection broker, repeat step a through step d.

f. Click **Next.**

The Repository Configuration dialog box opens.

8. Select **Upgrade Existing Repository,** select the repository, and then click **Next.**
9. Leave the **Enable Content Storage Services** checkbox unselected and click **Next.**
10. Leave the **Enable Collaborative Services or Collaborative Services with Rooms** checkbox unselected and click **Next.**
11. Leave the **Enable Retention Policy Services** checkbox unselected and click **Next.**
12. Provide connection information for the connection broker to which the repository projects, as follows:

    a. Accept the default connection broker port number (1489) or specify a different port.

    The port number is the port where the connection broker listens.

    b. Type the connection broker host name.

    c. To test the server's ability to connect to the connection broker, select **Test Connection Broker.**

    d. Click **Next.**

13. Choose the mode in which clients are to connect to the repository.

    For unsecure connections, select **Native.**

    For secure connections, select **Secure.**

    If you want clients to be able to use either connection mode, select **Native and Secure.**

    The configuration program prompts you to indicate whether you want to accept or modify the `server.ini` and `webcache.ini` files.

14. Leave the options unselected and click **Next.**

    The repository service starts.

15. To use the current repository as the global registry repository, select **Use this Repository.**

    The global registry repository is used to store SBOs and network locations. For more information on the global registry, see *Preparing for the Global Registry* on page 141 and *Enabling a Repository as a Global Registry* on page 143.

Note

It is safe to leave the checkbox unchecked if you do not have the necessary information. You can set the information manually in the `dfc.properties` file later.

    a. Accept the user login name of the global registry user or type a new user login name.

    b. Type the password of the global registry user.

    c. Confirm the global registry user password, and then click **Next.**

16. To use a different repository as the global registry repository, select **Specify a Different Repository** and click **Next.**

    The global registry repository is used for storing SBOs and network locations. For information on the global registry see *Preparing for the Global Registry* on page 141 and *Enabling a Repository as a Global Registry* on page 143.

    a. Type the repository name.

       The repository must be known to the connection broker you specified for DFC installation.

    b. Type the user login name of the global registry user for the repository.

    c. Type the user password for the global registry.

17. To designate the global registry repository later, select **Do Later,** and then click **Next.**

    The configuration program prompts you to indicate whether you want to accept or modify the repository configuration scripts.

18. Click **Next.**

19. Do one of the following:
    - To configure additional repositories, select the checkbox, click **Next,** and then return to step 8.
    - To continue, click **Exit from server configuration,** and then click **Next.**
20. Click **Finish.**

    The upgrade is complete.
21. Start the Tomcat instance running the Java method server and ACS server

    For information on how to do this, see *Starting and Stopping the Java Method Server On UNIX and Linux* on page 145

    The Tomcat instance starts automatically only on Windows hosts that restart after installation.
22. If you created new full-text indexes before you upgraded the server, perform the following tasks:
    a. Shut down and delete the existing index agent, which is running in migration mode.

       For information on how to do this, see *Starting and Stopping the Index Agent* on page 88.
    b. Create a new index agent, which will be in migration mode.
    c. Provide the host name and base port values that were provided when you installed the index server during the migration process.

Preparing for the Global Registry

During repository configuration, you are prompted to choose from the following options:

- **Use the current repository as a global registry.** If you select this option, you must provide the user login name and password for the global registry user in the current repository (the repository you are currently configuring). Record that information and provide the user login name and password to any DFC instance that requires access to this global registry repository. The DFC instance on the current host is also configured to access this global registry.

- **Specify a different repository as a global registry.** If you select this option, you must provide the repository name, and the user login name and password of the global registry user in that repository. The DFC instance on the current host is configured to access the remote global registry repository.
- **Do later.** If you select this option, you can delete the `dfc.bof.registry.repository`, `dfc.bof.registry.username`, and `dfc.bof.registry.password` from the `dfc.properties` file and rerun the DFC installer on this host to designate the global registry repository later.

Whether or not you designate the repository as a global registry, the global registry user is created in the repository. The global registry user, who has the user name of `dm_bof_registry`, is the repository user whose account is used by DFC clients to connect to the repository to access required service-based objects or network locations stored in the global registry. This user has Read access to objects in the `/System/Modules and /System/NetworkLocations` only, and no other objects.

This user is created in all repositories, regardless of whether the repository is configured as a global registry:

- If you configure the repository as a global registry, you provide the user login name and password for the user and the user state is set to active. This can be any arbitrary user login name and password. Record the user login name and password, and provide the user login name and password during DFC installation on client hosts. Do not use the repository owner's credentials or the installation owner's credentials.
- If you do not configure the repository as a global registry, the user is created with a default value for the user login name and the user state is set to Inactive. If you later enable the repository as a global registry, use Documentum Administrator to change the user state to Active and provide the user with a user login name and password that you choose. For instructions on how to enable the repository as a global registry, see *Enabling a Repository as a Global Registry* on page 143

Enabling a Repository as a Global Registry

To enable a repository as a global registry after configuration, you must activate the `dm_bof_registry` user.

To enable a repository as a global registry:

1. Access Documentum Administrator in a browser and connect to the repository.
2. Click **Administration > User Management > Users.**
3. Locate the user named `dm_bof_registry` and click the **Information** icon.
4. Ensure that the user name attribute is set to `dm_bof_registry`.
5. Change the user login name to a new value (optional).
6. Change the user password.
7. Set the `dm_bof_registry` user status to active.
8. To save the user, click **OK.**
9. During DFC installation on client machines, provide the user login name and password.

   This updates the `dfc.properties` file and enables that DFC installation to contact the global registry.
10. To manually modify the `dfc.properties` file to designate a global registry repository and user credentials:
    a. On the DFC host, navigate to `$DOCUMENTUM/config` (UNIX or Linux) or `%DOCUMENTUM%\config` (Windows).
    b. From a command prompt, execute the following command to generate the encrypted form of the global registry user password:

       ```
       java -cp dfc.jar
       com.documentum.fc.tools.RegistryPasswordUtils
       <user_password>
       ```

       where <*user_password*> is the global registry user's clear-text password. In step d, you enter the encrypted form of this password in the `dfc.properties` file.
    c. Open the `dfc.properties` file in a text editor.

d. Modify the following attributes:

```
dfc.bof.registry.repository=<global_registry_repository_
name>
dfc.bof.registry.username=<user_login_name>
dfc.bof.registry.password=<encryped_user_password>
```

where <*encryped_user_password*> is the encrypted password you generated in step b.

e. Save the `dfc.properties` file.

Starting and Stopping Servers and Connection Brokers

On Windows systems, start and stop servers and connection brokers from the Server Manager tool. To access the tool, select **Start > Programs > Documentum > Server Manager.** On the correct tab, select the repository or connection broker and click **Start** or **Stop.** You can also launch the server configuration program from Server Manager and use it to create additional repositories or connection brokers in the installation, uninstall existing repositories or connection brokers, and modify the existing repositories.

On UNIX and Linux systems, start servers with the `m_start_repository` script, which is located in the `$DOCUMENTUM/dba` directory. To stop servers, use the `dm_shutdown_repository` script in the same directory.

If you create a custom passphrase after server installation, any time you restart the server host you must run the `dm_crypto_boot` utility.

Starting and Stopping the Java Method Server

Apache Tomcat, in which the Java method server and ACS server run, is installed when Content Server is installed. Tomcat does not start automatically after installation except on a Windows host that restarts after installation. You must manually start the Java method server on UNIX and Linux host and on any Windows host that does not restart after installation. Use the instructions in this section to start and stop Tomcat.

Starting and Stopping the Java Method Server On Windows

On Windows, Tomcat is installed and runs as a service. It is installed with the startup property set to Automatic.

To start Apache Tomcat:

1. Select **Start > Programs > Administrative Tools > Services.**
2. In the Services dialog box, scroll to **Documentum Java method server.**
3. Right-click **Documentum Java method server,** and then click **Start** on the shortcut menu.
4. Close the Services dialog box.

To stop Apache Tomcat:

1. Select **Start > Programs > Administrative Tools> Services.**
5. In the Services dialog box, scroll to **Documentum Java method server.**
6. Right-click **Documentum Java method server,** and then click **Stop** on the shortcut menu.
7. Close the Services dialog box.

Starting and Stopping the Java Method Server On UNIX and Linux

On UNIX and Linux, you must use scripts to start and stop Tomcat. It does not start or stop automatically after the host is restarted.

To start Apache Tomcat:

1. Connect to the Content Server host as the Documentum installation owner.
2. Navigate to the `$DM_HOME/tomcat/bin/` directory.
3. Run the `startup.sh` script.

To stop Apache Tomcat:

1. Connect to the Content Server host as the Documentum installation owner.
2. Navigate to the `$DM_HOME/tomcat/bin/` directory.
3. Run the `shutdown.sh` script.

Overview of Full-Text Indexing Migration

The full-text indexing implementation consists of Content Server, the index agent, and the index server. Content Server manages the objects in a repository, generates the events that trigger full-text indexing operations, queries the full-text indexes, and returns query results client applications.

Each repository in which full-text indexing is enabled requires its own index server and index agent. For a complete description of the full-text indexing process, see the *Content Server Administrator's Guide.*

Index Agent

The index agent exports documents from a repository and prepares them for indexing. The index agent is a Web application that runs on an instance of the Apache Tomcat servlet container. Installing the index agent also installs Tomcat. Each index agent runs in its own Tomcat instance.

A particular index agent runs against only one repository. The index agent can be installed on the Content Server host or remotely. If installed remotely, the index agent must be on a supported operating system. For a list of the supported operating systems, see the *Content Server Release Notes.*

Index Agent Modes

The index agent is typically run in one of two modes, migration mode or normal mode. In migration mode, the index agent prepares all indexable objects for indexing in object ID order. A single queue item records the ID of the most recent object indexed. The index agent reads the value in the queue item, exports the next object, and updates the queue item. The index agent can run in migration mode to create new indexes against a 5.2.5.x, or 5.3 repository.

Content Servers 5.3 and later generate a queue item any time an event such as a check-in or save requires that a new or modified object must be indexed. In normal mode, the index agent reads the queue item, prepares the object for indexing, and updates the queue item. When the index agent submits the object for indexing, the index agent deletes the queue item from the repository. The index agent can run in normal mode only against a 5.3 repository.

Index Server

The index server creates full-text indexes and responds to full-text queries from Content Server. A particular index server runs against only one repository.

The index server's operations are processor- and memory-intensive, and Mercury therefore recommends that you install the index server on a host remote from the Content Server host. The index server must be installed on a supported operating system. For a list of the supported operating systems, see the *Content Server Release Notes.*

About Indexing

The indexing process is not destructive to existing content or attributes in the repository. In migration mode, indexing is governed by a single queue item. In normal mode, indexing is governed by queue items generated by repository operations such as Save operations.

The content files and attributes are read during the indexing process. For a complete description of full-text indexing, see the *Content Server Administrator's Guide.*

Full-Text Indexing Components Configuration Options

Documentum supports the following two configurations for the full-text indexing components:

- Content Server, repository, index agent, and index server on a single host
- Content Server and repository on one host with the index agent and index server on a separate host

  The index agent and index server must be installed on the same operating system as Content Server. For example, if Content Server and the repository are on a Windows host, the index agent and index server for that repository must also be on Windows. Each repository requires its own index agent and index server. For example, if you have multiple repositories in a single Content Server installation, you must install a separate index agent and index server for each repository.

Sharing the Drives Where Content Files Reside

The index server requires access to the content files in a repository. If the index server is not installed on the Content Server host, for performance reasons, it is recommended, but not required, that you mount or share the drive or drives where the repository's file stores are located with the host where the index server is located. You can share or mount the drive or drives so that the content files are read-only. Refer to the documentation for your operating system for instructions on sharing or mounting drives.

Index Agent Ports

The index agent runs in the Apache Tomcat servlet container. When an index agent instance is configured, you must designate two ports for the index agent and Tomcat to use. The default ports for the first index agent on a host are 9081 and 9008. If the index agent is on the Content Server host, ensure that the ports are not the ports used for the Java method server.

Index Server Ports

The index server requires a contiguous range of 4000 (four thousand) free ports. You must designate which ports to use during installation. The default range is from 13000 to 17000.

Planning for Full-Text Migration

Content Server releases before 5.3 used the Verity full-text engine for full-text indexing.

Content Server releases 5.3 and later use the index agent and index server for full-text indexing. Migrating to Content Server 5.3 or later requires that you make some decisions before migrating to the new implementation.

Do not run the 5.3 SP2 indexing software against a 5.2.5 repository except for migration purposes.

When to Migrate the Full-Text Indexing System

Two models are supported for migrating to the new full-text implementation:

- Pre-upgrade migration

  In this model, the index agent and the index server are installed and run against the pre-5.3 production repository or a copy of the pre-5.3 production repository. The new index is created before Content Server is upgraded. After the index is created, the server and repository are upgraded and the new index is used. The full-text indexing system is completely available and up-to-date after Content Server is upgraded.

  Pre-upgrade migration is recommended for very large repositories or for any repository where it is a business requirement that the full-text system is available in a consistent state immediately after the upgrade.

- Post-upgrade migration

  In this model, you upgrade Content Server and the repository first. You then install the index agent and the index server and create the new index. The full-text system is in an inconsistent state until the index is created.

  Post-upgrade migration is recommended for small repositories (fewer than 100,000 documents) or for any repository for which it is acceptable for the full-text system to be in an inconsistent state immediately following the repository upgrade.

Planning for Pre-Upgrade Migration

If you use pre-upgrade migration, you must make the following decisions:

- **Do you index the production repository or a copy of the repository?**

  To test a Content Server upgrade, Mercury recommends that you create a copy of the repository and test the server and repository upgrade on the copy. You can create the new full-text index by indexing either the copy or the production repository.

  If the repository is extremely large, creating new indexes takes a significant amount of time. You may prefer to test the time and space required for creating new indexes on a copy or on the production repository.

You are not required to create a copy of the content files, even if you create a repository copy. Instead, the file stores can be shared with the repository copy. The content can be made available to the index agent and index server in read-only mode. If you create the new index on the repository copy, you use the index with the production repository. When you create an index agent in normal mode, you point it to the existing index server and the production repository, and the production repository then uses the index.

- **Which hardware do you use to run the index agent and the index server?**

  Mercury recommends that you use a host other than the Content Server host for the index agent and index server and strongly recommend that you choose hardware that will host the index and indexing software for the production repository. If you index the production repository rather than a copy, this is recommended because creating new indexes is processor-intensive.

  The index agent and the index server must be installed on an operating system that is supported for 5.3 SP2. If the production repository is installed on an older operating system that is not supported for the index agent and the index server, you must install the index agent and the index server a remote host.

  The index agent and the index server may be installed on a different supported operating system from the Content Server. The index agent and index server are required to be installed on the same host.

  On Windows systems, you cannot install more than one version of DFC. The index agent requires DFC 5.3 SP2. To create the new index for the older repository, you must therefore install the index agent and index server on a host other than the Content Server host.

  On UNIX and Linux systems, you can install the index agent and index server on the Content Server host, provided the operating system is supported. You must ensure that the environment variables are set so that the existing Content Server continues to use the older DFC with which it was installed and the index agent uses DFC 5.3 SP2.

- **Which configuration do you use?**

  For more information, see Chapter 3, *Installing Content Server Full-Text Indexing Software,* on page 65.

- **Do you mount or share the drives where content files are located?**

  For more information, see *Sharing the Drives on which Content Files Reside* on page 71.

Installation Order for a Pre-Upgrade Migration

If you use pre-upgrade migration, the overall procedure, including decisions to be made, is:

1. Decide whether to index the production repository or a copy.
2. Choose hardware for the index agent and index server.
3. If you are indexing a repository copy, create the copy.

   Use the instructions in the section "Creating a Repository Copy to Test an Upgrade" in the *Content Server Installation Guide.*
4. Install the index server and configuration program.

   For instructions, see Chapter 3, *Installing the Full-Text Indexing Components on a Windows System,* on page 73.
5. Configure an index agent in migration mode.
6. Create the full-text indexes.
7. Run the `ftintegrity` tool.

   Use the instructions in the section "Verifying the full-text indexes," in the *Content Server Full-text Indexing Installation Guide.*
8. If any documents were not indexed, resubmit those documents for indexing.

   Use the instructions in the section "Resubmitting objects to the index agent," in the *Content Server Full-text Indexing Installation Guide.*

9. Upgrade the repository.

   For instructions, see *Configuring the Connection Broker, Server, and the Repository on Windows* on page 130 and *Configuring the Connection Broker, Server, and the Repository on UNIX and Linux* on page 138.

10. Shut down the migration-mode index agent change it to a normal-mode index agent.

Planning for Post-Upgrade Migration

Post-upgrade migration assumes that new full-text indexes are created using the production repository, not a copy. If you choose post-upgrade migration, you must make the following decisions:

- **Which hardware do you use to run the index agent and the index server?**

  Mercury recommends that you use a host other than the Content Server host for the index server. If you index the production repository rather than a copy, this is strongly recommended because creating new indexes is processor-intensive. The index agent and the index server must be installed on an operating system that is supported for 5.3 SP2. The index agent and the index server may be installed on a different supported operating system from the Content Server. The index agent and index server must be installed on the same host.

- **Which configuration do you use?**

  For information, see *Installing Content Server Full-Text Indexing Software* on page 65

- **Do you mount or share the drives on which content files are located?**

  For information, see *Sharing the Drives on which Content Files Reside* on page 71.

Installation Order for a Post-Upgrade Migration

If you use post-upgrade migration, the overall procedure, including decisions to be made, is:

1. Choose hardware for the index agent and index server.
2. Upgrade the repository.

   For instructions, see *Configuring the Connection Broker, Server, and the Repository on Windows* on page 130 and *Configuring the Connection Broker, Server, and the Repository on UNIX and Linux* on page 138.
3. Install the index server and index agent configuration program.
4. Configure an index agent in migration mode.
5. Create the full-text index.
6. Run the `ftintegrity` tool.
7. If any documents were not indexed, resubmit those documents for indexing.
8. Shut down the migration-mode index agent change it to a normal-mode index agent.

Chapter

What Document Management Users Need to Know

In This Chapter:

Attaching Documents to Entities

This section contains the procedure you use to attach documents to a Mercury IT Governance Center entity such as a request or a project (in a Mercury IT Governance Center instance containing the document management module).

You can attach documents to an entity in one of following two ways:

- If the entity has a custom attachment field, you can attach a document directly to that field.
- You can add a document as a reference to any entity that supports references.

The following section provides the steps you use to attach documents to an entity that supports references.

Adding a Document as a Reference

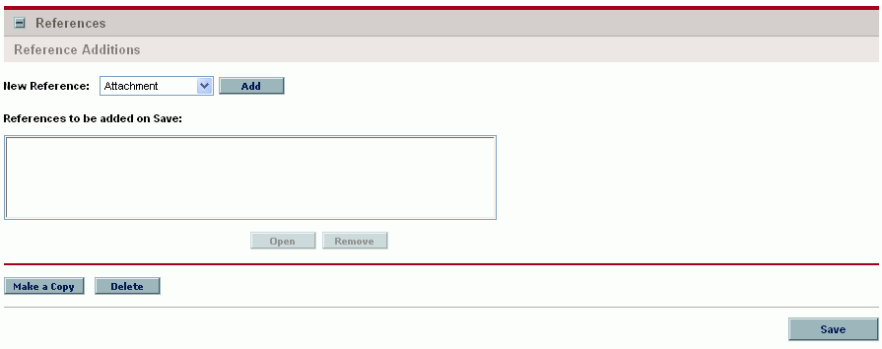
To attach a document to an entity that supports references:

1. From the standard interface, open the entity to which you want to attach a document.

Note

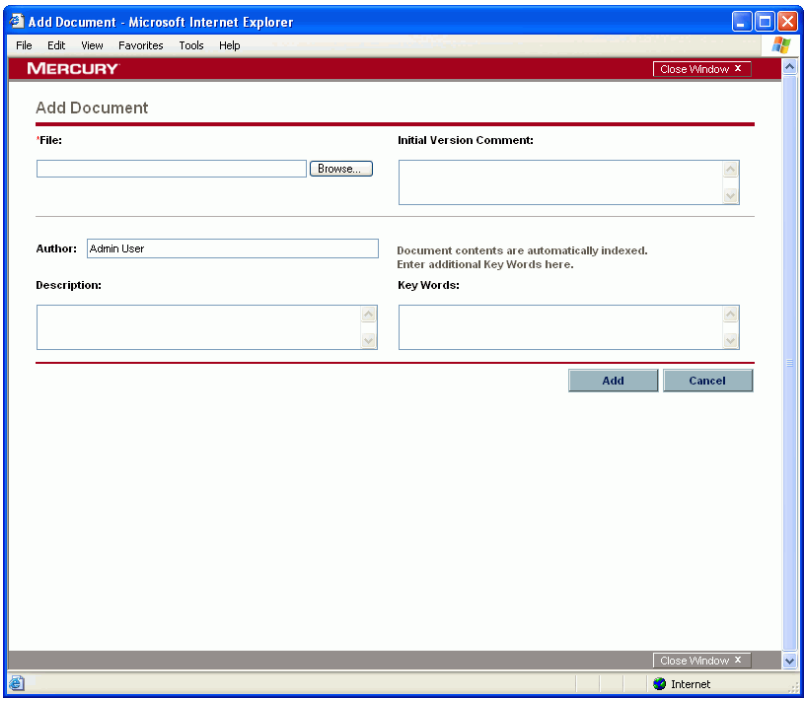
You can also attach a document to an entity that you are creating and have not yet submitted.

2. At the bottom of the page, expand the **References** section.



3. In the **New Reference** list, leave **Attachment** selected.

4. Click **Add.**



The Add Document window opens.

5. In the **File** field, enter the full directory path of the file to attach.
6. You can specify the following optional information:
   - In the **Initial Version Comment** field, type notes on the initial version of the document you are attaching.
   - In the **Author** field, type the name of the document author or authors.
   - In the **Description** field, type a description of the document and its purpose.
   - In the **Key Words** field, type keywords to add to an index of document contents.

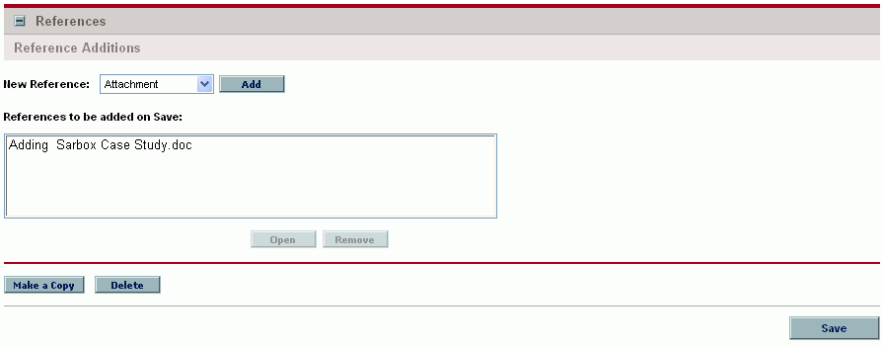
     The keywords you add to attachments help users search for entities with attachments that contain those words.

Note

Because the Content Server automatically indexes the contents of text-based files, there is no need to specify keywords for text-based documents. However, users cannot search non-text attachments such as image files unless you specify keywords.

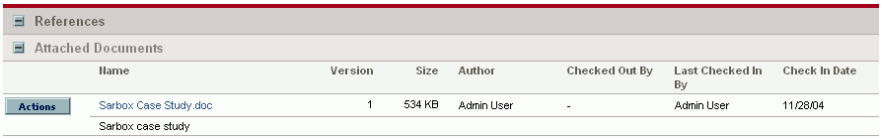
For information about the full-text indexing feature, see the section “The Indexes” in the Documentum document *Content Server Administrator’s Guide.*

7. Click **Add.**



The **References to be added on Save** field lists the document file you specified.

8. Click **Save.**



The document, which was loaded into the document management system after you clicked **Save,** is now listed in the **Attached Documents** section of the entity page.

The information displayed for the attached document also includes:

- Document version, size, and author
- Who (if anyone) has the document checked out
- When and by whom the document was last checked in

If Mercury IT Governance Center is running, and the Content Server stops, users can continue to use Mercury IT Governance Center, but cannot add or access attachments until the Content Server is up and running again.

Note

Any errors that occur while the Mercury IT Governance Server communicates with the Content Server are recorded in a log file. Server log files are stored in the `<ITG_Home>/server/kintana/log` directory. Server log files are named `serverLog.txt` and `serverLog_timestamp.txt`.

Active Mercury IT Governance Servers log their output to the `serverLog.txt` file. The `serverLog_timestamp` files are archived versions of the `serverLog.txt` file. For more information about Mercury IT Governance Server log files, see "Getting Information from Log Files," in Chapter 8, "Maintaining the System," in *System Administration Guide and Reference.*

Editing Document Attachment Information

You can edit document attachment information in one of the following two ways:

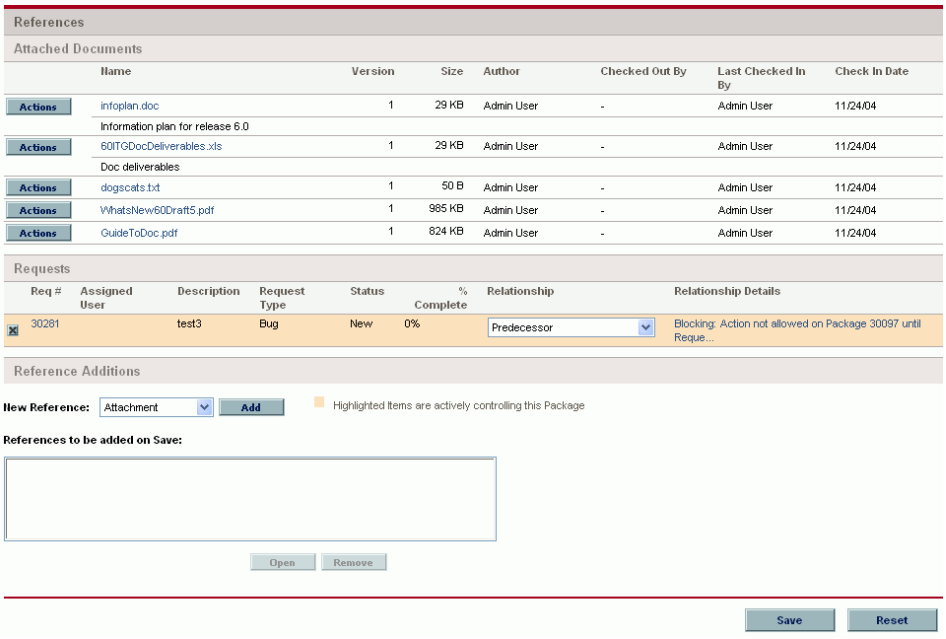
- If the entity has a custom attachment field, you can edit the information from that field.
- You can edit the information from the **References** section of the entity page.

  For detailed information about this procedure, see the following sections.

Editing Document Attachment Information

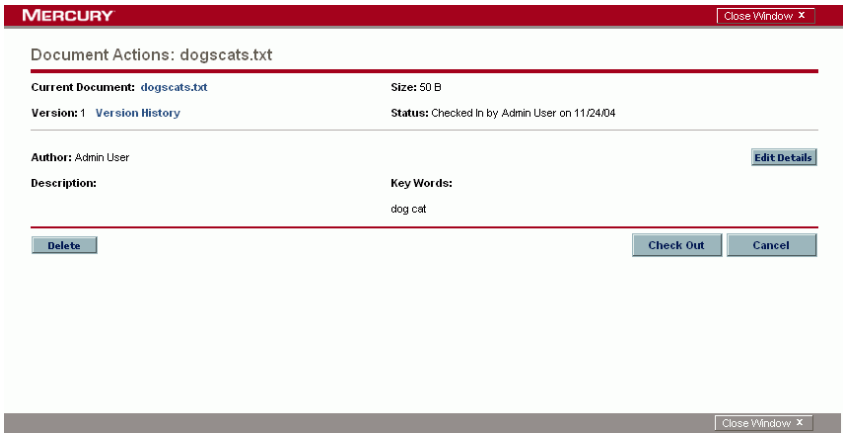
To edit document attachment information from the References section of an entity page:

1. In the standard interface, on an entity page, expand the **References** section.

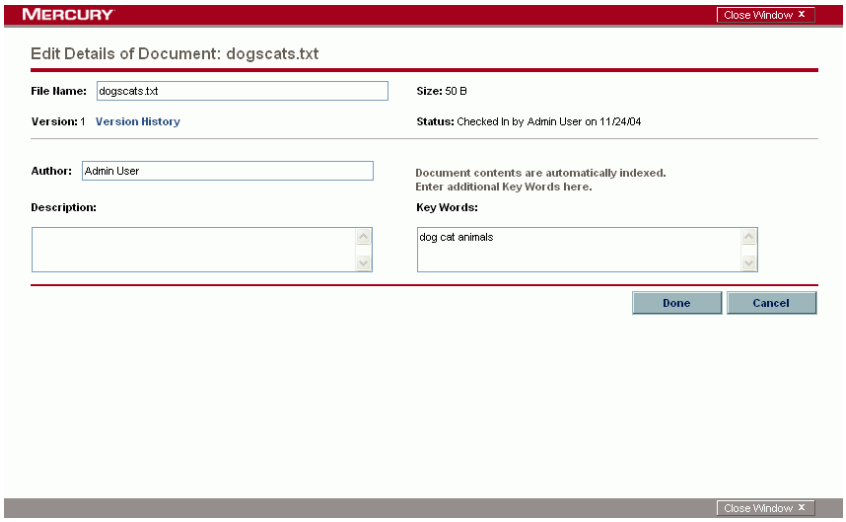


2. Under **Attached Documents,** to the left of the name of the document that has associated attachment information you want to edit, click **Actions.**

   The Document Actions window opens. From this window, you can view document information, open the Edit Details window, check out the document, or remove the document from the **Attached Documents** section.



3. Click **Edit Details.**



The Edit Details of Document window opens. You can use this window to change descriptive information about the document.

4. Make the required changes to the document information.

5. Click **Done.**

Checking Attached Documents Out and In

To check an attached document in or out, use one of the following methods:

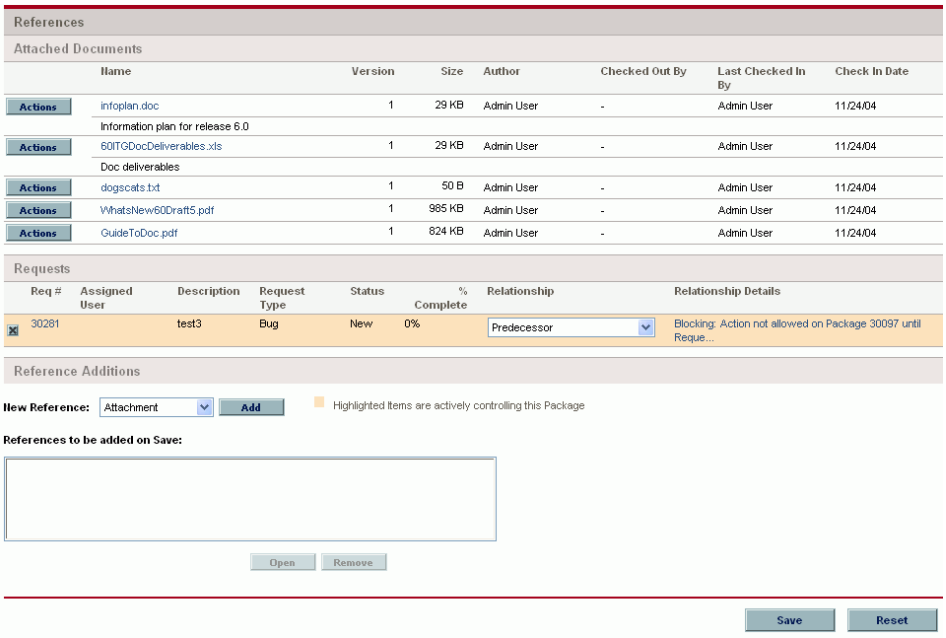
- If the entity has a custom attachment field, use that field to check the document in or out.
- Use the **References** section of the entity page.

Checking a Document Out and In from the References Section

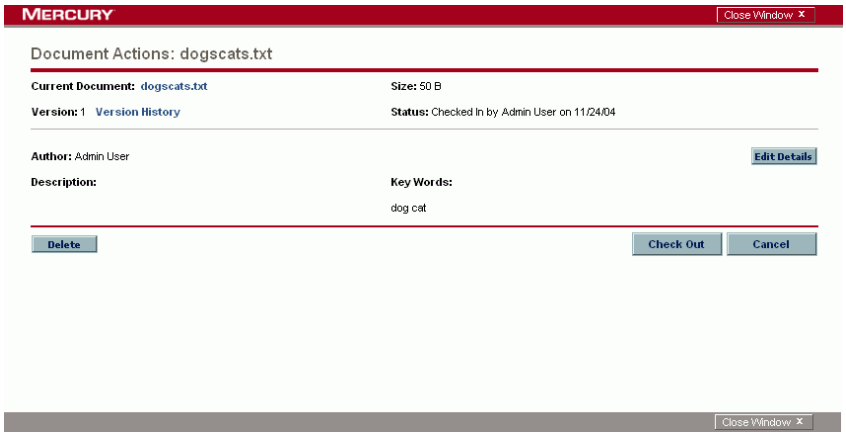
To check out an attached document, edit it, and check it back in:

1. In the standard interface, on an entity page, expand the **References** section.

   To expand a page section, to the left of the section heading, click **Restore** .



2. Under **Attached Documents,** to the left of the name of the document you want to check out, click **Actions.**



   The Document Actions window opens.

3. Click **Check Out.**

   The document opens for editing.

4. After you finish making changes, save and close the document.

   The Document Actions window now displays the **Check In** button.

5. Click **Check In.**

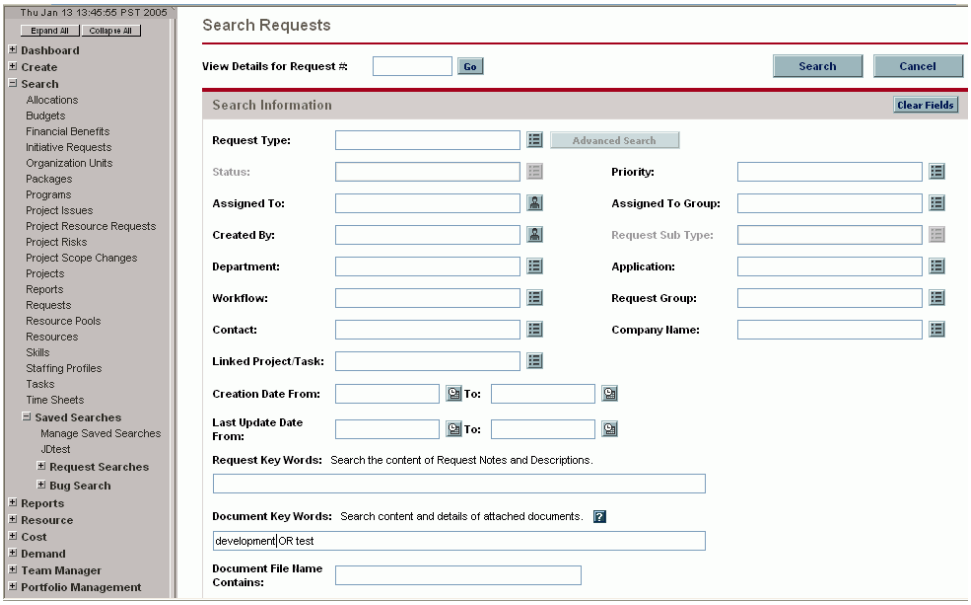
   Note that, in the **References Added** section of the entity page, the document version number displayed is now incremented by one.

Searching for Entities by Document Key Word

In the Mercury IT Governance Center standard interface, you can search for entities by document key words. The Content Server searches the descriptive fields for the document (author, description, title, and key words) and content of the document. The descriptive field searches are relevant to all documents types (text and binary). Content searches apply only to text-based documents.

To search for an entity using document key words:

1. Open a page for an entity that supports document management.



2. In the **Document Key Words** field, type one or more words to use as search criteria, separated by a space.

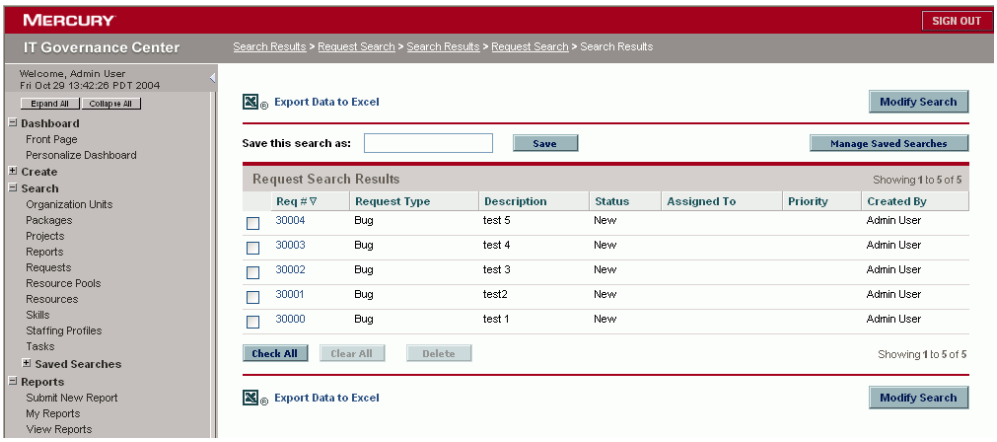
   By default, the ***<entity name>* Search Results** section only lists attached documents that include all of your search terms.For example, a search for "development test" is treated as "development and test." A document must have both "development" and "test" in its content or its descriptive fields to qualify as a match. To search for documents that contain either "development" or "test," type **`development OR test.`**

   For information about other ways to specify search terms, see *Specifying Search Terms* on page 165.

Note

Keyword searches are not case-sensitive.

3. Scroll to the bottom of the entity page and click **Search.**



The search returns a list of all entities (of the selected type) that have one or more attached documents containing key words that match your search terms. A document that you just attached may not show up on the Search Results page for several minutes. Before a content search can find a document, the document content must first be indexed. Although indexing is automatic, the process is periodic, and so may require several minutes to complete.

Specifying Search Terms

In addition to searches based on the AND and OR operators, you can search for exact phrases, exclude documents based on a key word, or search by combining queries. This section provides information on how to specify the key words for these search types.

Searching by Phrase

To search for an exact phrase, type double quotation marks (“key words”) at either end of the phrase. The content server returns a list of entities with attached documents that include all of the words inside the quotation marks, in the same order as you typed them.

Excluding Documents that Contain a Specific Text String

To exclude documents that contain a particular key word, type a minus character (-) in front of the key word. For example, to include documents with “development” or “test,” but not those with “production,” type **`development OR test -production.`**

Combination Queries

You can use any of the search formats described above in combination. OR queries take precedence over AND queries. For example, if you search for **finance development OR test,** the search first finds documents that contain either the word "development" or the word "test," and then finds documents from that list that also contain the word "finance."

Tokens Associated with Document Management

Table 7-1 lists the tokens related to document management. You can use these tokens to reference documents, version history, and metadata. Except for `DOC_HISTORY`, these tokens are also valid in systems without document management.

Warning

These tokens only work for custom fields, and not for reference attachments.
These tokens do not support client-side token parsing.

Table 7-1. Tokens associated with document management (page 1 of 2)

Token	Description
DOC_LINK	Resolves to a URL which, when clicked, opens the latest version of the document. Forces user authentication before delivering the document.
DOC_HISTORY	Resolves to a URL which, when clicked, displays a view of the document's version history. Forces user authentication before delivering the information.
AUTHOR	Resolves to the author descriptive field stored with the document.
DESCRIPTION	Resolves to the descriptive field stored with the document.

Table 7-1. Tokens associated with document management (page 2 of 2)

Token	Description
LAST_CHECK_IN_DATE	Resolves to the timestamp of the last check-in.
LAST_CHECKED_IN_BY_ NAME	Resolves to the full name of the Mercury IT Governance Center user who added or last checked in the document.
LAST_CHECKED_IN_BY	Resolves to the ID of the Mercury IT Governance Center user who added or last checked in the document.

For more information about tokens and how to use them, see the Mercury document *Commands, Tokens, and Validations Guide and Reference.*

Index

I

K

L

M

O

P

Q

R

S

T

U

V

W