

IBM WebSphere Information Integrator
OmniFind Edition



Administering Enterprise Search

Version 8.3

IBM WebSphere Information Integrator
OmniFind Edition



Administering Enterprise Search

Version 8.3

Before using this information and the product it supports, be sure to read the general information under "Notices."

This document contains proprietary information of IBM. It is provided under a license agreement and Copyright law protects it. The information contained in this publication does not include any product warranties, and any statements provided in this manual should not be interpreted as such.

You can order IBM publications online or through your local IBM representative:

- To order publications online, go to the IBM Publications Center at www.ibm.com/shop/publications/order.
- To find your local IBM representative, go to the IBM Directory of Worldwide Contacts at www.ibm.com/planetwide.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 2004, 2005. All rights reserved.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

About these topics vii

Who should read these topics vii

What is enterprise search? 1

Data source types supported by enterprise search 2

Enterprise search component overview 2

Enterprise search crawlers 3

Enterprise search parsers 4

Enterprise search indexes 6

Search servers for enterprise search 7

Enterprise search administration console 8

Monitoring an enterprise search system 9

Enterprise search log files 9

Customizing enterprise search 10

Sample search application for enterprise search 11

The enterprise search data flow 11

Enterprise search administration

overview 15

Logging in to the administration console 18

Changing the enterprise search administrator

password in a single server configuration 19

Changing the enterprise search administrator

password in a multiple server configuration 21

Changing the port number for the enterprise search

system 24

Enterprise search collections 27

Creating a collection by using the Collection wizard 27

Creating a collection by using the Collections view 28

Editing a collection 30

Deleting a collection 31

Enterprise search crawler

administration 33

Creating a crawler 35

Editing crawler properties 36

Editing a crawl space 37

Deleting a crawler 37

Content Edition crawlers 38

Server mode access to WebSphere II Content

Edition repositories 39

Direct mode access to WebSphere II Content

Edition repositories 39

Configuring the crawler server on UNIX for

WebSphere II Content Edition 40

Configuring the crawler server on Windows for

WebSphere II Content Edition 41

DB2 crawlers 42

Configuring WebSphere Information Integrator

Event Publisher Edition for DB2 crawlers 43

Configuring WebSphere MQ for DB2 crawlers 45

Configuring the crawler server on UNIX to use

event publishing 47

Configuring the crawler server on Windows to

use event publishing 48

DB2 Content Manager crawlers 49

Configuring the crawler server on UNIX for DB2

Content Manager 50

Configuring the crawler server on Windows for

DB2 Content Manager 51

Domino Document Manager crawlers 52

Exchange Server crawlers 54

Verifying access to secure Exchange Server

documents 54

NNTP crawlers 55

Notes crawlers 55

Tips for crawling Lotus Domino databases 57

Configuring the crawler server on UNIX to crawl

Lotus Domino sources 58

Configuring the crawler server on Windows to

crawl Lotus Domino sources 60

Configuring servers that use the DIIOP protocol 61

Configuring the I/O completion port on AIX to

crawl Lotus Domino sources 62

QuickPlace crawlers 64

Configuring the QuickPlace server to use Local

User security 65

Configuring Directory Assistance on a

QuickPlace server 66

UNIX file system crawlers 67

Web crawlers 67

User agent configuration 68

Support for JavaScript 69

Rules to limit the Web crawl space 69

Tests of URL connections with the Web crawler 73

Recrawl interval settings in the Web crawler 74

Options for visiting URLs with the Web crawler 74

How the Web crawler handles soft error pages 74

Support for crawling secure Web sites 76

Web sites that are served by proxy servers 78

Cookie administration 79

Global Web crawl space configuration 80

No-follow and no-index directives 82

Overriding no-follow and no-index directives in

Web pages 83

WebSphere Portal crawlers 83

Deploying the enterprise application for the

WebSphere Portal crawler 84

Copying the URL to crawl from WebSphere

Portal 85

Windows file system crawlers 86

Configuring support for Data Listener applications 87

Custom crawler plug-ins 88

URI formats in an enterprise search index 89

Enterprise search parser

administration 97

Working with categories 98

| | |
|---|------------|
| Rule-based categories | 99 |
| Model-based categories | 101 |
| Category trees | 101 |
| Selecting the categorization type | 102 |
| Configuring categories | 103 |
| Working with XML search fields | 105 |
| XML search fields | 105 |
| Mapping XML elements to search fields | 105 |
| Working with HTML search fields | 107 |
| HTML search fields | 107 |
| Mapping HTML metadata elements to search fields | 107 |
| Custom text processing | 108 |
| Adding text analysis engines to the system | 110 |
| Associating a text analysis engine with a collection | 111 |
| Mapping XML elements to a common analysis structure | 111 |
| Mapping a common analysis structure to the index | 113 |
| Mapping a common analysis structure to JDBC tables | 113 |
| Configuring threads for the parser service | 114 |
| Enabling advanced analysis for compound terms | 115 |
| Enabling support for native XML search | 116 |
| Linguistic analysis of Chinese, Japanese, and Korean documents | 116 |
| N-gram segmentation | 117 |
| Removing new line characters from non-ASCII character ranges | 117 |
| Document types associated with collection parsers and Stellent sessions | 118 |
| Associating document types with a collection parser | 118 |
| Default collection parser service rules | 120 |
| Associating document types with a Stellent session | 120 |
| Default parsing rules for Stellent sessions | 123 |
| Enterprise search index administration | 125 |
| Scheduling index builds | 126 |
| Changing the index schedule | 127 |
| Enabling and disabling the index schedules | 127 |
| Configuring concurrent index builds | 128 |
| Options that influence the searchable view of the index | 129 |
| Indexed options for searching documents | 129 |
| Wildcard characters in queries | 131 |
| Configuring options for wildcard characters in queries | 133 |
| Scopes | 134 |
| Configuring scopes | 135 |
| Collapsed URIs | 136 |
| Collapsing URIs in the search results | 137 |
| Removing URIs from the index | 137 |
| Search server administration for enterprise search | 139 |
| Search caches | 140 |

| | |
|--|-----|
| Configuring a search cache | 140 |
| Custom synonym dictionaries | 140 |
| Adding synonym dictionaries to the system | 142 |
| Associating a synonym dictionary with a collection | 142 |
| Custom stop word dictionaries | 143 |
| Adding stop word dictionaries to the system | 144 |
| Associating a stop word dictionary with a collection | 144 |
| Dynamic summarization | 145 |
| Customizing document summaries in the administration console | 145 |
| Customizing document summaries by editing properties | 146 |
| Working with quick links | 146 |
| Quick links | 147 |
| Configuring quick links | 147 |

Document ranking in enterprise search 149

| | |
|---|-----|
| Text-based scoring | 149 |
| Static ranking | 150 |
| Custom boost word dictionaries | 150 |
| Adding boost word dictionaries to the system | 151 |
| Associating a boost word dictionary with a collection | 152 |
| Document ranking that is based on URI patterns | 152 |
| Influencing the scores of documents that match URI patterns | 153 |
| Document ranking that is based on boost classes | 154 |
| Mapping fields to boost classes | 156 |
| Configuring boost factors for boost classes | 157 |
| Default boost class values | 157 |

Search applications for enterprise search 161

| | |
|---|-----|
| Associating search applications with collections | 162 |
| Sample search application functions | 162 |
| Sample search application properties | 163 |
| Editing the sample search application properties | 172 |
| Accessing the sample search application | 173 |
| Enabling security for the sample search application | 174 |

Enterprise search external sources 177

| | |
|---|-----|
| Adding external sources to the system | 177 |
| Associating search applications with external sources | 178 |

Enterprise search security. 181

| | |
|--|-----|
| Administrative roles | 182 |
| Configuring administrative users | 183 |
| Authentication versus access control | 184 |
| Disabling security for an enterprise application in WebSphere Application Server | 185 |
| Collection-level security | 186 |
| Duplicate document analysis | 186 |
| Anchor text analysis | 187 |
| Indexing the anchor text in links to forbidden documents | 187 |
| Security with search application IDs | 188 |

| | | | |
|--|------------|--|------------|
| Document-level security | 189 | Monitoring index activity for a collection | 230 |
| Validation by stored security tokens | 189 | Monitoring the enterprise search index queue | 231 |
| Validation of current credentials during query processing | 190 | Monitoring the search servers | 232 |
| Enforcement of document-level security for Windows file system documents | 193 | Monitoring the Data Listener | 233 |
| Enforcement of document-level security for Lotus Domino documents | 195 | Document tracking | 234 |
| Disabling document-level security | 197 | Configuring log files for document tracking | 235 |
| | | Viewing reports about dropped documents | 235 |
| Enterprise search integration with WebSphere Portal | 199 | Enterprise search log files and alerts | 237 |
| Document-level security with the Portal Search Engine | 201 | Alerts | 237 |
| Deploying the Search portlet | 201 | Configuring collection-level alerts | 238 |
| Configuring the WebSphere Portal Search and Browse portlet for enterprise search | 203 | Configuring system-level alerts | 239 |
| Installing the enterprise search adapter for the Search Center | 204 | Configuring log files | 240 |
| Installing the enterprise search registration portlet for the Search Center | 205 | Configuring SMTP server information | 241 |
| | | Receiving e-mail about logged messages | 242 |
| | | Viewing log files | 244 |
| Migration from WebSphere Portal to enterprise search. | 207 | Backing up and restoring an enterprise search system | 245 |
| Migrating a model-based taxonomy from WebSphere Portal | 207 | Backing up the enterprise search system | 245 |
| Migrating a collection from WebSphere Portal | 209 | Restoring the enterprise search system | 246 |
| Migrated collection settings. | 210 | Restoring enterprise search system files to new servers | 247 |
| Migration wizard log file | 212 | Enterprise search commands, return codes, and session IDs | 249 |
| Starting and stopping the enterprise search servers | 213 | Enterprise search documentation. | 277 |
| Starting the enterprise search servers | 213 | WebSphere II OmniFind Edition accessibility | 279 |
| Stopping the enterprise search servers | 215 | Glossary of terms for enterprise search | 281 |
| Monitoring enterprise search activity | 217 | Accessing information about WebSphere Information Integration | 291 |
| Estimating the number of documents in a collection | 217 | Providing comments on the documentation. | 293 |
| Checking the availability of system resources. | 218 | Contacting IBM | 295 |
| Monitoring a collection | 219 | Trademarks | 297 |
| Viewing details about a URI | 220 | Index | 303 |
| Monitoring crawlers | 221 | | |
| Viewing details about Web crawler activity | 223 | | |
| Web crawler thread details | 223 | | |
| Web crawler active sites | 224 | | |
| Web crawler crawl rate | 225 | | |
| Creating Web crawler reports | 225 | | |
| Web crawler HTTP return codes | 227 | | |
| Monitoring the parser | 229 | | |

About these topics

Use this information when you administer an IBM® WebSphere® Information Integrator OmniFind™ Edition Version 8.3 system.

WebSphere Information Integrator OmniFind Edition (WebSphere II OmniFind Edition) provides a technology called *enterprise search*. The components for enterprise search are installed when you install the WebSphere II OmniFind Edition product. The term *enterprise search* is used throughout the WebSphere II OmniFind Edition documentation except for references to installation paths and product packaging labels.

The administration documentation for enterprise search covers the following topics:

- An introduction to enterprise search concepts, components, and capabilities
- Instructions on how to create, monitor, and administer collections
- Information about how data is parsed, and the ways that you can customize parsing activities to optimize search and retrieval
- Information about how parsed data is indexed, and the ways that you can administer indexing activity
- Information about how the search servers find, rank, and return search results
- Information about associating your custom search applications with collections
- An overview of the different levels of security that are available in enterprise search
- Information about how enterprise search integrates with IBM WebSphere Portal
- Instructions on how to migrate WebSphere Portal taxonomies and collections to enterprise search category trees and collections
- Instructions for backing up and restoring the system
- Instructions for creating and viewing log files
- Instructions on how to use enterprise search commands and interpret the information that is returned

Who should read these topics

This information is intended for system administrators and system operators who are responsible for creating, monitoring, and administering enterprise search collections.

Use this information to create collections, select content for the collection, and configure options for making the content searchable. Also use this information to monitor collection and system activity, enroll users as enterprise search administrators, and associate collections and external searchable sources with search applications.

To use this information effectively, you must be familiar with Web applications and have experience with the data sources that you want to search.

What is enterprise search?

An enterprise search system provides extensive capabilities for searching any number of structured and unstructured data sources with a single query. The enterprise search system provides fast query response times and a consolidated, ranked result set that enables you to easily locate the information that you need.

The enterprise search components, which are installed with IBM WebSphere Information Integrator OmniFind Edition (WebSphere II OmniFind Edition), collect information from throughout your enterprise and make that information available for searching. By entering a query in a Web browser, you can search local and remote databases, collaboration systems, content management systems, file systems, and internal and external Web sites at the same time.

Designed to integrate seamlessly with your existing systems, an enterprise search system handles the logistics that are required to collect data from diverse sources and index the data for fast retrieval. By applying linguistic analysis and other types of analysis to the data, enterprise search can deliver highly relevant search results. You do not need to learn different interfaces to search various repository types.

You can add support for searching data sources that you do not want to include in an enterprise search index. With the federated search capability of enterprise search, you can search these external sources at the same time that you search indexed data sources.

Search quality

To ensure that users find the information that they seek, WebSphere II OmniFind Edition supports the IBM Unstructured Information Management Architecture (UIMA). UIMA is an open framework that defines a common, standard interface for text analytics. With extensive text analysis, enterprise search can identify concepts, latent meanings, relationships, facts, and other relevant data that is often hidden in unstructured text. The information that is extracted during analysis can be used to enhance the quality of search results, or be used to enhance the quality of other applications, such as business intelligence and data mining.

Security

Security is an integral element for enterprise search. Only users who are authorized to administer the system can do so. With the security mechanisms available in IBM WebSphere Application Server, you can configure administrative roles and control which users have access to various administrative functions.

You can also specify options to associate security tokens with data as the data is being collected. If your search applications enable security, you can use these tokens, which are stored with documents in the index, to enforce access controls and ensure that only users with the proper credentials are able to query the data and view search results.

For certain types of data sources, you can configure options to validate a user's login credentials with current access controls during query processing. This extra layer of security ensures that a user's privileges are validated in real time with the

native data source. This option can protect against instances in which a user's credentials changed after a document and its security tokens were indexed.

Related concepts

Enterprise search security

"Custom text analysis integration" in "Text Analysis Integration"

"Unstructured information management architecture (UIMA)" in "Text Analysis Integration"

Data source types supported by enterprise search

Predefined support is available for searching a variety of data source types.

After you install IBM WebSphere Information Integrator OmniFind Edition, you can begin collecting data from the following types of data sources:

- IBM DB2[®] Content Manager item types (documents, resources, and items)
- IBM DB2 Universal Database[™] databases
- IBM Domino[®] Document Manager (formerly Domino.Doc[®]) databases
- IBM Lotus Notes[®] databases
- IBM Lotus[®] QuickPlace[®] databases
- Microsoft[®] Exchange Server public folders
- Microsoft Windows[®] file systems
- Network news transfer protocol (NNTP) news groups
- UNIX[®] file systems
- IBM WebSphere Information Integrator Content Edition repositories: Documentum, FileNet Panagon Content Services, FileNet P8 Content Manager, Hummingbird[®] Document Management (DM), OpenText Livelink, and Portal Document Manager (PDM) item classes
- IBM WebSphere Information Integrator nickname tables for IBM DB2 Universal Database for z/OS[®], IBM Informix[®], Microsoft SQL Server, Oracle, and Sybase databases
- IBM WebSphere Portal sites
- Web sites on the Internet or in your intranet

You can also add support for searching the following types of external sources without adding documents from these sources to the enterprise search index:

- Java[™] database connectivity (JDBC) database tables (IBM DB2 Universal Database (DB2 UDB) and Oracle databases only). A separate external source is created for each table in a JDBC database.
- Lightweight Directory Access Protocol (LDAP) servers. One external source is created for each LDAP server.

For the latest information about supported data source types, see the WebSphere Information Integrator OmniFind Edition Web site.

Related concepts

Enterprise search external sources

Enterprise search component overview

The enterprise search components collect data from throughout your enterprise; analyze, parse, and categorize the information; and create an index that users can search.

An enterprise search *collection* represents the set of sources that users can search with a single query. When you create a collection, you specify which sources you want to include and configure options for how users can search the indexed data.

You can create multiple collections, and each collection can contain data from a variety of data sources. For example, you might create a collection that includes documents from IBM DB2 Universal Database, IBM Lotus Notes, and IBM DB2 Content Manager databases. When users search this collection, the search results potentially include documents from each of the data sources.

Support for federated searching enables users to search more than one collection with a single query. The search results potentially include documents from all collections and external sources in your enterprise search system.

Creating and administering a collection involves the following activities:

Collecting data

The *crawler* components collect documents from data sources, either on a continual basis or according to a schedule that you specify. Frequent crawling ensures that users always have access to the latest information.

Analyzing data

The *parser* components extract text from documents, and do linguistic analysis and other types of analysis on each document that a crawler crawls. The detailed content analysis improves the quality of search results.

Indexing data

The *index* components run on a regularly scheduled basis to add information about new and changed documents to the index. The index components also do global analysis of the documents in a collection to enhance the quality of the search results.

Searching data

The *search* components search the index and work with your search applications to process search requests and return search results.

Other WebSphere Information Integrator OmniFind Edition components enable you to specify security preferences, monitor system activity, and troubleshoot problems that occur. The product also provides a working sample search application that you can use as a template for creating your own search applications.

Related concepts

- Enterprise search crawler administration
- Enterprise search parser administration
- Enterprise search index administration
- Search server administration for enterprise search

Enterprise search crawlers

Enterprise search crawlers collect documents from data sources so that the documents can be analyzed, indexed, and searched.

The crawler component that is provided with WebSphere Information Integrator OmniFind Edition has the following functions:

- When you configure a crawler, the *discovery* processes find information about the sources that are available to be crawled, such as the names of all of the views and folders in a Lotus Notes database or the names of all file systems on a UNIX server.
- After you select the sources that you want to crawl and start the crawler, the crawler components collect data from the sources so that the data can be analyzed and indexed.

A single collection can have multiple crawlers, and each crawler is designed to gather data from a particular type of data source. For example, you might create three crawlers to combine data from file systems, Notes® databases, and relational databases in the same collection. Or, you might create several crawlers of the same type, and set up different crawling schedules for them according to how frequently the data that is being crawled by each crawler changes.

The crawlers for Web, WebSphere Portal, and Network News Transfer Protocol (NNTP) sources run continuously. After you specify which uniform resource locators (URLs) or NNTP news groups you want to crawl, the crawler returns periodically to check for data that is new and changed. You can start and stop other types of crawlers manually, or you can set up crawling schedules. If you schedule a crawler, you specify when it is to run initially and how often it needs to visit the data sources to crawl new and changed documents.

Crawler properties are a set of rules that govern the behavior of a particular crawler when it crawls. For example, you specify rules to control how the crawler uses system resources. The set of sources that is eligible to be crawled constitutes the *crawl space* of a crawler. After you create a crawler, you can edit the crawler properties at any time to alter how the crawler collects data. You can also edit the crawl space to change the crawler schedule, add new sources, or remove sources that you no longer want to search.

Related concepts

Enterprise search crawler administration

Related tasks

Monitoring crawlers

Enterprise search parsers

An enterprise search parser analyzes documents that were collected by a crawler and prepares them for indexing.

The parser component that is provided with WebSphere Information Integrator OmniFind Edition analyzes document content and document metadata. It stores the results of the analysis in a data store for access by the indexing component. The parser does the following tasks:

- Extracts text from whatever the format a document is in. For example, the parser extracts text from the tags in XML and HTML documents. By using Stellent for IBM WebSphere Information Integrator OmniFind Edition Outside In Viewer Technology, the parser also extracts text from binary formats such as Microsoft Word and Adobe Acrobat portable document format (PDF) documents.
- Detects the character set encoding of each document. Before doing any linguistic analysis, the parser uses this information to convert all text to Unicode.
- Detects the source language of each document.
- Applies parsing rules that you specify for the collection. When you configure the parser, you can configure:

Field mapping rules for XML and HTML documents

This option enables users to search structured and unstructured content in XML and HTML documents. If you map XML elements or HTML metadata elements to search fields in the enterprise search index, users can specify the field names in queries and search specific parts of XML and HTML documents. (Queries that search specific fields can provide more precise search results than free text queries that search all document content.)

Categories

This option enables users to search documents by the categories that the documents belong to. Users can also select categories in the search results and browse only documents that belong to that same category.

When you create a collection, you choose the type of categories that you want to use, if any. If you use *rule-based* categories, documents are associated with categories according to rules that you define. You can configure rule-based categories with enterprise search collections that you create and with collections that you migrate from IBM WebSphere Portal.

If you use *model-based* categories, documents are associated with model-based categories that exist in your WebSphere Portal system. To use this option, WebSphere Portal must be installed on the enterprise search index server. You must also use the categorization tools in WebSphere Portal to administer the categories.

Custom text analysis

Application developers can create custom analysis programs to perform complex linguistic analysis of the data that you need to search. You can plug these programs into the enterprise search system and use them to annotate the content of your collections. By indexing the annotations, you enable collections for semantic search.

For example, users can search for query terms that occur in proximity to each other or that occur in the same sentence, or they can search for relationships between query terms (such as documents that discuss an IBM salesperson named Smith, not an IBM engineer named Smith).

Support for n-gram segmentation

To enhance the retrievability of documents that were written in Chinese, Japanese, or Korean, you can enable the n-gram segmentation method of lexical analysis. This form of analysis does not use white space to delimit words. (You cannot change the segmentation method after you create a collection.)

Support for searching XML documents with native XML search

A native XML search can provide more precise search results by searching XML markup. For example, a query might specify that a word must occur in a particular XML element.

Classes to boost the relative importance scores of fields

When you map fields to boost classes, you can influence how documents are ranked in the search results. For example, you might want to boost the score of title fields to ensure that when a query term occurs in the title, documents with that term in their titles are ranked higher in the search results.

- Extracts text and adds tokens to enhance the retrievability of data. During this phase, the parser does the following tasks:

- Character normalization, such as normalizing capitalization and diacritical marks such as the German umlaut.
- Analyzing the structure of paragraphs, sentences, words, and white space. Through linguistic analysis, the parser decomposes compound words and assigns tokens that enable dictionary and synonym lookup.

Related concepts

Working with categories

XML search fields

HTML search fields

"Custom text analysis integration" in "Text Analysis Integration"

"Text analysis included in enterprise search" in "Text Analysis Integration"

Related tasks

Monitoring the parser

Enterprise search indexes

The enterprise search indexing components run on regular schedules to add information about new and changed documents to the index.

To ensure that users always have access to the latest information in the sources that they search, building an index involves two stages:

Reorganizing the index

When an index is reorganized, the entire index is rebuilt so that the structure has an optimal organization. The indexing processes read all of the data that was collected by crawlers and analyzed by the parser.

Refreshing the index

When an index is refreshed, information that was crawled since the last time the index was reorganized is added to the index.

When you configure index options for a collection, you can specify schedules for reorganizing and refreshing the index. The frequency with which you reorganize and refresh the index depends on your system resources and whether the sources being indexed contain static or dynamic content.

To ensure the availability of new information, schedule the index to be refreshed frequently. Periodically schedule a reorganization of the index to consolidate all of the new information, analyze new content, and optimize the performance of the index.

You can also start the indexing processes without scheduling them. For example, if you change certain parsing rules and want those changes to become available to your search applications, you can start an index reorganization after the data is recrawled and parsed instead of waiting for the index reorganization to start at its scheduled time.

To control resource usage, you control how many collections can share the indexing processes and submit index build requests at the same time. Building indexes concurrently helps ensure that the reorganization of a very large index does not block the refreshing of other indexes. Index building can be a resource-intensive process, so for large systems, you must monitor system loads to adjust the index reorganization and refresh rates.

When building an index, the indexing processes do global document analysis. During this phase, algorithms are applied to identify duplicate documents, to analyze the link structure of documents, and to do special processing on anchor text (the text that describes the target page in a hypertext link) in Web documents.

You can specify options for the following indexing activities:

- To enable users to specify wildcard characters, you can build support for expanding the query terms into the index, or you can specify that the query terms are to be expanded during query processing. The decision that you make involves a trade-off between resource usage and query response time.
- You can configure scopes. A *scope* enables you to limit what users can see in the collection. For example, you might create one scope that includes the URIs for documents in your Technical Support department and another scope for the URIs of documents in your Human Resources department. If the search application supports scopes, users can search and retrieve documents from only those subsets of the collection.
- You can specify options for collapsing search result documents that have the same URI prefix. You can also specify a group name so that documents with different URI prefixes can be collapsed together in the search results.
- After an index is built, you can remove URIs that you want to prevent users from searching.

Related concepts

Enterprise search index administration

Wildcard characters in queries

Scopes

Collapsed URIs

Document ranking that is based on URI patterns

Related tasks

Scheduling index builds

Configuring concurrent index builds

Removing URIs from the index

Monitoring index activity for a collection

Monitoring the enterprise search index queue

Search servers for enterprise search

The search servers for enterprise search work with your search applications to process queries, search the index, and return search results.

The search servers for enterprise search are installed when you install WebSphere Information Integrator OmniFind Edition. When you configure the search servers for a collection, you can specify options for how the collection is to be searched:

- You can configure a search cache to hold frequently requested search results. A search cache can improve search and retrieval performance.
- You can specify a default language for searching documents in the collection.
- If your application developers create custom dictionaries, you can associate the dictionaries with collections:
 - When users query a collection that uses a *synonym dictionary*, documents that contain synonyms of the query terms are included in the search results.
 - When users query a collection that uses a *stop word dictionary*, the stop words are removed from the query before the query is processed.

- When users query a collection that uses a *boost word dictionary*, the importance of documents that contain the boost words is increased or decreased, depending on the boost factor that is associated with the word in the dictionary.
- If you predetermine that certain documents are relevant to certain queries, you can configure quick links. A *quick link* associates a specific URI with specific keywords and phrases. If a query contains any of the keywords or phrases that specify in a quick link definition, the associated URI is returned automatically in the search results.

In a multiple server configuration, failure protection is available at the collection level, not just at the server level. If a collection on one search server becomes unavailable for any reason, then the queries for that collection are routed automatically to the other search server.

Related concepts

- Search applications for enterprise search
- Search caches
- Custom synonym dictionaries
- Custom stop word dictionaries
- Custom boost word dictionaries
- Quick links

Related tasks

- Monitoring the search servers

Enterprise search administration console

The enterprise search administration console runs in a browser, which means administrative users can access it from any location at any time. Security mechanisms ensure that only those users who are authorized to access administrative functions do so.

The administration console for enterprise search is installed on the search servers when you install WebSphere Information Integrator OmniFind Edition.

The administration console includes wizards that can help you do several of the primary administrative tasks. For example, the Collection wizard helps you create a collection and allows you to save your work in draft mode. Crawler wizards are specific to a data source type and help you select the sources that you want to enable users to search.

For other administrative tasks, you can select individual items that you want to administer. For example, when you edit a collection, you can select the Index page to change the index schedule or select the Parse page to modify a rule for parsing XML documents.

Related concepts

- Enterprise search administration overview
- Administrative roles

Related tasks

- Logging in to the administration console

Monitoring an enterprise search system

You can use the enterprise search administration console to monitor system activities and adjust operations as needed.

After you install WebSphere Information Integrator OmniFind Edition and create at least one collection, you can view detailed statistics for each major activity (crawling, parsing, indexing, and searching). The information includes average response times and progress information, such as how many documents were crawled or indexed during a specific crawl or index building session.

You can stop and start most activities. For example, you can pause an activity, change its configuration or troubleshoot a problem, and restart processing when you are ready to allow the activity to proceed.

You can also configure alerts, which enable you to receive e-mail about certain monitored activities whenever a monitored event occurs. For example, you can receive an alert if the search response time exceeds a specified threshold.

If a document was dropped from the enterprise search system, you can track the document and determine when, where, and why the document was dropped. For example, the parser might not be able to parse a document or an administrator might remove a document from the index.

Related concepts

- Monitoring enterprise search activity

- Starting and stopping the enterprise search servers

Enterprise search log files

Log files are created for individual collections and for system-level sessions.

When you configure logging options for an enterprise search collection or for the system, you specify the types of messages that you want to log (such as error messages and warning messages). You also specify how often you want the system to rotate older log files to make room for recent messages. You can choose options to receive e-mail about specific messages (including alerts), or all error messages, whenever they occur.

When you view log files, you select the log file that you want to view (the file name includes information about when the file was created and which component issued the messages). You can also specify viewing filters. For example, you can choose to see only error messages or only messages from a particular enterprise search session.

Related concepts

- Enterprise search log files and alerts

- Alerts

- Messages for enterprise search

Related tasks

- Configuring log files

- Configuring SMTP server information

- Receiving e-mail about logged messages

- Viewing log files

Customizing enterprise search

The application programming interfaces for enterprise search enable you to create custom search applications, custom applications to update the content of collections, custom programs for text analysis, and custom dictionaries for synonyms, stop words, and boost words.

After installing WebSphere Information Integrator OmniFind Edition, the following families of APIs are available for extending enterprise search collections:

Search and Index API (SI-API)

Use this API to build custom search applications and a custom administration interface.

Data Listener API

Use this API to receive data from external crawlers. The external crawlers can connect to the enterprise search Data Listener, then add data to a collection or remove data from a collection.

Crawler plug-ins

Use APIs for Web and non-Web crawlers to add metadata to documents while they are being crawled or to associate security tokens that enforce your organization's business and security rules.

You can enhance the retrievability of information by integrating custom programs for linguistic analysis with your enterprise search collections. After you add custom text analysis engines to the system, you can associate the engines with collections. When users query a collection, they benefit from the word associations that your custom programs build into the index. For example, users can search for concepts and relationships between terms, not just on the terms themselves.

You can also enhance the retrievability of information by integrating custom dictionaries that reflect, for example, acronyms, abbreviations, and vocabulary terms that are specific to your industry. After you add dictionaries to the system, you can associate the dictionaries with collections. When users query a collection, they benefit in the following ways:

- If a query includes words that are defined as synonyms, documents that contain synonyms of the query terms will be included in the search results.
- If a query includes stop words, the stop words will be removed from the query so that irrelevant documents are not returned in the search results.
- If a query includes boost words, documents that contain the boost words will be ranked higher or lower in the search results, depending on the boost value that is associated with the word in the dictionary.

Related concepts

Search applications for enterprise search

Custom synonym dictionaries

Custom stop word dictionaries

Custom boost word dictionaries

"Search and index API overview" in "Programming Guide and API Reference for Enterprise Search"

"Data listener" in "Programming Guide and API Reference for Enterprise Search"

"Crawler plug-ins" in "Programming Guide and API Reference for Enterprise Search"

Related tasks

Sample search application for enterprise search

You can use the sample search application for enterprise search as provided or use it as a template for developing custom search applications.

A sample search application is installed when you install WebSphere Information Integrator OmniFind Edition. The sample search application demonstrates most of the search and retrieval functions that are available for enterprise search. The application is also a working example that enables you to search all active collections and external sources in your enterprise search system. You can use the sample application to test new collections and external sources before you make the collections or external sources available to users.

The sample search application demonstrates support for federated search by enabling you to search one or more collections and external sources at a time.

If you enable security for a collection, and enable WebSphere II OmniFind Edition identity management for the system, users can create profiles when they use the search application. The user profile stores the credentials that users specify to log in to various domains.

During query processing, the search processes use the stored credentials (which are encrypted in a secure store that is managed by WebSphere II OmniFind Edition) to determine whether a user has permission to search the secure domains. If the credentials are missing or not valid for a domain, documents from that domain are excluded from the search results.

For information about using a sample search application, click **Help** while you are using the application. To create a custom search application, use the Search and Index API for enterprise search.

Related concepts

Search applications for enterprise search

Sample search application functions

"Search and index API overview" in "Programming Guide and API Reference for Enterprise Search"

Related tasks

Accessing the sample search application

Editing the sample search application properties

Enabling security for the sample search application

The enterprise search data flow

The enterprise search components that you install with WebSphere Information Integrator OmniFind Edition closely interact to ensure the flow of data through the system.

Crawlers gather documents from data sources throughout your enterprise. The parser extracts useful information from the crawled documents and generates tokens that can, for example, associate documents with categories and help determine the relevance of documents to the terms in a search request. The index stores the data for efficient retrieval.

By using a Web browser and a search application, users search indexed collections and external sources. The search application can display a list of results that users can click in a browser, or the application can be more sophisticated and return dynamically generated content that is based on information in different sources.

For example, a catalog search application can customize the display of products that satisfies a search request. A single query can search through documents from different types of data sources, such as a combination of documents from IBM DB2 Content Manager and Lotus Notes repositories.

Administrators determine what data will be collected and how it will be crawled, parsed, indexed, and searched. By monitoring system activity, administrators also make adjustments to optimize data throughput.

The following diagram shows the flow of information through an enterprise search system.

|

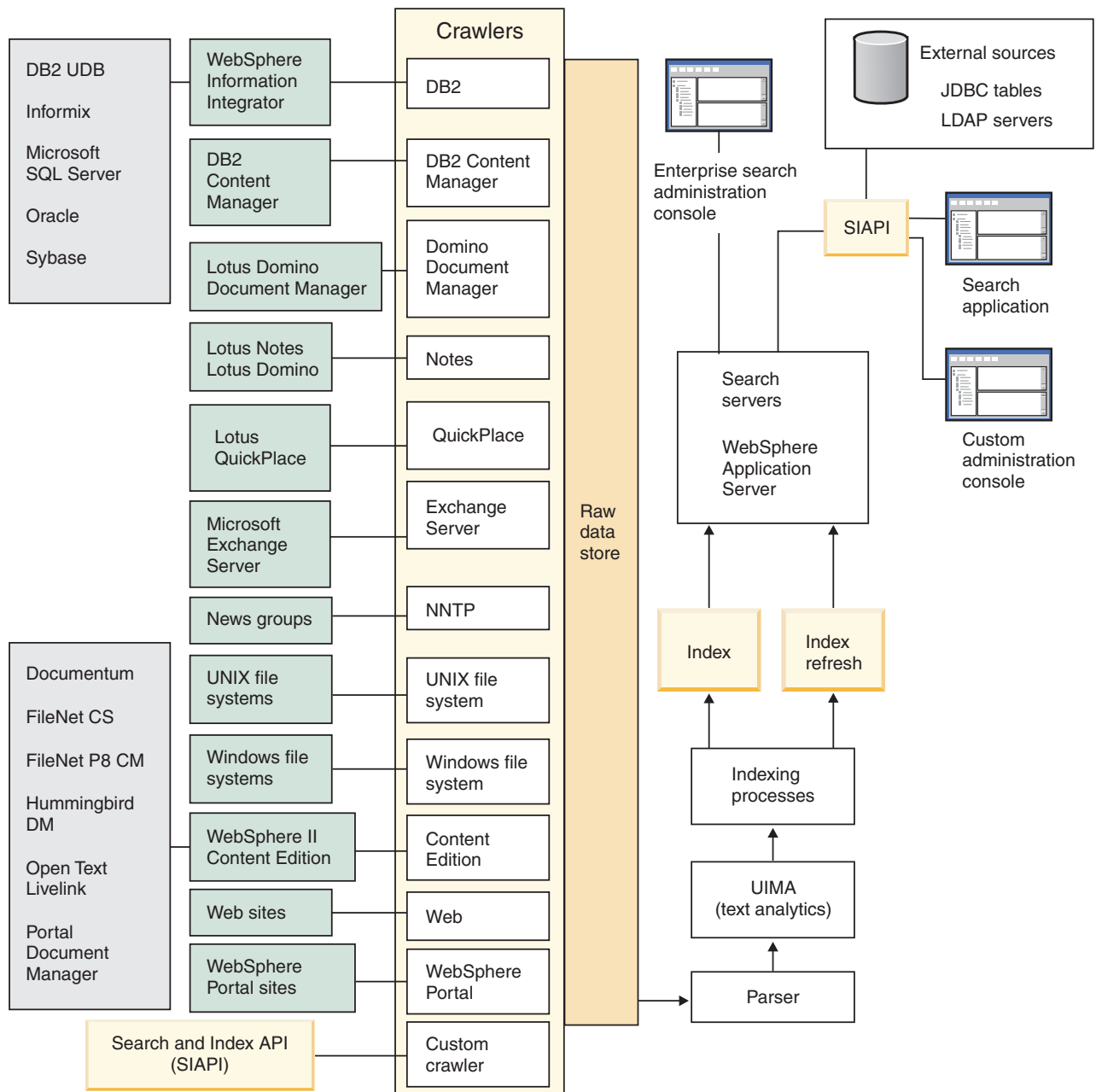


Figure 1. How data flows through an enterprise search system


Enterprise search administration overview

You use the enterprise search administration console to create and administer collections and external sources, start and stop components, monitor system activity and log files, configure administrative users, associate search applications with collections and external sources, and specify information to enforce security.

Collections view

Use the following guidelines to create your first collection and administer the system.

Log in Log in to the enterprise search administration console. The Collections view is the entry point for creating and administering collections.

Tip: For assistance with the administration console, click **Help** on the toolbar or **Help for this page** at any time. If detailed explanations and corrective actions are available for messages, you can click the  **More information** icon at the end of the message to see the details.


Create a collection

Choose one of the following approaches to create a new, empty collection:

- To create a collection by using the collection wizard, click **Collection Wizard** and follow the wizard prompts.
- To create a collection by using the Collections view, click **Create Collection**, fill in the fields on the Create a Collection page, then click **OK**.

Configure the collection

You must edit a new, empty collection to add content to it and to specify options for how you want to crawl data and make the data searchable.

Click  **Edit** for your new collection, then select a page to specify options for the collection.

Attention: If you click the Back or Refresh button in the Web browser, inconsistent results and a potential loss of data can occur. When you configure collections and need to go to the previous page or to refresh information in the administration console, click the **Previous** and **Refresh** buttons in the enterprise search administration console instead of the Back and Refresh buttons in the Web browser.

- On the General page, you can specify options that apply to the entire collection:
 - You can edit general options to change the name or description of the collection, or change the estimated size of the collection.
 - You can view information about the collection that you cannot change, such as the collection ID or the static ranking method for ranking documents in the search results.
 - If security was enabled for the collection when it was created, you can enable or disable document-level security controls.
- On the Crawl page, configure at least one crawler.

A single collection can contain data from a variety of data sources. You must configure at least one crawler for each type of data source that you

want to include. When you create a crawler, a wizard that is specific to the type of data being crawled helps you configure the crawler.

- On the Parse page, you can configure options for how the crawled data is to be parsed so that it can be effectively searched:
 - You can specify whether XML documents are to be parsed so that they can be searched with native XML search.
 - You can associate documents with categories, which enables users to search a subset of the collection or browse search result documents by the categories that they belong to.
 - You can map XML elements and HTML metadata elements to search fields in the index, which enables users to specify the field names in queries and search specific parts of documents.
 - If you added custom text analysis engines to the enterprise search system, you can choose one to use with the collection, and then specify text processing options to enhance the retrievability of information.
 - You can associate fields with boost classes to influence how fields that match the query terms are ranked in the search results.
- On the Index page, configure schedules for reorganizing and refreshing the index. Schedule the index builds to occur frequently so that your users always have access to the latest information. You can also do the following optional activities:
 - Enable users to specify wildcard characters in query terms.
 - Configure scopes, so that users search a limited part of the collection instead of all documents in the index.
 - Collapse search results, so that all documents from the same source are grouped and collapsed together in the search results.
 - Assign boost factors to influence how documents that match a URI pattern are ranked in the search results.
 - Remove URIs from the index. For example, you might need to prevent users from seeing certain documents after the collection is created.
- On the Search page, you can specify options for searching documents in the collection:
 - You can set aside cache space for search results and change the default language of the collection.
 - If you added custom dictionaries for synonyms, stop words, and boost words to the enterprise search system, you can select the dictionaries to use when users search the collection.
 - You can specify a display length for document summaries in the search results.
 - If you want specific URIs to appear automatically in the search results whenever a query includes particular keywords or phrases, you can configure quick links.
- On the Log page, you can do the following activities:
 - Specify options for the types of messages that you want to log and how often you want the log files to be recycled.
 - Specify options for receiving alerts about collection activity. For example, an alert can inform you when the average search response time is exceeding a specified limit.

- Specify options for receiving e-mail whenever certain messages or certain types of messages are logged.
- Specify options for logging information that enables you to determine when, where, and why a document was dropped from the enterprise search system.

Start the components

After you specify the data sources to crawl and options for collecting and searching the data, you can start the processes for building the collection. The order in which you start components is critical. Crawlers must crawl data before it can be parsed, parsers must analyze the crawled data before it can be indexed, and the index must be refreshed or reorganized before the search servers can start to process search requests.

External Sources view

If you want to search data sources without crawling or indexing them, you can click **External Sources** on the toolbar to specify options for making the data sources searchable. You must specify information that enables your Java Database Connectivity (JDBC) databases and Lightweight Directory Access Protocol servers to be accessed for enterprise search. After you associate the external sources with search applications, users can search these sources at the same time that they search collections with data that was crawled, parsed, and indexed.

System view

If you are a member of the enterprise search administrator role, you can click **System** on the toolbar to do the following activities (collection administrators, operators, and monitors can access this view only if an enterprise search administrator grants them permission to do so):

- Check the availability of system resources.
- Configure client Data Listener applications so that they can update collections.
- Add custom text analysis engines to the system.
- Add custom dictionaries for synonyms, stop words, and boost words to the system.
- Specify how many collections can build indexes in parallel, and specify whether refresh and reorganization requests for a single collection can run concurrently.
- Configure alerts for system-level events.
- Specify options for logging messages that are produced by system-level sessions.
- Specify information about your mail server so that you can receive e-mail about enterprise search activities.


Security view

If you are a member of the enterprise search administrator role, you can click **Security** to specify security options. Collection administrators, operators, and monitors cannot access this view.

If you enable security in IBM WebSphere Application Server, you can use the Security view to configure administrative roles. By configuring administrative roles, you can allow more users to administer the system, yet restrict each user's access to specific functions and collections.

Until you create your own search applications, you can use the sample search application to search all collections and external sources. After you create a custom search application, use the Security view to associate your application with the collections and external sources that it can search.

Monitor view

You can click  **Monitor** to monitor the system or collection components at any time. If your administrative role permits, you can also start and stop component processes while you monitor them.

Related concepts

“Enterprise search crawler administration” on page 33

You configure crawlers for the different types of data that you want to include in a collection. A single collection can contain any number of crawlers.

“Monitoring enterprise search activity” on page 217

When you monitor system and collection activities, you can view the status of various processes, watch for potential problems, or adjust configuration settings to enhance performance.

Related tasks

“Starting the enterprise search servers” on page 213

To enable users to search a collection, you must start the system processes and then start the servers that crawl, parse, index, and search the collection.

“Stopping the enterprise search servers” on page 215

You might need to stop and restart an enterprise search server if you make changes to its configuration or if you need to troubleshoot problems.

Creating a collection by using the Collection wizard

If you are new to enterprise search, a wizard can help you with creating a collection. The wizard provides details about each step in the process and enables you to save your settings as you progress.

“Creating a collection by using the Collections view” on page 28

Use the Collections view to create an empty collection. You can then edit the collection to specify options for adding data to the collection and making the collection searchable.

Logging in to the administration console

To administer an enterprise search system, you specify a URL in a Web browser and then log in to the administration console.

Before you begin

You must log in with a user ID that is authorized to access the enterprise search administration console:

- If you do not enable global security in WebSphere Application Server, only the enterprise search administrator that was specified when WebSphere II OmniFind Edition was installed can access the administration console.
- If you enable global security in WebSphere Application Server, you can use the enterprise search administration console to configure administrative roles. The user IDs that you configure must exist in a WebSphere Application Server user registry. When you configure administrative roles, you allow more users to log in to the administration console, but you can control the functions and collections that each administrative user can access.

Procedure

To log in to the enterprise search administration console:

1. Type the URL for the administration console in your Web browser. For example:

```
http://SearchServer.com/ESAdmin/
```

SearchServer.com is the host name of the search server for enterprise search.

Depending on your Web server configuration, you might also need to specify the port number. For example:

```
http://SearchServer.com:9080/ESAdmin/
```

2. On the welcome page, type your user ID and password and click **Log in**.

The Collections view, which is your entry point for administering the system and collections, is displayed. If you use administrative roles, the actions that you can take and the collections that you see depend on your administrative role.

If your session is inactive for a period of time, the system logs you out automatically. To continue administering the system, log in again.

After you finish administering collections, you can click **Logout** to log out of the console. You can then log in with a different ID and password, or you can close the Web browser to exit the administration console.

Related concepts

“Administrative roles” on page 182

Enterprise search uses the concept of roles to control access to various functions in the administration console.

Related tasks

“Starting the enterprise search servers” on page 213

To enable users to search a collection, you must start the system processes and then start the servers that crawl, parse, index, and search the collection.

Changing the enterprise search administrator password in a single server configuration

The password for the enterprise search administrator is stored in an encrypted format. To change the password, use the `eschangePW` script.

Before you begin

The enterprise search administrator ID and password must be valid on your operating system and must have authority to access and configure DB2 Universal Database.

About this task

The password for the initial enterprise search administrator ID is specified when WebSphere II OmniFind Edition is installed.

To change the password, you must run the `eschangePW` script to disseminate the change throughout the enterprise search system. The installation program creates two environment variables that you can use with the `eschangePW` script:

ES_INSTALL_ROOT

The enterprise search installation directory.

ES_NODE_ROOT

The enterprise search data directory. The password for the enterprise search administrator ID is stored in the es.cfg file in this directory.

Because the eschangepw script is installed in the ES_INSTALL_ROOT/bin directory, you can run it from anywhere in the system.

Procedure

To change the enterprise search administrator password in a single server configuration:

1. Log in as the enterprise search administrator.
2. Stop the server by entering this command: `esadmin stop`
3. Open the WebSphere Application Server Administrative Console and stop the server1 and ESSearchServer enterprise applications.
4. Change the system password for the enterprise search administrator user ID by using UNIX operating system commands or the Microsoft Windows change password facility.
5. Run the following script, where *newValue* is the password that you specified in step 4:

| Operating system | Command |
|------------------|-------------------------------------|
| UNIX | <code>eschangepw.sh newValue</code> |
| Windows | <code>eschangepw newValue</code> |

6. In the WebSphere Application Server Administrative Console, start the server1 and ESSearchServer applications.
7. Recycle the WebSphere II OmniFind Edition common communication layer (CCL) by entering the following commands:

| Operating system | Commands |
|--------------------------------------|--|
| UNIX | <code>stopccl.sh</code> , then <code>startccl.sh -bg</code> |
| Windows command prompt | <code>stopccl</code> , then <code>startccl</code> |
| Windows Services administrative tool | <ol style="list-style-type: none">1. Launch Windows Services.2. Right-click WebSphere Information Integrator OmniFind Edition and select Stop.3. Right-click WebSphere Information Integrator OmniFind Edition again and select Properties.4. Click the Log On tab.5. Change the password by specifying the <i>newValue</i>, and click OK.6. Right-click WebSphere Information Integrator OmniFind Edition again, and select Start. |

8. Restart enterprise search by entering this command: `esadmin start`.

Related reference

“Enterprise search commands, return codes, and session IDs” on page 249
You can use commands to diagnose problems, determine the status of the different parts of the system, start and stop sessions, or start and stop the system.

Changing the enterprise search administrator password in a multiple server configuration

The password for the enterprise search administrator is stored in an encrypted format. To change the password, use the `eschangepw` script to change it on all computers in your enterprise search system.

Before you begin

The enterprise search administrator ID and password must be valid on your operating system and must have authority to access and configure DB2 Universal Database.

The password for the enterprise search administrator must be the same on all computers that belong to this installation of WebSphere II OmniFind Edition.

About this task

The password for the initial enterprise search administrator ID is specified when WebSphere II OmniFind Edition is installed.

To change the password, and to disseminate the change throughout the enterprise search system, you must run the `eschangepw` script on each computer in your multiple server configuration. The installation program creates two environment variables that you can use with the `eschangepw` script:

ES_INSTALL_ROOT

The enterprise search installation directory.

ES_NODE_ROOT

The enterprise search data directory. The password for the enterprise search administrator ID is stored in the `es.cfg` file in this directory.

Because the `eschangepw` script is installed in the `ES_INSTALL_ROOT/bin` directory, you can run it from anywhere in the system.

Procedure

To change the enterprise search administrator password in a multiple server configuration:

1. On the computer where you installed the index server, log in as the enterprise search administrator.
2. Start the system maintenance mode by entering this command: `esadmin system maintenance`
3. Change the system password for the enterprise search administrator user ID by using UNIX operating system commands or the Microsoft Windows change password facility.
4. Run the following script, where *newValue* is the password that you specified in step 3:

| Operating system | Command |
|------------------|-------------------------------------|
| UNIX | <code>eschangepw.sh newValue</code> |
| Windows | <code>eschangepw newValue</code> |

5. Recycle the WebSphere II OmniFind Edition common communication layer (CCL) by entering the following commands on the index server:

| Operating system | Commands |
|--------------------------------------|---|
| UNIX | <code>stopccl.sh</code> , then <code>startccl.sh -bg</code> |
| Windows command prompt | <code>stopccl</code> , then <code>startccl</code> |
| Windows Services administrative tool | <ol style="list-style-type: none"> 1. Launch Windows Services. 2. Right-click WebSphere Information Integrator OmniFind Edition and select Stop. 3. Right-click WebSphere Information Integrator OmniFind Edition again and select Properties. 4. Click the Log On tab. 5. Change the password by specifying the <i>newValue</i>, and click OK. 6. Right-click WebSphere Information Integrator OmniFind Edition again, and select Start. |

6. On the computer where you installed the crawler server, log in as the enterprise search administrator.
7. Change the system password for the enterprise search administrator user ID by using UNIX operating system commands or the Windows change password facility. This password must match the password that you specified in step 3 on page 21.
8. Run the following script, where *newValue* is the password that you specified in step 3 on page 21:

| Operating system | Command |
|------------------|-------------------------------------|
| UNIX | <code>eschangepw.sh newValue</code> |
| Windows | <code>eschangepw newValue</code> |

9. Recycle CCL on the crawler server by entering the following commands:

| Operating system | Commands |
|------------------------|---|
| UNIX | <code>stopccl.sh</code> , then <code>startccl.sh -bg</code> |
| Windows command prompt | <code>stopccl</code> , then <code>startccl</code> |

| Operating system | Commands |
|--------------------------------------|---|
| Windows Services administrative tool | <ol style="list-style-type: none"> 1. Launch Windows Services. 2. Right-click WebSphere Information Integrator OmniFind Edition and select Stop. 3. Right-click WebSphere Information Integrator OmniFind Edition again and select Properties. 4. Click the Log On tab. 5. Change the password by specifying the <i>newValue</i>, and click OK. 6. Right-click WebSphere Information Integrator OmniFind Edition again, and select Start. |

10. On one of the computers where you installed a search server, log in as the enterprise search administrator.
11. Stop the IBM HTTP Web Server. (This step prevents the Network Dispatcher from routing queries to this server.)
12. Stop CCL on the search server by entering the following commands:

| Operating system | Commands |
|--------------------------------------|--|
| UNIX | <code>stopccl.sh</code> |
| Windows command prompt | <code>stopccl</code> |
| Windows Services administrative tool | <ol style="list-style-type: none"> 1. Launch Windows Services. 2. Right-click WebSphere Information Integrator OmniFind Edition and select Stop. |

13. Open the WebSphere Application Server Administrative Console and stop the `server1` and `ESSearchServer` applications.
14. Change the system password for the enterprise search administrator user ID by using UNIX operating system commands or the Windows change password facility. This password must match the password that you specified in step 3 on page 21.
15. Run the following script, where *newValue* is the password that you specified in step 3 on page 21:

| Operating system | Command |
|------------------|-------------------------------------|
| UNIX | <code>eschangepw.sh newValue</code> |
| Windows | <code>eschangepw newValue</code> |

16. Restart CCL on the search server by entering the following commands:

| Operating system | Commands |
|------------------------|------------------------------|
| UNIX | <code>startccl.sh -bg</code> |
| Windows command prompt | <code>startccl</code> |

| Operating system | Commands |
|--------------------------------------|--|
| Windows Services administrative tool | <ol style="list-style-type: none"> 1. Launch Windows Services. 2. Right-click WebSphere Information Integrator OmniFind Edition and select Properties. 3. Click the Log On tab. 4. Change the password by specifying the <i>newValue</i>, and click OK. 5. Right-click WebSphere Information Integrator OmniFind Edition again, and select Start. |

17. On the second search server, repeat steps 10 on page 23 through 16 on page 23.
18. Start the IBM HTTP Web Server.
19. In the WebSphere Application Server Administrative Console, start the server1 and ESSearchServer enterprise applications.
20. On the computer where you installed the index server, log in as the enterprise search administrator, and enter this command to start enterprise search:
esadmin start
21. Log in to the enterprise search administration console, monitor a collection, click the Search page, and then click **Stop** and **Start** to restart the search servers. Repeat this step for each collection in your enterprise search system.

Related reference

“Enterprise search commands, return codes, and session IDs” on page 249
You can use commands to diagnose problems, determine the status of the different parts of the system, start and stop sessions, or start and stop the system.

Changing the port number for the enterprise search system

If the port number that the enterprise search system uses for communication conflicts with a port number that is used by another product, you must change the enterprise search port number.

About this task

A port number for the enterprise search system is specified when WebSphere II OmniFind Edition is installed. (In a multi-server configuration, the same port number is specified on all servers.)

If the port number is unusable (for example, the port number might be assigned to another product on the same server), the conflict results in the following error message in the CCLServer_*date*.log file (where *date* specifies the date that the log file was created):

```
FFQ00273W Internal warning java.net.BindException: Address already in use: NET_Bind
at java.net.PlainSocketImpl.socketBind(Native Method)
at java.net.PlainSocketImpl.bind(PlainSocketImpl.java:357)
at java.net.ServerSocket.bind(ServerSocket.java:341)
at java.net.ServerSocket.<init>(ServerSocket.java:208)
at java.net.ServerSocket.<init>(ServerSocket.java:120)
```

Procedure

To change the port number that is used by enterprise search:

1. Enter the following command to stop the enterprise search system:
`esadmin stop`
2. Go to the server where the port number needs to be changed, and stop the common communication layer (CCL) by entering the following commands:

| Operating system | Commands |
|--------------------------------------|---|
| UNIX | <code>stopccl.sh,</code> |
| Windows command prompt | <code>stopccl</code> |
| Windows Services administrative tool | <ol style="list-style-type: none">1. Launch Windows Services.2. Right-click WebSphere Information Integrator OmniFind Edition and select Stop. |

3. Edit the `$ES_NODE_ROOT/nodeinfo/es.cfg` file (on UNIX) or `%ES_NODE_ROOT%\nodeinfo\es.cfg` file (on Windows), locate the following property, specify a new port number value, and then save and close the file:
`CCLPort=new_port_number`
4. Restart the CCL by entering the following commands:

| Operating system | Commands |
|--------------------------------------|--|
| UNIX | <code>startccl.sh -bg</code> |
| Windows command prompt | <code>startccl</code> |
| Windows Services administrative tool | <ol style="list-style-type: none">1. Launch Windows Services.2. Right-click WebSphere Information Integrator OmniFind Edition and select Start. |

5. Go to the index server and follow the instructions in step 2 to stop the CCL.
6. Edit the `$ES_NODE_ROOT/nodeinfo/es.cfg` file (on UNIX) or `%ES_NODE_ROOT%\nodeinfo\es.cfg` file (on Windows).
 - a. Locate the following property, where `computer_name` is the name of the server where you modified the port number in step 3. The `N` in the `nodeN` property is a number that identifies the server.
`nodeN.destination=computer_name`
 - b. Locate the following subproperty, specify the same port number here that you specified for the server in step 3, and then save and close the file:
`nodeN.port=new_port_number`

7. Follow the instructions in step 4 to restart the CCL.
8. Enter the following command to restart the enterprise search system:
`esadmin start`

After this command finishes, the new port number will be updated on all of the enterprise search servers.

Related reference

“Enterprise search commands, return codes, and session IDs” on page 249
You can use commands to diagnose problems, determine the status of the different parts of the system, start and stop sessions, or start and stop the system.

Enterprise search collections

An enterprise search collection contains the entire set of sources that users can search with a single query. Through federation, users can search multiple collections with a single query.

When you create a collection, you specify options that apply to the entire collection. The collection is empty until you add content to it.

You can add collections to an enterprise search system in two ways:

- If you are not familiar with the enterprise search administration console, or if you are still learning how the collection components work together, you might want to use the Collection wizard to create a collection. The Collection wizard helps you progress through the tasks and allows you to save your work as a draft collection while it is being created.
- When you are more familiar with the administration console, you might prefer to create collections by selecting the specific pages that you want to administer in the Collections view.

After you create a collection, you use controls in the Collections view to edit and monitor the collection, the enterprise search system, and security options.

Collection federation

If support for federation is built into the search application, users can search multiple collections at the same time. Federation also enables you to scale beyond the size limitation for a collection, which is 20 000 000 documents per collection. For example, users can search two collections that each contain 20 000 000 documents.

Search quality is dependent on the scores that are generated by individual collections, which are then merged to produce the final result set. The results are the same as submitting two separate searches and then merging and ranking the results.

Related tasks

“Monitoring a collection” on page 219

You can view general information about the status of each component in a collection or select options to view detailed information about individual components and URIs.

“Migrating a collection from WebSphere Portal” on page 209

To migrate collections from WebSphere Portal to enterprise search, prepare the collections in WebSphere Portal, then use the migration wizard to migrate them.

Creating a collection by using the Collection wizard

If you are new to enterprise search, a wizard can help you with creating a collection. The wizard provides details about each step in the process and enables you to save your settings as you progress.

Before you begin

To create a collection, you must be a member of the enterprise search administrator role.

To add content to a collection or to specify options for how content in the collection can be parsed, indexed, or searched, you must be an enterprise search administrator or a collection administrator for the collection.

About this task


While you create a collection, you can save it in a draft state. While it is in a draft state, any administrator who has authority to administer the collection can make changes to it. For example, you might want a collection administrator who has experience with Lotus Notes sources to configure a Notes crawler. Later, a collection administrator who has experience with UNIX systems might edit the draft collection to configure a UNIX file system crawler.

Procedure

To use the Collection wizard to create a collection:

1. Click **Collections** to open the Collections view.
2. Click **Collection Wizard**.
3. Follow the instructions in the wizard to create an empty collection and add content to it.

You must configure general information about the collection and create at least one crawler. You can accept the default values for the remaining configuration options, or specify options for your new collection.

4. To save a collection before you finish creating it, click **Save as Draft**.
Your collection is listed with other draft collections on the Collections view. If you enabled security for the collection, the  **Collection security is enabled** icon is displayed next to the collection name.
5. To return to a collection that you are still creating, click **Return to wizard** on the Collections view.
6. Click **Finish** to create the collection.

Your new collection is listed with other collections on the Collections view.

After you create a collection, you must start the processes for crawling, parsing, indexing, and searching the collection. Until you are ready to associate the collection with the search applications that can search it, you can use the sample search application (named Default) to search your new collection.

Creating a collection by using the Collections view

Use the Collections view to create an empty collection. You can then edit the collection to specify options for adding data to the collection and making the collection searchable.

Before you begin

To create a collection, you must be a member of the enterprise search administrator role.

To add content to a collection or to specify options for how content in the collection can be parsed, indexed, or searched, you must be an enterprise search administrator or a collection administrator for the collection.


About this task

For information about the values that you can specify for a new collection, click **Help** while you are creating the collection.

Procedure

To create a collection from the Collections view:

1. On the Collections view, click **Create Collection**.
2. On the Create a Collection page, provide information or make selections in the following fields:
 - **Collection name.** Specify a descriptive name for the content or purpose of the collection.
 - **Collection security.** Specify whether you want to enable security for the collection. After you create the collection, you cannot change this setting. If collection security is enabled, you can later specify options for enforcing document-level access controls.
 - **Document importance (static ranking model).** Specify a strategy for assigning a static ranking factor that will be used to rank documents in the search results. After you create the collection, you cannot change this value.
 - **Categorization type.** Specify whether you want to be able to search for documents by the categories that they belong to.
 - **Language to use.** Specify the default language for searching documents in the collection.
3. Accept the default values for the following fields, or specify options that you want to use with this collection:
 - **Description.** By default, no description is created.
 - **Estimated number of documents.** The default estimated size of the collection is 1 000 000 documents. The system uses this value to estimate the memory and disk resources for the collection, not to limit the size of the collection.
 - **Location for collection data.** The default location for collection-related files is on the index server. After you create the collection, you cannot change this value.
 - **Collection ID.** The default collection ID is based on the collection name. After you create the collection, you cannot change this value. (If you specify a custom collection ID, your search applications call the collection with this identifier instead of the potentially cryptic identifier that the system creates.)
 - **N-gram segmentation.** The default segmentation method is Unicode-based, white space segmentation. Select the option to use n-gram segmentation only if your collection includes Chinese, Japanese, or Korean documents and you want the parser to use n-gram segmentation to delimit words instead. After you create the collection, you cannot change this value.
4. Click **OK**.

The Collections view lists your new collection with other collections in your enterprise search system. If you enabled security for the collection, the  **Collection security is enabled** icon is displayed next to the collection name.

The collection is empty until you add content to it. To add content to a new collection, select the collection in the Collections view, edit it, create at least one crawler, and specify options for how you want data to be parsed, indexed, and searched.

You must then start the processes for crawling, parsing, indexing, and searching the collection. You can use the sample search application to search your new collection until you are ready to make it available for users to search with your custom search applications.

Related concepts

“Enterprise search administration overview” on page 15

You use the enterprise search administration console to create and administer collections and external sources, start and stop components, monitor system activity and log files, configure administrative users, associate search applications with collections and external sources, and specify information to enforce security.

Editing a collection

You edit collections to specify information about the documents that you want to include in a collection.

Before you begin


To edit a collection, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

About this task

When you edit a collection, you specify options for crawling data sources, parsing documents, reorganizing and refreshing the index, searching the indexed content, and logging error messages. When you create a collection, you must edit it to add content to it. Later, you can edit the collection to update the content or to change the way information is crawled, parsed, indexed, searched, or logged.

Procedure

To edit a collection:

1. Click **Collections** in the toolbar to open the Collections view.
2. Locate the collection that you want to edit in the list of collections, and click  **Edit**.
3. Make changes to any of the following pages:

General

Specify general information about the collection and view settings that you cannot change. If security was enabled for the collection when it was created, you can configure document-level security options.

Crawl Specify the data sources that you want to crawl and specify options for how the content is to be crawled. Every collection must contain at least one crawler, and a single collection can contain data from multiple types of data sources. You must configure at least one crawler for each type of data source that you want to include in the collection.

Parse Specify options for how you want documents that were crawled to be parsed and analyzed. You can configure categories, which enable users to search subsets of the collection, and you can configure rules that enable users to search specific parts of XML and HTML documents. If you add custom text analysis engines to the enterprise search system,

you can select one to use for analyzing and annotating content in this collection. You can also associate fields with boost classes to influence document ranking.

Index Specify schedules for reorganizing the entire index and refreshing the index with new and changed content. You can also configure options for using wildcard characters in queries, limiting the view of the collection to a range of URIs, collapsing search results from the same Web site, and removing URIs from the index.

Search

Specify options for the search processes, such as configuring a search cache and selecting a search language. You can also configure quick links, which is a function that ensures the return of predetermined URIs whenever a user includes specific words or phrases in a query. If you added custom dictionaries to the enterprise search system, you can select the dictionaries that you want to use for searching this collection.

Log

Specify the types of messages that you want to log and options for creating and rotating log files. You can configure alerts so that you can be notified when certain events occur, and specify options for receiving e-mail whenever certain messages, or certain types of messages, are logged. You can also specify options for logging information about documents that are dropped from the enterprise search system.

Deleting a collection

Deleting a collection completely removes all information about the collection from your enterprise search system.

Before you begin

To delete a collection, you must be a member of the enterprise search administrator role.

You must stop all processes associated with the collection before you can delete it.


About this task

Deleting a collection can be a time-consuming process. After you confirm that you want to delete the collection, the system deletes all data in the system that relates to the collection.

Tip: You might see a message about the requested operation timing out even though the process is still running in the background. To determine whether the task has completed, click **Refresh** in the administration console (do not click **Refresh** in the Web browser). The delete process is finished when the collection name no longer appears in the list of collections.

Procedure

To delete a collection:

1. Click **Collections** to open the Collections view.
2. In the list of collections, locate the collection that you want to delete and click  **Delete**.

Enterprise search crawler administration

You configure crawlers for the different types of data that you want to include in a collection. A single collection can contain any number of crawlers.

Configuring crawlers

You use the enterprise search administration console to create, edit, and delete crawlers. Typically, an expert in the type of data being crawled configures the crawler. For example, to set up a crawler to crawl Lotus Notes data sources, the collection administrator should either be a Notes administrator or work closely with someone who is knowledgeable about the databases that are being crawled.

When you create a crawler, a wizard for the type of data that is being crawled helps you specify properties that control how the crawler uses system resources. The wizard also helps you select the sources that you want to search.

You can make changes to existing crawlers at any time. You can edit crawler properties or parts of the crawl space as needed. Crawler wizards also help you to make these changes.

Populating a new crawler with base values

You can create a crawler by using the system default values or by copying values that are specified for another crawler of the same type. If you use an existing crawler as the base for a new crawler, you can quickly create multiple crawlers that have similar properties and then configure them, for example, to crawl different sources or operate on different crawling schedules.

By copying a crawler, you can divide the crawling workload among multiple crawlers that use the same crawling rules. For example, you might copy a Notes crawler because you want to use the same properties and field crawling rules with a different Lotus Notes server. The only differences might be the databases that each crawler crawls and document-level security settings.

Combining crawler types in a collection

Enterprise search crawlers are designed to gather information from specific types of data sources. When you configure crawlers for a collection, you must decide how to combine these different data source types so that users can easily search your enterprise data. For example, if you want users to be able to search Microsoft Windows file systems and Microsoft Exchange Server public folders with a single query, create a collection that includes Windows file system crawlers and Exchange Server crawlers.

When you combine multiple types of crawlers in a single collection, ensure that all of the crawlers can use the same static ranking method. (You specify the static ranking method when you create the collection.) For example, if you combine Web sources (which use document links as a ranking factor) and NNTP sources (which typically use the document date as a ranking factor), the quality of the search results might be degraded.

Document-level security

If you enable security for a collection when you create it, you can configure document-level security options. Each crawler can associate security tokens with the documents that it crawls. If you specify that you want to use document-level security when you configure the crawler, the crawler associates the security tokens that you specify with each document, and these tokens are added to the index with the documents.

If you enable security in your custom search applications, your applications can use the security tokens that the crawlers associated with documents to authenticate users. This capability enables you to restrict access to some documents in a collection and to allow other documents to be searched by all users. For example, in one collection you might allow all users to access all of the documents in your Microsoft Exchange Server public folders, but allow only users with specific user IDs to access documents in your Lotus Notes databases.

You can apply custom business rules to determine the value of the security tokens by encoding the rules in a Java class. When you configure crawler properties, you specify the name of the plug-in that you want the crawler to use when it crawls documents. The security tokens that your plug-in adds are stored in the index and can be used to control access to documents.

When you configure certain types of crawlers, you can specify additional security controls. For example, you can specify that you want to validate users during query processing. If you enable this option, the user's credentials are compared to current access control lists that are maintained by the data sources to be searched. This validation of current credentials can be done instead of or in addition to validation that is based on security tokens in the enterprise search index.

Scheduling crawlers

Crawlers that you create for Web, NNTP, and WebSphere Portal sources run continuously. After you start such crawlers, you typically do not need to stop them unless you change the crawler's configuration.

For all other crawler types, you specify a crawling schedule when you configure the crawler. For some data source types, a single schedule controls when the crawler visits all data sources in the crawl space. For other data source types, you can specify different schedules for specific data sources. For example, you can specify different schedules for crawling each Lotus Notes database that the crawler crawls.

When you configure the schedule, you specify the type of crawl that is to be done. You can schedule a full crawl of the all documents in the crawl space, schedule a crawl that includes all updates to the crawl space (new documents, modified documents, and deleted documents), or schedule a crawl that includes only new and modified documents. A full crawl takes the most time. A crawl that removes deleted documents takes longer than a crawl that ignores deleted documents.

When you edit a crawler's crawl space, you can specify a second crawling schedule. For example, you might want to configure one schedule to crawl all documents in the crawl space every Saturday night, and configure a second schedule that runs more frequently to crawl new and modified documents.

By creating multiple crawler schedules, you can better control when the crawler visits the target sources. For example, to crawl databases in different time zones, you can schedule the crawler for times when users are most likely to be finished with their work for the day.

Related concepts

“Enterprise search administration overview” on page 15

You use the enterprise search administration console to create and administer collections and external sources, start and stop components, monitor system activity and log files, configure administrative users, associate search applications with collections and external sources, and specify information to enforce security.

“Document-level security” on page 189

If security is enabled for a collection when it is created, you can configure document-level security controls. Document-level security ensures that users who search collections are able to access only the documents that they are allowed to see.

Related tasks

“Monitoring crawlers” on page 221

You can view general information about the status of each crawler in a collection or select options to view detailed information about a crawler activity.

Related reference

“URI formats in an enterprise search index” on page 89

The uniform resource identifier (URI) of each document in an enterprise search index indicates the type of crawler that added the document to the collection.

Creating a crawler

When you create a crawler, you specify the type of crawler that you want to create. A wizard helps you specify information about the data that you want to include in the collection.

Before you begin

To create a crawler, you must be a member of the enterprise search administrator role or be a collection administrator for the collection that owns the crawler.

About this task

You must create at least one crawler for a collection. The type of crawler that you create depends on the type of data that you want to include in the collection. A wizard for the type of crawler that you create helps you specify options for the crawler. For example, the wizard helps you specify options for how the crawler is to use system resources. The wizard also helps you select the data sources that you want to include in the collection.

Procedure

To create a crawler:

1. Edit a collection, select the Crawl page, and click **Create Crawler**.
2. Select the crawler type and base values for the crawler:
 - a. Select the type of crawler that supports the type of data that you want to crawl, such as Web sites, Lotus Notes databases, or UNIX file systems.

After you select a crawler type, options for how you want to create it are displayed.

- b. Select the base values for the crawler:

Use the system default values for the new crawler

Populates the initial crawler settings with the installation default values.

If you select this option, click **Next** to begin configuring your new crawler.

Clone the values of an existing crawler for the new crawler

Populates the initial crawler settings with values that are configured for another crawler of this type.

If you select this option, a list of crawlers that match this crawler type is displayed. Select the crawler that you want to use for the new crawler, then click **Next** to begin configuring your new crawler.

A wizard for the type of crawler that you are creating opens. Follow the wizard prompts to create the crawler. Click **Help** on any page in the wizard to learn more about the options that you can specify for that type of crawler.

Your new crawler is listed on the Crawl page with other crawlers that belong to the collection. You can click options to edit the crawler properties and the crawl space any time that you need to make changes to the crawler.

Editing crawler properties

You can change information about the crawler and how it crawls data. For example, you can change how the crawler uses system resources.

Before you begin


To edit crawler properties, you must be a member of the enterprise search administrator role or be a collection administrator for the collection that owns the crawler.

About this task

When you edit crawler properties, click **Help** to learn about the types of changes that you can make. The properties that you can edit depend on the crawler type.

Procedure

To edit the properties for a crawler:

1. Edit a collection, select the Crawl page, locate the crawler that you want to edit, and click  **Crawler properties**.
2. Change the crawler properties, then click **OK**.
3. For the changes to become effective, stop and restart the crawler. (If you change only the crawler description, you do not need to restart the crawler.)

Editing a crawl space

You can change information about the data sources that a crawler crawls. For example, you can add data sources, remove data sources, change the crawling schedule, and change the rules for crawling documents in a specific data source.

Before you begin


To edit a crawl space, you must be a member of the enterprise search administrator role or be a collection administrator for the collection that owns the crawler.

About this task

To learn about the changes that you can make for the type of crawler that you are administering, click **Help** while you edit the crawl space.

Procedure

To edit a crawl space:

1. Edit a collection, select the Crawl page, locate the crawler that you want to edit, and click  **Crawl space**.
2. Change the crawl space by selecting the options that you want to change.
The options that you can choose depend on the crawler type. For some options, such as adding data sources to the collection, a wizard for the crawler type opens to help you change the crawl space.
3. For the changes to become effective, stop and restart the crawler.

Deleting a crawler

Deleting a crawler removes all information about the crawler from your enterprise search system. Information that was previously crawled by the crawler remains in the index until you reorganize the index.

Before you begin

To delete a crawler, you must be a member of the enterprise search administrator role or be a collection administrator for the collection that owns the crawler.


About this task

Deleting a crawler can be a time-consuming process. After you confirm that you want to delete the crawler, the system deletes all data in the system that relates to the crawler.

Tip: Because this task takes time to complete, you might see a message about the requested operation timing out even though the process is still running in the background. To determine whether the task has completed, intermittently click **Refresh** in the administration console (do not click **Refresh** in the Web browser). The delete process is finished when the crawler name no longer appears in the list of crawlers.

Procedure

To delete a crawler:

1. Edit a collection and select the Crawl page.
2. Locate the crawler that you want to delete and click  **Delete**.

Content Edition crawlers

To include IBM WebSphere Information Integrator Content Edition repositories in an enterprise search collection, you must configure a Content Edition crawler.

You can use the Content Edition crawler to crawl Documentum, FileNet Panagon Content Services, FileNet P8 Content Manager, Hummingbird Document Management (DM), OpenText Livelink, and Portal Document Manager (PDM) repositories.

When you configure the crawler, you specify options for how the crawler is to crawl all repositories in the crawl space. You also select the item classes that you want to crawl in each repository.

To create or change a Content Edition crawler, log in to the enterprise search administration console. You must be a member of the enterprise search administrator role or be a collection administrator for the collection that owns the crawler.

When you create the crawler, a wizard helps you do these tasks:

- Specify properties that control how the crawler operates and uses system resources. The crawler properties control how the crawler crawls all of the repositories in the crawl space.
- Specify whether the crawler uses direct mode or server mode to access repositories. For server mode, you must also specify information that enables the crawler to access the Web application server.
- Select the repositories that you want to crawl.
- Specify user IDs and passwords that enable the crawler to access content in the selected repositories.
- Set up a schedule for crawling the repositories.
- Select the item classes that you want to crawl in each repository.
- Specify options for making the properties of item classes searchable. For example, you can exclude certain types of documents from the crawl space or specify that you want to crawl a particular version of a repository.
- Configure document-level security options. If security was enabled when the collection was created, the crawler can associate security data with documents in the index. This data enables search applications to enforce access controls based on the access control lists or security tokens.

For Documentum, FileNet Panagon Content Services, and Portal Document Manager item classes, you can also select an option to validate user credentials when a user submits a query. In this case, instead of comparing user credentials to indexed security data, the system compares the credentials to current access control lists that are maintained by the original data source. (This type of current credential validation is not available for the other repository types.)

Click **Help** while you are creating the crawler to learn about the fields in the wizard and how to provide the crawler with the information that the crawler needs to crawl data.

Server mode access to WebSphere II Content Edition repositories

You can configure the Content Edition crawler to access repositories in server mode.

In server mode, the WebSphere Information Integrator Content Edition connector that the crawler uses to access data is installed as an enterprise application on WebSphere Application Server, and the crawler accesses repositories through the server. This approach enables you to take advantage of J2EE application server environments.

Before you configure the crawler to access WebSphere Information Integrator Content Edition repositories in server mode, you must run a script on the crawler server. This script, which is provided with WebSphere Information Integrator OmniFind Edition, enables the Content Edition crawler to access repositories on the server.

Before you use the enterprise search administration console to configure a Content Edition crawler to use server mode, complete the task that is appropriate for your environment:

- “Configuring the crawler server on UNIX for WebSphere II Content Edition” on page 40.
- “Configuring the crawler server on Windows for WebSphere II Content Edition” on page 41.

Direct mode access to WebSphere II Content Edition repositories

You can configure the Content Edition crawler to access repositories in direct mode.

In direct mode, the crawler uses a WebSphere Information Integrator Content Edition connector that is installed on the crawler server when WebSphere II OmniFind Edition is installed.

Procedure

To configure the system so that repositories can be accessed in direct mode:

1. Confirm that the VBR_HOME and JAVA_HOME environment variables in the *iice_install_root/bin/config.sh* file (on UNIX) or *iice_install_root\bin\config.bat* file (on Microsoft Windows) specify the correct directory.
2. To configure the WebSphere Information Integrator Content Edition administration console to run in direct mode, add the **vbr.as.operationMode=direct** Java system property to the *iice_install_root/bin/Admin.bat* file (on UNIX) or *iice_install_root\bin\Admin.bat* file (on Windows).
3. Start the WebSphere Information Integrator Content Edition administration console in direct mode and configure the connector for the WebSphere II OmniFind Edition crawler server. (See the WebSphere Information Integrator Content Edition documentation for instructions.)
4. Select the direct mode option when you use the WebSphere II OmniFind Edition administration console to configure the Content Edition crawler.

Configuring the crawler server on UNIX for WebSphere II Content Edition

If you install WebSphere II OmniFind Edition on a computer that is running IBM AIX®, Linux®, or the Solaris operating environment, and you configure the Content Edition crawler to use server mode when accessing repositories, you must run a script to configure the crawler server. The script enables the Content Edition crawler to access WebSphere Information Integrator Content Edition repositories.

About this task

The Content Edition crawler uses Java libraries of WebSphere Information Integrator Content Edition as a Java client. In server mode, these Java libraries require EJB-related Java libraries of WebSphere Application Server. To ensure that the Content Edition crawler can work with the Java libraries, you must run a setup script that WebSphere II OmniFind Edition provides on the crawler server after you install WebSphere Application Server.

WebSphere Information Integrator Content Edition is installed on the crawler server when WebSphere II OmniFind Edition is installed. To be able to use the Content Edition crawler in server mode, you must copy the `vbr_access_services.jar` file from the WebSphere Information Integrator Content Edition server to the crawler server.

Procedure

To configure the crawler server so that it can crawl WebSphere Information Integrator Content Edition repositories:

1. If WebSphere II OmniFind Edition is installed in a multiple server configuration, install and bind the WebSphere Application Server Java libraries.
2. Run the setup script for the Content Edition crawler on the crawler server:
 - a. Log in as the enterprise search administrator.
 - b. Start the following script, which is installed in the `$ES_INSTALL_ROOT/bin` directory), and answer the prompts:
`escrvbr.sh`
3. Stop and restart the enterprise search system, including all sessions on the enterprise search common communication layer (CCL):

```
esadmin stop
stopccl.sh
startccl.sh -bg
esadmin start
```

4. Copy the `vbr_access_services.jar` file from the WebSphere Information Integrator Content Edition server to the crawler server.

Copy from:

The `vbr_access_services.jar` file is in the following default location:

`was_install_root/installedApps/server_name/application_name`

`was_install_root` is the WebSphere Application Server installation directory, `server_name` is the name that you specified for the server, and `application_name` is the name that you specified for the WebSphere Information Integrator Content Edition application in WebSphere Application Server.

|

| **Copy to:**

| The target directory on the crawler server is *iice_install_root/lib*, where
| *iice_install_root* is the WebSphere Information Integrator Content Edition
| installation directory on the crawler server.

|

| **Configuring the crawler server on Windows for WebSphere II**
| **Content Edition**

| If you install WebSphere II OmniFind Edition on a Microsoft Windows computer,
| and you configure the Content Edition crawler to use server mode when accessing
| repositories, you must run a script to configure the crawler server. The script
| enables the Content Edition crawler to access WebSphere Information Integrator
| Content Edition repositories.

|

| **About this task**

| The Content Edition crawler uses Java libraries of WebSphere Information
| Integrator Content Edition as a Java client. In server mode, these Java libraries
| require EJB-related Java libraries of WebSphere Application Server. To ensure that
| the Content Edition crawler can work with the Java libraries, you must run a setup
| script that WebSphere II OmniFind Edition provides on the crawler server after
| you install WebSphere Application Server.

| WebSphere Information Integrator Content Edition is installed on the crawler
| server when WebSphere II OmniFind Edition is installed. To be able to use the
| Content Edition crawler in server mode, you must copy the *vbr_access_services.jar*
| file from the WebSphere Information Integrator Content Edition server to the
| crawler server.

|

| **Procedure**

| To configure the crawler server so that it can crawl WebSphere Information
| Integrator Content Edition repositories:

- | 1. If WebSphere II OmniFind Edition is installed in a multiple server
| configuration, install and bind the WebSphere Application Server Java libraries.
- | 2. Run the setup script for the Content Edition crawler on the crawler server:
 - | a. Log in with the enterprise search administrator ID (this user ID was
| specified when WebSphere II OmniFind Edition was installed).
 - | b. Start the following script, which is installed in the
| `%ES_INSTALL_ROOT%\bin` directory, and answer the prompts:
| `escrvbr.vbs`
- | 3. Stop and restart the enterprise search system, including all sessions on the
| enterprise search common communication layer (CCL):
 - | a. At a command prompt, stop the enterprise search system:
| `esadmin stop`
 - | b. Select **Start** → **Programs** → **Administrative Tools** → **Services**, then restart the
| IBM WebSphere Information Integrator OmniFind Edition service.
 - | c. At a command prompt, start the enterprise search system:
| `esadmin start`
- | 4. Copy the *vbr_access_services.jar* file from the WebSphere Information
| Integrator Content Edition server to the crawler server.

|

| **Copy from:**

| The *vbr_access_services.jar* file is in the following default location:

was_install_root\installedApps*server_name**application_name*

was_install_root is the WebSphere Application Server installation directory, *server_name* is the name that you specified for the server, and *application_name* is the name that you specified for the WebSphere Information Integrator Content Edition application in WebSphere Application Server.

Copy to:

The target directory on the crawler server is *iice_install_root*\lib, where *iice_install_root* is the WebSphere Information Integrator Content Edition installation directory on the crawler server.

DB2 crawlers

You use the DB2 crawler to include IBM DB2 Universal Database databases in a collection. You can also use the DB2 crawler to include nickname tables that you create for IBM DB2 Universal Database for z/OS, IBM Informix, Oracle, and Microsoft SQL Server databases.

You must configure a separate crawler for each database server that you want to crawl. When you configure the crawler, you specify options for how the crawler is to crawl all databases on the same server. You also select the specific tables that you want to crawl in each database.

The tables that you select for crawling should be database tables, nickname tables, or views. The DB2 crawler does not support joined tables.

Event publishing

If you use WebSphere Information Integrator Event Publisher Edition, and if you associate the databases that you want to crawl with publishing queue maps, the DB2 crawler can use the maps to crawl updates to the database tables.

A publishing queue map identifies a WebSphere MQ queue that receives XML messages when updates to a database table are published. The crawler listens to the queue for information about these published events and updates the crawl space when tables are updated (the first time that the crawler crawls a table, the crawler crawls all of the documents).

Event publishing allows new and changed documents to become available for searching on a faster basis than documents that the crawler crawls according to the crawler schedule.

If some or all of the tables are configured to use event publishing, you can specify information that enables the crawler to access WebSphere MQ and the publishing queue maps when you configure the crawler.

You must also ensure that WebSphere MQ and WebSphere Information Integrator Event Publisher Edition are configured on the server to be crawled, and that the WebSphere MQ client module is configured on the crawler server. Complete the following tasks to use event publishing with a DB2 crawler:

- “Configuring WebSphere MQ for DB2 crawlers” on page 45.
- “Configuring WebSphere Information Integrator Event Publisher Edition for DB2 crawlers” on page 43.
- “Configuring the crawler server on UNIX to use event publishing” on page 47.

- “Configuring the crawler server on Windows to use event publishing” on page 48.

Configuration overview

To create or change a DB2 crawler, log in to the enterprise search administration console. You must be a member of the enterprise search administrator role or be a collection administrator for the collection that owns the crawler.

When you create the crawler, a wizard helps you do these tasks:

- Specify properties that control how the crawler operates and uses system resources. The crawler properties control how the crawler crawls all of the databases on a particular database server.

- Specify information about the types of databases that you want to crawl.

If you plan to crawl remote databases that are not cataloged on the local database server, you must start the DB2 Administration Server on the remote server before you can use the DB2 crawler to crawl those databases. You must also specify the host name and port of the remote database server when you configure the crawler.

- Select the database that you want to crawl.
- Specify user IDs and passwords that enable the crawler to access databases that use access controls.
- Set up a schedule for crawling the databases.
- Select the tables that you want to crawl in each database.

Attention: To optimize the performance of the discovery processes (and to prevent the crawler configuration process from timing out), choose to crawl all tables only if the database does not contain many tables or if the tables do not contain many columns. If you select some tables to crawl now, you can edit the crawl space later and add more tables to the collection.

- Select the tables that are to be crawled when updates to them are published in an event publishing queue, and specify information that enables the crawler to access the event publishing queue.
- Specify options for making the columns in specific tables searchable. For example, you can enable certain columns to be used in parametric queries or specify which columns can be returned in the search results.
- Configure document-level security options. If security was enabled when the collection was created, the crawler can associate security data with documents in the index. This data enables search applications to enforce access controls based on the stored access control lists or security tokens.

Click **Help** while you are creating the crawler to learn about the fields in the wizard and how to provide the crawler with the information that the crawler needs to crawl data.

Configuring WebSphere Information Integrator Event Publisher Edition for DB2 crawlers

Before you configure a DB2 crawler to use event publishing, ensure that IBM WebSphere Information Integrator Event Publisher Edition is configured on the server to be crawled.

About this task

Use the following guidelines when you configure WebSphere Information Integrator Event Publisher Edition for use with the DB2 crawler:

- Both changed and unchanged columns in the source tables must be selected for publishing.
- Deleted rows in the source tables must be selected for publishing.
- An event publishing queue cannot be shared among multiple databases.
- A single database can have multiple queue maps and queues.
- A table should have one XML publication associated with one publishing queue map. (A table should not have more than one XML publication associated with a single publishing queue map. A table can have more than one XML publication if each XML publication is associated with a different publishing queue map.)

Procedure

Complete the following steps to configure a database server so that the DB2 crawler can access table updates that are published in an event publishing queue. (See the WebSphere Information Integrator Publisher Edition documentation for assistance with these steps.)

1. Install WebSphere Information Integrator Event Publisher Edition on the database server to be crawled.
2. Start the Replication Center Launchpad:

| Operating system | Command |
|------------------|---|
| UNIX | db2rc |
| Windows | Select Start → IBM DB2 Replication Center |

3. Create Q Capture control tables:
 - a. Select **Event publishing** as the launchpad view, select **Create Q Capture Control Tables**, and then click **Next**.
 - b. In the **Q Capture server** field, select the server that you want to use as the Q Capture server from the list of available database servers, and click **OK**.
 - c. Specify a user ID and password that is authorized to access the selected Q Capture server. Change the Q Capture schema or accept the default schema name, and click **Next**.
 - d. Specify the names of the queue manager, administration queue, and restart queue that you specified when you configured WebSphere MQ on this database server, and click **Next**.
 - e. Click **Finish**. After a page with messages and SQL scripts is displayed, click **Close**.
 - f. For the processing option, select **Run now** and click **OK**. After a message that indicates that the SQL scripts are finished is displayed, click **Close**.
4. Create an XML publication:
 - a. In the Replication Center Launchpad, select **Event publishing** as the launchpad view, select **Create an XML Publication**, and then click **Next**.
 - b. On the Start page, click **Next**.
 - c. On the Server and Queue Map page, confirm that the Q Capture server and Q Capture schema are correct, then click the button next to the **Publishing queue map** field, and click **New** to create a publishing queue map.
 - d. On the General page, type a name for the queue map.
 - e. On the Properties page, specify the name of the send queue (such as the name of the data queue that you specified when you configured WebSphere

- MQ on this server), select either **Row operation** or **Transaction** for the type of message content, clear the check boxes for sending heartbeat messages and adding JMS message headers, and click **OK**.
- f. After a page with messages and SQL scripts is displayed, click **Close**.
 - g. For the processing option, select **Run now** and click **OK**. After a message that indicates that the SQL scripts are finished is displayed, click **Close**.
 - h. On the Select Publishing Queue Map page, select the queue map that you created and click **OK**.
 - i. On the Server and Queue Map page, confirm that the queue map name is correct, and click **Next**.
 - j. On the Source Table page, click **Add**, click **Retrieve All**, select a table that you want to enable for event publishing, click **OK**, and then click **Next**.
 - k. On the Columns and Rows page, select the columns that you want the DB2 crawler to crawl (or all columns) and select key columns. On the page where you select the rows to crawl (or all rows), select the option to publish source table deletes. After you finish configuring these options, click **Next**.
 - l. On the Message Content page, select the option to include both changed and unchanged columns for the column data, and select the option for new data values only. Ensure that the check box for starting XML publications automatically is selected, and click **Next**.
 - m. On the Review XML Publications page, click **Next**.
 - n. On the Summary page, click **Finish**. After a page with messages and SQL scripts is displayed, click **Close**.
 - o. For the processing option, select **Run now** and click **OK**. After a message that indicates that the SQL scripts are finished is displayed, click **Close**.
5. Start the Q Capture server:
 - a. Close the Replication Center Launchpad and start the Replication Center.
 - b. In the object tree, click **Q Replication** → **Definitions** → **Q Capture Servers**.
 - c. Right-click the icon for the Q Capture server that you configured and select **Enable Database for Q Replication**.
 - d. After a warning message is displayed, click **OK**.
 - e. After a page with DB2 messages is displayed, click **Close**.
 - f. In the object tree, right-click the icon for the Q Capture server and select **Start Q Capture program**.
 - g. For the processing option, select **Run now**, specify the system name, the user ID and password for the DB2 user, the path for the directory where logs are stored, and the DB2 instance name, then click **OK**.
 - h. After a message that indicates that the request was submitted is displayed, click **Close**.
 - i. In the object tree, right-click the icon for the Q Capture server and select **Check status**.
 The Q Capture server status is displayed. If errors occurred, a status message states that the server is presumed down. To review the logs and determine the cause of any errors, enter the following command on a command line:


```
asnqcap Capture_Server=capture server name LOGSTDOUT=y
```

Configuring WebSphere MQ for DB2 crawlers

Before you configure a DB2 crawler to use event publishing, ensure that IBM WebSphere MQ is configured on the server that the crawler will listen to.

Before you begin

Ensure that DB2 UDB, WebSphere Information Integrator Event Publisher Edition, and WebSphere MQ are installed on the target database server.

Restrictions

If the target database server is installed on a Linux computer, all DB2 Universal Database (DB2 UDB) users, WebSphere MQ users, and WebSphere II OmniFind Edition users must set the following environment variable:

```
export LD_ASSUME_KERNEL=2.4.19
```

This environment variable enables LinuxThread threading implementations to be exported from any shell where installation is performed, WebSphere MQ control commands are issued, or WebSphere MQ applications are run. WebSphere MQ requires this environment variable to be exported.

About this task

The DB2 crawler supports client connection mode to the WebSphere MQ server. The crawler listens for XML messages that are published in an event publishing queue. The crawler cannot listen for XML messages that are transported through more than one queue.

After you configure WebSphere MQ, the DB2 crawler uses the queue manager name, the queue name, the server host name, the server port number, and the server channel name to obtain XML messages from a publishing queue. The crawler parses the messages and updates the crawl space with information about updated tables.

Procedure

Complete the following steps to configure a database server so that the DB2 crawler can listen to an event publishing queue. (See the WebSphere MQ documentation for assistance with these steps.)

1. Log in as the WebSphere MQ Administrator role and enter the following commands to create a queue manager and queues.
 - a. On a command line, enter the following command:

```
crtmqm QM1
```
 - b. After the Setup completed message is displayed, enter the following command:

```
strmqm QM1
```
 - c. After the 'QM1' started message is displayed, enter the following command:

```
runmqsc QM1
```
 - d. After the Starting MQSC for queue manager QM1 message is displayed, enter the following command to create an administration queue:

```
DEFINE QLOCAL('ASN.QM1.ADMINQ')
```
 - e. After the WebSphere MQ queue created message is displayed, enter the following command to create a restart queue:

```
DEFINE QLOCAL(' ASN.QM1.RESTARTQ')
```
 - f. After the WebSphere MQ queue created message is displayed again, enter the following command to create a data queue:


```
DEFINE QLOCAL(' ASN.QM1.DATAQ')
```

- g. After the WebSphere MQ queue created message is displayed again, enter the following command to exit:
end

2. Enter the following command to start the MQ Listener on the database server (the MQ Listener must be running when you create a DB2 crawler that uses event publishing). In this example, 1414 is the server's port number and the default channel, SYSTEM.DEF.SVRCONN is used:

```
runmqtsr -m QM1 -t TCP -p 1414 &
```

3. Enter the following commands to authorize a DB2 UDB user to access the queue manager and the queues through the Message Queuing Interface (MQI) for event publishing (in this example, the user ID is db2inst1):

```
setmqaut -m QM1 -t qmgr -p db2inst1 +allmqi  
setmqaut -m QM1 -t queue -n ASN.QM1.DATAQ -p db2inst1 +allmqi  
setmqaut -m QM1 -t queue -n ASN.QM1.ADMINQ -p db2inst1 +allmqi  
setmqaut -m QM1 -t queue -n ASN.QM1.RESTARTQ -p db2inst1 +allmqi
```

4. Enter the following commands for the user ID that is used to create and run the DB2 crawler with event publishing. These commands authorize the user ID to access the queue manager and the queues through the Message Queuing Interface (MQI) for event publishing. In this example, the user ID is esuser:

```
setmqaut -m ASN.QM1.QM2 -t qmgr -p esuser +allmqi  
setmqaut -m ASN.QM1.QM2 -t queue -n ASN.QM1.DATAQ -p esuser +allmqi
```

Configuring the crawler server on UNIX to use event publishing

If you install WebSphere II OmniFind Edition on a computer that is running IBM AIX, Linux, or the Solaris operating environment, and you configure the DB2 crawler to use event publishing, you must run a script to configure the crawler server. The script enables the crawler to access WebSphere MQ queue managers and queues.

About this task

The DB2 crawler uses the WebSphere MQ 5.3 modules for Java Messaging to access WebSphere MQ queue managers and queues. You must install these modules on the crawler server.

To ensure that the DB2 crawler can use event publishing, you must also run a setup script that WebSphere II OmniFind Edition provides on the crawler server after you install the WebSphere MQ modules.

Procedure

To configure the crawler server to use event publishing:

1. Install the WebSphere MQ 5.3 modules for Java Messaging on the crawler server:
 - a. Log in as the root user and enter the following command:
export LD_ASSUME_KERNEL=2.4.19
 - b. Insert the WebSphere MQ CD.

- c. Change to the directory where the MQ modules for Java Messaging are located.
- d. Enter the following command to install the modules:


```
rpm -i MQSeriesJava-5.3.0-1.i386.rpm
```
- 2. Run the setup script for the DB2 crawler on the crawler server:
 - a. Log in as the enterprise search administrator (this user ID was specified when WebSphere II OmniFind Edition was installed).
 - b. Start the following script, which is installed in the `$ES_INSTALL_ROOT/bin` directory, and answer the prompts:


```
escrdb2.sh
```
- 3. Stop and restart the enterprise search system, including all sessions on the enterprise search common communication layer (CCL):

```
esadmin stop
stopccl.sh
startccl.sh -bg
esadmin start
```

Configuring the crawler server on Windows to use event publishing

If you install WebSphere II OmniFind Edition on a computer that is running Microsoft Windows, and you configure the DB2 crawler to use event publishing, you must run a script to configure the crawler server. The script enables the crawler to access WebSphere MQ queue managers and queues.

About this task

The DB2 crawler uses the WebSphere MQ 5.3 modules for Java Messaging to access WebSphere MQ queue managers and queues. You must install these modules on the crawler server.

To ensure that the DB2 crawler can use event publishing, you must also run a setup script that WebSphere II OmniFind Edition provides on the crawler server after you install the WebSphere MQ modules.

Procedure

To configure the crawler server to use event publishing:

1. Install the WebSphere MQ 5.3 modules for Java Messaging on the crawler server:
 - a. Insert the WebSphere MQ CD.
 - b. Start the WebSphere MQ installer.
 - c. In the Choose Product Features window, select **Java Messaging** for the installation option.
2. Run the setup script for the DB2 crawler on the crawler server:
 - a. Log in with the enterprise search administrator ID (this user ID was specified when WebSphere II OmniFind Edition was installed).
 - b. Start the following script, which is installed in the `%ES_INSTALL_ROOT%\bin` directory, and answer the prompts:


```
escrdb2.vbs
```

3. Stop and restart the enterprise search system, including all sessions on the enterprise search common communication layer (CCL):
 - a. At a command prompt, stop the enterprise search system:

```
esadmin stop
```
 - b. Select **Start** → **Programs** → **Administrative Tools** → **Services**, then restart the IBM WebSphere Information Integrator OmniFind Edition service.
 - c. At a command prompt, start the enterprise search system:

```
esadmin start
```

DB2 Content Manager crawlers

To include IBM DB2 Content Manager item types in an enterprise search collection, you must configure a DB2 Content Manager crawler.

Crawler server configuration

Before you can crawl a DB2 Content Manager server, you must run a script on the crawler server. This script, which is provided with WebSphere Information Integrator OmniFind Edition, enables the DB2 Content Manager crawler to communicate with DB2 Content Manager servers.

Before you use the enterprise search administration console to configure a DB2 Content Manager crawler, complete the task that is appropriate for your environment:

- “Configuring the crawler server on UNIX for DB2 Content Manager” on page 50.
- “Configuring the crawler server on Windows for DB2 Content Manager” on page 51.

Configuration overview

You can use the DB2 Content Manager crawler to crawl any number of DB2 Content Manager servers. When you configure the crawler, you specify options for how the crawler is to crawl all DB2 Content Manager servers in the crawl space. You also select the specific item types that you want to crawl on each server.

To create or change a DB2 Content Manager crawler, log in to the enterprise search administration console. You must be a member of the enterprise search administrator role or be a collection administrator for the collection that owns the crawler.

When you create the crawler, a wizard helps you do these tasks:

- Specify properties that control how the crawler operates and uses system resources. The crawler properties control how the crawler crawls all of the item types on all DB2 Content Manager servers in the crawl space.
- Select the DB2 Content Manager servers that you want to crawl.
- Specify user IDs and passwords that enable the crawler to access content on the DB2 Content Manager servers.
- Set up a schedule for crawling the servers.
- Select the item types that you want to crawl on each DB2 Content Manager server.

- Specify options for making the attributes in some item types searchable. For example, you can exclude certain types of documents from the crawl space and specify which attributes can be returned in the search results.
- Configure document-level security options. If security was enabled when the collection was created, the crawler can associate security data with documents in the index. This data enables search applications to enforce access controls based on the stored access control lists or security tokens.

Click **Help** while you are creating the crawler to learn about the fields in the wizard and how to provide the crawler with the information that the crawler needs to crawl data.

Configuring the crawler server on UNIX for DB2 Content Manager

If you install WebSphere II OmniFind Edition on a computer that is running IBM AIX, Linux, or the Solaris operating environment, you must run a script to configure the crawler server. The script enables the DB2 Content Manager crawler to communicate with IBM DB2 Content Manager servers.

About this task

The DB2 Content Manager crawler uses the Java connector for DB2 Content Manager Version 8 to access DB2 Content Manager servers. You install this connector by installing IBM DB2 Information Integrator for Content Version 8.2 or later on the crawler server. To ensure that the DB2 Content Manager crawler can work with DB2 Content Manager, you run a setup script that WebSphere II OmniFind Edition provides on the crawler server after you install the connector.

Procedure

To configure the crawler server so that it can crawl DB2 Content Manager servers:

1. Install the Java connector for DB2 Content Manager Version 8 on the crawler server:
 - a. On the crawler server, log in as the root user:


```
su - root
```
 - b. Run the db2profile file. For example:


```
./home/db2inst/sqllib/db2profile
```
 - c. Export the JAVAHOME environment variable. For example:


```
export JAVAHOME=/usr/IBMJava2-141
```
 - d. Add the Java directory to the PATH environment variable:


```
export PATH=$PATH:$JAVAHOME/bin
```
 - e. Insert the DB2 Information Integrator for Content installation CD and run the installation wizard.
 - f. In the Component Selection window, take the following actions. (If you are working with Information Integrator for Content Version 8.3, you can see the Component Selection window with Custom install option.)
 - 1) Select **Local connectors** from the **Components** list, then select **Content Manager V8 connector** from the **Subcomponents** list.
 - 2) Select **Connector toolkits and samples** from the **Components** list, then select **Content Manager V8 connector** from the **Subcomponents** list.

- g. Specify a database name, user name, and password for the DB2 Content Manager library, and accept the default settings for the remaining windows.
- 2. On the crawler server, log in with a user ID that is in the DB2 administration group.
- 3. Catalog the remote DB2 Content Manager library server database, and verify that the crawler server can connect to the DB2 Content Manager server:

```
db2 catalog tcpip node node_name remote hostname server port
db2 catalog database database_name as alias at node node_name
```

- 4. Optional: Log in as the root user and test the database connection:

```
. Information_Integrator_for_Content_install_directory/bin/cmbenv81.sh
cd Information_Integrator_for_Content_install_directory/samples/java/icm
javac *.java
java SConnectDisconnect ICMdatabase_name CMadmin_ID CMadmin_password
```

- 5. Run the setup script for the DB2 Content Manager crawler on the crawler server:

- a. Change to the ES_INSTALL_ROOT/bin directory:

```
cd $ES_INSTALL_ROOT/bin
```

- b. Start the following script and answer the prompts:

```
escrcm.sh
```

- 6. Stop and restart the enterprise search system, including all sessions on the enterprise search common communication layer (CCL):

```
esadmin stop
stopccl.sh
startccl.sh -bg
esadmin start
```

Configuring the crawler server on Windows for DB2 Content Manager

If you install WebSphere II OmniFind Edition on a Microsoft Windows computer, you must run a script to configure the crawler server. The script enables the DB2 Content Manager crawler to communicate with IBM DB2 Content Manager servers.

About this task

The DB2 Content Manager crawler uses the Java connector for DB2 Content Manager Version 8 to access DB2 Content Manager servers. You install this connector by installing IBM DB2 Information Integrator for Content Version 8.2 or later on the crawler server. To ensure that the DB2 Content Manager crawler can work with DB2 Content Manager, you run a setup script that WebSphere II OmniFind Edition provides on the crawler server after you install the connector.

Procedure

To configure the crawler server so that it can crawl DB2 Content Manager servers:

- 1. Install the Java connector for DB2 Content Manager Version 8 on the crawler server:
 - a. Insert the DB2 Information Integrator for Content installation CD. The installation program begins automatically.

The DB2 Content Manager Enterprise Information Portal installation wizard opens.

- b. In the Component Selection window, take the following actions. (If you are working with Information Integrator for Content Version 8.3, you can see the Component Selection window with Custom install option.)
 - 1) Select **Local connectors** from the **Components** list, then select **Content Manager V8 connector** from the **Subcomponents** list.
 - 2) Select **Connector toolkits and samples** from the **Components** list, then select **Content Manager V8 connector** from the **Subcomponents** list.
 - c. Specify a database name, user name, and password for the DB2 Content Manager library, and accept the default settings for the remaining windows.
2. Catalog the remote DB2 Content Manager library server database and verify that the crawler server can connect to the DB2 Content Manager server. Enter the following commands at a command prompt on the crawler server:

```
db2 catalog tcpip node node_name remote hostname server port  
db2 catalog database database_name as alias at node node_name
```

3. Optional: Test the database connection by opening an command prompt and entering the following commands:

```
cmbenv81.bat  
cd Information_Integrator_for_Content_install_directory\samples\java\icm  
javac *.java  
java SConnectDisconnect ICMdatabase_name CMadmin_ID CMadmin_password
```

4. Run the setup script for the DB2 Content Manager crawler on the crawler server:
- a. Change to the ES_INSTALL_ROOT\bin directory:

```
cd %ES_INSTALL_ROOT%\bin
```
 - b. Start the following script and answer the prompts:

```
escrcm.vbs
```
5. Stop and restart the enterprise search system, including all sessions on the enterprise search common communication layer (CCL):
- a. At a command prompt, stop the enterprise search system:

```
esadmin stop
```
 - b. Select **Start** → **Programs** → **Administrative Tools** → **Services**, then restart the IBM WebSphere Information Integrator OmniFind Edition service.
 - c. At a command prompt, start the enterprise search system:

```
esadmin start
```

Domino Document Manager crawlers

To include Domino Document Manager libraries and cabinets in an enterprise search collection, you must set configure a Domino Document Manager crawler.

Crawler server configuration

If a Domino Document Manager server that you plan to crawl uses the Notes remote procedure call (NRPC) protocol, you must run a script on the crawler server. This script, which is provided with WebSphere Information Integrator OmniFind Edition, enables the Domino Document Manager crawler to communicate with the servers that use NRPC.

If a Domino Document Manager server that you plan to crawl uses the Domino Internet Inter-ORB Protocol (DIIOP), you do not need to run a setup script on the crawler server. However, you must configure the Domino Document Manager server so that the Domino Document Manager crawler can access the server.

If WebSphere II OmniFind Edition was installed on an IBM AIX system, you must ensure that the I/O Completion Port module is installed and available on the crawler server.

Before you use the enterprise search administration console to configure a Domino Document Manager crawler, complete the tasks that are appropriate for your environment:

- “Configuring the crawler server on UNIX to crawl Lotus Domino sources” on page 58.
- “Configuring the crawler server on Windows to crawl Lotus Domino sources” on page 60.
- “Configuring servers that use the DIIOP protocol” on page 61.
- “Configuring the I/O completion port on AIX to crawl Lotus Domino sources” on page 62.

Configuration overview

You can use the Domino Document Manager crawler to crawl any number of Domino Document Manager libraries. When you create the crawler, you select the libraries to crawl from a single Domino Document Manager server. Later, when you edit the crawl space, you can add documents from another Domino Document Manager server that you want to include in the same crawl space. When you create or edit the crawler, you can specify whether you want to crawl all of the cabinets in the libraries that you select for crawling, or whether you want to crawl specific cabinets.

To create or change a Domino Document Manager crawler, log in to the enterprise search administration console. You must be a member of the enterprise search administrator role or be a collection administrator for the collection that owns the crawler.

When you create the crawler, a wizard helps you do these tasks:

- Specify properties that control how the crawler operates and uses system resources. The crawler properties control how the crawler crawls all of the documents in the crawl space.
- Identify the Domino Document Manager server and communications protocol.
- Select the libraries that you want to crawl.
- Set up a schedule for crawling the libraries.
- Select the documents that you want to crawl. The crawler can crawl all of the cabinets in a library, or crawl only the documents that are in cabinets that you select.
- Specify options for making the fields in various libraries and cabinets searchable. For example, you can exclude certain fields from the crawl space and specify options for searching attachments.
- Configure document-level security options. If security was enabled when the collection was created, the crawler can associate security data with documents in the index. This data enables search applications to enforce access controls based on the stored access control lists or security tokens.

You can also select an option to validate user credentials at the time a user submits a query. In this case, instead of comparing the user's credentials to indexed security data, the system compares the credentials to current access control lists that are maintained by the original data source.

Click **Help** while you are creating the crawler to learn about the fields in the wizard and how to provide the crawler with the information that the crawler needs to crawl data.

Exchange Server crawlers

To include Microsoft Exchange Server public folders in an enterprise search collection, you must configure an Exchange Server crawler.

You can use the Exchange Server crawler to crawl any number of folders and subfolders on Exchange Server public folder servers. When you create a crawler, you select the content that you want to crawl on a public folder server. Later you can edit the crawl space to add content from another public folder server.

To create or change an Exchange Server crawler, log in to the enterprise search administration console. You must be a member of the enterprise search administrator role or be a collection administrator for the collection that owns the crawler.

When you create the crawler, a wizard helps you do these tasks:

- Specify properties that control how the crawler operates and uses system resources. The crawler properties control how the crawler crawls all subfolders on all servers in the crawl space.
- Specify information about the Exchange Server public folder server that you want to crawl.

You must specify a user ID and password so that the crawler can access content on the server. If the server uses the Secure Sockets Layer (SSL) protocol, you can specify options that enable the crawler to access the keystore file on the crawler server.

- Set up a schedule for crawling the public folder server.
- Select folders and subfolders to crawl.
- Specify options for making documents in subfolders searchable. For example, you can exclude certain types of documents from the crawl space.
- Configure document-level security options. If security was enabled when the collection was created, the crawler can associate security data with documents in the index. This data enables search applications to enforce access controls based on the stored access control lists or security tokens.

Click **Help** while you are creating the crawler to learn about the fields in the wizard and how to provide the crawler with the information that the crawler needs to crawl data.

Verifying access to secure Exchange Server documents

To use an Exchange Server crawler to crawl documents that are protected by a firewall, you must verify that the crawler server is able to access the Microsoft Exchange Server public folder server.

About this task

If the crawler server is not able to access a secure Exchange Server server, you receive HTTP code 501 (Not Implemented) from the server. You might also see messages that indicate that an unexpected HTTP response was received.

Procedure

To ensure that the crawler server can access documents behind the firewall:

1. Launch a Web browser on the crawler server.
2. Go to the URL for the Exchange Server public folder server that you want to crawl. For example: `http://exchange.yourCompany.com/public/`
3. Verify that you can open the Exchange Server page.

If you are not able to access the Exchange Server server, contact the server administrator for your organization.

NNTP crawlers

To include articles from NNTP news groups in an enterprise search collection, you must configure an NNTP crawler.

You can use the NNTP crawler to crawl any number of NNTP servers. When you configure the crawler, you select the news groups on each server that you want to crawl. You can also specify patterns for the news groups that you want to exclude. With this design, you can easily allow the crawler to crawl the majority of news groups on a server, and forbid the crawler from crawling a few news groups that you do not want users to search.

For example, you can specify rules to include all of the news groups on a specific NNTP server, then specify that you want to exclude news groups on that server if their names include the string private.

To create or change an NNTP crawler, log in to the enterprise search administration console. You must be a member of the enterprise search administrator role or be a collection administrator for the collection that owns the crawler.

When you create the crawler, a wizard helps you do these tasks:

- Specify properties that control how the crawler operates and uses system resources. The crawler properties control how the crawler crawls all news groups in the crawl space.
- Specify patterns to include news groups, and specify patterns to exclude certain news groups from the crawl space.
- Configure document-level security options. If security was enabled when the collection was created, the crawler can associate security data with documents in the index. This data enables search applications to enforce access controls based on the stored access control lists or security tokens.

Click **Help** while you are creating the crawler to learn about the fields in the wizard and how to provide the crawler with the information that the crawler needs to crawl data.

Notes crawlers

To include IBM Lotus Notes databases in an enterprise search collection, you must configure a Notes crawler.

Crawler server configuration

If a Lotus Notes server that you plan to crawl uses the Notes remote procedure call (NRPC) protocol, you must run a script on the crawler server. This script, which is provided with WebSphere Information Integrator OmniFind Edition, enables the Notes crawler to communicate with the servers that use NRPC.

If a Lotus Notes server that you plan to crawl uses the Domino Internet Inter-ORB Protocol (DIIOP), you do not need to run a setup script on the crawler server. However, you must configure the Lotus Notes server so that the Notes crawler can access the server.

If WebSphere II OmniFind Edition was installed on an IBM AIX system, you must ensure that the I/O Completion Port module is installed and available on the crawler server.

Before you use the enterprise search administration console to configure a Notes crawler, complete the tasks that are appropriate for your environment:

- “Configuring the crawler server on UNIX to crawl Lotus Domino sources” on page 58.
- “Configuring the crawler server on Windows to crawl Lotus Domino sources” on page 60.
- “Configuring servers that use the DIIOP protocol” on page 61.
- “Configuring the I/O completion port on AIX to crawl Lotus Domino sources” on page 62.

Document-level security

If collection security is enabled, and a Lotus Notes server that you plan to crawl uses the NRPC protocol, you must configure a Lotus Domino Trusted Server on the crawler server. The Trusted Server is used to enforce document-level access controls. Before you make the collection available for users to search, complete the following tasks:

- “Configuring Lotus Domino Trusted Servers to validate user credentials” on page 195.
- Enabling global security in WebSphere Application Server and configuring the search application to use security. This step ensures that users will be prompted to specify credentials when they attempt to use the search application. The search servers can then use these credentials to verify each user’s authority to access to Lotus Notes documents.

Configuration overview

You can use the Notes crawler to crawl any number of standard Lotus Notes databases (.nsf files). When you create the crawler, you select the databases or directories to crawl from a single Lotus Notes server. Later, when you edit the crawl space, you can add documents from another Lotus Notes server that you want to include in the same crawl space. When you create or edit the crawler, you can specify whether you want to crawl all databases or directories on the server, or whether you want to crawl specific databases, views, and folders.

To create or change a Notes crawler, log in to the enterprise search administration console. You must be a member of the enterprise search administrator role or be a collection administrator for the collection that owns the crawler.

When you create the crawler, a wizard helps you do these tasks:

- Specify properties that control how the crawler operates and uses system resources. The crawler properties control how the crawler crawls all of the documents in the crawl space.
- Identify the Lotus Notes server host name, port, and communications protocol.
- Select the databases or directories that you want to crawl.
- Set up a schedule for crawling the databases or directories.
- Select the documents that you want to crawl. You can crawl all documents in a directory, all documents in a database, or documents from selected views and folders of a database.
- Specify options for making the fields in various databases, views, and folders searchable. For example, you can exclude certain fields from the crawl space and specify options for searching attachments.
- Configure document-level security options. If security was enabled when the collection was created, the crawler can associate security data with documents in the index. This data enables search applications to enforce access controls based on the stored access control lists or security tokens.

You can also select an option to validate user credentials at the time a user submits a query. In this case, instead of comparing the user's credentials to indexed security data, the system compares the credentials to current access control lists that are maintained by the original data source.

Click **Help** while you are creating the crawler to learn about the fields in the wizard and how to provide the crawler with the information that the crawler needs to crawl data.

Related concepts

"Enforcement of document-level security for Lotus Domino documents" on page 195

If the Lotus Notes server to be crawled uses the Notes remote procedure call (NRPC) protocol, you must configure the crawler server so that document-level access controls can be enforced.

Related tasks

"Configuring Lotus Domino Trusted Servers to validate user credentials" on page 195

To enforce security for documents that were crawled by a Notes crawler that uses the Notes remote procedure call (NRPC) protocol, the Domino servers to be crawled must be configured to be Lotus Domino Trusted Servers.

Tips for crawling Lotus Domino databases

Review guidelines for crawling for Lotus Domino databases before you configure a Notes crawler.

- Notes databases that are based on standard templates (such as a discussion database) are the best type of database to crawl.
- The Notes crawler applies the following field mapping rules:
 - The major field names from Domino standard templates are initially registered.
 - Values from Notes fields that are specified in the mapping rule table are used as document summaries in the search results.
 - Values from Notes fields that are not specified in the mapping rule table are not used in the document summaries.

- Values from Notes fields that are mapped to the Title field are used as the document title in the search results.
- The fields in the following table are mapped to search field names by default:

Table 1. Default field mapping rules

| Notes database field name | Search field name |
|---------------------------|-------------------|
| Title | Title |
| EventTitle | Title |
| Subject | Title |
| Body | Body |
| Mission | Body |
| From | Creator |
| Author | Creator |
| Keywords | Categories |
| Categories | Categories |
| TeamRoomName | Organization |
| TeamName | Organization |
| Department | Organization |

- The Notes crawler can crawl all types of fields except for computed for display fields.
- Static text and images that are placed on a Notes form are not crawled.
- When you configure the crawler, select the **Crawl All** check box to crawl all fields and maximize the field data to be crawled (you can use the **Crawl all fields except** field to limit the fields to be crawled).
To minimize the crawling of unnecessary fields, clear the **Crawl** check box for all fields except for the fields that are mapped to search fields.

Configuring the crawler server on UNIX to crawl Lotus Domino sources

If you install WebSphere II OmniFind Edition on a computer that is running IBM AIX, Linux, or the Solaris operating environment, and you plan to crawl servers that use the Notes remote procedure call (NRPC) protocol, you must run a script to configure the crawler server. The script enables the Notes, QuickPlace, and Domino Document Manager crawlers to communicate with the database servers.

Restrictions

A Domino Server cannot run at the same time, on the same computer, with a Notes, QuickPlace, or Domino Document Manager crawler that is configured to use the NRPC protocol. If you try to start one of these crawlers while the Domino Server is running, an error occurs and the crawler stops.

About this task

The crawlers that use the NRPC protocol use Domino libraries as a client. You install these libraries by installing Lotus Domino Server Version 6.0.2 or later on the crawler server. To ensure that the crawlers can work with the Domino libraries, you run a setup script that WebSphere II OmniFind Edition provides on the crawler server after you install the Domino libraries.

Procedure

To configure the crawler server so that it can crawl Lotus Notes, Lotus QuickPlace, and Domino Document Manager servers:

1. Create the user notes and the group notes on the crawler server:
 - a. Log in as the root user:

```
su - root
```
 - b. Add a user:

```
useradd notes
```
 - c. Add a password for this user:

```
passwd notes
```

You will be prompted to change the password.
2. Install Lotus Domino Server on the crawler server:
 - a. Insert the Domino Server CD, version 6.0.2 or later, and mount it. (If you do not have a CD, you can download the image.)
 - b. Change to the folder for your operating system.

```
AIX: cd /mnt/cdrom/aix
Linux: cd /mnt/cdrom/linux
Solaris: cd /mnt/cdrom/solaris
```
 - c. Start the installation program:

```
./install
```
 - d. Answer the prompts and accept the default values or specify your preferred installation settings (such as paths for the installation directory and data directory).

Consult the Domino documentation if you need assistance with installing Domino Server.
3. Run the setup script provided by WebSphere II OmniFind Edition on the crawler server:
 - a. Log in as the enterprise search administrator (this user ID was specified when WebSphere II OmniFind Edition was installed).
 - b. Start the following script, which is installed in the `$ES_INSTALL_ROOT/bin` directory:

```
escrnote.sh
```
 - c. Answer the prompts:
 - For the following prompt, answer Y if Domino Server is installed in the default directory, and answer N if it is not:

```
The Lotus Notes directory path /opt/lotus/notes/latest/linux was found.
Is this the correct Lotus Notes directory path?
```

```
The default path for AIX is /opt/lotus/notes/latest/ibmpow.
The default path for Linux is /opt/lotus/notes/latest/linux.
The default path for Solaris is /opt/lotus/notes/latest/sunspa.
```
 - If Domino Server is not installed into the default directory on the crawler server, specify where Domino is installed in response to the following prompt:

```
Enter the path for the Lotus Notes directory
```

For example, on a Linux computer you might specify `/opt/lotus/notes/latest/linux`.

- For the following prompt, answer Y if the Domino Server data directory is installed in the default directory, and answer N if it is not:

The Lotus Notes data directory path /local/notesdata was found.
Is this the correct Lotus Notes data directory path?

The default path is /local/notesdata.

- If the Domino Server data directory is not deployed in the default location on the crawler server, specify the Domino data path in response to the following prompt:

Enter the path for the Lotus Notes data directory.

4. Stop and restart the enterprise search system, including all sessions on the enterprise search common communication layer (CCL):

```
esadmin stop
stopccl.sh
startccl.sh -bg
esadmin start
```

Configuring the crawler server on Windows to crawl Lotus Domino sources

If you install WebSphere II OmniFind Edition on a Microsoft Windows computer, and you plan to crawl servers that use the Notes remote procedure call (NRPC) protocol, you must run a script to configure the crawler server. The script enables the Notes, QuickPlace, and Domino Document Manager crawlers to communicate with the database servers.

Restrictions

Lotus Domino Server and the Lotus Notes client cannot run at the same time, on the same computer, with a Notes, QuickPlace, or Domino Document Manager crawler that is configured to use the NRPC protocol. If you try to start one of these crawlers while the Domino Server is running, an error occurs and the crawler stops.

About this task

The crawlers that use the NRPC protocol use the Lotus Notes client libraries. You install these libraries by installing Lotus Notes Release 6.0.2 or later on the crawler server. To ensure that the crawlers can work with the Lotus Notes client libraries, you run a setup script that WebSphere II OmniFind Edition provides on the crawler server after you install the Lotus Notes client libraries.

Procedure

To configure the crawler server so that it can crawl Lotus Notes, Lotus QuickPlace, and Domino Document Manager servers:

1. On the crawler server, log in with a user ID that is a member of the Administrators group. Ensure that the user ID has authority to install Lotus Notes.
2. Install Lotus Notes:
 - a. Insert the Lotus Notes CD, Release 6.0.2 or later. (If you do not have a CD, you can download the image.)
 - b. Start the installation program: setup.exe

- c. Answer the prompts and accept the default values or specify your preferred installation settings (such as paths for the installation directory and data directory).

Consult the Lotus Notes documentation if you need assistance.

3. Run the setup script that is provided by WebSphere II OmniFind Edition on the crawler server:

- a. Log in with the enterprise search administrator ID (this user ID was specified when WebSphere II OmniFind Edition was installed).

- b. Start the following script, which is installed in the %ES_INSTALL_ROOT%\bin directory:

```
escrnote.vbs
```

- c. Answer the prompts:

- For the following prompt, answer Y if Lotus Notes is installed in the default directory, and answer N if it is not:

```
The Lotus Notes directory path c:\lotus\notes was found.  
Is this the correct Lotus Notes directory path?
```

The typical installation path on a Windows computer is c:\lotus\notes or c:\lotus\domino.

- If Lotus Notes is not installed in the default directory on the crawler server, specify where Lotus Notes is installed in response to the following prompt:

```
Enter the path for the Lotus Notes directory
```

- For the following prompt, answer Y if the Lotus Notes data directory is deployed in the default location, and answer N if it is not:

```
The Lotus Notes data directory path c:\lotus\notes\data was found.  
Is this the correct Lotus Notes data directory path?
```

The typical path on a Windows computer is c:\lotus\notes\data or c:\lotus\domino\data.

- If the Lotus Notes data directory is not deployed in the default location on the crawler server, specify the data directory path in response to the following prompt:

```
Enter the path for the Lotus Notes data directory.
```

4. Stop and restart the enterprise search system, including all sessions on the enterprise search common communication layer (CCL):

- a. At a command prompt, stop the enterprise search system:

```
esadmin stop
```

- b. Select **Start** → **Programs** → **Administrative Tools** → **Services**, then restart the IBM WebSphere Information Integrator OmniFind Edition service.

- c. At a command prompt, start the enterprise search system:

```
esadmin start
```

Configuring servers that use the DIIOP protocol

To crawl servers that use the Domino Internet Inter-ORB Protocol (DIIOP), you must configure the server so that the Notes, QuickPlace, and Domino Document Manager crawlers can use the protocol.

Before you begin

The server that you want to crawl must be running the DIIOP and HTTP tasks.

Procedure

To configure servers that uses the DIIOP protocol:

1. Configure the server document:
 - a. Open the server document on the Lotus Notes, Lotus QuickPlace, or Domino Document Manager server that you want to crawl. This document is stored in the Domino directory.
 - b. On the Configuration page, expand the **server** section.
 - c. On the Security page, in the **Programmability Restrictions** area, specify the appropriate security restrictions for your environment in the following fields:
 - **Run restricted Lotus Script/Java agents**
 - **Run restricted Java/Javascript/COM**
 - **Run unrestricted Java/Javascript/COM**For example, you might specify an asterisk (*) to allow unrestricted access by Lotus Script/Java agents, and specify user names that are registered in the Domino Directory for the Java/Javascript/COM restrictions.

Important: The crawler that you configure to crawl this server with the DIIOP protocol must be able to use the user names that you specify in these fields.
 - d. Open the Internet Protocol page, then open the HTTP page, and set the **Allow HTTP clients to browse database** option to **Yes**.
2. Configure the user document:
 - a. Open the user document on the Lotus Notes, Lotus QuickPlace, or Domino Document Manager server that you want to crawl. This document is stored in the Domino directory.
 - b. On the Basics page, in the **Internet password** field, specify a password. When you use the enterprise search administration console to configure options for crawling this server, specify this user ID and password on the page where you identify the server to crawl. The crawler uses these credentials to access the server.
3. Restart the DIIOP task on the server.

Configuring the I/O completion port on AIX to crawl Lotus Domino sources

Before you can use the Notes, QuickPlace, or Domino Document Manager crawlers on an IBM AIX system, you must install the I/O completion port (IOCP) module and configure it for use by the crawler.

About this task

Without the IOCP module, the discovery processes will fail when you try to create a crawler. The following error message is displayed:

```
FFQM0105E Recieved an error from the server -  
Message: FFQG0024E An unexpected exception was caught: discover
```

The following message, which includes the ENOEXEC error, is written to the \$ES_NODE_ROOT/logs/system_YYYYMMDD.log file. (Some of the message text is split across multiple lines to improve readability.)


```

5/20/05 18:08:52.423 JST [Error] [ES_ERR_EXCEPTION_DEFAULT_MESSAGE] [] [discovery]
ies10.yamato.ibm.com:0:2108088751:control:ComponentDiscoveryW.java:
com.ibm.es.control.discovery.server.ComponentDiscoveryW.discover:86
FFQ00277E An exception was caught with the detail 'java.lang.UnsatisfiedLinkError:
/opt/lotus/notes/65010/ibmpow/liblsxbe_r.a:
load ENOEXEC on shared library(s) /opt/lotus/notes/latest/ibmpow/libnotes_r.a'
and a stack trace of 'java.lang.UnsatisfiedLinkError:
/opt/lotus/notes/65010/ibmpow/liblsxbe_r.a:
load ENOEXEC on shared library(s) /opt/lotus/notes/latest/ibmpow/libnotes_r.a
  at java.lang.ClassLoader$NativeLibrary.load(Native Method)
  at java.lang.ClassLoader.loadLibrary0(ClassLoader.java:2120)
  at java.lang.ClassLoader.loadLibrary(ClassLoader.java:1998)
  at java.lang.Runtime.loadLibrary0(Runtime.java:824)
  at java.lang.System.loadLibrary(System.java:908)
  at lotus.domino.NotesThread.load(NotesThread.java:306)
  at lotus.domino.NotesThread.checkLoaded(NotesThread.java:327)
  at lotus.domino.NotesThread.sinitThread(NotesThread.java:181)
  at com.ibm.es.crawler.discovery.notes.NotesLibrary$NotesOperation.discover
    (Unknown Source)
  at com.ibm.es.crawler.discovery.api.DiscoveryAPI.discover(Unknown Source)
  at com.ibm.es.control.discovery.server.ComponentDiscoveryW.discover
    (ComponentDiscoveryW.java:72)
  at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
  at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:85)
  at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:58)
  at sun.reflect.DelegatingMethodAccessorImpl.invoke
    (DelegatingMethodAccessorImpl.java:60)
  at java.lang.reflect.Method.invoke(Method.java:391)
  at com.ibm.es.ccl.sessionwrapper.CallThread.run(CallThread.java:77)

```

Procedure

To install the IOCP module and ensure that it is correctly installed on the crawler server:

You must

1. Install the IOCP module (bos.iocp.rte) from the AIX product CD on the crawler server.

After you install the IOCP module, and before you create a Notes, QuickPlace, or Domino Document Manager crawler, apply a software fix for the module. See the information at the following link for instructions:

<http://www.ibm.com/support/docview.wss?uid=swg21086556>

2. Enter the following command to ensure that the IOCP module is installed on the crawler server:

```
$ lsllpp -l bos.iocp.rte
```

The output from the lsllpp command should be similar to the following example:

| Fileset | Level | State | Description |
|-------------------------|----------|-----------|--------------------------|
| ----- | | | |
| Path: /usr/lib/objrepos | | | |
| bos.iocp.rte | 5.2.0.10 | COMMITTED | I/O Completion Ports API |
| ----- | | | |
| Path: /etc/objrepos | | | |
| bos.iocp.rte | 5.2.0.10 | COMMITTED | I/O Completion Ports API |

3. Enter the following command to ensure that the status of the IOCP port is **Available**:

```
$ lsdev -Cc iocp
```

The output from the lsdev command should match the following example:

```
iocp0 Available I/O Completion Ports
```

4. If the IOCP port status is **Defined**, change the status to **Available**:
 - a. Log in to the crawler server as root and issue the following command:

```
# smit iocp
```
 - b. Select **Change / Show Characteristics of I/O Completion Ports** and change **STATE to be configured at system restart** from **Defined** to **Available**.
 - c. Reboot the crawler server.
 - d. Enter the `lsdev` command again and confirm that the status of the IOCP port was changed to **Available**.

QuickPlace crawlers

To include Lotus QuickPlace places and rooms in an enterprise search collection, you must configure a QuickPlace crawler.

Crawler server configuration

If a QuickPlace server that you plan to crawl uses the Notes remote procedure call (NRPC) protocol, you must run a script on the crawler server. This script, which is provided with WebSphere Information Integrator OmniFind Edition, enables the QuickPlace crawler to communicate with the servers that use NRPC.

If a QuickPlace server that you plan to crawl uses Domino Internet Inter-ORB Protocol (DIIOP), you do not need to run a setup script on the crawler server. However, you must configure the QuickPlace server so that the QuickPlace crawler can access the server.

If a QuickPlace server that you plan to crawl uses a Lightweight Directory Access Protocol (LDAP) server, then the QuickPlace server must be configured to use the DIIOP protocol (the QuickPlace crawler cannot use the NRPC protocol to crawl LDAP data). You must also configure a Directory Assistance database and configure the QuickPlace server to use the LDAP server as a secondary Domino server.

If WebSphere II OmniFind Edition was installed on an IBM AIX system, you must ensure that the I/O Completion Port module is installed and available on the crawler server.

Before you use the enterprise search administration console to configure a QuickPlace crawler, complete the tasks that are appropriate for your environment:

- “Configuring the crawler server on UNIX to crawl Lotus Domino sources” on page 58.
- “Configuring the crawler server on Windows to crawl Lotus Domino sources” on page 60.
- “Configuring servers that use the DIIOP protocol” on page 61.
- “Configuring the QuickPlace server to use Local User security” on page 65.
- “Configuring Directory Assistance on a QuickPlace server” on page 66.
- “Configuring the I/O completion port on AIX to crawl Lotus Domino sources” on page 62.

Configuration overview

You can use the QuickPlace crawler to crawl any number of QuickPlace places. When you create the crawler, you select the places to crawl from a single

QuickPlace server. Later, when you edit the crawl space, you can add documents from another QuickPlace server that you want to include in the same crawl space. When you create or edit the crawler, you can specify whether you want to crawl all of the rooms in the places that you select for crawling, or whether you want to crawl specific rooms.

To create or change a QuickPlace crawler, log in to the enterprise search administration console. You must be a member of the enterprise search administrator role or be a collection administrator for the collection that owns the crawler.

When you create the crawler, a wizard helps you do these tasks:

- Specify properties that control how the crawler operates and uses system resources. The crawler properties control how the crawler crawls all of the documents in the crawl space.
- Identify the QuickPlace server and communications protocol.
- Specify information about the user directory that is associated with the server (the crawler needs this information so that access controls can be enforced when users search the collection).
- Select the places that you want to crawl.
- Set up a schedule for crawling the places.
- Select the documents that you want to crawl. The crawler can crawl all of the rooms in a place, or crawl only the documents that are in rooms that you select.
- Specify options for making the fields in various places and rooms searchable. For example, you can exclude certain fields from the crawl space and specify options for searching attachments.
- Configure document-level security options. If security was enabled when the collection was created, the crawler can associate security data with documents in the index. This data enables search applications to enforce access controls based on the stored access control lists or security tokens.

You can also select an option to validate user credentials at the time a user submits a query. In this case, instead of comparing the user's credentials to indexed security data, the system compares the credentials to current access control lists that are maintained by the original data source.

Click **Help** while you are creating the crawler to learn about the fields in the wizard and how to provide the crawler with the information that the crawler needs to crawl data.

Configuring the QuickPlace server to use Local User security

If you plan to configure a QuickPlace crawler to use the Local User option for implementing security, you must configure the Domino Directory on the Lotus QuickPlace server before you create the crawler.

About this task

When you configure a QuickPlace crawler, you select a security mode for the crawler to use for enforcing document-level security. If you select the Local User mode, you must ensure that all of the local user IDs and local groups are registered in the Domino Directory (the Domino Directory hierarchy must correspond to the QuickPlace hierarchy).

You must also ensure that the user ID and password that you specify for the crawler to use is registered in the Domino Directory and has permission to read the database to be crawled.

To use QuickPlace, only the user name is required. To crawl QuickPlace sources, however, the fully expanded user ID is required. The expanded user ID is in the following format:

`username/placename/QP/domainname`

Use this procedure to determine the fully expanded version of the user ID, ensure that this user ID is authorized to read the QuickPlace database, and add the user ID to the Domino Directory. The Domino Directory must contain the user ID that will be used to crawl QuickPlace databases and all of the QuickPlace local users and local groups (the Domino Directory hierarchy must correspond to the QuickPlace hierarchy).

Procedure

To configure the QuickPlace server to use Local User security:

1. Confirm the user ID permissions:
 - a. Open the Server document on the QuickPlace server.
 - b. Open the Files page and then open the access control list (ACL) for the database that you want crawl.
 - c. Confirm that the Local User ID that the crawler will be configured to use exists in the ACL and that this user ID has permission to read the database.
You must specify the fully expanded form of this user ID in step 2.
2. Add the user to the Domino Directory:
 - a. Open the Server document on the QuickPlace server.
 - b. On the People and Groups page, in the people tree item, add the fully expanded user ID that you confirmed in step 1.
 - c. In the **Internet password** field, specify the password for this user ID.

Configuring Directory Assistance on a QuickPlace server

If you plan to configure a QuickPlace crawler to use an LDAP directory for implementing security, you must create a Directory Assistance database on the Lotus QuickPlace server before you configure the crawler.

Restrictions

The QuickPlace server that you want to crawl must be running the DIIOP and HTTP tasks.

Procedure

To configure LDAP Directory Assistance on a QuickPlace server:

1. Create a Directory Assistance database:
 - a. Open the Server document on the QuickPlace server.
 - b. Create a database by using the **Directory Assistance(6)** template. This template is on the server.
 - c. Click **Add Directory Assistance** to create a document in the database.
 - d. Open the Basic tab and, in the **DomainType** field, select **LDAP**.

- e. Open the Naming Contexts tab and ensure that the **Trusted for credentials** check box is selected.
 - f. Open the LDAP tab and specify information about the LDAP server.
 - g. Save and close the Server document.
 2. Configure the QuickPlace server to use the Directory Assistance database:
 - a. Open the Server document on the QuickPlace server.
 - b. Open the Basic tab and, in the **Directory assistance database name** field, specify the name of the database that you created in step 1.
 - c. Save and close the Server document.
- The QuickPlace server can now use the LDAP server as a secondary Domino directory.

UNIX file system crawlers

To include documents that are stored in UNIX file systems in an enterprise search collection, you must configure a UNIX file system crawler.

You can use the UNIX file system crawler to crawl any number of UNIX file systems. When you configure the crawler, you select the local and remote directories and subdirectories that you want to crawl.

If you install the crawler server on a Windows computer, you cannot use that server to crawl UNIX file system sources (the UNIX file system crawler does not appear in the list of available crawler types).

To create or change a UNIX file system crawler, log in to the enterprise search administration console. You must be a member of the enterprise search administrator role or be a collection administrator for the collection that owns the crawler.

When you create the crawler, a wizard helps you do these tasks:

- Specify properties that control how the crawler operates and uses system resources. The crawler properties control how the crawler crawls all subdirectories in the crawl space.
- Set up a schedule for crawling the file systems.
- Select the subdirectories, and the levels of subdirectories, that you want the crawler to crawl.
- Specify options for making documents in subdirectories searchable. For example, you can exclude certain types of documents from the crawl space.
- Configure document-level security options. If security was enabled when the collection was created, the crawler can associate security data with documents in the index. This data enables search applications to enforce access controls based on the stored access control lists or security tokens.

Click **Help** while you are creating the crawler to learn about the fields in the wizard and how to provide the crawler with the information that the crawler needs to crawl data.

Web crawlers

To include pages from Web sites in an enterprise search collection, you must configure a Web crawler.

You can use the Web crawler to crawl any number of Hypertext Transfer Protocol (HTTP) servers and secure HTTP (HTTPS) servers. The crawler visits a Web site and reads the data on the site. It then follows links in documents to crawl additional documents. The Web crawler can crawl and extract links from individual pages or *framesets* (pages that are created with HTML frames).

The crawled data can be in one of many common formats, and comes from various sources within your intranet or the Internet. Common formats include HTML, PDF, Microsoft Word, Lotus WordPro, Extensible Markup Language (XML), and so on.

To create or change a Web crawler, log in to the enterprise search administration console. You must also be a member of the enterprise search administrator role or be a collection administrator for the collection that owns the crawler.

When you create the crawler, a wizard helps you do these tasks:

- Specify properties that control how the crawler operates and uses system resources. The crawler properties control how the crawler crawls all Web pages in the crawl space.
- Specify rules to allow and forbid visits to Web sites. When you specify crawling rules, you can test the rules and verify that the crawler is able to access the sites that you want to include in the crawl space.
- Specify options to include certain types of files and exclude files with certain file extensions.
- Specify rules for how the Web crawler handles soft error pages.
- Configure document-level security options. If security was enabled when the collection was created, the crawler can associate security data with documents in the index. This data enables search applications to enforce access controls based on the stored access control lists or security tokens.
- Specify options for crawling password-protected Web sites (the Web servers to be crawled must use HTTP basic authentication or HTML forms to prompt for passwords).
- Specify options to crawl Web sites that are served by a proxy server.

Click **Help** while you are creating the crawler to learn about the fields in the wizard and how to provide the crawler with the information that the crawler needs to crawl data.

User agent configuration

The Web crawler complies with the Robots Exclusion protocol. To crawl a Web site that uses this protocol, ensure that the robots.txt file on the Web site allows the user agent name that you configure for the Web crawler to access the Web site.

When the enterprise search system is started, the Web crawler loads the user agent name that you configure for it. Before the crawler downloads a page from a Web site that it has not previously visited (or that it has not visited for some time), the crawler first tries to download a file called robots.txt. This file is in the root directory of the Web site.

If the robots.txt file does not exist, the Web site is open to unrestricted crawling. If the file does exist, it specifies what areas of the site (directories) are off limits to crawlers. The robots.txt file specifies permissions for crawlers by identifying their user-agent name.

The Robots Exclusion protocol is voluntary, but the enterprise search Web crawler complies with it:

- If a robots.txt file contains an entry for the user agent name that is configured for the Web crawler, then the Web crawler complies with the restrictions on that user agent.
- If the user agent name does not appear in the robots.txt file, but the last entry specifies `User-agent: *` (which means any user agent) and the restriction is `Disallow: /` (which means do not allow any crawling, starting at the root of the Web site), then the Web crawler is barred from crawling that site.
- If the user agent name does not appear in the robots.txt file, but the last entry specifies `User-agent: *` and the restriction is `Allow: /`, then the Web crawler is allowed to crawl that site.

Web site administrators often specify a final entry that bars access to all crawlers that are not explicitly granted access. If you are configuring a new Web crawler and you know that some of the Web sites that you want to crawl use the Robots Exclusion protocol, ask the Web site administrators to add an entry for your crawler to their robots.txt files.

Be sure to specify the same user agent name in the Web crawler's properties and in all robots.txt files that belong to the Web sites of interest.

If none of the Web sites to be crawled use the Robots Exclusion protocol, then the value that you specify for the user agent property typically does not matter. However, some application servers, JSPs, and servlets tailor their responses to the user agent name. For example, different responses exist to handle browser incompatibilities. The user agent name that you specify for the Web crawler might matter in these situations, regardless of the Robots Exclusion protocol. If you need to crawl these types of sites, consult with the Web site administrators to ensure that the Web crawler is allowed access.

Support for JavaScript

The Web crawler for enterprise search can find some links (URLs) that are contained in the JavaScript™ portions of Web documents.

The Web crawler can find both relative and absolute links. If an HTML document contains a BASE element, the crawler uses that element to resolve relative links. Otherwise, the crawler uses the document's own URL.

Support for JavaScript is limited to link extraction. The crawler does not parse JavaScript, does not build a DOM (Document Object Model), and does not interpret or execute JavaScript statements. The crawler looks for strings in the document content (including, but not limited to the JavaScript portions) that are likely to be URLs in JavaScript statements. This means two things:

- Some URLs will be found that are ignored by the stricter HTML parser. The crawler will reject anything that is not a syntactically valid URL, but some of the valid URLs returned by the scanning step might be of low interest for searching.
- Document content that is generated by JavaScript, such as when a human user views a page with a browser and the browser executes some JavaScript, will not be seen by the Web crawler, and thus will not be indexed.

Rules to limit the Web crawl space

To ensure that users access only the Web sites that you want them to search, you specify rules to limit what the Web crawler can crawl.

When a Web crawler crawls a Web page, it discovers links to other pages and puts those links in a queue to be crawled next. Crawling and discovery can be repeated as long as time and memory resources permit. When you configure a Web crawler, you specify where the crawler is to begin crawling. From these initial URLs (which are called *start URLs*) the Web crawler can reach any document on the Web that is connected by direct or indirect links.

To limit the crawl space, configure the Web crawler to crawl certain URLs thoroughly and ignore links that point outside the area of interest. Because the crawler, by default, accepts any URL that it discovers, you must specify rules that identify which URLs you want to include in the collection, and eliminate the rest of the pages.

You can specify in several ways what you want the Web crawler to crawl and not crawl. You can specify:

- A list of start URLs where the crawler is to begin crawling
- Three types of crawling rules: domain, Internet Protocol (IP) address, and URL prefix
- A list of MIME types for documents that you want to include
- A list of file extensions for documents that you want to exclude
- The maximum number of directories in a URL path

Crawling rules have the form:

```
action type target
```

action is forbid or allow; type is domain, IP address, or URL prefix (HTTP or HTTPS); and target depends on the value of type. You can specify an asterisk (*) as a wildcard character, in limited ways, to specify targets that match a pattern.

Domain rules

The target of a domain rule is a DNS domain name. For example, you can specify that the entire `www.ibm.com` domain is to be crawled:

```
allow domain www.ibm.com
```

The first character in the target can be an asterisk, which causes the rule to apply to any host name that ends with the rest of the pattern. For example, you can specify that no domains that match the following pattern are to be crawled:

```
forbid domain *.ibm.com
```

Host name matching is case sensitive, whether you specify an explicit domain name or a domain name pattern. For example, `*.user.ibm.com` matches `joe.user.ibm.com` and `mary.smith.user.ibm.com`, but not `joe.user.IBM.com`.

A domain rule that does not specify a port number applies to all ports on that domain. In the following example, all ports on the sales domain are allowed:

```
allow domain sales.ibm.com
```

If a domain rule specifies a port number, then the rule applies only to that port. In the following example, only port 443 on the sales domain is allowed:

```
allow domain sales.ibm.com:443
```


Prefix rules

A prefix rule controls the crawling of URLs that begin with a specified string. The target is a single URL, which typically contains one or more asterisks to signify a pattern. For example, an asterisk is often specified as the final character in the prefix string.

A prefix rule enables you to crawl all or part of a Web site. You can specify a directory path or pattern, and then allow or forbid everything from that point on in the directory tree. For example, the following rules work together to allow the crawler to crawl everything in the public directory at sales.ibm.com, but forbid the crawler from accessing any other pages on the site:

```
allow prefix http://sales.ibm.com/public/*
forbid prefix http://sales.ibm.com/*
```

When you specify prefix rules, you can specify more than one asterisk and you can specify them anywhere in the prefix string, not just in the last position. For example, the following rule forbids the crawler from crawling any documents in a top-level directory of the sales.ibm.com site if the directory name ends in fs. (For example, you might have file system mounts that do not contain information that would be useful in the search index.)

```
forbid http://sales.ibm.com/*fs/*
```

Address rules

An address rule enables you to control the crawling of entire hosts or networks by specifying an IP address (IPv4 only) and netmask as the target. For example:

```
allow address 9.0.0.0 255.0.0.0
```

The netmask enables you to specify pattern matching. For an address rule to apply to a candidate IP address, the IP address in the rule and the candidate IP address must be identical, except where masked off by zeros in the netmask. The address rule defines a pattern, and the netmask defines the significant bits in the address pattern. A zero in the netmask acts as a wildcard and signifies that any value that is specified in that same bit position in the address matches.

In the preceding example, the allow rule applies to any IP address with 9 in the first octet, and any value at all in the last three octets.

A useful rule to include as the final address in your list of rules is `forbid address 0.0.0.0 0.0.0.0`. This rule matches any IP address because the netmask makes all bits insignificant (the rule forbids all addresses that are not allowed by a preceding rule in your list of rules).

Restrictions for proxy servers: If you plan to crawl Web sites that are served by a proxy server, do not specify IP address rules. A proxy server is typically used when a user agent (browser or crawler) does not have direct access to the networks where the Web servers are. For example, an HTTP proxy server can relay HTTP requests from a crawler to a Web server, and convey the responses back to the crawler.

When a Web crawler uses a proxy server, the IP address of the proxy server is the only IP address that the crawler has for another host. If IP address

rules are used to constrain the crawler to a subnet of IP addresses, the constraint causes almost all URLs to be classified with return code 760 (which indicates that they are forbidden by the Web space).

Crawling rule order

The crawler applies the crawling rules at various times during the process of discovering and crawling URLs. The order of the rules is important, but only within the rules of a each type. It makes a difference whether an address rule comes before or after another address rule, but it makes no difference whether an address rule comes before or after a prefix rule, because the crawler does not apply the rules at the same time.

Within the set of rules for a single type, the crawler tests a candidate domain, address, or URL against each rule, from the first specified rule to the last, until it finds a rule that applies. The action specified for the first rule that applies is used.

The dependency on order leads to a typical structure for most crawling rules:

- The set of domain rules typically begins with forbid rules that eliminate single domains from the crawl space. For example, the collection administrator might determine that certain domains do not contain useful information.
- The list of forbid rules is typically followed by a series of allow rules (with wildcard characters) that enable the crawler to visit any domain that ends in one of the high-level domain names that define an enterprise intranet (such as *.ibm.com and *.lotus.com).

End the set of domain rules with the following default rule, which eliminates domains that were not allowed by a preceding rule:

```
forbid domain *
```

This final rule is critical, because it prevents the crawl space from including the entire Internet.

- The set of address rules typically begins with a small number of allow rules that enable the crawler to crawl the high-level (class-A, class-B, or class-C) networks that span an enterprise intranet.

The address rules typically end with the following rule, which prevents the crawler from crawling Web sites that are outside the corporate network.

```
forbid 0.0.0.0 0.0.0.0
```

- The set of prefix rules is usually the largest, because it contains arbitrarily detailed specifications of allowed and forbidden regions that are specified as trees and subtrees. A good approach is to allow or forbid more tightly localized regions first, and then specify the opposite rule, in a more general pattern, to allow or forbid everything else.

The prefix section does not typically end with a typical rule. The suggested final domain and address rules can ensure that the crawler does not crawl beyond the enterprise network more efficiently than by testing URL prefixes.

The crawler can apply prefix rules more efficiently if you group the rules by action (forbid or allow). For example, instead of specifying short sequences of allow and forbid rules that alternate with each other, specify a long sequence of rules that stipulate one action and then specify a long sequence of rules that stipulate the other action. You can interweave allow and forbid rules to achieve the goals of your crawl space. But grouping the allow rules together and the forbid rules together can improve crawler performance.

File extensions, MIME types, and maximum crawl depth

These options provide additional ways for you to specify content for the crawl space. You can exclude certain types of documents based on document's file extension, and you can include certain types of documents based on the document's MIME type. When you specify which MIME types you want the crawler to crawl, consider that the MIME type is often set incorrectly in Web documents.

The maximum crawl depth is the number of slashes in a URL from its site root. This option enables you to prevent the crawler from being drawn into recursive file system structures of infinite depth. The crawl depth does not correspond to the levels that the crawler traverses when it follows links from one document to another.

Start URLs

Start URLs are the URLs that the crawler begins crawling with, and these URLs are inserted into the crawl every time the crawler is started. If the start URLs were already discovered, they will not be crawled or recrawled sooner than other Web sites that you allow in the crawling rules.

A start URL is important the first time that a Web crawler is started and the crawl space is empty. A start URL is also important when you add a URL that was not previously discovered to the list of start URLs in a crawl space.

Start URLs must be fully qualified URLs, not just domain names. You must specify the protocol and, if the port is not 80, the port number.

The following URLs are valid start URLs:

```
http://w3.ibm.com/  
http://sales.ibm.com:9080/
```

The following URL is not a valid start URL:

```
www.ibm.com
```

You must include the start URLs in your crawling rules. For example, the crawler cannot begin crawling with a specified start URL if the crawling rules do not allow that URL to be crawled.

Tests of URL connections with the Web crawler

After you specify URLs for the Web crawler to crawl, you can test the configuration of the crawling rules.

The test results show whether the crawler is able to access URLs with the user agent name that is specified in the crawler properties. The test results also show whether a URL cannot be crawled because of exclusion rules (for example, a document might not be crawled because it has a file extension that matches an extension that is excluded from the crawl space).

After a site is crawled at least once, you can test URLs to obtain additional information. For example, the test report can provide the most recent HTTP return code (which indicates whether a crawl of the URL was successful), show when the URL was last crawled and when it is scheduled to be crawled again, and show whether the user agent is using the Web server's current robots.txt file.

Recrawl interval settings in the Web crawler

To influence how frequently the Web crawler revisits URLs, you specify options in the Web crawler properties.

Most of the other crawler types in an enterprise search system run according to schedules that an administrator specifies. In contrast, after you start a Web crawler, it typically runs continuously. To control how often it revisits URLs that it previously crawled, you specify minimum and maximum recrawl intervals.

When you use the enterprise search administration console to create a Web crawler or to edit Web crawler properties, you can select an option to configure advanced properties. On the Advanced Web Crawler Properties page, you specify minimum recrawl interval and maximum recrawl interval options. The Web crawler uses the values that you specify to calculate an interval for recrawling data.

The first time that a page is crawled, the crawler uses the date and time that the page is crawled and an average of the specified minimum and maximum recrawl intervals to set a recrawl date. The page will not be recrawled before that date. The time that the page will be recrawled after that date depends on the crawler load and the balance of new and old URLs in the crawl space.

Each time that the page is recrawled, the crawler checks to see if the content has changed. If the content has changed, the next recrawl interval will be shorter than the previous one, but never shorter than the specified minimum recrawl interval. If the content has not changed, the next recrawl interval will be longer than the previous one, but never longer than the specified maximum recrawl interval.

Options for visiting URLs with the Web crawler

You can force the Web crawler to visit specific URLs as soon as possible.

If you need to refresh the crawl space with information from certain Web sites, you can monitor the crawler, select the **URLs to visit or revisit** option, then specify the URLs or URL patterns of the pages that need to be crawled or recrawled.

For example, if your Communications department adds a Web page to your intranet or revises a page to reflect an important policy change, you can specify the URL of the new or changed page. If the crawler is running, the crawler queues the specified URL for crawling the next time that it checks for pages that are waiting to be visited (typically every ten minutes). If the crawler is not running, it queues the specified URL so that it can be crawled the next time that the crawler is started.

Ensure that the crawling rules include a rule that allows the crawler to visit the URLs that you specify. The crawler can visit the URLs that you specify sooner than it normally would. However, for a URL to be crawled at all, a crawling rule must exist that allows the URL to be crawled.

The newly crawled data becomes available for searching the next time that the index is reorganized or refreshed.

How the Web crawler handles soft error pages

You can configure the Web crawler to handle custom pages that Web site administrators create when they do not want to return a standard error code in response to requests for certain pages.

If an HTTP server cannot return the page that a client requests, the server normally returns a response that consists of a header with a return code. The return code indicates what the problem is (such as error 404, which indicates that the file could not be found). Some Web site administrators create special pages that explain the problem in more detail and configure the HTTP server to return these pages instead. These custom pages are called *soft error pages*.

Soft error pages can distort the Web crawler's results. For example, instead of receiving a header that indicates a problem, the crawler receives a soft error page and the return code 200, which indicates the successful download of a valid HTML page. But this downloaded soft error page is not related to the requested URL, and its content is nearly identical each time it is returned in place of a requested page. These irrelevant and near-duplicate pages distort the index and search results.

To handle this situation, you can specify options for handling soft error pages when you configure the Web crawler. The Web crawler needs the following information about each Web site that returns soft error pages:

- A URL pattern for a site that uses soft error pages. This URL pattern consists of the protocol (HTTP or HTTPS), the host name, port number (if non standard), and path name. You can use an asterisk (*) as a wildcard character to match one or more characters up to the next occurrence of a non-wildcard character in the pattern. The pattern that you specify is case sensitive.
- A title pattern for text that corresponds to the <TITLE> tag of an HTML document. You can use the asterisk (*) as a wildcard character to specify this pattern. This pattern that you specify is case sensitive.
- A content pattern for text that corresponds to the content of an HTML document. The content is not just the content of the <BODY> tag, if a <BODY> tag is present. The content is everything that comes after the HTTP header in the file. You can use the asterisk (*) as a wildcard character to specify this pattern. This pattern that you specify is case sensitive.
- An integer that represents the return code to use for documents that match the URL, title, and content patterns that you specified.

Example

This following configuration tells the Web crawler to compare all valid HTML pages (return code 200) that are returned from the `http://www.mysite.com/hr/*` Web site to the specified title and content patterns. If the <TITLE> tag of a page begins with "Sorry, the page" and the content of the document contains anything (*), then the crawler handles the page the same way it would a return code 404 (the page was not found).

Table 2. Soft error page example

| URL pattern | Title pattern | Content pattern | HTTP return code |
|---|------------------|-----------------|------------------|
| <code>http://www.mysite.com/hr/*</code> | Sorry, the page* | * | 404 |

You can create multiple entries for the same Web site to handle different return codes. Each return code from the same Web site requires its own entry in the Web crawler's configuration.

Using wildcard characters

The URL, title, and content patterns are not regular expressions. The asterisk character matches any characters up to the next occurrence of any non-wildcard character. For example:

*404 matches *any characters*404
404: * matches 404: any characters
http://*.mysite.com/* matches http://*any host*.mysite.com/*any file*
* matches *any characters*

Performance impact

When you configure options for handling soft error pages, you increase the amount of crawler processing time because all successfully crawled pages must be checked. More processing time is required to check for pattern matches and determine whether a page or a replacement return code should be returned.

Support for crawling secure Web sites

By specifying credentials in the enterprise search administration console, you can enable the Web crawler to access restricted content, such as documents that require a password for access.

If a Web server uses HTTP basic authentication or HTML form-based authentication to restrict access to Web sites, you can specify credentials in the Web crawler's configuration that enable pages on the password-protected Web sites to be crawled. You can also specify options for manually configuring cookie files.

Web sites protected by HTTP basic authentication

If a Web server uses HTTP basic authentication to restrict access to Web sites, you can specify authentication credentials that enable the Web crawler to access password-protected pages.

To determine whether a user (or client application) has permission to access pages on a Web site, many Web servers use a client authentication scheme called HTTP basic authentication to establish the user's identity. Typically, this interaction is interactive:

- When an HTTP user agent (such as a Web browser) requests a page that is protected by HTTP basic authentication, the Web server responds with a 401 return code, which indicates that the requestor is not authorized to access the requested page.
- The Web server also challenges the requestor to present credentials that can be used to verify whether the user is allowed to access the restricted content.
- The Web browser presents the user with a dialog that requests a user name, password, and any other information that is required to constitute the user's credentials.
- The Web browser encodes the credentials, then includes them when it repeats the request for the protected page.
- If the credentials are valid, the Web server responds with a 200 return code and the contents of the requested page.
- Subsequent requests for pages from the same Web server typically include the same credentials, enabling the authorized user to access additional restricted content without being challenged to specify credentials with each request.

Once a user's identity is established, the Web server and HTTP user agent typically exchange tokens, called *cookies*, that enable knowledge of the user's login status to be maintained between HTTP requests.

Because the Web crawler does not run interactively, the credentials that enable it to crawl password-protected pages must be specified before the crawler begins crawling. When you create a Web crawler or edit the crawl space, specify information about each secure Web site that needs to be crawled.

To specify this information, you must work closely with the administrators for the Web sites or Web servers that are protected by HTTP basic authentication. They must provide you with the security requirements for the Web sites to be crawled, including all information that is used to authenticate the Web crawler's identity and determine that the crawler has permission to crawl the restricted pages.

If security was enabled for the collection when the collection was created, you can specify security tokens, such as user IDs, group IDs, or user roles, to control access to documents when you configure the crawler. The Web crawler associates these security tokens with every document that it crawls in the file system tree for the specified root URL. The tokens are used in addition to any document-level security tokens that you configure for the entire Web crawl space.

The order of the URLs is important. After you add information about a password-protected Web site, you must position it in the order that you want the crawler to process it. List the more specific URLs first, and put the more generic URLs lower in the list. When the Web crawler evaluates a candidate URL, it uses the authentication data that is specified for the first URL in the list that matches the candidate URL.

Web sites protected by form-based authentication

If a Web server uses HTML forms to restrict access to Web sites, you can specify authentication credentials that enable the Web crawler to access password-protected pages.

To determine whether a user (or client application) has permission to access pages on a Web site, many Web servers use HTML forms to establish the user's identity. Typically, this interaction is interactive:

- When an HTTP user agent (such as a Web browser) requests a page that is protected by form-based authentication, the Web server checks to see whether the request includes a cookie that establishes the user's identity.
- If the cookie is not present, the Web server prompts the user to enter security data into a form. When the user submits the form, the Web server returns the required cookies, and the request for the password-protected page is allowed to proceed.
- Future requests that include the required cookies are also allowed to proceed. The authorized user is able to access additional restricted content without being challenged to fill in a form and specify credentials with each request.

Because the Web crawler does not run interactively, the credentials that enable it to crawl password-protected pages must be specified before the crawler begins crawling. When you create a Web crawler or edit the crawl space, specify information about each secure Web site that needs to be crawled.

The fields that you specify correspond to the fields that an interactive user fills in when prompted by the Web browser, and any hidden or static fields that are required for a successful login.

To specify this information, you must work closely with the administrators for the Web sites or Web servers that are protected by form-based authentication. They must provide you with the security requirements for the Web sites to be crawled, including all information that is used to authenticate the Web crawler's identity and determine that the crawler has permission to crawl the restricted pages.

The order of the URL patterns is important. After you add information about a password-protected Web site, you must position it in the order that you want the crawler to process it. List the more specific URL patterns first, and put the more generic URL patterns lower in the list. When the Web crawler evaluates a candidate URL, it uses the form data that is specified for the first URL pattern in the list that matches the candidate URL.

Web sites that are served by proxy servers

If the Web crawler is not permitted direct access to a network, you can configure the crawler to use an HTTP proxy server to access the content that you want to crawl.

If access to a TCP/IP network is not available on the computer where the Web crawler is to run, or if access is restricted to privileged processes, you can configure the Web crawler to use an HTTP proxy server. An HTTP proxy is a process that listens at a specified port on a specified host for HTTP requests. The proxy server relays requests to the Web server, and relays responses from the Web server to the requesting client (the Web crawler). A proxy server can run on the same computer with the Web crawler, or run on a different computer.

In non-proxy crawling, a request for a URL is sent directly to the host. With proxy crawling, the request is sent to the proxy server.

When you create a Web crawler or edit the crawl space, specify information about the proxy servers that the Web crawler is to use when crawling pages in the proxy server's domain. Obtain the following information before you add a proxy server to the crawl space:

Proxy server domains

The domains that are served by the proxy server. You can use an asterisk (*) as a wildcard character. For example, * matches all domains that are served by this proxy server, and *.resource.com matches all domains that end in resource.com.

Restriction: You cannot specify IP address rules to crawl a proxy server because the IP address of the proxy server is the only IP address that the crawler has for another host. If IP address rules are used to constrain the crawler to a subnet of IP addresses, the constraint causes almost all URLs to be classified with return code 760 (which indicates that they are forbidden by the Web space).

Proxy server host name or IP address

The DNS host name or dotted IP address of the proxy server.

Proxy server port number

The TCP/IP port number where the proxy server listens for HTTP proxy requests.

After you add a proxy server, you must select it and position it in the order that you want the crawler to process it. List the more specific domain names first, and put the more generic domain names lower in the list. When the Web crawler evaluates a candidate URL, it uses the proxy server data that is specified for the first domain in the list that matches the candidate URL. (URLs that do not match any proxy rule are assumed to be directly accessible to the crawler.)

Cookie administration

Typically, cookie administration occurs automatically, with no action required from an enterprise search administrator. If necessary, you can manually specify cookies for a Web crawling session.

Cookies are opaque tokens that a Web server returns to a user agent as part of an HTTP response header. They are meaningful only to the Web server that issued them, and they are used to maintain state between HTTP requests. For example, during client authentication, the Web server might return a cookie that enables the server to determine that an authenticated user is already logged in. The presence of the cookie enables the user to issue additional requests for pages on that Web server without being prompted to log in again.

The Web crawler retains cookies that are received from Web servers and uses them for the duration of the crawler instance. It stores the cookies in a cookies.ini file, which is rewritten by the crawler at the end of every crawler session. When the Web crawler stops, it saves all unexpired cookies, then reloads them at the start of the next session.

If you manually specify cookies, store them in a separate file, and then merge them with the cookies in the cookies.ini file when needed. The crawler does not discard unexpired cookies, but if a problem prevents the writing of the entire cookie collection, you do not want to lose the cookies that you manually specified. You must merge your cookies with the cookies that the crawler automatically maintains before the start of a crawling session.

Cookie format

Cookies that you plan to merge with the enterprise search cookies.ini file must be in a particular format.

- Each cookie must be on a single line. Blank lines and comments are permitted, but they will not be preserved in the cookies.ini file.
- Each cookie must have the following format:

```
CookieN(cookie_length,URL_length)cookie_text,validation_URL
```

Cookie

A required keyword that indicates the start of a cookie entry.

The Cookie keyword cannot contain blanks and it must have a single digit appended to it, either 0, 1, or 2. The digit indicates the cookie type: version-0 (Netscape), version-1 (RFC2109), or version-2 (RFC2965). Port lists are not supported in RFC2965 cookies.

cookie_length

The length in characters of the associated cookie text.

URL_length

The length in characters of the associated validation URL.

cookie_text

The content of the cookie that is to be sent to the originating Web server. This string (which represents the right side of the Set-Cookie directive in an HTTP response header) specifies the cookie's name and value pair and any other content (such as a path, security setting, and so on) to be sent with the cookie. This string is followed by a comma (,) separator.

validation_URL

The URL at which this cookie was discovered. This URL is used to determine where to send the cookie (for example, by supplying a domain name and path name). The validation URL must satisfy the originating Web server's security and privacy restrictions on cookies.

The following example is shown on two lines for readability; cookies that you specify must be on a single line:

```
Cookie0(53,40)ASPSESSIONIDQSQTACSD=SLNSIDFNLSIDNFLSINFLSNL;path=/  
https://www.ibm.com:443/help/solutions/
```

Configuring cookies for the Web crawler

You can manually specify cookies for a Web crawling session, and merge them with cookies that the Web crawler maintains.

Before you begin

To manually configure cookies for the Web crawler to use, you must be an enterprise search administrator.

Procedure

To manually configure cookies for a Web crawler:

1. From the enterprise search administration console, monitor the collection that you want to specify cookies for, and stop the Web crawler.
2. Log in as the enterprise search administrator on the crawler server (this user ID was specified when WebSphere II OmniFind Edition was installed).
3. Change to the data directory for the crawler that you want to configure, where *crawler_session_ID* is an ID that was assigned to the crawler session by the enterprise search system. For example:
`ES_NODE_ROOT/data/col_56092.WEB_88534`
4. Edit the cookies.ini file, append the cookie entries that you manually specified to the ones that are already listed, then save and exit the file. Ensure that your cookies do not override any that are already present.
5. From the enterprise search administration console, restart the Web crawler that you stopped.

Global Web crawl space configuration

You can configure a global crawl space for Web crawlers, which enables you to better control the removal of URLs from the index.

Each Web crawler is configured with a crawl space that defines the URLs that are to be crawled or not crawled. Discovered URLs that are in the crawl space are retained (in a database) for later crawling; URLs that are not in the crawl space are

discarded. If the crawler starts with an empty database, the crawl space definition and database remain consistent while the crawler runs.

Sometimes a crawler is stopped, and its crawl space is reduced (for example, by new rules that forbid pages to be crawled). When the crawler is restarted, its crawl space definition and database become inconsistent. The database contains URLs (some crawled and some not crawled) which are not in the new, smaller crawl space.

If a collection has only one Web crawler, the Web crawler can restore consistency by changing the HTTP return codes for these URLs to 760 (which specifies that they are to be excluded) and requesting the removal of the now-excluded pages from the index.

If you divide the crawl space between two or more Web crawlers (for example, to ensure some pages are crawled more often than the rest), each Web crawler maintains independent database tables (initially empty), and they each crawl a different part of the Web crawl space. The original crawler's crawl space is then reduced to whatever is left after the parts to be crawled by other crawlers are removed. Problems arise when the original crawler attempts to restore consistency by removing the moved pages from the index. Because the moved pages are now being crawled by other crawlers, the pages should remain in the index.

By configuring a higher level, global crawl space you can identify URLs that are not to be crawled by the original crawler, but are not to be removed from the index, either. URLs that are no longer in any crawler's crawl space continue to be marked for exclusion by the discovery processes, and are removed from the index when they are recrawled.

The global crawl space is defined by a configuration file named `global.rules`, which must exist in the crawler configuration directory (the presence of a `global.rules` file enables the global crawl space function). If this file exists, it is read during crawler initialization. If this file does not exist, the crawler operates with a single-level crawl space, and removes documents from the index as necessary to maintain consistency between its crawl space definition and database.

If a global crawl space exists, the crawler rules URLs in or out as before, but will request the removal of a URL from the index only if the URL is not in any Web crawl space.

The `global.rules` file has the same syntax as the local `crawl.rules` file, except that it can contain only domain name rules. This restriction enables a crawl space to be partitioned between crawlers only on the basis of DNS host names, not IP addresses or HTTP prefix patterns. URLs that are excluded by URL prefix or IP address rules in the local crawl space (as defined in the `crawl.rules` file) are unaffected by the global crawl space; such URLs are still excluded.

The global crawl space is used only to prevent the removal of URLs, which are excluded from one crawler's crawl space by a local domain rule, from the index. The following rules apply in the following order:

1. If a URL from the crawler's database is excluded by a local prefix rule or address rule, the URL is assigned return code 760 and it is removed from the index. The URL will not be crawled again.

2. If a URL from the crawler's database is excluded by a local domain rule, and there is no global crawl space, the URL is assigned return code 760, and it is removed from the index. The URL will not be crawled again.
3. If a URL from the crawler's database is excluded by a local domain rule, but explicitly allowed by a rule in the global crawl space, the URL is assigned return code 761. The crawler will not crawl the URL again, but it is not removed from the index (it is assumed to be in some other crawler's local crawl space).
4. If a URL from the crawler's database is excluded by a local domain rule, and not explicitly allowed by a rule in the global crawl space, the URL is assigned return code 760, and removed from the index.

Because the global crawl space is consulted only to prevent the deletion of URLs that have already been excluded by the local crawl space, the default result from the global crawl space, if no rule applies to a candidate URL, is to forbid it from being crawled.

The `global.rules` file must exist in the `master_config` directory of every crawler that shares the global crawl space. You must carefully edit all copies of the `global.rules` file and the individual local `crawl.rules` files to ensure that they remain mutually consistent.

No-follow and no-index directives

You can improve search quality by specifying directives for the Web crawler that control whether links on pages are followed and whether pages are indexed.

Some Web pages have no-follow or no-index directives, which instruct robots (such as the Web crawler) to not follow links found in those pages, to not include the contents of those pages in the index, or to not do either of these actions.

Controlling these settings can improve the quality of the crawl. For example, some directory pages can contain thousands of links but no other useful content; those pages should be crawled, and their links followed, but there is no benefit to indexing the directory pages themselves.

There might also be times when you want the crawler to go no lower in a hierarchy, but the desired leaf pages contain links and do not contain no-follow directives. Because some of these pages are automatically generated, they have no owners who might insert the required directives.

To specify rules for crawling such pages, you create or edit a configuration file named `followindex.rules`. Use the following guidelines when you specify rules in this file:

- The rules that you configure must specify URL prefixes (you cannot identify Web sites by IP address or DNS host name).
- The URL prefixes can include asterisks (*) as a wildcard character to allow or forbid multiple sites with similar URLs.
- Order is significant (the crawler applies the first rule that matches a candidate URL).
- The rules, which explicitly allow and forbid following or indexing, override other settings, including those in the target document.

Overriding no-follow and no-index directives in Web pages

You can specify rules in a configuration file to control whether the Web crawler follows links to pages or indexes pages that contain no-follow or no-index directives.

Before you begin

To specify no-follow and no-index directives for the Web crawler, you must be an enterprise search administrator. The directives that you specify override directives that exist in the pages to be crawled.

Procedure

To override no-follow and no-index directives:

1. From the enterprise search administration console, monitor the collection that you want to configure rules for, and stop the Web crawler.
2. Log in as the enterprise search administrator on the crawler server (this user ID was specified when WebSphere II OmniFind Edition was installed).
3. Change to the configuration directory for the crawler that you want to configure, where *crawler_session_ID* is an ID that was assigned to the crawler session by the enterprise search system. For example:

```
ES_NODE_ROOT/master_config/col_56092.WEB_88534
```

4. Create or edit a file named followindex.rules.
5. Type rules for the crawler in the following format, where *URLprefix* is the beginning characters for the Web sites that you want to allow or forbid to be followed or indexed:

```
forbid follow URLprefix
allow follow URLprefix
forbid index URLprefix
allow index URLprefix
```

6. Save, then exit the file.
7. From the enterprise search administration console, restart the Web crawler that you stopped.

WebSphere Portal crawlers

To include pages from an IBM WebSphere Portal site in an enterprise search collection, you must configure a WebSphere Portal crawler.

WebSphere Portal server configuration

Before you create a WebSphere Portal crawler, you must deploy an enterprise application, ESPACServer.ear, in WebSphere Portal. This enterprise application is installed on the search servers when WebSphere Information Integrator OmniFind Edition is installed. To deploy this enterprise application, complete the following task:

- “Deploying the enterprise application for the WebSphere Portal crawler” on page 84.

Configuration overview

You can use the WebSphere Portal crawler to crawl a single WebSphere Portal site. When you configure the crawler, you specify the URL for the portal site to be crawled. The crawler then downloads the portlets that are available on the specified portal.

To create or change a WebSphere Portal crawler, log in to the enterprise search administration console. You must be a member of the enterprise search administrator role or be a collection administrator for the collection that owns the crawler.

When you create the crawler, a wizard helps you do these tasks:

- Specify properties that control how the crawler operates and uses system resources. The crawler properties control how the crawler crawls all pages on the site.
- Specify the URL for the portal site to be crawled and information that enables the crawler to connect to the site. Because these types of URLs can be long and include encoded non-ASCII characters, you might want to copy the URL from the WebSphere Portal server and paste it in the enterprise search administration console.
- Configure document-level security options. If security was enabled when the collection was created, the crawler can associate security data with documents in the index. This data enables search applications to enforce access controls based on the stored access control lists or security tokens.

You can also select an option to validate user credentials at the time a user submits a query. In this case, instead of comparing the user's credentials to indexed security data, the system compares the credentials to current access control lists that are maintained by the original data source.

Click **Help** while you are creating the crawler to learn about the fields in the wizard and how to provide the crawler with the information that the crawler needs to crawl data.

Deploying the enterprise application for the WebSphere Portal crawler

Before you create a WebSphere Portal crawler, you must deploy an enterprise application, `ESPACServer.ear`, in WebSphere Portal.

About this task

The `ESPACServer.ear` file is installed in the `ES_INSTALL_ROOT/bin` directory on the search servers when WebSphere II OmniFind Edition is installed. The default installation paths are as follows:

UNIX systems:

`/opt/IBM/es/bin/ESPACServer.ear`

Windows systems:

`C:\Program Files\IBM\es\bin\ESPACServer.ear`

Procedure

To deploy the enterprise application that enables the WebSphere Portal crawler to crawl WebSphere Portal sites:

1. Stop the WebSphere_Portal server instance.
2. If it is not already started, start the WebSphere Application Server server1 server instance.
3. On the WebSphere Portal server, start the WebSphere Application Server Administrative Console. If you are prompted to log in, log in.
You can open the Administrative Console in the following ways:
 - Use the Windows **Start** menu to select the program.
 - For WebSphere Application Server version 5, open a Web browser and go to `http://hostname:port/admin`, where *hostname* is the host name of the search server and *port* is the port number for the WebSphere Application Server Administrative Console. Typically, the Administrative Console port is 9090.
 - For WebSphere Application Server version 6, open a Web browser and go to `http://hostname:port/ibm/console`, where *hostname* is the host name of the search server and *port* is the port number for the WebSphere Application Server Administrative Console. Typically, the Administrative Console port is 9060.
4. Click **Applications** and then click **Install new application**.
5. Click **Browse** and select the ESPACServer.ear file from your system.
6. Click **Next** twice. If you receive a warning about policy files, click **Continue**.
7. Click **Next** until you see the Map modules to application servers page, make the following selections, and click **Apply**:
 - a. In the **Clusters and Servers** text box, select **WebSphere:cell=cell_name, node=node_name, server=WebSphere Portal**.
 - b. Select the check box next to the **ESPACServer.ear** module.
8. Continue clicking **Next** until you see the Summary page, then click **Finish**.
9. Click the **Save to Master Configuration** link, then click the **Save** button to save your changes to the WebSphere Application Server configuration.
10. Restart the WebSphere Portal server.

Related concepts

“Enterprise search integration with WebSphere Portal” on page 199
You can expand the search capabilities of IBM WebSphere Portal by deploying enterprise search portlets in WebSphere Portal and the WebSphere Portal Search Center.

Copying the URL to crawl from WebSphere Portal

To reduce the possibility of typing an incorrect URL, you can copy and paste the URL of the WebSphere Portal site that you want to crawl into the appropriate field when you configure a WebSphere Portal crawler.

About this task

When you create a WebSphere Portal crawler, you specify the URL of the portal on the WebSphere Portal server that you want to crawl. Because the URLs are long and usually contain encoded non-ASCII characters, you might want to use this procedure to copy the URL from the WebSphere Portal server and paste it into the enterprise search administration console.

Procedure

To specify the URL that you want the WebSphere Portal crawler to crawl:

1. When you see the WebSphere Portal Server to Crawl page in the enterprise search administration console, ensure that WebSphere Portal server is started, and then log in to WebSphere Portal as an administrator.
2. Select **Administration** in the upper right corner.
3. Select **Portal Settings** in the navigation area on the left, and then select **Search Administration**.
4. On the Manage Search Collections page, select **PortalCollection** in the Search Collections area. (You can select another collection, if other collections are available.)
5. In the Content Sources in the Collection area, click **Add Content Source**.
6. For **Crawl source type**, select **Portal site**. The site URL is displayed in the **Collect documents linked from this URL** field.
7. Copy the URL to the clipboard. (For example, highlight the URL and then hold the Ctrl key while you press the Insert key.)
8. Return to the enterprise search administration console and paste the URL that you copied into the **WebSphere Portal site URL** field.

Windows file system crawlers

To include documents that are stored in Microsoft Windows file systems in an enterprise search collection, you must configure a Windows file system crawler.

You can use the Windows file system crawler to crawl any number of Windows file systems. When you configure the crawler, you select the local and remote directories and subdirectories that you want to crawl.

If you install the crawler server on a UNIX computer, you cannot use that server to crawl Windows file system sources (the Windows file system crawler does not appear in the list of available crawler types).

To create or change a Windows file system crawler, log in to the enterprise search administration console. You must be a member of the enterprise search administrator role or be a collection administrator for the collection that owns the crawler.

When you create the crawler, a wizard helps you do these tasks:

- Specify properties that control how the crawler operates and uses system resources. The crawler properties control how the crawler crawls all subdirectories in the crawl space.
- Set up a schedule for crawling the file systems.
- Select subdirectories to crawl.

You can specify how many levels of subdirectories that you want the crawler to crawl. To crawl remote file systems, you also specify a user ID and password that enables the crawler to access data.

- Specify options for making documents in subdirectories searchable. For example, you can exclude certain types of documents from the crawl space or specify a user ID and password that enables the crawler to access files in a particular subdirectory.
- Configure document-level security options. If security was enabled when the collection was created, the crawler can associate security data with documents in the index. This data enables search applications to enforce access controls based on the stored access control lists or security tokens.

You can also select an option to validate user credentials at the time a user submits a query. In this case, instead of comparing the user's credentials to indexed security data, the system compares the credentials to current access control lists that are maintained by the original data source.

To enforce document-level security, you must ensure that user and domain account information is configured correctly on the crawler server.

Click **Help** while you are creating the crawler to learn about the fields in the wizard and how to provide the crawler with the information that the crawler needs to crawl data.

Related concepts

"Enforcement of document-level security for Windows file system documents" on page 193

To enable current credentials to be validated when a user searches documents that were crawled by a Windows file system crawler, you must configure domain account information on both the crawler server and Microsoft Windows server.

Configuring support for Data Listener applications

You can extend enterprise search by using the Data Listener API to create an external crawler. Your custom Data Listener applications can add data to a collection, remove data from a collection, or instruct a Web crawler to visit and revisit URLs.

Before you begin

To configure Data Listener applications, you must be a member of the enterprise search administrator role.

About this task


A client Data Listener application enables the crawling of data source types that cannot be crawled by the default crawlers for enterprise search. Before you can use a Data Listener application, you must configure credentials that enable the application to access and update collections.



When your client Data Listener application connects to the Data Listener, it must pass in the client application ID and password and the ID of the collection to be updated. This information must match the information that you configure for the application in the administration console.

The Data Listener is started automatically when the enterprise search system is started. If you change the port number after you configure the application in the administration console, you must restart the Data Listener.

Procedure

To configure Data Listener applications:

1. Click **System** to open the System view.
2. Click  **Edit** to change to the system editing view.
3. On the Data Listener page, click **Configure Data Listener applications**.

4. On the Data Listener Applications page, specify the number of threads that the Data Listener can create for processing requests from client applications and the port number where the Data Listener listens for requests.
5. Click **Add Data Listener Application** to add information about a client application.
6. On the Add Data Listener Application page, specify authentication information that enables your client Data Listener applications to access enterprise search collections. The Data Listener client IDs must be unique within the enterprise search system.
7. Select the collections that the Data Listener application can update:
 - Click **All collections** if you want the application to update all collections.
 - Click **Specific collections** if you want the application to update only the collections that you specify.When you select this option, a list of collection names is displayed. Select the **Select** check box for each collection that the application can update.
8. Click **OK**.
9. If you changed the Data Listener port number, restart the Data Listener:
 - a. Click  **Monitor** to change to the system monitoring view.
 - b. On the Data Listener page, click  **Restart**.

Related concepts

"Data listener" in "Programming Guide and API Reference for Enterprise Search"

Related tasks

"Monitoring the Data Listener" on page 233

Monitor the Data Listener to see its status and to view details about client Data Listener application activity.

Custom crawler plug-ins

When you configure properties for crawlers, you can specify a Java class to use to enforce document-level access controls and to associate metadata with documents in an enterprise search index.

A plug-in contains a Java class that is called for each document that the crawler crawls. The Java class is passed the document identifier (URI) from the enterprise search index, security tokens, and metadata. The class returns a new or modified set of security tokens and metadata, or the class can indicate that a document is to be ignored by the crawler.

After all of the documents in the crawl space are crawled once, the plug-in is called only for new or modified documents. To change the security tokens and metadata for documents that are in the enterprise search index, but that were not updated in the original data source, start a full crawl of all documents in the crawl space and then reorganize the index.

You cannot associate a plug-in with an existing crawler. You must specify the plug-in class name and class path when you configure properties for a new crawler.

Using a plug-in to enforce security

Document-level security is enforced by associating one or more security tokens (a comma-delimited string) with each document that a crawler crawls. Group identifiers are commonly used as the security tokens.

By default, each document is assigned a public token that makes the document available to everyone. The public token can be replaced with a value that is provided by the administrator or a value that is extracted from a field in the crawled document.

The plug-in allows you to apply your own business rules to determine the value of the security tokens for crawled documents. The security tokens that are associated with each document are stored in the index. They are used to filter documents that match the security tokens and ensure that only the documents that a user has permission to view are returned in the search results.

Using a plug-in to add metadata to documents

Document metadata, such as the date that a document was last modified, is created for all crawled documents. The crawler plug-in allows you to apply your own business rules to determine the value of the metadata that is to be indexed for each document.

The metadata is created as a name-value pair. Users can search the metadata with a free-text query or with a query that specifies the metadata field name.

Web crawler plug-in

With the application programming interfaces for the Web crawler, you can control how documents are crawled and how they are prepared for parsing. For example, you can add fields to the HTTP request header that will be used when the crawler requests a document. After a document is crawled, and before it is parsed and tokenized, you can change the content, security tokens, and metadata. You can also stop the document from being sent to the parser.

Related concepts

"Crawler plug-ins" in "Programming Guide and API Reference for Enterprise Search"

URI formats in an enterprise search index

The uniform resource identifier (URI) of each document in an enterprise search index indicates the type of crawler that added the document to the collection.

You can specify URIs or URI patterns when you configure categories, scopes, and quick links for a collection. You also specify the URI when you need to remove documents from the index, or to view detailed status information about a specific URI.

Search the collection to determine the URIs or URI patterns for a document. You can click the URIs in the search results to retrieve documents that you are interested in. You can copy the URI from the search results to use the URI in the enterprise search administration console. For example, you can specify a URI pattern to automatically associate documents that match that URI pattern with an enterprise search category.

Content Edition crawlers

The URI format for documents that are crawled by a Content Edition crawler in server access mode is:

```
vbr://Server_Name/Repository_System_ID/Repository_Persistent_ID  
      /Item_ID/Version_ID  
      /Item_Type/?[Page=Page_Number&] JNDI_properties
```

The URI format for documents that are crawled by a Content Edition crawler in direct access mode is:

```
vbr:///Repository_System_ID/Repository_Persistent_ID  
      /Item_ID/Version_ID  
      /Item_Type/[?Page=Page_Number]
```

Parameters

URL encoding is applied to all of the fields.

Server_Name

The name of the WebSphere Information Integrator Content Edition server.

Repository_System_ID

The system ID for the repository.

Repository_Persistent_ID

The persistent ID for the repository.

Item_ID

The ID for the item.

Version_ID

The ID for the version. If the version ID is blank, this value indicates the latest version of the document.

Item_Type

The type of the item (CONTENT or FOLDER).

Page_Number

The page number.

JNDI_properties

The JNDI properties for the J2EE application client. There are two types of properties:

java.naming.factory.initial

The name of the class for the application server that is used to create the EJB handle.

java.naming.provider.url

The URL to the naming service for the application server that is used to request the EJB handle.

Examples

Documentum:

```
vbr://vbrsrv.ibm.com/Documentum/c06b/094e827780000302//CONTENT/?  
java.naming.provider.url=iiop%3A%2F%2Fmyvbr.ibm.com%3A2809&  
java.naming.factory.initial=com.ibm.websphere.naming.WsnInitContextFactory
```

FileNet PanagonCS:

```
vbr://vbrsrv.ibm.com/PanagonCS/4a4c/003671066//CONTENT/?Page=1&  
java.naming.provider.url=iiop%3A%2F%2Fmyvbr.ibm.com%3A2809&  
java.naming.factory.initial=com.ibm.websphere.naming.WsnInitContextFactory
```

DB2 crawlers

The URI format for documents that are crawled by a DB2 crawler is:

```
db2://Database_Name/Table_Name
      /Unique_Identifier_Column_Name1/Unique_Identifier_Value1
      [/Unique_Identifier_Column_Name2/Unique_Identifier_Value2/...
      /Unique_Identifier_Column_NameN/Unique_Identifier_ValueN]
```

Parameters:

URL encoding is applied to all of the fields.

Database_Name

The internal name of the database or the alias for the database.

Table_Name

The name of the target table, including the name of the schema.

Unique_Identifier_Column_Name1

The name of the first Unique Identifier column in the table.

Unique_Identifier_Value1

The value of the first Unique Identifier column.

Unique_Identifier_Column_NameN

The name of the *n*th Unique Identifier column in the table.

Unique_Identifier_ValueN

The value of the *n*th Unique Identifier column.

Examples

Local, cataloged database:

```
db2://LOCALDB/SCHEMA1.TABLE1/MODEL/ThinkPadA20
```

Remote, uncataloged database:

```
db2://myserver.mycompany.com:50001/REMOTEDB/SCHEMA2.TABLE2/NAME/DAVID
```

DB2 Content Manager crawlers

The URI format for documents that are crawled by a DB2 Content Manager crawler is:

```
cm://Server_Name/Item_Type_Name/PID
```

Parameters

Server_Name

The name of the IBM DB2 Content Manager library server.

Item_Type_Name

The name of the target item type.

PID

The DB2 Content Manager persistent identifier.

Example

```
cm://cmsrvctg/ITEMTYPE1/92+3+ICM8+icmn1sdb12+ITEMTYPE159+26+A1001001A
03F27B94411D1831718+A03F27B+94411D183171+14+1018
```

Domino Document Manager crawlers

The URI format for documents that are crawled by a Domino Document Manager crawler is:

```
dominodoc://Server_Name:Port_Number/Database_Replica_ID/Database_Path_and_Name
            /View_Universal_ID/Document_Universal_ID
            /?AttNo=Attachment_Number&AttName=Attachment_File_Name
```

Parameters

URL encoding is applied to all of the fields.

Server_Name

The name of the Domino Document Manager server.

Port_Number

Optional: The port number for the Domino Document Manager server.

Database_Replica_ID

The identifier for the database replica.

Database_Path_and_Name

The path and file name for the document NSF database on the target Domino Document Manager server.

View_Universal_ID

The View Universal ID that is used to crawl Domino Document Manager documents.

Document_Universal_ID

The Document Universal ID that is defined in the crawled document.

Attachment_Number

Optional: A consecutive number, starting from zero, for each attachment.

Attachment_File_Name

Optional: The original name of the attachment file.

Examples

A Domino Document Manager document:

```
dominodoc://dominodocsvr.ibm.com/49256D3A000A20DE/domdoc%2FADMN-6FAJXL.nsf/8178B1C14B1E9B6B8525624F0062FE9F/0205F44FA3F45A9049256DB20042D226
```

A document attachment:

```
dominodoc://dominodocsvr.ibm.com/49256D3A000A20DE/domdoc%2FADMN-6FAJXL.nsf/8178B1C14B1E9B6B8525624F0062FE9F/0205F44FA3F45A9049256DB20042D226?AttNo=0&AttName=AttachedFile.doc
```

Exchange Server crawlers

The URI format for documents that are crawled by an Exchange Server crawler is:

exchange://*OWA_path*[?useSSL=true]

Parameters

OWA_Path

The Outlook Web Access (OWA) path, without the protocol.

useSSL=true

Added when the protocol of the original OWA path is HTTPS.

Examples

Document body:

```
exchange://exchangesvr.ibm.com/public/RootFolder1/Folder1/Document.EML
```

Document attachment:

```
exchange://exchangesvr.ibm.com/public/RootFolder1/Folder1/Document.EML/AttachedFile.doc
```

Enabled for SSL:

```
exchange://exchangesvr.ibm.com/public/TeamRoom/Folder1/Document.EML  
?useSSL=true
```

Notes crawlers

The URI format for documents that are crawled by a Notes crawler is:

```
domino://Server_Name[:Port_Number]/Database_Replica_ID/Database_Path_and_Name  
/[View_Universal_ID]/Document_Universal_ID  
[?AttNo=Attachment_Number&AttName=Attachment_File_Name]
```

Parameters

URL encoding is applied to all of the fields.

Server_Name

The name of the Lotus Notes server.

Port_Number

The port number for the Lotus Notes server. The port number is optional.

Database_Replica_ID

The identifier for the database replica.

Database_Path_and_Name

The path and file name for the NSF database on the target Lotus Notes server.

View_Universal_ID

The View Universal ID that is defined on the target database. This ID is specified only when the document is selected from a view or folder. If you do not designate a view or folder to crawl (for example if you specify that you want to crawl all documents in a database), the View Universal ID is not specified.

Document_Universal_ID

The Document Universal ID that is defined in the document that is crawled by the crawler.

Attachment_Number

A consecutive number, starting from zero, for each attachment. The attachment number is optional.

Attachment_File_Name

The original name of the attachment file. The attachment file name is optional.

Examples

A document that was selected for crawling by view or folder:

```
domino://dominosvr.ibm.com/49256D3A000A20DE/Database.nsf/  
8178B1C14B1E9B6B8525624F0062FE9F/0205F44FA3F45A9049256DB20042D226
```

A document that was not selected for crawling by view or folder:

```
domino://dominosvr.ibm.com/49256D3A000A20DE/Database.nsf//  
0205F44FA3F45A9049256DB20042D226
```

A document attachment:

```
domino://dominosvr.ibm.com/49256D3A000A20DE/Database.nsf//  
0205F44FA3F45A9049256DB20042D226?AttNo=0&AttName=AttachedFile.doc
```

QuickPlace crawlers

The URI format for documents that are crawled by a QuickPlace crawler is:
`quickplace://Server_Name:Port_Number/Database_Replica_ID/Database_Path_and_Name
/View_Universal_ID/Document_Universal_ID
/?AttNo=Attachment_Number&AttName=Attachment_File_Name`

Parameters

URL encoding is applied to all of the fields.

Server_Name

The name of the Lotus QuickPlace server.

Port_Number

Optional: The port number for the QuickPlace server.

Database_Replica_ID

The identifier for the database replica.

Database_Path_and_Name

The path and file name for the document NSF database on the target QuickPlace server.

View_Universal_ID

The View Universal ID that is used to crawl QuickPlace documents.

Document_Universal_ID

The Document Universal ID that is defined in the crawled document.

Attachment_Number

Optional: A consecutive number, starting from zero, for each attachment.

Attachment_File_Name

Optional: The original name of the attachment file.

Examples

A document:

```
quickplace://1twsvr.ibm.com/49257043000214B3/QuickPlace%5Csampleplace  
%5CPageLibrary4925704300021490.nsf  
/A7986FD2A9CD47090525670800167225  
/2B02B1DE3A82B2CE49257043001C2498
```

A page attachment:

```
quickplace://1twsvr.ibm.com/49257043000214B3/QuickPlace%5Csampleplace  
%5CPageLibrary4925704300021490.nsf  
/A7986FD2A9CD47090525670800167225  
/2B02B1DE3A82B2CE49257043001C2498  
?AttNo=0&AttName==QPCons3.ppt
```

UNIX file system crawlers

The URI format for documents that are crawled by a UNIX file system crawler is:
`file:///Directory_Name/File_Name`

Parameters

URL encoding is applied to all of the fields.

Directory_Name

The absolute path name for the directory.

File_Name
The name of the file.

Example

file:///home/user/test.doc

WebSphere Portal crawlers

The URI format for documents that are crawled by a WebSphere Portal crawler is:

wps://WPS_Access_Path?portletDefID=Portlet_Def_ID&pageID=Page_ID&useSSL=SSL

Parameters

URL encoding is applied to all of the fields.

WPS_Access_Path
The WebSphere Portal server path, without the protocol.

Portlet_Def_ID
The portlet definition identifier for the WebSphere Portal server.

Page_ID
The page identifier for the WebSphere Portal server.

SSL useSSL=true is added when the protocol of the path is HTTPS. Otherwise, useSSL=false is added.

Examples

Document body:

wps://wpsserver.ibm.com:9081/wps/myportal!/ut/p/kcxml/04_Sj9SPykssy0x+LKnPy1vM0Y_QjzKCN4g3cQbJgQi0-pFQAW99X4_83FT9AP2C5IhyR0dFRQD8qHRj/delta/base64xml/L01DU1kvd0NrQUpORUEvNFBVR0VoQSEvN18wXzZPLzZfMF80RA!!?portletDefID=3_0_3S&pageID=6_0_6J&useSSL=false

Examples

Enabled for SSL:

wps://wpsserver.ibm.com:9081/wps/myportal!/ut/p/kcxml/04_Sj9SPykssy0x+LKnPy1vM0Y_QjzKCN4g3cQbJgQi0-pFQAW99X4_83FT9AP2C5IhyR0dFRQD8qHRj/delta/base64xml/L01DU1kvd0NrQUpORUEvNFBVR0VoQSEvN18wXzZPLzZfMF80RA!!?portletDefID=7_0_A4&pageID=6_0_6J&useSSL=true

Windows file system crawlers

The URI formats for documents that are crawled by a Windows file system crawler are:

file:///Directory_Name/File_Name
file:///Network_Folder_Name/Directory_Name/File_Name

Parameters

URL encoding is applied to all of the fields.

Directory_Name
The absolute path name for the directory.

File_Name
The name of the file.

Network_Folder_Name
For documents on remote servers only, the name of the shared folder on a Windows network.

Examples

Local file system:

file:///d:/directory/test.doc

Network file system:

file:///filesvr.ibm.com/directory/file.doc

Related concepts

“Enterprise search crawler administration” on page 33

You configure crawlers for the different types of data that you want to include in a collection. A single collection can contain any number of crawlers.

Related tasks

“Configuring categories” on page 103

You can create any number of categories for a collection, and each category can contain any number of rules. The rules determine which documents are associated automatically with the category.

“Configuring scopes” on page 135

When you configure a scope for an enterprise search collection, you specify the URIs, or URI patterns, for a range of documents in the index that users are allowed to search.

“Removing URIs from the index” on page 137

To prevent users from searching documents in a collection, you can remove the URIs for those documents from the index.

“Configuring quick links” on page 147

To create a quick link for an enterprise search collection, you associate the URI of a document with the keywords that trigger its inclusion in the search results.

“Viewing details about a URI” on page 220

You can view detailed information about a URI. You can see current and historical information about how the document that is represented by this URI is crawled, indexed, and searched.

“Viewing reports about dropped documents” on page 235

You can view detailed information about documents that are dropped from an enterprise search system. This information is available only if you enabled document tracking for the collection.

Enterprise search parser administration

To enhance the retrievability of documents, you can specify options for how documents and metadata are to be parsed, analyzed, and categorized before they are added to the enterprise search index.

The options that you can specify for parsing document content and optimizing the retrievability of information include the following:

Configuring options for parsing Chinese, Japanese, and Korean documents

You can specify options for using n-gram segmentation to parse documents that are written in the Chinese, Japanese, and Korean languages. You can also remove new line characters from the white space in Chinese and Japanese documents.

Enabling native XML search

If your collection includes XML documents, you can enable them to be searched with native XML query syntax, such as XPath and XML fragments. A native XML search enables users to specify queries based on the relationships between various XML elements.

Configuring categories

You can group documents that share a similar URI pattern or that contain specific words into categories. When users search the collection, they can limit the search results to documents that belong to specific categories.

Configuring search fields

You can map elements in XML documents to search fields in the index. You can also map metadata elements in HTML documents to search fields. By creating search fields in the enterprise search index, you enable users to query specific parts of XML and HTML documents and improve the precision of the search results.

Configuring text processing options

If custom text analysis engines were added to the enterprise search system, you can select one to use with a collection. After you associate a analysis engine with a collection, you can specify options for mapping content so that it can be linguistically analyzed and annotated. You can also specify how the results of the analysis are to be mapped to the enterprise search index or to JDBC database tables.

Mapping fields to boost classes

You can specify that documents with fields that match the query terms are to be ranked higher in the search results than other documents that match the query terms. When you map fields to boost classes, you specify which content and metadata fields are to be boosted. You can also configure the scores that each boost class uses to rank documents.

Related concepts

"Linguistic support for semantic search" in "Text Analysis Integration"

"Text analysis included in enterprise search" in "Text Analysis Integration"

"Unstructured information management architecture (UIMA)" in "Text Analysis Integration"

"Semantic search applications" in "Text Analysis Integration"

"Semantic search query" in "Text Analysis Integration"

Working with categories

Categories enable you to group documents that share common characteristics, and search and retrieve only documents that meet the criteria for being members of that group.

If you associate documents with categories, and your search applications support this capability, users can search a subset of the collection by specifying the category name. If they search the entire collection, users can refine the search results and browse only the documents that are in the same category as one of the result documents.

Enterprise search supports two approaches for categorizing documents:

Rule-based

Use this approach if you want to associate documents with categories according to rules that you specify. For example, you can group documents that share a URI pattern or group documents based on document content (for example, documents that contain or exclude specific words and phrases).

Model-based

Use this approach if you use IBM WebSphere Portal and you want to use the predefined categories in WebSphere Portal to search enterprise search collections.

Applying categorization changes

To create and administer categories, you use the enterprise search administration console:

- You select the categorization type when you create a collection. You can choose to use no categories, use rule-based categories, or use model-based categories.
- When you configure parsing rules for the collection, you can change the categorization type, if necessary. If you change the categorization type after documents are crawled and indexed, search quality will be degraded until you recrawl all documents, reparse the documents, and reorganize the index.
- If you choose to use rule-based categories, you use the administration console to administer the category tree, categories, and category rules. If you change categories or category rules after documents are crawled and indexed, search quality will be degraded until you recrawl all documents, reparse the documents, and reorganize the index.

Document content fields

To create a collection with model-based categories or with rule-based categories that use document content rules (as opposed to URI pattern rules), ensure that the documents to be categorized contain content fields.

Model-based categories and category rules that are based on document content operate on the content fields of documents and do not take document metadata into account. Only documents that contain content fields, or that contain fields that can be configured to be content fields when the crawler is configured, can be categorized with these categorization options.

If you configure model-based categories or create category rules that are based on document content, the rules apply only to the content parts of a document. For the

following crawler types, the rules apply to files (such as the content of UNIX or Microsoft Windows files) or to the content of Web pages:

- Exchange Server
- NNTP
- UNIX file system
- Web
- WebSphere Portal
- Windows file system

For the following crawler types, you specify which fields are document content fields when you configure options for individual data sources. When you create a crawler, this option is available on the final page in the crawler wizard. If the crawler already exists, edit the crawl space, select a data source to edit options for, then specify which fields are document content fields in that data source.

- Content Edition
- DB2 Content Manager
- DB2
- Domino Document Manager
- Notes
- QuickPlace

Rule-based categories

You can configure rules to control which documents are associated with categories in an enterprise search collection.

You can create categories and category rules for collections that you create in enterprise search and for rule-based categories that you migrate from IBM WebSphere Portal collections. To configure rules for categorizing documents, you must specify that you want to use rule-based categories when you create the collection or when you specify parsing rules for the collection.

The parser uses the rules that you specify to associate documents with one or more categories:

- If a document passes at least one rule in a category, the parser associates the document with the category.
- If a document passes at least one rule in several categories, the parser associates the document with all of the categories.
- If a document does not pass any of the rules for a category, the parser does not associate the document with a category. Users can search for this document and retrieve it when they search the collection, but they cannot search a category and expect to retrieve the document.

When you administer the category tree (or taxonomy) for a collection, you decide where in the hierarchy of categories that you want to add a new category. You also use the category tree to select a category that you want to edit, and then add rules for categorizing documents, delete rules, or change the content of individual rules.

When you configure a rule for categorizing documents, you choose whether enterprise search is to use the URI of a document or content in the document to determine whether the document belongs to the category:

URI pattern

A URI rule applies to the document's URI. You specify a partial URI (a pattern), and documents that have the specified pattern in their URIs pass the rule.

For example, if you specify that the rule text is `/hr/`, then the first URI below passes the rule, and the second URI does not:

```
file:///corporate/hr/medicalform.doc
http://company.com/human_resources/medicalform.htm
```

Because all URIs are treated as patterns, the system ignores any asterisks that you specify as wildcard characters at the start or end of the pattern. For example, `*/hr/*` and `/hr/` match the same set of URIs.

URI pattern rules are not case sensitive. If a URI contains spaces, the URI pattern must adhere to the enterprise search rules for encoding URIs. The following example shows correct and incorrect ways to specify a URI for a Windows file system path:

```
Incorrect URI: file:///c:/program files/
Correct URI: file:///c:/program+files/
```

Document content

You express document content rules in the same format as a query. If the document is valid for the query, it passes the rule. When you configure the rule, you specify the words and phrases that documents must contain or exclude, and you choose a language for applying word stemming rules.

For example, the following rule specifies that if a document contains either the word `hr` or the phrase `human resources`, the document passes the rule:

```
hr "human resources"
```

For another example, the following rule specifies that if a document contains the word `hr` but not the word `benefits`, the document passes the rule:

```
+hr -benefits
```

Content rules undergo the same linguistic normalizations as Search and Index API (SI-API) queries. However, the syntax for content rules supports a subset of the operations available in the SI-API query syntax. Only the following query operators are allowed:

- + Precede a term with a plus sign to indicate that the term must occur in the document.
- Precede a term with a minus sign to indicate that the term must not occur in the document.
- " Enclose two or more terms in quotation marks to indicate that the exact phrase must occur in the document.

Document content rules apply only to the content parts of a document. For the following crawler types, the rules apply to files (such as the content of UNIX or Microsoft Windows files) or to the content of Web pages:

- Exchange Server
- NNTP
- UNIX file system
- Web

- WebSphere Portal
- Windows file system

For the following crawler types, you specify which fields are document content fields when you configure options for individual data sources. When you create a crawler, this option is available on the final page in the crawler wizard. If the crawler already exists, edit the crawl space, select a data source to edit options for, then specify which fields are document content fields in that data source.

- Content Edition
- DB2 Content Manager
- DB2
- Domino Document Manager
- Notes
- QuickPlace

Related tasks

“Migrating a collection from WebSphere Portal” on page 209

To migrate collections from WebSphere Portal to enterprise search, prepare the collections in WebSphere Portal, then use the migration wizard to migrate them.

Model-based categories

If you use model-based categories in your IBM WebSphere Portal system, you can continue to use those categories with enterprise search collections.

WebSphere Portal provides a predefined taxonomy that includes over 2,300 subjects. These subjects are grouped into main business category areas, such as Computers, Finance, and Transportation. Portal users can create applications that automatically determine which documents match these subject areas, and they can customize the categories for their own business needs.

If you want to use the WebSphere Portal categories with enterprise search, you must:

- Use the migration wizard to import model-based taxonomy files to enterprise search.
- Specify that you want to use model-based categories when you create a collection or when you configure parsing rules for a collection.
- Ensure that WebSphere Portal is installed on the enterprise search index server.
- Use the categorization tools in WebSphere Portal to administer the categories. You cannot administer model-based categories with the enterprise search administration console.

Related tasks

“Migrating a model-based taxonomy from WebSphere Portal” on page 207

You can select which model-based taxonomy you want to use with an enterprise search collection by using the WebSphere Portal Taxonomy Management Portlet. Collections that you already migrated to enterprise search are not affected by a new taxonomy migration.

Category trees

A category tree enables you to view all of the rule-based categories in a collection. You use the category tree to create categories, delete categories, and edit the rules that associate documents with categories.

A category tree, which is also called a taxonomy, is arranged in a hierarchy. The tree starts with the root category, and all other categories stem from the root category. You can nest any number of categories and subcategories to provide users with different choices for browsing and retrieving documents.

For example, if a document passes the rules in several categories, it is associated with all of those categories. When users search a category, or browse documents that belong to a category when they browse search results, the fact that a document belongs to multiple categories enhances the likelihood that users will find it.

When you administer the category tree, you can control which documents belong to one or more categories by nesting new categories under existing categories. When you create a category, you specify whether it is to be created at the root level or as a subcategory of another category. You also use the category tree to delete categories from the collection and to change the rules for associating documents with categories. When you edit a category, you can rename the category, add or delete categorization rules, or modify the content of individual rules.

When you administer the category tree, use the following descriptions of search and browse behavior as a guideline:

- If a user searches a high-level category, that category and all of its subcategories are searched for documents that match the search criteria. If a user searches a category that has no additional subcategories, only that category is searched.
- If a user is browsing search results and selects an option to browse documents that belong to a specific category, only the documents in that category are displayed. The names of any subcategories are also displayed in the search results, so that the user can navigate between categories and view subsets of documents at a time.

Related tasks

“Migrating a collection from WebSphere Portal” on page 209

To migrate collections from WebSphere Portal to enterprise search, prepare the collections in WebSphere Portal, then use the migration wizard to migrate them.

Selecting the categorization type

When you select a categorization type, you specify the approach that you want to use to associate documents with categories in the collection.

Before you begin

To change the categorization type, you must be a member of the enterprise search administrator role or be a collection administrator for the collection that you are changing.

About this task

The categorization type is specified when the collection is created. If necessary, you can change how documents are categorized for a collection. You can use rule-based categories that you configure specifically for a collection, use model-based categories that exist in your IBM WebSphere Portal system, or use no categories.

Important: If you change the categorization type after you crawl data and create an index for a collection, the index becomes inconsistent. To ensure the

accuracy of search results, recrawl all documents in the collection and, after the documents are parsed, reorganize the index.

Procedure

To select the categorization type:

1. Edit a collection, select the Parse page, and click **Select a categorization type**.
2. On the Select a Categorization Type page, select one of the following options:

None Select this option if you do not want to categorize documents in this collection.

Rule-based

Select this option if you want to use a taxonomy that contains category rules that you configure specifically for this collection.

- If you are configuring a collection that you created for enterprise search, select this option to specify category names and rules for categorizing documents.
- If you are configuring a collection that you migrated from WebSphere Portal, select this option to use or change the rule-based categories that you imported.

Model-based

Select this option if you want to associate documents with model-based categories that exist in a WebSphere Portal system. To use this option, WebSphere Portal must be installed on the enterprise search index server. You must also use the categorization tools in WebSphere Portal to administer the categories.

3. Click **OK**.

Configuring categories

You can create any number of categories for a collection, and each category can contain any number of rules. The rules determine which documents are associated automatically with the category.

Before you begin

To configure categories, you must be a member of the enterprise search administrator role or be a collection administrator for the collection that the categories belong to.

The option to use rule-based categories must be selected as the categorization type.

For examples of how to specify rules for associating documents with categories, click **Help** while you are creating or editing a category.

About this task

If your search applications enable support for categories, users can search a subset of the collection by specifying the category name. Users can also select a category in the search results and browse only the documents that belong to the selected category.

Important: If you change categories or category rules after you crawl data and create an index for a collection, the index becomes inconsistent. To

ensure the accuracy of search results, recrawl all documents in the collection, reparse the documents, and reorganize the index.

Procedure

To configure a category:

1. Edit a collection, select the Parse page, and click **Configure the category tree**.
2. On the Category Tree page, select the location in the tree where you want to add a category and click **Create a category**.

If you select the root, the new category is created at the root level. If you select a category name, the new category is nested below the selected category in the category tree.

A wizard opens to help you specify rules for associating documents with the new category:

- a. On the Create a Category page, type a descriptive name for the category, then click **Next**.
- b. On the Create Category Rules page, click **Add Rule**.
- c. On the Create a Category Rule page, type a unique name for the rule in the **Rule name** field. This name must be unique across all categories in the collection.
- d. Specify the rule that you want to use for associating documents with this category, then click **OK**.

- If you want enterprise search to use the URI of a document when determining whether the document belongs to the category, click **URI pattern** and then specify the URI pattern.

If the text that you specify exists in the URI, the document is associated with the category.

For example: `file:///c:/program+files/finance`

- If you want enterprise search to analyze words in the content fields of documents when determining whether the document belongs to the category, click **Document content**, select the language of the documents, and then specify the words that must or must not appear in the document content. You express the rule in the same format as a query (only the +, -, and " " query operators are allowed).

If a document includes or excludes the words that you specify, the document is associated with the category.

For example: `+finance -accounting +"fiscal year"`

- e. Click **Finish**.

Your new category is listed on the Category Tree page with the other categories that belong to this collection.

Related concepts

"Migration from WebSphere Portal to enterprise search" on page 207
Enterprise search provides a migration wizard that you can use to migrate taxonomies and collections from IBM WebSphere Portal to enterprise search.

Related tasks

"Migrating a model-based taxonomy from WebSphere Portal" on page 207
You can select which model-based taxonomy you want to use with an enterprise search collection by using the WebSphere Portal Taxonomy Management Portlet. Collections that you already migrated to enterprise search are not affected by a new taxonomy migration.

Related reference

“URI formats in an enterprise search index” on page 89

The uniform resource identifier (URI) of each document in an enterprise search index indicates the type of crawler that added the document to the collection.

Working with XML search fields

Map XML elements to search fields if you want to enable users to search specific parts of XML documents.

You use the enterprise search administration console to map XML elements to search fields. Linux

XML search fields

XML search fields enable users to query specific parts of XML documents.

XML documents are becoming more common because they contain both semi-structured and unstructured text. The structure of XML is encapsulated and uses a context that is explicitly defined by XML elements that surround the text. For example, an author’s name might appear as follows:

```
<author>John Smith</author>
```

In this context, the text John Smith identifies the author of an XML document.

Enterprise search can associate, or map, the text inside XML elements with search field names. When you configure parsing options for a collection, you specify which XML elements are to be mapped to which search field names. By mapping XML elements to search fields, you enable users to search those elements by specifying the mapped field names in queries. Queries that search specific fields can provide more precise search results than free-text queries that search all document content.

For example, if your collection includes XML documents, and you specify that the title and author elements are to be marked as search fields in the index, users can query these specific elements. A search for `author:Smith` finds XML documents that have Smith in the author element.

Mapping XML elements to search fields

When you map an XML element to a search field, you specify which XML elements users can search by specifying a field name in a query.

Before you begin

To map XML elements to search fields, you must be a member of the enterprise search administrator role or be a collection administrator for the collection that the XML documents belong to.

About this task

When you create an XML field mapping, or add, change, or delete fields in an existing XML field mapping, the change becomes effective after you stop and restart the parser. The new and changed mappings apply to data that is parsed after you restart the parser. The new and changed mappings have no effect on data that is already parsed and indexed.

This task uses the following sample XML document to show how you might map personnel records and enable users to directly query certain elements.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<personnel>
  <personnelrecord>
    <phone>5555</phone>
    <email>joe@us.ibm.com</email>
    <jobroles>Manager, architect
      <jobrole>Managing Search Development Group</jobrole>
      <jobrole>Architecting Search Technology</jobrole>
    </jobroles>
    <location>New York</location>
    <section id="expertise">
      <text>Linguistics</text>
    </section>
  </personnelrecord>
</personnel>
```

Procedure

To map XML elements in this example to search fields:

1. Edit a collection, select the Parse page, and click **Map XML elements to fields**.
2. On the XML Field Mappings page, click **Create XML Mapping**. The Create an XML Field Mapping page opens.
3. In the **XML root element name** field, type the root element name: `personnel`.
Ensure that the name that you specify here exactly matches the root element in the XML documents that you want to search. When parsing and indexing XML documents, enterprise search selects which mapping to use according the root element name.
4. In the **XML mapping name** field, type a name for this set of XML field mapping rules.
After you create a set of XML mapping rules, this name is displayed on the XML Field Mappings page, and you select this name to add, delete, or change the mapping rules.
5. Map the XML element `jobrole` to a search field named `jobrole`:
 - a. In the **Field name** field, type `jobrole`.
 - b. In the **XML element name** field, type `jobrole`.
 - c. To enable users to query the `jobrole` field and view the job roles in search results, select the **Fielded search** and **Search results** check boxes.
6. Map the XML element `jobroles` to the same search field:
 - a. Click **Add field** to add a blank line to the list of field mapping rules.
 - b. In the **Field name** field, type `jobroles`.
 - c. In the **XML element name** field, type `jobrole`.

Tip: The XML element names do not need to match the search field names, and multiple XML elements can map to the same search field.
 - d. To enable users to query the `jobrole` field and view the job roles in search results, select the **Fielded search** and **Search results** check boxes.
7. Map the XML element `section` with the attribute `expertise` to a search field named `expertise`:
 - a. Click **Add field** to add a blank line to the list of field mapping rules.
 - b. In the **Field name** field, type `expertise`.
 - c. In the **Field name** field, type `section`.

- d. In the **XML attribute name** field, type `id`.
 - e. In the **XML attribute value** field, type `expertise`.
 - f. To enable users to query the `expertise` field and view the `expertise` values in search results, select the **Fielded search** and **Search results** check boxes.
8. Click **OK**.

Examples:

To find everyone in an organization who work on search products, specify the following query:

```
jobrole:search
```

To find everyone in an organization who has expertise in linguistics, specify the following query:

```
expertise:linguistics
```

Working with HTML search fields

Map HTML metadata elements to search fields in the index if you want to enable users to search specific metadata sections of HTML documents.

You use the enterprise search administration console to map HTML metadata elements to search fields.

HTML search fields

HTML search fields enable users to query attributes of HTML documents.

Metadata elements in HTML documents are similar to document attributes in that they provide information about the document, how it is formatted, and how it is allowed to be accessed on the Web. For example:

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" /> ]  
<meta name="copyright" content="(C) Copyright IBM Corporation 2005" />  
<meta name="content.owner" content="(C) Copyright IBM Corporation 2005" />  
<meta name="security" content="public" />  
<meta name="abstract" content="This topic describes an IBM product." />  
<meta name="format" content="XHTML" />
```

Enterprise search can associate, or map, the names of HTML metadata elements with search field names. When you configure parsing options for a collection, you specify which HTML metadata elements are to be mapped to which search field names. By mapping HTML metadata elements to search fields, you enable users to find documents with those elements by specifying the search field names in queries. Queries that search specific fields can provide more precise search results than free-text queries that search all document content.

For example, if your collection includes HTML documents, and you specify that the `copyright` and `abstract` metadata elements are to be indexed as search fields, users can query these specific elements. A search for `copyright:IBM` finds HTML documents that have IBM in the `copyright` metadata.

Mapping HTML metadata elements to search fields

When you map an HTML metadata element to a search field, you specify which HTML metadata elements users can search by specifying a field name in a query.

Before you begin

To map HTML metadata elements to search fields, you must be a member of the enterprise search administrator role or be a collection administrator for the collection that the HTML documents belong to.

About this task

When you create an HTML field mapping, or add, change, or delete fields in an existing HTML field mapping, the changes become effective after you stop and restart the parser. The changes apply only to data that is parsed after you restart the parser. To update documents that have already been parsed and indexed, you must crawl and parse the documents again, and then reorganize the index.

Procedure

To map HTML metadata elements to search fields:

1. Edit a collection, select the Parse page, and click **Map HTML metadata to fields**.
2. On the HTML Field Mappings page, click **Add field** to add a blank line to the list of field mapping rules.
3. In the **Field name** field, type a name that you want to associate with the HTML metadata element that you are mapping. Users can specify this field name when they query HTML documents in this collection.
4. In the **HTML metadata element name** field, type the name of the HTML metadata element that you want to map.
5. To enable users to query this field and view the field in the search results, select the **Fielded search** and **Search results** check boxes.
6. If the data type of this field is DECIMAL, DOUBLE, INTEGER, SHORT, TIME, or TIMESTAMP, and you want to enable users to specify parametric queries when searching this field, select the **Parametric search** check box.
7. Click **OK**.

Examples:

Users can now query the mapped field names to find HTML documents with specific metadata. For example, if you mapped an HTML metadata element named `description` to a search field named `abstract`, users might enter a query similar to the following to find documents about Thinkpad computers:

```
abstract:thinkpad
```

Custom text processing

You can improve the quality and precision of search results by integrating custom text processing algorithms with enterprise search collections.

WebSphere Information Integrator OmniFind Edition supports the IBM Unstructured Information Management Architecture (UIMA), which is a framework for creating, discovering, composing, and deploying text analysis functions. Application developers create and test analysis algorithms for the content to be searched, then create a processing engine archive (.pear file) that includes all of the resources required to use the archive for enterprise search. To be

able to search collections with your custom analysis algorithms, you must add the archive (which contains the text analysis engine) to the enterprise search system.

The analysis logic component in a text analysis engine is called an *annotator*. Each annotator performs specific linguistic analysis tasks. A text processing engine can contain any number of annotators, or it can be a composite of several text analysis engines, each of which contain their own custom annotators.

The information produced by the annotators is referred to as the *analysis results*. Analysis results, which correspond to the information that you want to search for, are written to a data structure called a *common analysis structure*.

When you configure text processing options for a collection, you do the following tasks:

- Select the text analysis engine that you want to use for annotating documents in the collection.
- If your collection contains XML documents with meaningful markup, and you want to use this markup in your custom text analysis, you can associate XML mapping files with the collection and map the output of the XML mapping to a common analysis structure.

For example, you can map the content of <addressee> and <customer> elements to Person annotations in the common analysis structure. These annotations can then be accessed by your custom annotators, which might detect additional information (for example, they might detect the gender of the Person). You can also map Person annotations to the enterprise search index, which allows users to search for Persons without having to know the original XML elements.

If you want to allow users to specify the original XML elements in queries, then you do not need to define any XML mappings. Instead, you can configure parsing options and enable native XML search for the collection.

- Map data structures in a common analysis structure to the enterprise search index, which enables the annotated documents to be searched with semantic search.

For example, depending on the entities and relationships that are detected by the annotators, users can search for concepts that occur in the same sentence (such as a specific person and any competitor name), or a keyword and a concept (such the term Alex and a phone number).

- Map data structures in a common analysis structure to database tables that are Java Database Connectivity (JDBC) capable. You can map data to IBM DB2 Universal Database (DB2 UDB) or Oracle tables. This type of mapping enables the results of analysis to be used in database applications such as data mining. It also enables you to use SQL queries to search the data outside of enterprise search.

Related concepts

"Custom text analysis integration" in "Text Analysis Integration"

"Unstructured information management architecture (UIMA)" in "Text Analysis Integration"

"Workflow for custom analysis integration" in "Text Analysis Integration"

"Text analysis algorithms" in "Text Analysis Integration"

"Semantic search applications" in "Text Analysis Integration"

"Semantic search query" in "Text Analysis Integration"

Adding text analysis engines to the system

If you create a custom text analysis engine, you must add it to the system before you can use it for enterprise search. Collections can use the engine to analyze and annotate documents and improve the precision of search results.

Before you begin

To add text analysis engines to the system, you must be a member of the enterprise search administrator role.

About this task


Application developers can create a processing engine archive (.pear) that adheres to the UIMA framework for text analysis. The archive includes all of the resources required to search enterprise search collections. To be able to search collections with your custom analysis algorithms, you must add the archive (which contains the text analysis engine) to the enterprise search system.

After you add a text analysis engine to the system, you can change its display name and select an option to view the XML source. (The XML source shows you what information is produced by this engine.)

If a text analysis engine is associated with a collection, you cannot remove the text analysis engine from the system.

Procedure

To add a custom text analysis engine to the enterprise search system:

1. Click **System** to open the System view.
2. Click  **Edit** to change to the system editing view.
3. On the Parse page, click **Configure text analysis engines**.
4. On the Text Analysis Engines page, click **Add Text Analysis Engine**.
5. On the Add a Text Analysis Engine page, type a descriptive name for the new engine. The system uses this display name to identify the text analysis engine throughout the administration console.
6. Specify the location of the .pear file. If the file is smaller than 8 MB, the file can be on your system. If the file is larger than 8 MB, the file must be on the index server.
7. Click **OK**. Your text analysis engine is listed on the Text Analysis Engines page.

Related concepts

"Workflow for custom analysis integration" in "Text Analysis Integration"

"Custom text analysis integration" in "Text Analysis Integration"

"Unstructured information management architecture (UIMA)" in "Text Analysis Integration"

"Approaches for using XML markup in analysis and search" in "Text Analysis Integration"

Related tasks

"Creating an XML to UIMA types mapping configuration file" in "Text Analysis Integration"

Associating a text analysis engine with a collection

If custom text analysis engines are associated with the enterprise search system, you can select one to use with a collection. Users can then specify semantic queries when searching the collection, and improve the quality and precision of the search results.

Before you begin

To associate a text analysis engine with a collection, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

About this task

If a text analysis engine is already associated with this collection, the following actions occur when you associate a different engine:

- If you select **No custom analysis**, then all text analysis mappings that you previously defined for the collection are reset. The collection begins using the system default values.
- If you select the name of a different custom text analysis engine, then all text analysis mappings that you previously defined for the collection are retained. For example, if you change from engine_1 to engine_2, then engine_2 inherits the XML mapping files that you configured for engine_1.

Procedure

To associate a text analysis engine with a collection:

1. Edit a collection, select the Parse page, and click **Configure text processing options**.
2. Click **Select a text analysis engine**. If no custom text analysis engines were added to the enterprise search system, or if the collection uses the default analysis algorithms, the engine name is **Default**.
3. On the Select a Text Analysis Engine for this Collection page, select the name of the engine that you want to use with this collection. If no text analysis engines are available, or if you select **No custom analysis**, then the parser applies default text analysis rules as it annotates documents and prepares documents for the index.
4. Click **OK**.

Related concepts

"Workflow for custom analysis integration" in "Text Analysis Integration"

"Custom text analysis integration" in "Text Analysis Integration"

"Unstructured information management architecture (UIMA)" in "Text Analysis Integration"

Mapping XML elements to a common analysis structure

If your collection contains XML documents with meaningful markup, and you want to use this markup to enable users to search the enterprise search index or JDBC tables with semantic search, you can map the XML elements to a common analysis structure.

Before you begin

To map XML elements to a common analysis structure, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

To enable custom text analysis processes to access specific elements in XML documents, or to map several XML elements to a common Type for use in semantic search, you can create custom XML mapping files. The XML mapping files must adhere to the UIMA framework for text analysis.

When you add XML mapping files to a collection that uses a custom text analysis engine, you enable XML elements in source documents to be mapped to annotations in the common analysis structure. These annotations can then be used by your custom text analysis engine. You can then map the analysis results (in the common analysis structure) to the index, and enable users to query the annotations when they search the collection with semantic search

For example, you can map the content of addressee and customer elements to Person annotations in the common analysis structure. These annotations can then be accessed by your custom annotators, which might detect additional information (for example, they might detect the gender of the Person). You can also map Person annotations to the enterprise search index, which allows users to search for Persons without having to know the original XML elements.

If you want to allow users to specify the original XML elements in queries, then you do not need to define any XML mappings. Instead, you can configure parsing options and enable native XML search for the collection.

Procedure

To map XML elements to a common analysis structure:

1. Edit a collection, select the Parse page, and click **Configure text processing options**.
2. In the **Map XML elements to common analysis structures** area, click **Add Mapping**.
3. On the Map XML Elements to a Common Analysis Structure page, type a descriptive display name for the XML mapping file.
4. Specify the location of the file. If the XML mapping file is smaller than 8 MB, you can type the path or browse for the file. If the XML mapping file is larger than 8 MB, ensure that the file is on the index server and specify the fully qualified path for the file.
5. Click **OK**. Your new XML mapping file is added to the list of XML mapping files on the Text Processing Options page.

Related concepts

"Workflow for custom analysis integration" in "Text Analysis Integration"

"Custom text analysis integration" in "Text Analysis Integration"

"Unstructured information management architecture (UIMA)" in "Text Analysis Integration"

"Approaches for using XML markup in analysis and search" in "Text Analysis Integration"

Related tasks

"Creating an XML to UIMA types mapping configuration file" in "Text Analysis Integration"

Mapping a common analysis structure to the index

You can specify which common analysis structure you want to when users query a collection with semantic search.

Before you begin

To map a common analysis structure to the index, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

About this task

By mapping a common analysis structure to the enterprise search index, you enable users to specify semantically precise queries and improve the quality of the search results.

For example, depending on the entities and relationships detected by the annotators, users can search for concepts that occur in the same sentence (such as a specific person and any competitor name), or a keyword and a concept (such as the term Alex and a phone number).

Procedure

To map a common analysis structure to the index:

1. Edit a collection, select the Parse page, and click **Configure text processing options**.
2. In the **Map one common analysis structure to the index** area, click **Select a common analysis structure**.
3. On the Select a Common Analysis Structure for this Collection page, select the mapping that you want to use with the enterprise search index:
 - To use the default system rules for mapping data structures in a common analysis structure to the enterprise search index, select **Default**.
 - To map a specific common analysis structure to the index, specify the location of the common analysis structure file. If the file is smaller than 8 MB, the file can be on your system. If the file is larger than 8 MB, the file must be on the index server.
4. Click **OK**. The common analysis structure that you specified is displayed on the Text Processing Options page.

Related concepts

"Workflow for custom analysis integration" in "Text Analysis Integration"

"Custom text analysis integration" in "Text Analysis Integration"

"Unstructured information management architecture (UIMA)" in "Text Analysis Integration"

"Approaches for indexing custom analysis results" in "Text Analysis Integration"

Related tasks

"Creating the index build configuration file" in "Text Analysis Integration"

Mapping a common analysis structure to JDBC tables

You can specify which common analysis structure you want to map to JDBC tables for use in database applications.

Before you begin

To map a common analysis structure to Java Database Connectivity (JDBC) database tables, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

About this task

By mapping a common analysis structure to JDBC tables, you enable the data to be used by database applications. For example, users can specify SQL queries outside of enterprise search to search the annotations that were added by the common analysis structure. You can also use the information for further text processing, such as using the information in data mining applications.

Procedure

To map a common analysis structure to JDBC tables:

1. Edit a collection, select the Parse page, and click **Configure text processing options**.
2. In the **Map common analysis structures to JDBC tables** area, click **Add Mapping**.
3. On the Map a Common Analysis Structure to JDBC Tables page, type a descriptive display name for the common analysis structure that you want to use to map information to JDBC tables.
4. Specify the location of the common analysis structure file. If the file is smaller than 8 MB, the file can be on your system. If the file is larger than 8 MB, the file must be on the index server.
5. Click **OK**. Your new file is added to the list of files on the Text Processing Options page. The date and time that you added the file to the system is also displayed.

Related concepts

"Workflow for custom analysis integration" in "Text Analysis Integration"

"Custom text analysis integration" in "Text Analysis Integration"

"Unstructured information management architecture (UIMA)" in "Text Analysis Integration"

"Approaches for mapping analysis results to a database" in "Text Analysis Integration"

Related tasks

"Creating the XML mapping configuration file" in "Text Analysis Integration"

Configuring threads for the parser service

If you have sufficient memory resources, you can increase the number of threads that are available to the parser for parsing documents.

Before you begin

If you have a large number of collections, you might want to increase the number of parser threads. Ensure that your system has sufficient memory to support additional threads. A parser with one thread requires 200 MB memory. An additional 50 MB of memory is required for each additional thread.

To configure the number of threads that are started for the parser, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

Procedure

To configure the number of parser threads:

1. Edit a collection, select the Parse page, and click **Configure parsing options**.
2. Specify the maximum number of parser threads that are to be started when the parser is started and click **OK**.
3. Restart the parser.

Enabling advanced analysis for compound terms

You can enhance search quality by enabling the parser to use advanced analysis for compound terms. With advanced analysis, the compound terms are decomposed so that each part can be treated like a single term.

Before you begin

To specify options for parsing compound terms, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

About this task

Some languages accumulate multi-word terms into a single words without spaces (*compound terms*). Advanced analysis and decomposition of the compound terms is helpful for searching languages like German and critical for searching languages like Korean.

If you enable advanced analysis for compound terms, users can search for terms without having to use wildcard characters to find compound forms of the query terms. For example, a search for `Organ` (organ) might return documents that contain `Organspender` (organ donor) but it will not return not documents that contain `Organisation` (organization). Unlike the wildcard query `Organ*`, which can return any string that follows `Organ`, the search matches only the full linguistic subwords within the larger compound term.

User-defined vocabulary terms, like synonyms and boost words, also apply to compound parts that are used as individual words in the query.

Procedure

To enable advanced analysis of compound terms:

1. Edit a collection, select the Parse, and click **Configure parsing options**.
2. Select the **Enable advanced analysis for compound terms** check box, and click **OK**.

Related concepts

"Linguistic support for semantic search" in "Text Analysis Integration"

"Text analysis included in enterprise search" in "Text Analysis Integration"

Enabling support for native XML search

If a collection includes XML documents, you can enable users to use the XML markup when searching for documents by enabling native XML search for the collection.

Before you begin

To enable support for searching XML documents with native XML search, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

About this task

A native XML search, such as XPath or XML fragments, can provide more precise search results by exploiting the XML markup of the documents within the query. For example, users can specify that a query term must occur within a certain XML element or attribute.

For example, invoices from a computer retailer that are in XML format might contain <order> entries that include <company> and <computertype> elements. To retrieve invoices that contain orders for IBM notebooks, a keyword search for IBM and notebook might retrieve documents that include Dell notebook computers and IBM desktop models. By using XML search, you can specify that IBM must appear within the <company> element, that notebook must appear in the <computertype> element, and that both elements must be under the same <order> element. This way, you retrieve invoices that specifically discuss IBM notebooks.

Procedure

To enable users to search a collection with native XML search:

1. Edit a collection, select the Parse page, and click **Configure parsing options**.
2. Select **Enable users to search XML documents with native XML search**.
3. Click **OK**.

Related concepts

"Linguistic support for semantic search" in "Text Analysis Integration"

"Text analysis included in enterprise search" in "Text Analysis Integration"

"Semantic search applications" in "Text Analysis Integration"

"Semantic search query" in "Text Analysis Integration"

Linguistic analysis of Chinese, Japanese, and Korean documents

To enhance the retrievability of documents written in the Chinese, Japanese, and Korean languages, you can specify linguistic analysis options.

For Chinese, Japanese, and Korean documents, you can specify that the parser is to use the n-gram segmentation method for lexical analysis. For Chinese and Japanese documents, you can also configure the parser to remove new line characters from white space.

N-gram segmentation

When you create a collection, you select the type of lexical analysis that you want to use for parsing documents that are written in languages that do not use blank space to delimit words.

Unicode-based white space segmentation uses blank space as the delimiter between words. N-gram segmentation considers overlapping sequences of any number of characters as a single word. For languages like Chinese, Japanese, and Korean, which do not use blanks as word delimiters, n-gram segmentation can return better search results than Unicode-based white space segmentation.

You choose the segmentation method that you want to use for parsing documents when you create a collection. After the collection is created, you can view the setting by viewing parsing options, but you cannot change it.

Removing new line characters from non-ASCII character ranges

In languages where white space is not used to delimit word boundaries, such as Chinese and Japanese, you can configure the parser to remove certain white space characters that cause line breaks.

Before you begin

To complete this task, you must log in as an enterprise search administrator.

About this task

If this option is enabled for a collection, and a document is written in Chinese or Japanese, then the parser will remove sequences of white space characters that separate two letter characters. The letter characters must be from a non-ASCII Unicode character range. The following characters are removed:

- Tab (0x09)
- LF or line feed (0x0A)
- CR or carriage return (0x0D)

For the change to become effective, stop and restart the parser. To apply the change to documents that are already parsed and stored in the index, crawl and parse the documents again, and then reorganize the index.

Procedure

To remove new line characters from white space:

1. On the index server, log in as the enterprise search administrator. This user ID was specified when WebSphere II OmniFind Edition was installed.
2. Use a text editor to edit the following file, where *collection_ID* is the ID that was specified for the collection (or that was assigned by the system) when the collection was created:
`collection_ID.parserdriver/collection.properties`
3. Change the value of the `removeCjNewlineChars` property from `false` to `true`.

Document types associated with collection parsers and Stellent sessions

To ensure that documents in a crawl space are accurately and efficiently parsed, you can create configuration files to specify which types of documents are to be parsed by the collection parser and which are to be parsed by a Stellent document filtering session.

In an enterprise search collection, most document formats are processed by built-in HTML or XML parsers. Certain types of documents are typically not parsed (such as Postscript documents), and other types of documents are handled by Stellent parsing functions (such as Microsoft Word, Microsoft Excel, Microsoft PowerPoint, Lotus Freelance, Lotus 123, PDF, RT, and Ichitaro document types).

Because metadata can be misleading, plain text and HTML documents might be sent to the Stellent session in error, and then sent back to one of the built-in parsers, a situation that can impact performance. To avoid this situation, you can create configuration files to control where and how different types of documents are parsed.

Associating document types with the collection parser and Stellent session involves the following tasks:

1. Configuring document types for the collection parser. This step involves creating a configuration file that maps document types to the parser that is used by a collection. You can create one of these configuration files per collection.
2. Configuring document types for the Stellent session. This step involves creating a configuration file that maps document types to the Stellent document filters that are used by a collection. You can create one of these configuration files per collection.
3. Stopping and restarting the parser. For the changes to become effective, use the enterprise search administration console to monitor the collection for which you configured document types, then stop and restart the parser.

Associating document types with a collection parser

To associate particular types of documents with a collection parser, you create a configuration file. There is no support for this task in the enterprise search administration console.

Before you begin

To complete this task, you must log in as an enterprise search administrator.

About this task

If the configuration file does not exist, the collection parser uses the default parser service rules. If the configuration file exists, rules in the file specify:

- Which documents you want to send to the Stellent session, depending on the file extension or the content type.
- How to parse documents whose type is unknown because of incomplete metadata.

The format of the file is a sequence of lines, where each line is one of the following rules:

EXTENSION *extension parser*

All documents whose URL ends on the specified extension will be processed by the specified parser. Do not include the period in the extension; comparison is not case sensitive.

CONTENTTYPE *type/subtype parser*

All documents whose content type matches the specified type/subtype will be processed by the specified parser. Given the content type t/s of a document, a match occurs if t equals type, and either s equals subtype or the subtype is a wildcard character (the asterisk, *).

UNKNOWN *parser*

All documents whose extension and content type are not known (that is, not made available by the crawler), will be processed by the specified parser.

DEFAULT *parser*

All documents that are not covered by any of the other rules will be processed by the specified parser.

In all cases, *parser* must specify `html`, `xml`, `stellent`, or `none`, where `none` means that documents of that type are not to be parsed.

If more than one rule matches a document, then the more specific rule prevails, disregarding the order in which the rules appear:

- An **EXTENSION** rule is more specific than a **CONTENTTYPE** rule.
- A **CONTENTTYPE** rule that includes a subtype is more specific than one with a wildcard character. For example, a rule for content type `application/postscript` has priority over a rule for `application/*`.
- There should not be two rules for the same extension or content type. In that case, it is up to the implementation which of the rules is given priority.

Procedure

To associate document types with the collection parser:

1. On the index server, log in as the enterprise search administrator. This user ID was specified when WebSphere II OmniFind Edition was installed.
2. Create the configuration file as follows, where *collection_ID* identifies the collection that you want to configure:
`ES_NODE_ROOT/master_config/collection_ID.parserdriver/parserTypes.cfg`
3. Use a text editor to edit the file and specify parser service rules, then save and exit the file.
4. For the changes to become effective, use the enterprise search administration console to monitor the parser for the collection, and stop and restart the parser.

Example

In this example, the built-in HTML parser processes all documents with the extension `txt`, `htm` or `html`, with a content type that begins with `text/`, or with an unknown extension and content type. The built-in XML parser processes all documents with extension `xml` or with content type `text/xml`. All other documents, including those with a content type that starts with `application/`, are sent to the Stellent session.

```
EXTENSION doc stellent
EXTENSION txt html
EXTENSION htm html
```

```

EXTENSION html html
EXTENSION xml xml
EXTENSION ps none
CONTENTTYPE text/xml xml
CONTENTTYPE text/* html
CONTENTTYPE application/* stellent
UNKNOWN html
DEFAULT stellent

```

Default collection parser service rules

If you do not create a configuration file to map file types and content types to the parser for a collection, default rules are used to parse documents.

The default rules used by the collection parser are as follows:

```

EXTENSION pdf stellent
EXTENSION ppt stellent
EXTENSION prz stellent
EXTENSION lwp stellent
EXTENSION doc stellent
EXTENSION rtf stellent
EXTENSION xls stellent
EXTENSION 123 stellent
EXTENSION vsd stellent
EXTENSION vdx stellent
EXTENSION jxw stellent
EXTENSION jsw stellent
EXTENSION jtw stellent
EXTENSION jaw stellent
EXTENSION juw stellent
EXTENSION jbw stellent
EXTENSION jvw stellent
EXTENSION jfw stellent
EXTENSION jtt stellent
EXTENSION jtd stellent
EXTENSION jttc stellent
EXTENSION jtdc stellent
EXTENSION jtdx stellent
EXTENSION ps none
EXTENSION xml xml
EXTENSION txt text
EXTENSION htm html
EXTENSION html html
EXTENSION shtml html
EXTENSION xhtml html
EXTENSION asp html

CONTENTTYPE application/postscript none
CONTENTTYPE application/* stellent
CONTENTTYPE text/rtf stellent
CONTENTTYPE text/richtext stellent
CONTENTTYPE text/xml xml
CONTENTTYPE text/html html
CONTENTTYPE text/plain text

UNKNOWN html
DEFAULT html

```

Associating document types with a Stellent session

To specify which types of documents are to be parsed by Stellent document filters, you create a configuration file. There is no support for this task in the enterprise search administration console.

Before you begin

To complete this task, you must log in as an enterprise search administrator.

About this task

The configuration file specifies:

- Which file types are to be parsed by the Stellent session. A file type corresponds to one of the file types recognized by the Stellent library.
- Which file types are to be sent back to the collection parser for processing with one of the built-in parsers. (This action is needed because the collection parser might send a document to the Stellent session in error, due to misleading metadata.)
- Which file types are to be rejected because they are not supported in enterprise search.

If the configuration file was specified but does not exist, the parser will fail to start. If no configuration file was specified for the `OutsideInSupportedTypes` property in the `stellent.properties` file, then the default parsing rules for Stellent sessions will be used.

The configuration file lists document types and how they are to be handled. The format of the file is a sequence of lines, where each line is one of the following rules:

```
accept DEFAULT
accept ALL doctype
accept type doctype
native DEFAULT
native type doctype
reject type
```

Where:

doctype

Is the value to be used for the doctype query token. Documents can be searched by document type. For example, a user might specify `$doctype::pdf` to search PDF documents.

type Is one of the `FI_` values in the Stellent library, and *doctype* is the value to be used for the doctype token if a rule is applied.

DEFAULT

Means that the list of accepted or native types, depending on the type of the rule, includes the default list. This option enables you to extend the default configuration instead of replacing it.

A11 Means that all types that are not explicitly listed are accepted with the specified doctype token.

The rules in the configuration file are processed as follows:

- If there is an accept rule for *type*, including the default list if accept DEFAULT was specified, it is accepted.
- Else, if there is a reject rule for *type*, it is not accepted.
- Else, if accept ALL was specified, it is accepted.
- Otherwise, it is not accepted.

If the document type is accepted, then the *doctype* value that was specified in the rule that was applied is used. This value is sent back to the collection parser in addition to the parsed content. If the document type is not accepted, the following behavior occurs:

- If there is a native rule for *type* (including the default parsing rules if `native DEFAULT` was specified), the document is sent back to the built-in parser in addition to the value for the *doctype* token that is specified by this rule. The value of *doctype* must be either `txt`, `htm` or `xml`, indicating plain text, HTML or XML, respectively.
- Otherwise, the document is rejected and will not be parsed.

Procedure

To associate document types with the Stellent session:

1. On the index server, log in as the enterprise search administrator. This user ID was specified when WebSphere II OmniFind Edition was installed.
2. Edit the `ES_NODE_ROOT/master_config/collection_ID.stellent/stellent.properties` file, where *collection_ID* identifies the collection that you want to configure.
3. For the `OutsideInSupportedTypes` property, specify the absolute path of the configuration file that you are creating.

For example, you might create the following configuration file for a single collection, and store it with other collection-specific files:

```
ES_NODE_ROOT/master_config/collection_ID.stellent/stellenttypes.cfg
```

As another example, you might create the following configuration file so that you can use the same settings with all collections, and store this file with other system-level files. (If you use this approach, be sure to specify this path in the `stellent.properties` file for each collection, as specified in step 2.)

```
ES_INSTALL_ROOT/default_config/stellent/stellenttypes.cfg
```

4. Use a text editor to create the configuration file and specify Stellent parsing rules, then save and exit the file.
5. For the changes to become effective, use the enterprise search administration console to monitor the parser for the collection, and stop and restart the parser.

Example

In the following configuration file, the Stellent session accepts, in addition to the default list, the Microsoft Visio format.

```
accept DEFAULT
accept FI_VISI03 visio
accept FI_VISI04 visio
accept FI_VISI05 visio
accept FI_VISI06 visio
```

In the following configuration file, Postscript documents will be supported and searchable with a document type of `ps`; the X pixmap format (XPM) will be sent back to the built-in text parser; the PNG image format will be rejected; and all other file types will be accepted and made searchable with a document type of `other`.

```
accept DEFAULT
accept FI_POSTSCRIPT ps
native FI_XPIXMAP txt
accept ALL other
reject FI_PNG
```

Default parsing rules for Stellent sessions

If you do not create a configuration file to map file types to Stellent session document filters, default rules are used to parse documents.

The default rules used by the Stellent session are as follows:

| | |
|-----------------------------|-----|
| ACCEPT FI_WORD4 | doc |
| ACCEPT FI_WORD5 | doc |
| ACCEPT FI_RTF | rtf |
| ACCEPT FI_WINWORD1 | doc |
| ACCEPT FI_WINWORD1COMPLEX | doc |
| ACCEPT FI_WINWORD2 | doc |
| ACCEPT FI_WORD6 | doc |
| ACCEPT FI_WINWORD6 | doc |
| ACCEPT FI_ICHITAR03 | jxw |
| ACCEPT FI_ICHITAR04 | jsw |
| ACCEPT FI_WINWORD1J | doc |
| ACCEPT FI_WINWORD5J | doc |
| ACCEPT FI_RTFJ | rtf |
| ACCEPT FI_WINWORD7 | doc |
| ACCEPT FI_WORDPRO | lwp |
| ACCEPT FI_WINWORD97 | doc |
| ACCEPT FI_ICHITAR08 | jtd |
| ACCEPT FI_WORDPRO97 | lwp |
| ACCEPT FI_WINWORD2000 | doc |
| ACCEPT FI_WINWORD2002 | doc |
| ACCEPT FI_WINWORD2003 | doc |
| ACCEPT FI_123R1 | 123 |
| ACCEPT FI_123R2 | 123 |
| ACCEPT FI_123R3 | 123 |
| ACCEPT FI_EXCEL | xls |
| ACCEPT FI_EXCEL3 | xls |
| ACCEPT FI_EXCEL4 | xls |
| ACCEPT FI_123R4 | 123 |
| ACCEPT FI_EXCEL5 | xls |
| ACCEPT FI_123R6 | 123 |
| ACCEPT FI_EXCEL97 | xls |
| ACCEPT FI_123R9 | 123 |
| ACCEPT FI_EXCEL2000 | xls |
| ACCEPT FI_EXCEL2002 | xls |
| ACCEPT FI_EXCEL2003 | xls |
| ACCEPT FI_FREELANCE | prz |
| ACCEPT FI_POWERPOINT4 | ppt |
| ACCEPT FI_POWERPOINT3 | ppt |
| ACCEPT FI_POWERPOINT7 | ppt |
| ACCEPT FI_FREELANCE3 | prz |
| ACCEPT FI_POWERPOINTMAC3 | ppt |
| ACCEPT FI_POWERPOINTMAC4 | ppt |
| ACCEPT FI_PDF | pdf |
| ACCEPT FI_EXTPOWERPOINT4 | ppt |
| ACCEPT FI_EXTPOWERPOINTMAC4 | ppt |
| ACCEPT FI_POWERPOINTMACB3 | ppt |
| ACCEPT FI_POWERPOINTMACB4 | ppt |
| ACCEPT FI_POWERPOINT97 | ppt |
| ACCEPT FI_PDFMACBIN | pdf |
| ACCEPT FI_POWERPOINT9597 | ppt |
| ACCEPT FI_POWERPOINT2000 | ppt |
| ACCEPT FI_POWERPOINT2 | ppt |
| NATIVE FI_HTML | htm |
| NATIVE FI_HTML_LATIN2 | htm |
| NATIVE FI_HTML_JAPANESESJIS | htm |
| NATIVE FI_HTML_JAPANESEEUC | htm |
| NATIVE FI_HTML_CHINESEBIG5 | htm |
| NATIVE FI_HTML_CHINESEEUC | htm |
| NATIVE FI_HTML_CHINESEGB | htm |

| | |
|-----------------------------|-----|
| NATIVE FI_HTML_KOREANHANGUL | htm |
| NATIVE FI_HTML_CYRILLIC1251 | htm |
| NATIVE FI_HTML_CYRILLICKO18 | htm |
| NATIVE FI_CYRILLIC1251 | txt |
| NATIVE FI_CYRILLICKO18 | txt |
| NATIVE FI_W2KHTML | htm |
| NATIVE FI_XL2KHTML | htm |
| NATIVE FI_PP2KHTML | htm |
| NATIVE FI_XML | xml |
| NATIVE FI_WML | xml |
| NATIVE FI_HTML_JAPANESEJIS | htm |
| NATIVE FI_WML_CHINESEBIG5 | xml |
| NATIVE FI_WML_CHINESEEEUC | xml |
| NATIVE FI_WML_CHINESEGB | xml |
| NATIVE FI_WML_CYRILLIC1251 | xml |
| NATIVE FI_WML_CYRILLICKO18 | xml |
| NATIVE FI_WML_JAPANESEJIS | xml |
| NATIVE FI_WML_JAPANESEJIS | xml |
| NATIVE FI_WML_JAPANESEEEUC | xml |
| NATIVE FI_WML_JAPANESEEEUC | xml |
| NATIVE FI_WML_KOREANHANGUL | xml |
| NATIVE FI_WML_LATIN2 | xml |
| NATIVE FI_HTMLUNICODE | htm |
| NATIVE FI_XML_DOCTYPE_HTML | htm |
| NATIVE FI_XHTML | htm |
| NATIVE FI_ASCII | txt |
| NATIVE FI_ANSI | txt |
| NATIVE FI_UNICODE | txt |
| NATIVE FI_ASCII8 | txt |
| NATIVE FI_ANSI8 | txt |
| NATIVE FI_MAC | txt |
| NATIVE FI_MAC8 | txt |
| NATIVE FI_SHIFTJIS | txt |
| NATIVE FI_CHINESEGB | txt |
| NATIVE FI_HANGEUL | txt |
| NATIVE FI_CHINESEBIG5 | txt |
| NATIVE FI_LATIN2 | txt |
| NATIVE FI_JAPANESE_EUC | txt |
| NATIVE FI_HEBREW_OLDCODE | txt |
| NATIVE FI_HEBREW_PC8 | txt |
| NATIVE FI_HEBREW_E0 | txt |
| NATIVE FI_HEBREW_WINDOWS | txt |
| NATIVE FI_ARABIC_710 | txt |
| NATIVE FI_ARABIC_720 | txt |
| NATIVE FI_ARABIC_WINDOWS | txt |
| NATIVE FI_7BITTEXT | txt |
| NATIVE FI_JAPANESE_JIS | txt |
| NATIVE FI_CENTRALEU_1250 | txt |
| NATIVE FI_UTF8 | txt |
| NATIVE FI_EBCDIC_37 | txt |
| NATIVE FI_EBCDIC_273 | txt |
| NATIVE FI_EBCDIC_277 | txt |
| NATIVE FI_EBCDIC_278 | txt |
| NATIVE FI_EBCDIC_280 | txt |
| NATIVE FI_EBCDIC_284 | txt |
| NATIVE FI_EBCDIC_285 | txt |
| NATIVE FI_EBCDIC_297 | txt |
| NATIVE FI_EBCDIC_500 | txt |
| NATIVE FI_EBCDIC_870 | txt |
| NATIVE FI_EBCDIC_871 | txt |
| NATIVE FI_EBCDIC_1026 | txt |

Enterprise search index administration

To ensure that users always have access to the latest information, enterprise search creates an index for each collection and maintains that index by periodically refreshing and reorganizing the content.

To make the data that is collected by crawlers searchable, you must create indexes. When you first create a collection, enterprise search creates an index for all of the data that was initially crawled. When the crawlers crawl new and changed data sources, enterprise search refreshes the index with new content. Eventually, the refreshed content needs to be merged into the base index. This merging process is called reorganization. Whenever the index is refreshed or reorganized, the new content is copied to the search servers and made available for searching.

Crawlers collect data continuously or on a regularly scheduled basis. If you refresh the index frequently, you enable users to search the most current data. Eventually, an index that is continuously refreshed must be reorganized. As a refreshed index grows larger, it consumes more system resources. Therefore, to maintain optimal performance, you should reorganize indexes regularly.

How often you reorganize an index depends on:

- System resources (file system space, processor speed, and memory)
- How many documents need to be crawled and recrawled
- The type of data that to crawl
- How often you change category rules (the changes do not become effective until the index is reorganized)
- How often you force a crawler to start instead of running at a scheduled time
- How often external crawlers remove or add URIs (these types of crawlers interact with enterprise search through the Data Listener API)

For collections with several million documents that are built with mostly Web documents, you should reorganize the index approximately once a day, and refresh the index every one or two hours.

To maintain a current, searchable index, you do the following tasks:

- Specify schedules for refreshing and reorganizing the index
- Change the index schedule
- Enable and disable the index schedule
- Configure concurrent index builds

To specify options that influence the user's view of the index, you can also do the following tasks:

- Configure support for wildcard characters in queries
- Configure scopes to limit the range of documents that users can search
- Collapse documents from the same source in the search results
- Remove URIs from the index

Related tasks

“Monitoring index activity for a collection” on page 230

Monitor the index for a collection when you need to see the progress of an index that is being built, enable or disable the index schedule, or start and stop indexing activity.

“Monitoring the enterprise search index queue” on page 231

You can view the status of all index builds in the index queue, stop an index that is being built, or delete an index from the queue.

Scheduling index builds

You can specify schedules for reorganizing an index and refreshing an index with new content.

Before you begin

To schedule an index build, you must be a member of the enterprise search administrator role or a collection administrator for that collection.

About this task

To ensure that users always have access to the latest information in the sources that they search, schedule the index to be reorganized and refreshed on a regular basis. When an index is reorganized, the entire index is rebuilt. The indexing processes read all of the data that was collected by crawlers and analyzed by the parser. When an index is refreshed, information that was crawled since the last time the index was reorganized is made searchable.

By default, the option to schedule index builds is selected. This option tells the scheduler process to schedule tasks to refresh and reorganize the index whenever the enterprise search system is started. You can clear the **Enable when system starts** check box at any time if you need to prevent a scheduled index build from running. For example, you might need to disable the schedule to troubleshoot problems.

Procedure

To schedule index builds:

1. Edit a collection, select the Index page, and click **Schedule index builds**.
2. To specify how often the index is to be refreshed with new content, specify the following options on the Schedule Index Builds page in the **Specify a schedule to refresh the index** area:
 - a. In the **Start on** area, in the **Year, Month, Day, Hour, and Minute** fields, specify when you want the index to be refreshed the first time.
 - b. In the **Update interval** area, in the **days, hours, and minutes** fields, specify how often you want the index to be refreshed.

Typically, you should refresh the index frequently, such as every hour or two. Depending on how often the source content changes, specify a lower or higher interval. For example, you might specify every hour (0 days and 1 hour) or every 12 hours (0 days and 12 hours).
3. To specify how often the index is to be completely rebuilt, specify the following options in the **Specify a schedule to reorganize the index** area:
 - a. In the **Start on** area, in the **Year, Month, Day, Hour, and Minute** fields, specify when you want the index to be reorganized the first time.

- b. In the **Update interval** area, in the **days**, **hours**, and **minutes** fields, specify how often you want the index to be reorganized.
Typically, you should reorganize the index regularly, such as every 24 hours. Depending on how often the source content changes, specify a lower or higher interval. For example, you might specify every 12 hours (0 days and 12 hours) or every two and a half days (2 days and 12 hours).
4. Click **OK**.

Changing the index schedule

You can change the schedule for reorganizing or refreshing an index.

Before you begin

To change an index schedule, you must be a member of the enterprise search administrator role or be a collection administrator for that collection.

Procedure

To change the index schedule:

1. Edit a collection, select the **Index** page, and change the appropriate values in the **Month**, **Day**, **Year**, and **Hour** fields. Specify how often the index is to be refreshed with new content and how often the index is to be reorganized.
2. Click **Apply**.

Enabling and disabling the index schedules

You can enable and disable the schedules for refreshing and reorganizing the index.

Before you begin

To enable or disable an index schedule, you must be a member of the enterprise search administrator role or be a collection administrator for that collection.




About this task

You can disable a schedule for an index if you need to prevent a scheduled index build from running. For example, you might want to disable the schedule to prevent an index from being built at the scheduled date and time so that you can troubleshoot problems.

You can enable and disable the schedule while you are editing a collection, and you can enable or disable the schedule while you are monitoring a collection.

Procedure

1. To enable or disable the schedule for an index by editing a collection, take the following steps:
 - a. Edit the collection that you want to change.
 - b. On the **Index** page, select or clear the **Enable when system starts** check box to enable or disable the schedule for refreshing the index.
 - c. To enable or disable the schedule for reorganizing the index, select or clear the **Enable when system starts** check box.
 - d. Click **Apply**.

2. To enable or disable the schedule for an index by monitoring a collection, take the following steps:
 - a. Monitor the collection that you want to change.
 - b. On the Index page, if an index is scheduled, and you do not want it to be built at the scheduled date and time, click  **Disable schedule**. The index is not built until you enable the schedule or click  **Start** to start the index building process.
 - c. If an index is scheduled, but the schedule for building it is disabled, click  **Enable schedule**. The index is queued for building at the date and time that you specified in the index schedule.

Configuring concurrent index builds

You control the use of indexing resources by specifying how many collections can have their index build requests processed at the same time. If you have sufficient system resources, you can improve search quality by enabling the index for a collection to be refreshed at the same time that it is being reorganized.

Before you begin

To specify index building options for the system, you must be a member of the enterprise search administrator role.

About this task


Enterprise search can build multiple indexes at a time by sharing resources among collections, which enables index build requests for multiple collections to be processed in parallel. By sharing the processes, you can ensure that the reorganization of a very large index does not block the availability of other indexes that are waiting in the queue to be built.

When an index build is requested or scheduled, it enters the index queue and waits for its turn to be processed. Because each collection has its own index, several index build requests from various collections might be in the index queue at the same time. When you configure indexing options for the system, you specify how many collections can share indexing resources and have their requests processed in parallel.

You can also specify that requests to refresh a collection's index are to be processed at the same time that the collection's index is being reorganized. If you enable this option, the search servers will be refreshed with the latest documents (through an index refresh) while the more slow running index reorganization is being processed. However, index building is a resource-intensive process. A large amount of system memory and disk space is consumed while an index is being built. If you enable this option, and you have insufficient disk space or memory, overall system performance might be degraded.

Procedure

To specify index building options for the system:

1. Click **System** to open the System view.
2. Click  **Edit** to change to the system editing view.
3. On the Index page, click **Configure indexing options**.

4. On the System-Level Indexing Options page, type the number of collections that can share system resources and have their index build requests processed in parallel.
The number of collections that share indexing resources cannot exceed the number of collections in your enterprise search system. For example, if you have five collections, you must enter a number that is less than or equal to five.
5. If you have sufficient system resources to support multiple concurrent index builds for individual collections, you can select the option that enables requests to refresh and reorganize the index to run concurrently.
6. Click **OK**.

Options that influence the searchable view of the index

After documents are indexed, you can specify options that control how users can search for documents and view documents in the search results.

To specify options that influence the user's view of the index, you can do the following tasks:

- Configure support for wildcard characters in query terms. You can build support for wildcard queries into the index, or you can specify options to expand the query terms during query processing.
- Configure scopes to limit the range of documents that users can search. When users search the collection, they search only the documents that belong to the scope, not the entire index.
- Collapse documents from the same source in the search results. You can group documents that match a URI or URI pattern in the index, and show only the top result documents in the search results (users can specify options to see the collapsed result documents).
- Remove URIs from the index. You might need to temporarily prevent users from searching particular documents in the index.

Indexed options for searching documents

When you configure options for searching crawled data or external sources, or when you map XML and HTML metadata elements to search fields, you specify how the documents can be searched and shown in the search results.

When you edit options for crawlers that contain fields, you can specify the following options to control whether a field can be searched, how it can be searched, and whether it can be returned in the search results:

- Free text search
- Fielded search
- Parametric search
- Search results
- Document content

When you edit options for external sources, the option to mark the field as a document content field is not available.

When you configure the parser and specify that you want to map XML elements and HTML metadata elements to fields in the index, you specify whether the fields can be used in a fielded search, parametric search, or search results.

The options that you specify are stored with documents in the index. They enable you to restrict what users can query and what users can see in the search results.

Free text search

The enterprise search index is a full text index with content from various data sources. You can search the content by specifying a simple query in natural language. The search processes search the fields and document content to find documents that are relevant to the query.

Example:

A free text search can be as simple as the following query:

```
bicycle chain
```

To indicate which words must or must not appear in a document, you can include special notations. For example, you can precede a word by a plus sign (+) to specify that a document must contain that word for a match to occur. Precede a word by a minus sign (-) to exclude documents that contain that word from the search results. Enclose two or more words in quotation marks (") to search for an exact phrase.

Example:

In the following free text query, a match occurs only if a document contains the exact phrase science fiction and does not contain the word robot:

```
+"science fiction" -robot
```

Fielded search

A fielded search enables you to constrain the object of the query to specific fields and metadata of a document. For example, you can specify that certain words must exist in the title of a document.

To specify a fielded search in enterprise search, include the field name and the word or phrase that must exist in that field in your query.

Example:

The following query searches for documents that must contain the word `ibm` and the phrase `enterprise search` in the title field:

```
title:ibm title:"enterprise search"
```

To search a field by field name, you must enable the field for fielded searching when you configure the crawler.

Parametric search

A parametric search is a type of fielded search that enables you to do comparative or evaluative queries on numeric and date fields and metadata. For example, you can search for documents that are of a certain size or that were written after a certain date. You can also search for documents with attributes that are greater than, less than, or equal to a specified value.

Example 1:

The following query searches for items that cost exactly 50 dollars (or whatever currency unit is indexed for the price field):

```
#price::=50
```

Example 2:

The following query searches for documents that have a file size greater than 1024 but less than or equal to 2048:

```
#filesize::>1024<=2048
```

To search a field with a parametric query, you must enable the field for parametric searching when you configure the crawler.

Search results

You might want to search some fields but not show them in the search results, or you might want to see a field in the search results even though you do not query it. For example, you might need to query financial data to obtain a meaningful report, but you might now want to show employee salaries in results that also show employee names.

Document content

The content of a field that is marked as a document content field can be used to associate documents with a model-based category and with categories that specify rules that are based on document content.

Related concepts

"Query syntax" in "Programming Guide and API Reference for Enterprise Search"

"Search applications for enterprise search" on page 161

A search application enables you to search collections and external sources in the enterprise search system. You can create any number of search applications, and a single search application can search any number of collections and external sources.

Wildcard characters in queries

You can enable users to include a wildcard character in query terms and search for words that match a specified pattern.

A wildcard query term is a term that contains an asterisk (*). When a user submits a query that includes a wildcard character, the search results include all documents in the index that match the query term plus all documents in the index that match the pattern represented by the wildcard character. For example, the trailing wildcard character in the query term `sea*` can match `search`, `season`, and `seals`.

When you configure wildcard character options for an index, you choose whether you want to enable users to specify wildcard characters in queries and, if so, how this support is to be provided:

- You can enable all parts of a document to be searched for words that match the wildcard character pattern, or you can restrict the pattern matching to fields.

- You can enable all fields to support queries that contain wildcard characters, or you can limit the pattern matching to fields that you specify.
- You can restrict the wildcard character to the final character in a query term (a trailing wildcard character), or you can allow the wildcard character to occur anywhere in a query term. (The wildcard character cannot occur in a field name.)
- Depending on where you allow wildcard characters to occur, you can choose how the query terms are to be expanded (query terms that contain wildcard characters are expanded to all of the terms in the index that match). The index can store all possible expansions of terms, or the search processes can expand terms during query processing.

Any changes that you make to the wildcard character settings become effective the next time that the index is reorganized.

Index expansion

To include expansions of terms in the index, you specify how many leading characters in a word must match the wildcard character pattern in a query term for a match to occur. Only query terms that have at least this number of characters (excluding the *) return results. For example, if you specify 4, then the query term must specify four characters at a minimum for a match to occur.

If you specify the 4, then the word `technology` matches the query term `tech*` and the query term `techno*` but does not match the query term `te*`.

When the index is refreshed or reorganized, all possible expansions for each term in a document are indexed in addition to the original terms. An advantage of this approach is that no additional time is required to expand the terms during query processing. However, this approach increases the size of the index, which means you must have sufficient system resources available to accommodate the larger index.

This approach is most useful if the size of the collection is relatively small, or where the space and time to build the index are less important than query response time. For example, you might choose this approach to search a catalog or an employee directory.

This approach is available only if you enable support for trailing wildcard characters. If you enable support for wildcard characters that occur anywhere in a query term, you cannot select the option to include expansions of terms in the index.

Query expansion

To expand queries and apply pattern matching rules when users submit queries that contain wildcard characters, you specify how many variations of a query term constitute a match. For example, if you specify 50, then up to 50 variations of a query term can qualify as matches of the query term.

To illustrate this example, the query term `tech*` matches the words `technical`, `technique`, `technology`, and up to 50 different words that begin with the characters `tech`.

Although query expansion has only a minor impact on the size of the index, it can degrade query performance. The search processes must iterate over all possible expansions of the wildcard query term, up to the limit that you specify in the wildcard character settings.

This approach is most useful if the size of the collection is relatively large and the space and time to build the index must be minimized. For example, you might choose this approach for e-mail repositories, where the index must keep up with the rapidly changing documents, but query response time is less important.

This approach is available regardless of whether you enable support for trailing wildcard characters or enable support for wildcard characters that occur anywhere in a query term.

Support for wildcard characters in queries

The set of expansions for a wildcard query term contains all terms in the index that can be obtained by replacing the wildcard character with arbitrary sequences of characters. The set is determined as follows:

- If a collection supports wildcard characters that can occur anywhere in a query term, then any query term that contains an asterisk is interpreted as a wildcard term.
- The set contains, at most, the maximum number of expansions that are configured by the enterprise search administrator. If the index contains more than this number of expansions, they are ignored. (The search results indicate whether any wildcard expansions were ignored.)
- If wildcard character support is restricted to a set of fields, then the set contains only terms that appear in one of the specified fields. A term needs to appear in only one of the fields in at least one document in the index.
- If the query term is a fielded term, then the wildcard character must appear after the field specifier (for example, `fieldname:*sphere`). The field name cannot contain a colon (:).
- If wildcard character support is restricted to a set of fields, then the field name in the wildcard query term must be one of the fields that is specified in the enterprise search administration console. Otherwise, no expansions are found for the term.
- Wildcard characters are supported only on plain text terms, not on XML element names, attribute names, or attribute values. A term that consists solely of a wildcard character is not supported.

Configuring options for wildcard characters in queries

When you configure indexing options for an enterprise search collection, you can specify whether you want to enable users to include wildcard characters in query terms.

Before you begin

To configure options for wildcard characters, you must be a member of the enterprise search administrator role or be a collection administrator for the collection that the index belongs to.

About this task

When you specify wildcard character options, your changes become effective the next time the index is reorganized.

Procedure

To configure support for wildcard characters in queries:

1. Edit a collection, select the Index page, and click **Configure options for wildcard characters**.
2. On the Options for Wildcard Characters page, select the **Support wildcard characters in queries** check box.
3. Use the **Support wildcard characters in queries that do not search fields** check box to specify whether wildcard characters are supported in queries that search for text that is not in fields. For example, the free text query `tech*`, which does not search a named field, returns expanded results (such as `technology` or `technique`) only if this check box is selected.
4. Specify which fields support wildcard characters:
 - To enable all fields in a document to support queries that contain wildcard characters, select **All fields**.
 - To limit support for wildcard characters to some fields, select **Specific fields** and then type the field names. Expanded results are returned only for the fields that you specify. For example, the query `author:john*` returns expanded results only if you specify that the `author` field supports wildcard characters.
5. Specify whether the wildcard character must occur in the final position of a query term (a trailing wildcard character), or whether the wildcard character is unrestricted and can occur anywhere in a query term.

When you select a wildcard position and type, you must also specify how you want to enable support for wildcard characters. Click **Help** while you configure options for wildcard characters for details.
6. Click **OK**.

Scopes

Configure a scope when you want to present users with a limited view of a collection.

A scope is a group of related URIs in an index. When you configure a scope, you limit the documents that users can see in the collection. When users search the collection, they search only the documents in the scope, not the entire index. To use this capability, your search applications must include support for searching scopes.

When you create a scope, you specify a range of URIs in the index that users are able to search. Limiting the documents that users can search helps ensure that documents in the search results are specific to the information that users seek.

For example, you might create one scope that includes the URIs for your Technical Support department and another scope that includes the URIs for your Human Resources department. If your search application supports scopes, users in the Technical Support department will retrieve documents from the Technical Support scope, and users in the Human Resources department will retrieve documents from the Human Resources scope.

You can create as many scopes as you want, although creating too many scopes can affect performance. Configure scopes so that most search requests need to filter only on one or two scopes. Because scopes can contain entire URIs or URI patterns, the same document can belong to more than one scope.

When you configure scopes, you might need to reorganize the index twice before the changes become effective. If you configure scopes before the first index reorganization for the collection, users will be able to search the collection, but they will not be able to see the scope data in the search results. Reorganize the index again to ensure that search results reflect the range of URIs in the scope.

If you configure scopes after the index has been reorganized at least once, the changes become effective after the next index reorganization.

Configuring scopes

When you configure a scope for an enterprise search collection, you specify the URIs, or URI patterns, for a range of documents in the index that users are allowed to search.

Before you begin

To configure scopes, you must be a member of the enterprise search administrator role or be a collection administrator for the collection that the scopes belong to.

About this task

If your search applications enable support for scopes, users can search only the documents that match the URIs that define the boundaries of the scope when they search the collection.

When you configure scopes, you might need to reorganize the index twice before the changes become effective. If you configure scopes before the first index reorganization for the collection, users will be able to search the collection, but they will not be able to see the scope data in the search results. Reorganize the index again to ensure that search results reflect the range of URIs in the scope.

If you configure scopes after the index has been reorganized at least once, the changes become effective after the next index reorganization.

Procedure

To configure a scope:

1. Edit a collection, select the Index page, and click **Configure scopes**.
2. On the Scopes page, click **Create Scope**.
3. Specify a name for the scope and the URIs and URI patterns that define the boundaries of the scope. You can also specify URIs and URI patterns that you want to exclude from the scope.
4. Click **OK**.

Your new scope is listed on the Scopes page with the other scopes that belong to this collection.

Related reference

“URI formats in an enterprise search index” on page 89

The uniform resource identifier (URI) of each document in an enterprise search index indicates the type of crawler that added the document to the collection.

Collapsed URIs

Enterprise search can organize the search results so that documents from Web and NNTP sources that have the same URI prefix are displayed as a group and collapsed in the search results.

When results are collapsed, the top result typically appears flush left. One or more lower ranking results are grouped and indented below the top result.

To collapse result documents that have different URI prefixes as a single group, you can associate the URI prefixes with a group name that you create. For example, if you have three servers for managing financial data, you can group documents from all three servers in the search results and collapse the lower ranking results below the top result documents.

Search applications can use the URI prefix or the group name to collapse documents in the search results. In the sample search application for enterprise search, the top two search result documents are displayed. If more than two result documents with the same URI prefix (or that belong to the same URI group) are returned, you can select an option to see the collapsed results.

Users can use enterprise search query syntax (`samegroupas:URI prefix`) to search all documents that are in the same group as the specified URI prefix.

How to organize URI prefixes and group names

When you use the administration console to configure rules for collapsing search results, you specify the URI prefixes of the documents that you want to collapse and optionally associate the URI prefixes with a group name.

The order of the URI prefixes that you configure is important. The index server uses the order of the URI prefixes when it computes the value of each URI in a collection. For each URI:

1. The index server scans through the URI prefixes in the rules for collapsing search results sequentially.
2. When the index server finds the first URI prefix that matches a prefix of a document in the index, it associates the group name (or the URI prefix, if the rule does not specify a group name) as an extra search term for the document.
3. If a document cannot be matched to a URI prefix, then:
 - For Web URIs, the index server uses the host name of the URL as the URI prefix.
 - For NNTP URIs, the index server uses the first message ID in the value of the reference header as the URI prefix.

After you add a URI prefix to the list of those that are to be collapsed in the search results, you must position the URI prefix in the order that you want the index server to scan it and potentially associate it as an extra search term with documents in the index:

- When you add a URI prefix and do not associate it with a group name, you can select the individual URI prefix and move it up or down in the list.
- When you add a URI prefix and associate it with a group name, you move the entire group of URI prefixes that belong to the same group whenever you move a URI prefix up or down in the list. (The order of URI prefixes within a group does not matter; selecting an individual URI prefix automatically selects the entire group.)

Collapsing URIs in the search results

You can specify options for grouping and collapsing result documents from Web and NNTP sources that have the same URI prefix. You can also create a group name that enables result documents with different URI prefixes to be collapsed together.

Before you begin



To specify options for collapsing search results, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

About this task

The changes that you make for collapsing search results do not take effect until the next time that the index is reorganized.

Procedure

To specify options for collapsing search results:

1. Edit a collection, select the Index page, and click **Collapse search results**.
2. On the Collapse Search Results page, click **Add URI Prefix**.
3. On the Add a URI Prefix for Collapsing Results page, type the URI prefix for documents that you want to collapse in the search results. For example:
`http://finance/ROI/`
`http://server1.com/finance/`
4. You can type a descriptive group name that you want to associate with this URI prefix. To collapse result documents from several sources as a single group, type the same group name when you add each URI prefix.
5. Click **OK**.
6. On the Collapse Search Results page, position the new rule in the order that you want the index server to scan it:
 - If you added a URI prefix and did not associate it with a group name, the new URI prefix appears at the bottom of the list. Use the arrow keys to move it to the correct position.
 - If you associated the new URI prefix with a group name, the new URI prefix appears at the bottom of the set of URI prefixes that belong to the same group. Use the arrow keys to move the entire group of URI prefixes to the correct position.
7. To change the URI prefix or group name, select the URI prefix and click  **Edit**.
8. To remove a URI prefix from the list, select the URI prefix and click  **Remove**.

Removing URIs from the index

To prevent users from searching documents in a collection, you can remove the URIs for those documents from the index.

Before you begin

To remove URIs from the index, you must be a member of the enterprise search administrator role or a collection administrator for that collection.

About this task

If you specify a fully qualified URI, users stop seeing the URI in the search results. However, if a user submits the same query, and result documents for that query are in the search cache, then the cached result page for the URI that you removed continues to be returned in the search results. The search cache is not refreshed, and the URI is not removed from the index, until the next time the index is refreshed or reorganized.

If you specify a pattern to remove multiple URIs, users continue to see the URIs that match that pattern in the search results until the next time the index is refreshed or reorganized.

When you remove a URI from the index, you do not remove it from the crawl space. The next time that the crawler crawls the document, the URI is built into the index and becomes available for searching again. To remove a URI from the crawl space, you must update the crawling rules to exclude the document, and then stop and restart the crawler.

Procedure

To remove URIs for specific documents from the index:

1. Edit a collection, select the Index page, and click **Remove URIs from the index**.
2. On the Remove URIs from the Index page, type the URIs (or the URI patterns) that you want to remove from the index.

For example:

```
http://domain.org/hr/*  
db2://knowledgeManagement/ROI*  
cm://enterprise/finance*
```

3. Click **OK**.

Related reference

“URI formats in an enterprise search index” on page 89

The uniform resource identifier (URI) of each document in an enterprise search index indicates the type of crawler that added the document to the collection.

Search server administration for enterprise search

Options that you can specify for the search servers include using cache space for returning search results, controlling the maximum display length of document summaries in the search results, associating custom dictionaries to improve search quality, and returning predefined URIs in the search results when certain terms appear in the query.

When a user submits a query, the search servers use the index to quickly locate relevant documents. The search servers use the enterprise search data store, which contains the parsed and tokenized data, to retrieve metadata for the relevant documents. Metadata can include but is not limited to the document URI, title, description, date, data type, and so on.

When you configure the search servers for a collection, you specify options that influence how queries are processed, including options that can impact query performance:

Configuring a search cache

To optimize query performance, you can specify that result documents are to be stored in a cache, and you can configure the amount of space to allocate for cached documents.

Configuring a maximum display length for document summaries

Most result documents show a summary of the document content to help users decide whether the document is one that they want to retrieve. You can specify how much space is to be used in the search results to display this summary information.

Specifying a different default language

A default language for searching documents in the collection is specified when the collection is created, but you can specify a different language as needed.

Associating custom dictionaries

If your application developers created custom dictionaries for synonyms, stop words, or boost words, you can specify the dictionaries to use when users search the collection.

Configuring quick links

You can predetermine URIs to be returned for certain keywords and phrases. When users specify the keywords or phrases in a query, the predefined URI is returned with the search results. The quick link URIs are returned in addition to URIs that the search servers return by searching the index.

Related concepts

“Document ranking in enterprise search” on page 149

When a user searches a collection, the search processes return the most relevant results for the terms and conditions of the query.

“Custom boost word dictionaries” on page 150

To improve the quality of the search results, you can influence how documents are ranked in the search results by creating a custom boost word dictionary.

Search caches

When the load on the search servers is relatively high, you can enhance performance by caching search results.

When the search servers process search requests, they first check if results for the same query already exist in the cache. If the search servers find the appropriate result documents, they can quickly return search results to the user. If the search servers do not find the appropriate result documents, they search the index.

When the search cache fills, the oldest result documents and result documents for infrequent queries are cycled out to make room for new search results.

From the enterprise search administration console, you can enable search caching and also specify the capacity of the cache (the number of queries whose results can be cached simultaneously).

When you make changes to the search cache settings, you must restart the search servers for the changes to become effective.

Configuring a search cache

You can enable or disable the search cache for a collection. You can also specify options to control the size of the search cache.

Before you begin

To configure a search cache for a collection, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

Procedure

To configure the search cache:

1. Edit a collection, select the Search page, and click **Configure search server options**.
2. On the Search Server Options page, select the **Use the search cache** check box.
3. In the **Maximum number of entries in the cache** field, type the maximum number of search result sets that the search cache can hold.
4. Click **OK**.
5. For the changes to become effective, monitor the search servers and restart the server processes.

Custom synonym dictionaries

To improve the quality of the search results, you can enable users to search for synonyms of their query terms when they search a collection.

If you create a synonym dictionary, add it to the enterprise search system, and associate it with a collection, users can search for documents that contain synonyms of their query terms when they search the collection. By expanding queries in this manner, users are more likely to find all documents of interest, not just documents that match their precise query terms. Because you define which

words are synonyms of each other when you create the synonym dictionary, you help ensure that users find relevant documents without having to specify all variations of the query term.

For example, your organization might use acronyms and abbreviations to refer to departments, equipment, and so on, or the documents in your collections might contain vocabulary that is specific to your industry. By creating a synonym dictionary, you can ensure that a query that includes an acronym (such as ACL) returns documents that discuss the expansion of that acronym (such as ACLs, access control lists, access controls, and so on).

The enterprise search query language support synonyms by allowing users to prepend a tilde operator to a query term. For example, the query ~WAS might return documents that discuss WebSphere Application Server. Application developers can also make synonym support available through query properties, which do not require special syntax.

Synonym dictionaries contain variants of words and have the following characteristics:

- The words are not specific to a language, but they can be used in different languages. There is only one synonym dictionary per collection.
- The words are not inflected. All possible inflections must be added to the synonym list. For example, an inflection can be the singular and the plural form of the word (such as ACL and ACLs).

Most of the terms that you add to a synonym dictionary are strict semantic equivalents, which means that if term A is a synonym of term B, then B is a synonym of A. Every time A is used in a query, B can be used, and vice versa.

However, you can also add terms that correspond to different uses of a term, including generic or more specific variants of the term. For example, you can have one synonym group that includes both `building` and `house`, and another group that includes `bank`, `shore`, and `credit union`.

The less strict that the relationship is between the terms, the larger the search result, although some of the search results might not be relevant to the query. The Search and Index API provides methods that allow users to select the appropriate synonyms when they submit a search request, and methods that show users which query terms were expanded to which synonyms.

To create a synonym dictionary, an expert in the subject matter of the collection needs to create a synonym list in XML format (or work with an application developer to create the XML file). A tool that is provided with WebSphere II OmniFind Edition must be used to convert the XML file to a binary (`.dic`) file.

An enterprise search administrator uploads the binary file to the system and assigns it a display name. Collection administrators can select a synonym dictionary to use for searching documents in a collection when they configure search server options for a collection.

Restriction: After you add a custom synonym dictionary to the system, you cannot edit it. To revise the synonyms that are available to a collection, you must:

1. Update the source XML file.
2. Convert the XML source to a new dictionary file.

3. Remove the old synonym dictionary from the collections that use it.
4. Delete the old synonym dictionary from the system.
5. Add the new synonym dictionary to the system.
6. Associate the new synonym dictionary with the collections that are to use it.

Related concepts

"Synonym support in search applications" in "Text Analysis Integration"

Related tasks

"Creating an XML file for synonyms" in "Text Analysis Integration"

"Creating a synonym dictionary" in "Text Analysis Integration"

Adding synonym dictionaries to the system


If you create custom synonym dictionaries for searching the documents in a collection, you must associate the dictionaries with the enterprise search system. You can later choose which synonym dictionary you want to use for searching a collection.

Before you begin

To add your custom synonym dictionaries for use with enterprise search queries, you must be a member of the enterprise search administrator role.

Procedure

To associate synonyms with the enterprise search system:

1. Click **System** to open the System view.
2. Click  **Edit** to change to the system editing view.
3. On the Search page, click **Configure synonym dictionaries**.
4. On the Configure Synonym Dictionaries page, click **Add Synonym Dictionary**.
5. On the Add a Synonym Dictionary page, type a unique display name for the synonym dictionary and optionally type a description.
6. Specify the location of the .dic file. If the file is smaller than 8 MB, the file can be on your system. If the file is larger than 8 MB, the file must be on the index server.
7. Click **OK**. Your custom synonym dictionary is added to the enterprise search system and becomes available for searching collections.

Associating a synonym dictionary with a collection

If synonym dictionaries are associated with the enterprise search system, you can select one to use when searching a collection. If a query term matches a term in the dictionary, then result documents that contain synonyms of that term are also returned in the search results.

Before you begin

To select a synonym dictionary for a collection, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

Procedure

To associate a synonym dictionary with a collection:

1. Edit a collection, select the Search page, and click **Configure search server options**.
2. In the **Synonym dictionary name** field on the Search Server Options page, select the synonym dictionary that you want to use when users query this collection.

The list of available synonym dictionaries includes all synonym dictionaries that were added to the enterprise search system.

3. Click **OK**.

Custom stop word dictionaries

To improve the quality of the search results, you can specify that certain words are to be automatically removed from the query terms during query processing.

A stop word dictionary contains enterprise-specific terms that are frequently used, and thus are not useful as query terms. By excluding these words from queries, you can ensure that users are not inundated with result documents that are only marginally relevant (only documents that match other terms in the query will be returned). During query processing, the search servers remove the stop words from queries. The words that are removed include stop words in your custom dictionary and stop words that are predefined for enterprise search (such as common prepositions and articles).

WebSphere II OmniFind Edition performs language-specific stop word recognition by default. This process removes frequent common words like *a* and *the* from a query. You need to define a custom stop word dictionary only for enterprise or domain-specific stop words.

To create a stop word dictionary, an expert in the subject matter of the collection needs to create a stop word list in XML format (or work with an application developer to create the XML file). A tool that is provided with WebSphere II OmniFind Edition must be used to convert the XML file to a binary (.dic) file.

An enterprise search administrator uploads the binary file to the system and assigns it a display name. Collection administrators can select a stop word dictionary to use for searching documents in a collection when they configure search server options for a collection.

Restriction: After you add a custom stop word dictionary to the system, you cannot edit it. To revise the stop words that are available for query processing, you must:

1. Update the source XML file.
2. Convert the XML source to a new dictionary file.
3. Remove the old stop word dictionary from the collections that use it.
4. Delete the old stop word dictionary from the system.
5. Add the new stop word dictionary to the system.
6. Associate the new stop word dictionary with the collections that are to use it.

Related concepts

"Custom stop word dictionaries" in "Text Analysis Integration"

Related tasks

"Creating an XML file for stop words" in "Text Analysis Integration"

"Creating a stop word dictionary" in "Text Analysis Integration"

Adding stop word dictionaries to the system


If you create custom stop word dictionaries for removing words from queries, you must associate the dictionaries with the enterprise search system. You can later choose which stop word dictionary you want to use for searching a collection.

Before you begin

To add custom stop word dictionaries to the system, you must be a member of the enterprise search administrator role.

Procedure

To associate custom stop words with the enterprise search system:

1. Click **System** to open the System view.
2. Click  **Edit** to change to the system editing view.
3. On the Search page, click **Configure stop word dictionaries**.
4. On the Configure Stop Word Dictionaries page, click **Add Stop Word Dictionary**.
5. On the Add a Stop Word Dictionary page, type a unique display name for the dictionary.
6. Specify the location of the .dic file. If the file is smaller than 8 MB, the file can be on your system. If the file is larger than 8 MB, the file must be on the index server.
7. Click **OK**. Your custom stop word dictionary is added to the enterprise search system and becomes available for searching collections.

Associating a stop word dictionary with a collection

If stop word dictionaries are associated with the enterprise search system, you can select one to use when searching a collection. If a query term matches a term in the dictionary, then that term is removed from the query before it is processed.

Before you begin

To select a stop word dictionary for a collection, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

Procedure

To associate a stop word dictionary with a collection:

1. Edit a collection, select the Search page, and click **Configure search server options**.
2. In the **Stop word dictionary name** field on the Search Server Options page, select the stop word dictionary that you want to use when users query this collection.

The list of available dictionaries includes all stop word dictionaries that were added to the enterprise search system.

3. Click **OK**.

Dynamic summarization

Dynamic summarization is a technique that determines which phrases of a result document best represent the concepts that the user is searching for.

For enterprise search, dynamic summarization tries to capture sentences in documents that contain a large variety of the search terms. A few sentences, or parts of sentences, are selected and displayed in the search results. The search terms are highlighted through HTML rendering of the search results.

When configuring search server options for a collection, you can specify the maximum display length for document summaries in the search results. Because the summary includes highlighting characters, the buffer returned to the search application will be larger than the specified maximum value. The display length, however, will not exceed the specified maximum value, although the summary might be shorter (depending on the summary data extracted from the source document).

Customizing document summaries in the administration console

You can customize the amount of information that is shown in document summaries by specifying options for the search server in the enterprise search administration console.

Before you begin

To control the display length of summaries for a collection, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

About this task

The value that you specify for the maximum display length of document summaries works with the value that you specify for the number of sentences that each summary can contain. The value that results in the shortest document summary has precedence.

For example, if you specify a limit of four sentences, then the document summary contains only four sentences, even if the display length allows more characters than the total number of characters in those sentences. For another example, a limit of 10 sentences combined with a 500-character limit for the display length, might result in document summary that contains fewer than 10 sentences.

Procedure

To configure a display length for document summaries:

1. Edit a collection, select the Search page, and click **Configure search server options**.
2. On the Search Server Options page, specify a maximum display length for document summaries. When users view search results, the document summaries will not exceed the value that you specify.
3. Specify how many sentences each document summary can contain (summaries can contain up to 10 sentences).
4. Click **OK**.

5. For the changes to become effective, monitor the search servers and restart the server processes.

Customizing document summaries by editing properties

Each result document for an enterprise search query includes a summary. You can customize the amount of information that each summary contains by editing a properties file.

About this task

You can customize search result descriptions by changing values for the following properties in the `ES_NODE_ROOT/master_config/collection_ID.runtime.node1/runtime-generic.properties` file:

MinWordsPerSentence

The minimum number of words that a description sentence can contain.
The default value is 4.

MaxWordsPerSentence

The maximum number of words that a description sentence can contain.
The default value is 20.

NumberOfReturnedSentences

The number of sentences that constitute a document's description. The default value is 5.

MaxSentencesPerDocument

The maximum number of sentences in a document that will be considered as candidates in the process of creating the description. The default value is 1000.

Procedure

To customize document summaries in the search results:

1. On the search servers, log in as the enterprise search administrator. This user ID was specified when WebSphere II OmniFind Edition was installed.
2. Use a text editor to edit the following file, where `coll_ID` is the ID that was specified for the collection (or that was assigned by the system) when the collection was created:

```
ES_NODE_ROOT/master_config/coll_ID.runtime.node1/runtime-generic.properties
```

Tip: To determine the mapping between a collection name and its ID, see the `ES_NODE_ROOT/master_config/collections.ini` file.

3. Change the properties that you want to customize, then save and exit the file.
4. Stop and restart the search servers to apply the changes.

Working with quick links

Quick links are documents that are returned in the search results whenever a user submits a query that includes specific words and phrases.

You use the enterprise search administration console to configure quick links for a collection.

Quick links

Quick links enable you to provide users with links to documents that are predetermined to be relevant to the query terms.

A quick link is a URI that enterprise search automatically includes in the search results when a query includes certain words or phrases. Typically, the quick link URIs appear at the top of the result list, which helps ensure that users see the documents that you predetermined to be relevant to the query.

Quick links are returned in addition to other search results. The search processes search the index for documents that match the query terms, and return URIs for those documents in addition to the quick link URIs.

When you configure a quick link, you can specify a descriptive title and summary for the URI to help users recognize the document and quickly determine whether it is a document that they want to retrieve.

For example, for the URI `http://www.ibm.com/education/us/`, you might use a title such as *IBM Education in the United States*, and provide the summary *Solutions, products, and resources for professionals, educators, and students in the United States*.

To use quick links in enterprise search collections, the option for showing quick links must be available in the search application. In some search applications, users might have the ability to enable and disable the return of quick links when they search the collection.

Configuring quick links

To create a quick link for an enterprise search collection, you associate the URI of a document with the keywords that trigger its inclusion in the search results.

Before you begin

To configure quick links, you must be a member of the enterprise search administrator role or be a collection administrator for the collection that the quick link belongs to.

About this task

For examples of how to specify keywords and URIs for quick links, click **Help** while you are creating or editing a quick link.

You do not need to restart the search servers for your changes to become effective.

Procedure

To configure a quick link:

1. Edit a collection, select the Search page, and click **Configure quick links**.
2. On the Quick Links page, click **Create Quick Link**.
3. Specify the keywords and phrases that cause this quick link to be returned in the search results, the URI for the document that you predetermined is relevant to this query, and other options for this quick link.

You can specify one keyword, several keywords, or one phrase (two or more words enclosed in quotation marks) per line. Separate keywords with a space (you cannot use a comma to delimit keywords). Press the Enter key to start a new line.

4. Click **OK**.

Your new quick link is listed on the Quick Links page with the other quick links that belong to this collection.

Related reference

“URI formats in an enterprise search index” on page 89

The uniform resource identifier (URI) of each document in an enterprise search index indicates the type of crawler that added the document to the collection.

Document ranking in enterprise search

When a user searches a collection, the search processes return the most relevant results for the terms and conditions of the query.

The search servers support a rich query syntax and use several techniques to produce the most relevant search results, such as text-based scoring and static ranking. You can extend the default ranking behavior by configuring options that influence the importance of documents in the search results:

- You can create custom boost word dictionaries to influence how documents that contain the specified boost words are ranked in the search results.
- You can influence the scores of documents that match a specified URI pattern.
- You can influence the scores of documents that contain fields that are mapped to boost classes.

Related concepts

“Document ranking that is based on boost classes” on page 154

By mapping fields to boost classes, you can influence how documents are ranked in the search results.

“Document ranking that is based on URI patterns” on page 152

You can increase or decrease the importance of documents by assigning boost factors to URI patterns.

“Custom boost word dictionaries” on page 150

To improve the quality of the search results, you can influence how documents are ranked in the search results by creating a custom boost word dictionary.

Text-based scoring

Enterprise search dynamically calculates a score for each document that matches the terms in a query.

To calculate the text score of each document that matches a query, enterprise search considers many factors, such as:

- The frequency of each query term in the entire collection. In general, query terms that appear in most documents contribute less to a document’s score than query terms that appear in a more selective set of documents.
- The number of appearances of each query term in the matching document. In general, the more occurrences of query terms within a document, the higher its score is.
- The proximity with which query terms appear in each matching document. In general, query terms that appear in close proximity to each other in a document contribute more to that document’s score than the same terms with more distant occurrences.
- The context in which query terms appear in each matching document. For example, a query term that appears in the title of a document contributes more to the document’s score than the same term that appears in the plain text of the document.

The length of each document and the richness of its vocabulary are also factors in determining its score.

Static ranking

For certain types of documents, you can associate a static ranking factor that increases the importance of those documents in the search results.

When you create a collection, you specify whether you want to associate a static ranking factor with the documents in the collection. For Web content, the number of links to a document from other documents, and the origins of those links, can increase the relevance of that document in the search results.

For documents that include date fields or date metadata, you can use the date of the document to increase its relevance. For example, recent articles in NNTP news groups might be more relevant than older articles. If a data source includes multiple date values, you can choose which one is most important for determining the relevance of documents in the data source.

If you use static ranking with a collection, ensure that you do not mix data sources that use different ranking types in the same collection. For example, if you want to use the links to a document as the static ranking factor, ensure that the collection contains only Web documents. Search quality can be degraded when sources with different ranking models are combined in the same collection.

You must also ensure that the documents contain fields and values that enable static ranking to be applied. For example, if you specify that you want to use the document date as a factor for ranking documents in the collection, and the documents do not contain date fields or attributes, the search quality might be degraded.

Related tasks

“Migrating a collection from WebSphere Portal” on page 209

To migrate collections from WebSphere Portal to enterprise search, prepare the collections in WebSphere Portal, then use the migration wizard to migrate them.

Custom boost word dictionaries

To improve the quality of the search results, you can influence how documents are ranked in the search results by creating a custom boost word dictionary.

If a query specifies a word that is in the boost word dictionary, the importance of documents that contain that word will be increased or decreased according to the boost factor that is configured for the word in the dictionary.

The boost factors range from -10 to 10. During query processing, the search servers increase the importance of documents that contain words with positive boost factors, and decrease the importance of documents that contain words with negative boost factors.

For example, a document that matches query terms with high boost factors is ranked higher than it would be if the boost factor was not applied. (Only query terms that are boosted contribute to the document’s rank in the search results, and the boost factor is only one factor that contributes to the document’s score.)

When you create the dictionary, you can assign the same boost factor to any number of words. The dictionary can contain a single word term or a multiple word term (multiple word terms are matched as a phrase).

If a word that is weighted by a boost value is specified in a query that uses the OR operator (for example: this | that), a weighted average is calculated for the query terms. The resulting aggregated score is used for all occurrences of the OR query operands (different scores are not calculated for different OR query operands).

Boosting that is based on boost word dictionaries is not supported with fielded query terms. When the query terms are parsed, only the query text, not the field name, is used to calculate the document's score. (To apply boost factors to query terms that occur in fields, you can map field names to boost classes.)

To create a boost word dictionary, an expert in the subject matter of the collection needs to create a boost word list in XML format (or work with an application developer to create the XML file). A tool that is provided with WebSphere II OmniFind Edition must be used to convert the XML file to a binary (.dic) file.

An enterprise search administrator uploads the binary file to the system and assigns it a display name. Collection administrators can select a boost word dictionary to use for searching documents in a collection when they configure search server options for a collection.

Restriction: After you add a custom boost word dictionary to the system, you cannot edit it. To revise the boost words that are available for query processing, you must:

1. Update the source XML file.
2. Convert the XML source to a new dictionary file.
3. Remove the old boost word dictionary from the collections that use it.
4. Delete the old boost word dictionary from the system.
5. Add the new boost word dictionary to the system.
6. Associate the new boost word dictionary with the collections that are to use it.

Related concepts

"Document ranking in enterprise search" on page 149

When a user searches a collection, the search processes return the most relevant results for the terms and conditions of the query.

"Custom boost word dictionaries" in "Text Analysis Integration"

Related tasks

"Creating an XML file for boost words" in "Text Analysis Integration"

"Creating a boost word dictionary" in "Text Analysis Integration"

Adding boost word dictionaries to the system


If you create custom boost word dictionaries, you must associate the dictionaries with the enterprise search system. You can later choose which boost word dictionary you want to use for searching a collection.

Before you begin

To add custom boost word dictionaries to the system, you must be a member of the enterprise search administrator role.

Procedure

To associate custom boost words with the enterprise search system:

1. Click **System** to open the System view.
2. Click  **Edit** to change to the system editing view.
3. On the Search page, click **Configure boost word dictionaries**.
4. On the Configure Boost Word Dictionaries page, click **Add Boost Word Dictionary**.
5. On the Add a Boost Word Dictionary page, type a unique display name for the dictionary and optionally type a description.
6. Specify the location of the .dic file. If the file is smaller than 8 MB, the file can be on your system. If the file is larger than 8 MB, the file must be on the index server.
7. Click **OK**. Your custom boost word dictionary is added to the enterprise search system and becomes available for searching collections.

Associating a boost word dictionary with a collection

If boost word dictionaries are associated with the enterprise search system, you can select one to use when searching a collection. If a query term matches a term in the dictionary, then the importance of documents that contain that term will be raised or lowered according to the boost factor that is assigned to the term in the dictionary.

Before you begin

To select a boost word dictionary for a collection, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

Procedure

To associate a boost word dictionary with a collection:

1. Edit a collection, select the Search page, and click **Configure search server options**.
2. In the **Boost word dictionary name** field on the Search Server Options page, select the boost word dictionary that you want to use when users query this collection.

The list of available dictionaries includes all boost word dictionaries that were added to the enterprise search system.

3. Click **OK**.

Document ranking that is based on URI patterns

You can increase or decrease the importance of documents by assigning boost factors to URI patterns.

All documents are assigned a default static ranking score when they are added to the index. The default score varies according to whether static ranking was enabled for the collection and, if so, the static ranking type (by document date or, for Web documents, the number of other documents that link to it).

You can influence a document's relative importance by assigning boost factors to URI patterns. The boost factor is used with the default static ranking score and other factors to determine the document's final static score.

The order of the URI patterns that you configure is important. The index server evaluates the URI patterns in the order that they are listed when it computes the value of each document in a collection. For each URI:

1. The index server scans through the URI patterns sequentially.
2. When the index server finds the first URI pattern that matches a document in the index, it applies the boost factor that is configured for that URI pattern to the document.
3. If a document cannot be matched to a URI pattern, then the default static ranking score is used.

After you configure a boost factor for a URI pattern, you must position the URI pattern in the order that you want the index server to scan it.

Related concepts

“Document ranking in enterprise search” on page 149

When a user searches a collection, the search processes return the most relevant results for the terms and conditions of the query.

Influencing the scores of documents that match URI patterns

You can increase or decrease the importance of documents that match a URI pattern by applying a boost factor to the default static ranking score.

Before you begin

To influence the importance of documents that match a URI pattern, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

About this task

The boost factor that you configure is used with the default static ranking score to calculate a new static score for all documents that match the specified URI pattern.

The boost factors boost only static scores, and the factors are just one contributor to the calculation that determines the final rank of a document. For example, if a document has a high number of links to it (which results in a high initial score), then a document that has no links to it will always be ranked lower.

Procedure

To influence the scores of documents that match a URI pattern:

1. Edit a collection, select the Index page, and click **Influence scores by URI pattern matching**.
2. On the Influence Scores by URI Pattern Matching page, click **Add URI Pattern**.
3. Type a URI pattern for documents that you want to increase or decrease the importance of in the search results. For example:

```
http://domain.org/hr/*  
db2://*ROI*  
*/afs/*
```
4. Type a value between -10 and 10 for the boost factor. The final static score for all documents that match the URI pattern will be calculated on the basis of this boost factor.
5. Click **OK**.

6. On the Influence Scores by URI Pattern Matching page, position the new URI pattern in the order that you want the index server to scan it.

The index server calculates the static ranking scores in the order that you list the URIs. For best results, list the more specific URIs first. In the following example, the /forms subdirectory matches the `http://www.ibm.com/hr/*` URI pattern. To ensure that scores for documents in the /forms subdirectory are calculated correctly, list the URI pattern for the /forms subdirectory first:

```
http://www.ibm.com/hr/forms/* 8
http://www.ibm.com/hr/* -2
```

7. To change the URI pattern or boost factor, select the URI pattern and click

 **Edit.**

8. To remove a URI pattern from the list, select the URI pattern and click

 **Remove.**

9. To apply the boost factors to documents that were previously indexed, reorganize the index.

Related concepts

“Document ranking in enterprise search” on page 149

When a user searches a collection, the search processes return the most relevant results for the terms and conditions of the query.

Related reference

“URI formats in an enterprise search index” on page 89

The uniform resource identifier (URI) of each document in an enterprise search index indicates the type of crawler that added the document to the collection.

Document ranking that is based on boost classes

By mapping fields to boost classes, you can influence how documents are ranked in the search results.

When documents are parsed, the parser assigns *boost classes* to document tokens, according to the fields that the tokens belong to. These boost classes are included in the index and are used during query evaluation to calculate scores that contribute to how result documents are ranked.

To influence how the scores are calculated, you can configure numeric boost factors for the boost classes. If a query term matches a token in a field that is mapped to a boost class, the contribution of this occurrence of the token influences the total score of the document. The score is calculated by applying the boost factor that is configured in the boost class.

For example, you might want to boost the scores of title fields. When a query term occurs in the title, the occurrence has a high contribution to the document score and helps the document to be ranked higher in the search results.

To influence document ranking, you use the enterprise search administration console to specify boost factors for boost classes and to map fields to the boost classes. Sixteen boost classes are preconfigured for enterprise search. Eight of the boost classes are designed to be used with content fields, and the other eight boost classes are designed for metadata fields. You can edit the scores that are associated with the default boost classes, and you can associate different or additional fields with the boost classes.

If you change the field mappings, you must crawl and parse documents again so that your changes can be applied to documents that were previously indexed. If

you change the factors that are specified for a boost class, monitor the search servers, and stop and restart the search server processes for your changes to become effective.

Duplicate document detection

When you map a field to a boost class, you must specify whether the field is used to detect duplicate documents:

- If a field is used to detect duplicate documents, then the field is considered to be a content field, and only the boost classes that are designed for content fields are available for selection.
- If the field is not used to detect duplicate documents, then the field is considered to be a metadata field, and only the boost classes that are designed for metadata fields are available for selection. In this case, two documents that are the same in all ways but the specified field are considered duplicates of each other.

High and low recall values

When a query is evaluated, the search processes estimate the number of result documents that will be returned. Thresholds determine whether a query is considered to have low recall value or high recall value:

Low recall

If the estimated number of result documents is below the low threshold, the query is considered a low recall query.

High recall

If the estimated number of result documents is above the high threshold, the query is considered a high recall query.

Mixture

If the estimated number of documents is between the two thresholds, the recall value of the query is a mixture of the two thresholds.

Each boost class specifies boost factors that are associated with low recall queries and high recall queries during query processing. The low boost factor influences the relative importance of low recall queries, and the high boost factor influences the relative importance of high recall queries. A mixture of the two boost factors influences the relative importance of queries that have a mixed recall value.

The values of the boost factors control the relative importance of each occurrence of a query term in a document. Each occurrence of a query term in a document is counted according to the corresponding boost factor.

When you configure boost classes for a collection, you can edit the default boost factors. For example, you might specify boost factors to ensure that query terms that occur in title fields count five times more than query terms that occur in regular text.

Related concepts

“Document ranking in enterprise search” on page 149

When a user searches a collection, the search processes return the most relevant results for the terms and conditions of the query.

Mapping fields to boost classes

You can influence the relative importance of fields by mapping field names to boost classes.

Before you begin

To map fields to boost classes, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

About this task

The system uses the boost factor to influence the ranking of documents that include query terms within the fields that are mapped to boost classes.

Enterprise search reserves some field names so that it can calculate scores for fields and text that do not have any other defining characteristics (for example, a field that is not a title field and regular text that is not emphasized). You can map other fields to the boost classes that the reserved fields use, but you cannot edit or delete the reserved fields.

Procedure



To map fields to boost classes:

1. Edit a collection, select the Parse page, and click **Map fields to boost classes**.
2. On the Map Fields to Boost Classes page, click **Add Field**.
3. On the Add a Field to a Boost Class page, type the name of the field that you want to map to a boost class.

You can specify the name of a field that exists in a crawled source or in an external source, the name of a field that is mapped from an XML element, the name of a field that is mapped from an HTML metadata element, or one of the predefined field names.

4. Specify whether the field is used for duplicate document detection. If you select the check box, the list of available boost classes contains classes that apply to content fields. If you clear the check box, the list of available boost classes contains classes that apply to metadata fields.
5. Select a boost class and click **OK**.

The field that you added is displayed on the Map Fields to Boost Classes page. You can select an option to edit the boost class and configure different boost factors for determining the scores of documents that contain this field.

6. To change whether a field is used for duplicate document detection or to map the field to a different boost class, click  **Edit**. (You cannot edit fields that are reserved for use by enterprise search.)
7. To remove a field from a boost class, click  **Remove**. (You cannot remove fields that are reserved for use by enterprise search.)
8. To apply changes to documents that were previously indexed, crawl, parse, and index the documents again.

Related concepts

“Document ranking in enterprise search” on page 149

When a user searches a collection, the search processes return the most relevant results for the terms and conditions of the query.

Configuring boost factors for boost classes

The boost factors that you configure for boost classes represent your estimate of how relevant the presence of particular fields in result documents are to a query. Boost classes with high boost factors can increase the importance of result documents that contain fields that are mapped to the boost class.

Before you begin


To configure boost factors for boost classes, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

About this task

The system uses the boost factors that are configured for a boost class, the default static ranking score, and other factors to calculate a new score for result documents that contain fields that are mapped to the boost class.

Procedure

To configure boost factors for boost classes:

1. Edit a collection, select the Parse page, and click **Map fields to boost classes**.
2. On the Map Fields to Boost Classes page, click **Edit Boost Classes**.
3. On the Boost Classes page, locate the boost class that you want to change and click  **Edit**.
4. On the Edit a Boost Class page, specify new values for the high and low boost factors. You can type the same value for both factors.
5. Click **OK**.
6. For the changes to become effective, monitor the search servers and select the icons for stopping and restarting the search processes. When users submit queries, the relative importance of result documents that contain fields that are mapped to this boost class will be determined by the new boost factors.

Related concepts

“Document ranking in enterprise search” on page 149

When a user searches a collection, the search processes return the most relevant results for the terms and conditions of the query.

Default boost class values

Enterprise search provides 16 boost classes that you can use to influence how documents are ranked in the search results.

To calculate scores for fields and text that do not have any other defining characteristics, the following fields are reserved for use by enterprise search:

```
es_special_field.regular_text  
es_special_field.default_field
```

You can map other fields to the boost classes that the reserved fields use, but you cannot edit or delete the reserved fields.

For all other fields, you can edit the boost factors that the system uses to calculate a document's rank. You can also map any number of fields to any of the boost classes, including the boost classes that are used by the reserved fields.

The following table lists the boost class names, the default boost factors for queries that have low recall value, the default boost factors for queries that have high recall value, and the names of predefined fields that are mapped to the boost classes in the default configuration.

The default boost factors vary according to the static ranking method that was selected for the collection when the collection was created. The options include no static ranking, a rank that is determined by the number of links to a document (for Web sources), or a rank that is determined by the document date.

Table 3. Default boost class values

| Default low and high boost factors | | | | |
|------------------------------------|-------------------|--------------------|-------------------|---|
| Boost class name | No static ranking | Document links | Document date | Predefined field mappings |
| Content class A | Low: 4 High: 2 | Low: 6 High: 1 | Low: 4 High: 2 | es_special_field.regular_text |
| Content class B | Low: 5 High: 4 | Low: 7 High: 3 | Low: 5 High: 4 | es_special_field.html_emphasized_text Includes these HTML elements: b, big, caption, dfn, em, h4, h5, h6, strong |
| Content class C | Low: 7 High: 4 | Low: 9 High: 3 | Low: 7 High: 4 | es_special_field.html_headers Includes these HTML elements: h1, h2, h3 |
| Content class D | Low: 2 High: 5 | Low: 1 High: 5 | Low: 2 High: 5 | title |
| Content class E | Low: 1 High: 1 | Low: 5 High: 10 | Low: 1 High: 1 | es_special_field.anchor |
| Content class F | Low: 1 High: 1 | Low: 1 High: 1 | Low: 1 High: 1 | es_special_field.anchor_same_dir |
| Content class G | Low: 1 High: 1 | Low: 1 High: 1 | Low: 1 High: 1 | es_special_field.anchor_same_host |
| Content class H | Low: 1 High: 1 | Low: 1 High: 1 | Low: 1 High: 1 | |
| Metadata class A | Low: 1 High: 1 | Low: 1 High: 1 | Low: 1 High: 1 | es_special_field.default_field |
| Metadata class B | Low: 1 High: 1 | Low: 1 High: 1 | Low: 1 High: 1 | |
| Metadata class C | Low: 1 High: 1 | Low: 1 High: 1 | Low: 1 High: 1 | |
| Metadata class D | Low: 1 High: 1 | Low: 1 High: 1 | Low: 1 High: 1 | |
| Metadata class E | Low: 1 High: 1 | Low: 5 High: 1 | Low: 1 High: 1 | keywords |
| Metadata class F | Low: 1 High: 1 | Low: 3 High: 1 | Low: 1 High: 1 | es_special_field.urlhost |
| Metadata class G | Low: 1 High: 1 | Low: 1 High: 1 | Low: 1 High: 1 | es_special_field.urlpath |
| Metadata class H | Low: 1 High: 1 | Low: 1 High: 1 | Low: 1 High: 1 | description |

Related concepts

| “Document ranking in enterprise search” on page 149
| When a user searches a collection, the search processes return the most relevant
| results for the terms and conditions of the query.

Search applications for enterprise search

A search application enables you to search collections and external sources in the enterprise search system. You can create any number of search applications, and a single search application can search any number of collections and external sources.

Sample search application

The sample search application demonstrates many of the search and retrieval functions that are available for enterprise search. The sample application is also a working example that demonstrates how you can use the IBM Search and Index API (SI-API) to build interactive, custom search applications that reflect the goals of your enterprise.

Unless you change properties in the default configuration file, the sample search application enables you to search all active collections and external sources in your enterprise search system. You can use the sample search application to test new collections and external sources before you make them available to users.

The sample search application is automatically associated with all collections and external sources. In a production environment, enterprise search administrators control which search applications are allowed to search various collections.

Custom search applications

You can run the search applications that you create as stand-alone Web applications in an IBM WebSphere Application Server environment, or you can launch them as portlets in an IBM WebSphere Portal environment. By using the Search and Index API, you can design search applications that, like the sample search application, work seamlessly in both environments.

Related concepts

"Indexed options for searching documents" on page 129

When you configure options for searching crawled data or external sources, or when you map XML and HTML metadata elements to search fields, you specify how the documents can be searched and shown in the search results.

"Security with search application IDs" on page 188

To provide collection-level security, you specify which search applications can search each collection and external source.

"Document-level security" on page 189

If security is enabled for a collection when it is created, you can configure document-level security controls. Document-level security ensures that users who search collections are able to access only the documents that they are allowed to see.

"Document-level security with the Portal Search Engine" on page 201

You can use the IBM WebSphere Portal Search Engine to enforce document-level security when users search enterprise search collections.

"Search and index API overview" in *"Programming Guide and API Reference for Enterprise Search"*

"Query syntax" in *"Programming Guide and API Reference for Enterprise Search"*

Associating search applications with collections

Before you can use a new search application, you must associate it with the collections that it can search.

Before you begin

To associate search applications with the collections that they can search, you must be a member of the enterprise search administrator role.

Procedure

To associate a search application with one or more collections:

1. Click **Security** in the toolbar of the administration console.
2. On the Search Applications page, click **Configure search applications**.
3. On the Configure Search Applications page, click **Add Search Application**.
4. Type the name of the search application.
5. Select the collections that the application can search:
 - Click **All collections and external sources** if you want the search application to access all of the collections that you add to the system.
 - Click **Specific collections and external sources** if you want the search application to access only the collections that you specify.
When you select this option, a list of collection names and external source names is displayed. Select the **Select** check box for each collection that the application can search.
6. Click **OK**.

Sample search application functions

The sample search application for enterprise search demonstrates most of the search functions that you can build into your custom search applications.

You can use the sample search application to search one, several, or all collections and external sources at a time. Unless the default application properties are modified, you can use this application to search all of the collections and external sources in the enterprise search system.

Query functions

With these functions you can:

- Specify simple, free-text queries.
- Specify more complex queries to improve the precision of search results. For example, you can search specific fields in a document or use query syntax to search for documents that include or exclude specific words and phrases.
- Specify which collections and external sources you want to search.
- Search specific source types or all source types.
- Search specific types of documents. For example, you can search only Microsoft Word documents or only portable document format (PDF) documents.
- Specify which language your query terms are in. You can also specify the languages of the documents that you want to search.

- Search specific subsets of a collection. For example, a search application can limit your view to a predefined range of documents (a scope), or you can submit a query that searches only the documents that belong to a named category.
- Expand the query to include synonyms of your query terms. If a synonym dictionary is associated with the collection, documents that contain synonyms of your query terms are returned in the search results.

Search result functions

With these functions you can:

- See the search results that match your query terms.
- Control how many result documents appear on each page, and browse forward and backward through the result set.
- Hide and display details about the result documents. For example, you can view brief descriptions of the documents or view details such as the names of fields in each result document.
- Collapse documents from the same source. For example, if one source returns 100 documents, the two most relevant documents are shown grouped together in the result set. You can see the remaining 98 documents by selecting an option to view more documents from the same source.
- Sort documents by relevance, by document date in ascending order, or by document date in descending order.
- Be prompted with suggestions for spelling corrections if possibly misspelled words are detected in the query string.
- View information about the categories that a result document belongs to (if the collection uses categories), and browse only the documents that belong to a specific category.
- Specify additional query terms to search within the search results.

Document retrieval functions

With these functions you can:

- Retrieve documents by clicking the document URI. If client applications are available, you might also be able to view a result document with a native viewer. For example, if a Notes client application is installed, you can use that application to view documents from a Lotus Notes database.
If document-level security is configured for a crawler, only users who are authorized to access the secure content can retrieve documents.
- Retrieve documents by clicking quick links. A quick link associates keywords with URIs. If a query includes the specified keywords, the associated URIs (which were predetermined to be highly relevant for those keywords) appear at the top of the search results.

Sample search application properties

You can edit the config.properties file for the sample search application to specify options for your environment, change the appearance of the application, and control the options that are available to users after they start the search application.

Environment parameters

You can specify options that control the operation of the search application.

| **applicationName**

| Specifies the name of a valid search application. The default value is
| Default.

| Change the default value if want to use a different search application as
| the default application.

| **Tip:** When the application name is Default, you can use the sample search
| application to search all collections and external sources with a single
| query.

| **hostname**

| Specifies the fully qualified host name of the Web server that is configured
| to support your WebSphere Application Server instance. The default value
| is localhost.

| To ensure that the search application works correctly, change the default
| value to the fully qualified host name that WebSphere Application Server is
| configured to use. For example, if the computer host name is MyMachine
| and the Web server is configured to use www.ibm.com as its host name,
| specify www.ibm.com.

| **port**

| Specifies the port number of the Web server that is configured to support
| your WebSphere Application Server instance. The default value is 80.
| Change the default value only if you change the port for your Web server
| instance.

| **timeout**

| Specifies the number of seconds to wait for a response from the search
| server before a search request times out. This number must be an integer
| (such as 60, not 60.5 or sixty). If you do not specify a timeout value, the
| default value is 30 seconds.

| **username**

| Specifies a user name that enables enterprise search to enforce user
| authentication, and authenticate users with WebSphere Application Server,
| when users submit search requests. This field is used only if you enabled
| global security in WebSphere Application Server.

| **password**

| Specifies the password for the specified user name.

| **filter**

| Specifies a class that is to be used to retrieve documents that are listed in
| the search results. The default class is
| com.ibm.es.api.filters.SetDocumentURIFilterFetch. Change this value
| only if you have a custom class that you want to use for retrieving
| documents instead.

| **logging.level**

| Specifies the amount of detail to log:

| **OFF** No messages are logged.

| **INFO** Informational messages are logged.

| **FINE** Error messages are logged.

| **ALL** All messages are logged.

Source type icons

You can customize the images that represent the type of data source that a result document belongs to. The following source type icons, which identify the crawlers and external sources that are supported when WebSphere Information Integrator OmniFind Edition is installed, are predefined in the config.properties file.



documentSource.vbr.icon

Specifies the path and name of an image file that indicates that the document was crawled by a Content Edition crawler. The default icon is /images/sourceVBR.gif.



documentSource.db2.icon

Specifies the path and name of an image file that indicates that the document was crawled by a DB2 crawler. The default icon is /images/sourceDB2.gif.



documentSource.cm.icon

Specifies the path and name of an image file that indicates that the document was crawled by a DB2 Content Manager crawler. The default icon is /images/sourceCM.gif.



documentSource.dominodoc.icon

Specifies the path and name of an image file that indicates that the document was crawled by a Domino Document Manager crawler. The default icon is /images/sourceDominoDoc.gif.



documentSource.exchange.icon

Specifies the path and name of an image file that indicates that the document was crawled by an Exchange Server crawler. The default icon is /images/sourceExchange.gif.



documentSource.nntp.icon

Specifies the path and name of an image file that indicates that the document was crawled by an NNTP crawler. The default icon is /images/sourceNNTP.gif.



documentSource.notes.icon

Specifies the path and name of an image file that indicates that the document was crawled by a Notes crawler. The default icon is /images/sourceNotes.gif.



documentSource.quickplace.icon

Specifies the path and name of an image file that indicates that the document was crawled by a QuickPlace crawler. The default icon is /images/sourceWorkplace.gif.



documentSource.unixfs.icon

Specifies the path and name of an image file that indicates that the document was crawled by a UNIX file system crawler. The default icon is /images/sourceUnixFS.gif.



documentSource.web.icon

Specifies the path and name of an image file that indicates that the document was crawled by a Web crawler. The default icon is /images/sourceWeb.gif.



documentSource.wps.icon

Specifies the path and name of an image file that indicates that the document was crawled by a WebSphere Portal crawler. The default icon is /images/sourceWPS.gif.



documentSource.winfs.icon

Specifies the path and name of an image file that indicates that the document was crawled by a Windows file system crawler. The default icon is /images/sourceWindowsFS.gif.



documentSource.ldap.icon

Specifies the path and name of an image file that indicates that the document belongs to an external source that was created for an LDAP server. The default icon is /images/sourceLDAP.gif.



documentSource.jdbc.icon

Specifies the path and name of an image file that indicates that the document belongs to an external source that was created for a Java Database Connectivity (JDBC) database table. The default icon is /images/sourceJDBC.gif.

Client viewer icons

Result documents can be displayed in the Web browser. If a client application that is native to the result document type is available, the document can also be displayed with the client viewer. You can customize the images that represent the type of client viewer that is used to display the document. In the following example, the Lotus Notes icon is used to indicate that the document can be displayed with a Lotus Notes client application:

```
client.notes.icon=/images/notes.gif
```

In the search results, the icon and the link to the client application are displayed as follows:



Client Viewer

Document fields

For data source types that have fields, you can control which fields are displayed in the result documents.

fields.URI prefix=*space_separated_list_of_field_names*

You must escape the colon character (:) in the URI prefix by preceding it with a backward slash character (\). To continue a list of field names to another line, end the preceding line with a backward slash character (\). For example:


```

fields.db2\://=databasename tablename
fields.domino\://=servername databasename databasetitle filename extension \
createddate modifieddate
fields.cm\://=servername itemtypename createddate modifieddate mimetype
fields.file\://=directory filename extension modifieddate filesize title

```

Field icons

For data source types and documents that have fields, you can customize the images that represent fields. The following field icons are predefined in the config.properties file.



field.icon.databasetitle

Specifies the path and name of an image file that indicates that the field contains the document title. The default icon is /images/notesdb.gif.



field.icon.databasename

Specifies the path and name of an image file that indicates that the field contains the name of the database that the document belongs to. The default icon is /images/db2.gif.



field.icon.tablename

Specifies the path and name of an image file that indicates that the field contains the name of the table that the document belongs to. The default icon is /images/table.gif.



field.icon.directory

Specifies the path and name of an image file that indicates that field contains the name of the directory that the document belongs to. The default icon is /images/closedFolder.gif.

Default field icon

You can specify an image to use when no field icons are configured for fields that are displayed in the search results. The following default field icon is predefined in the config.properties file.



field.defaultIcon

Specifies the path and name of an image file that is the default icon for fields in the search results. The default icon is /images/database.gif.

Date fields

You can specify which fields are date fields. The field names that you specify here are formatted like date data in the search results. The format of the date matches the locale settings in the Web browser.

date.fields=space_separated_list_of_field_names

To continue a list of field names to another line, end the preceding line with a backward slash character (\).

Example:

```
date.fields=modifieddate createddate
```

Document titles

You can specify alternative titles for documents by substituting title text with more meaningful data (that is, you can *clean* the titles). For example, instead of seeing document titles with the uninformative label Slide 1, you can specify that Slide 1 is to be suppressed in the search results. (A more meaningful field, such as the file name, might be used to identify the result document instead.)

You can also specify alternative titles for documents by removing meaningless words from the document titles (that is, you can *truncate* the titles). For example, if a number of result documents begin with Microsoft Word -, you can improve the readability of the search results by suppressing the repetitive text.

titles.clean=comma_separated_list_of_titles

titles.truncatePrefix=comma_separated_list_of_prefixes

The comma-separated lists can contain spaces and other characters except for the comma. To continue a list to another line, end the preceding line with a backward slash character (\).

For example:

```
titles.clean=Slide 1, Layout 1, IBM Software Group Presentation Template, \
untitled, Untitled Document, PowerPoint Presentation, \
(no title for this page)
```

```
titles.truncatePrefix=Microsoft Word -, Microsoft Powerpoint -
```

Default values for user preferences

You can specify default values for the Preferences page in the search application. If a user changes the preferences, the new values are in effect for the user's current session only. The following preferences are predefined in the config.properties file.

preferences.resultsRange=10

Specifies that each page of the search results can list 10 result documents.

preferences.siteCollapsing=Yes

Specifies that if site collapsing rules were configured in the enterprise search administration console, then URIs that match a URI prefix rule are to be collapsed in the search results.

preferences.spellCorrections=Yes

Specifies that suggested spelling corrections are to be displayed when a user submits a query that contains a possibly misspelled word.

preferences.extendedHighlighting=No

Specifies that query terms will not be highlighted in extra fields (such as the document title) in addition to the document summary field.

Default collections and external sources

You can specify which collections and external sources are preselected on the Preferences and Advanced Search pages. Users can edit the default set to search fewer collections and external sources than those that you make available by default. If you restrict the set of collections and external sources here, users can select any collection or external source that is available to the search application when they modify their preferences or advanced search options.

preferences.defaultCollections=*

preferences.defaultCollections=space_separated_list_of_collection_IDs

Specify an asterisk (*) to enable all collections and external sources to be

searched. (The collections and external sources must be associated with the search application in the enterprise search administration console.) This is the default setting in the config.properties file.

To restrict what users will search if they do not modify their preferences or advanced search options, specify the collection IDs for the collections and external sources that you want users to search by default.

For example:

```
preferences.defaultCollections=*  
preferences.defaultCollections=coll_id1 coll_id2
```

Extra information for the search results

You can customize the amount of information that is provided with the search results and control whether users can filter the search results. The following settings are the default settings in the config.properties file.

refreshButton.show=false

Controls whether a **Refresh** button appears and the ability to refresh the search application. If you set this option to true, users can refresh the list of collections and external sources that are available to search (for example, if the search application was associated with additional collections or external sources in the enterprise search administration console).

You might want to show the **Refresh** button when you test changes that you make to the config.properties file. After you save your changes, you can click **Refresh** to see how the changes affect the search application. Without the **Refresh** button, you must restart the ESSearchApplication enterprise application in WebSphere Application Server before the changes become effective.

If no collections or external sources are available to search (for example, if the wrong host name is specified, the search servers were not started, or the ESSearchServer enterprise application was not started in WebSphere Application Server), then the **Refresh** button is displayed automatically to help when you troubleshoot the problem.

extraMessages.show=false

Controls the display of an area at the bottom of the search results where warning and informational messages are displayed. Error messages are always displayed at the top of the page. Set this option to true if you want to see additional messages.

builtQueryString.show=false

Controls the display of the fully expanded query syntax in an area that precedes the list of result documents. Set this option to true if you want to see the actual query that was processed.

refineResults.show=true

Controls whether users can refine the search results by specifying additional query terms. If you set this option to true, a query box with the label **Search within results** is displayed at the bottom of the search results page.

filter.showOnTwoLines=true

Controls whether the options for filtering results by source type and filtering results by file type are displayed on one or two lines in the search

results. (While viewing search results, users can select a source type and select a file type to see only the result documents that match the selected filters.)

To maximize the amount of space that is available for the display of search results, set this property to false. To improve the readability of the filters, especially if the available filters extend beyond one line, you might want to set this property to true so that each filter is displayed on a separate line.

Custom banner and logo

You can customize the images that display in the banner area at the top of the search application. For example, you might want to replace the default images for WebSphere II OmniFind Edition with images that reflect your enterprise branding. If you do not want to display a banner, make one or both of these lines comment lines. The banner.icon property identifies a graphic that is displayed on the left side of the banner area. The banner2.icon property identifies a graphic that is displayed on the right side of the banner area.

```
banner.icon=/images/WS_II_OFEdition.gif  
banner2.icon=/images/WS_II_mosaic.gif
```

Custom background image

You can customize the images that display in the background of pages in the search application. For example, you might want to replace the default images for WebSphere II OmniFind Edition with images that reflect your enterprise branding. If you do not want to display a background image on a page, make one or more of these lines comment lines.

```
search.backgroundImage=/images/IIOF_search.gif  
preferences.backgroundImage=/images/IIOF_options.gif  
advanced.backgroundImage=/images/IIOF_advanced.gif  
browse.backgroundImage=/images/IIOF_tree.gif  
myProfile.backgroundImage=/images/IIOF_profile.gif  
logoff.backgroundImage=/images/IIOF_logout.gif
```

Links

The properties in the Links area of the config.properties file enable the names of the search application pages to be shown as links on each page instead of being shown on the toolbar and on pages that have tabs. Viewing links is useful when you run the search application as a portlet and want to minimize the amount of space that is used to display the search application on a portal page.

If you prefer to navigate the search application by selecting options on the toolbar and on pages with tabs, comment out these lines.

Search tabs

The properties in the Search tabs area of the config.properties file specify the names of the Java Server Pages (JSPs) that are used for tabbed pages in the Searches view of the search application (Search, Advanced Search, and Category Tree). Do not edit these pages unless you have experience with Java programming and JSPs.

Examples of how you might customize this area include:

- Directing the search application to custom JSPs that provide a different appearance for the tabbed pages.

- Commenting out the Category Tree entries. For example, if you do not configure categories for your collections, there is no need to show the Category Tree page in the search application.
- Copying the entries for the tabbed pages to the Toolbars area of the config.properties file and commenting out these lines. For example, you might want to show only the toolbar and not show tabbed pages at all.

Toolbars

The properties in the Toolbars area of the config.properties file specify the names of the Java Server Pages (JSPs) that are used for the toolbar in the search application. Do not edit these pages unless you have experience with Java programming and JSPs.

Examples of how you might customize this area include:

- Directing the search application to custom JSPs that provide a different appearance for the toolbar.
- Commenting out toolbar entries for items that you do not want to display. For example, you might not want to include a link to the About page on the toolbar.
- Moving the function for displaying the Advanced Search page from the Search tabs area of the config.properties file so that this option is available only on the toolbar.

Meaningful document type labels

You can improve the readability of the document type filter by mapping the actual document type names to more concise and meaningful terms. The document types that are available to the search applications are defined by the AvailableDocumentTypes class of the Search and Index API (SI-API). For convenience, the available document types are also listed at the end of the config.properties file.

documentType.label=space_separated_list_of_document_types

Specifies the name that is displayed on the document type filter line in the search results, and a list of actual document types that are to be displayed when a user selects the filter.

For example, you might specify the label **html** and map the file extensions and MIME types for various Web documents to that name. When a user clicks **html** to filter the search results, only documents with the specified extensions and MIME types are displayed.

The following document type mappings are predefined in the config.properties file:

```
documentType.html=shtml text/html html xhtml htm
documentType.doc=doc application/msword
documentType.ppt=application/mspowerpoint ppt
documentType.xls=xls application/x-excel application/msexcel \
application/x-msexcel application/excel application/vnd.ms-excel
documentType.xml=xml text/xml
documentType.txt=txt text/plain
documentType.pdf=pdf application/pdf
```

Custom filters

You can specify custom queries to filter the display of result documents.

| **filterCustom.label=query_terms**

| Specifies the name that is displayed on the custom filter line in the search
| results, and a query that refines the search results when a user selects the
| filter. (While viewing search results, users can select a custom filter to see
| only the result documents that match the predefined query.)

| In the following example, the search results are filtered to show only
| documents that belong to the human resources (hr) database:

| `filterCustom.HR_database_only=databasename::hr`

| When a user clicks **HR_database_only** to filter the search results, the query
| `databasename::hr` is processed. When the search results are displayed, only
| documents from the hr database are listed.

| Several custom filters are commented out and provided as examples in the
| `config.properties` file.

Editing the sample search application properties

The sample search application for enterprise search can search all active collections and external sources in your system. You can edit a properties file to specify options for your Web server environment, use a different search application as the default application, or control which options are displayed when the search application is started.

About this task

The installation program deploys a sample search application for enterprise search into IBM WebSphere Application Server on the search servers for enterprise search. To configure this search application, you edit a properties file, `config.properties`, that is deployed with the application.

For your changes to become effective, you must stop and restart the `ESSearchApplication` enterprise application in WebSphere Application Server.

Procedure

To edit the sample search application properties:

1. Edit the `config.properties` file with a standard text editor.

The `config.properties` file is installed in the following location, where `ES_INSTALL_ROOT` is the WebSphere II OmniFind Edition installation directory on the search server:

```
ES_INSTALL_ROOT/installedApps/ESSearchApplication.ear/  
ESSearchApplication.war/WEB-INF/config.properties
```

2. Edit the properties to specify information about your Web server environment and search preferences, then save and close the file. (In the file, the pound sign character (#) indicates a comment line.)
3. Stop and restart the `ESSearchApplication` application:
 - a. On the search server, start the WebSphere Application Server Administrative Console.

You can open the Administrative Console in the following ways:

- Use the Windows **Start** menu to select the program.
- For WebSphere Application Server version 5, open a Web browser and go to `http://hostname:port/admin`, where `hostname` is the host name of the

search server and *port* is the port number for the WebSphere Application Server Administrative Console. Typically, the Administrative Console port is 9090.

- For WebSphere Application Server version 6, open a Web browser and go to `http://hostname:port/ibm/console`, where *hostname* is the host name of the search server and *port* is the port number for the WebSphere Application Server Administrative Console. Typically, the Administrative Console port is 9060.
- b. When you are prompted for a user ID and password, enter a user name and password that is registered in the WebSphere Application Server user registry. If you are using the local operating system as your user registry, you can specify the enterprise search administrator ID and password.
- c. After you log in to the Administrative Console, click **Applications** and then click **Enterprise Applications**.
- d. Select the check box for `ESSearchApplication` and click **Stop**.
- e. After the application stops, select the check box for the application again, and click **Start**.

Accessing the sample search application

You access the sample search application by specifying a URL in a Web browser.

Before you begin

You must configure the sample search application for your Web server environment.

About this task

The sample search application is installed on the search servers for enterprise search. You can use this application as provided to test collections and external sources before you make them available to users. You can also use the application as a model for creating your own search applications.

Procedure

To start the sample search application:

1. Type the URL for the search application in a Web browser. For example:

`http://SearchServer.com/ESSearchApplication/`

SearchServer.com is the host name of the search server.

If your Web server is not configured to use port 80, you also need to specify the correct port number. For example:

`http://SearchServer.com:9080/ESSearchApplication/`

2. If security is enabled in WebSphere Application Server, log in to the application with a valid user ID and password.

If any of the collections that are available to the search application are enabled for security, and if the secure collections include crawlers that are configured to validate user credentials during query processing, you can configure a user profile. On the My Profile page, specify credentials for accessing the secure domains. You can then search those domains without being challenged to log in to them.

3. On the Search page, select the collections and external sources that you want to search and submit a query.

Enabling security for the sample search application

If you enable global security in WebSphere Application Server and want to use the sample search application to search secure collections, you must change configuration settings in the sample application and in WebSphere Application Server.

Before you begin

- You must be a member of the enterprise search administrator role.
- You must enable global security in WebSphere Application Server. If you installed WebSphere II OmniFind Edition as a multiple server configuration, then enable global security on the search servers. See the WebSphere Application Server documentation for instructions on how to enable global security.
- If you choose Lightweight Directory Access Protocol (LDAP) for the user registry in WebSphere Application Server, then you must add the enterprise search administrator ID and password to your LDAP registry (this ID and password were specified when WebSphere II OmniFind Edition was installed).

For example, if the enterprise search administrator ID is adminUser, then the user entry in the LDAP registry might be
uid=adminUser,ou=Employees,o=IBM,c=US. See the LDAP server documentation for instructions.

Procedure

To enable security for the sample application for enterprise search:

1. Update search application properties in the administration console:
 - a. Log in to the enterprise search administration console as a user with enterprise search administrator privileges.
 - b. Click **Security** on the toolbar.
 - c. On the Search Applications page, click **Configure search applications**.
 - d. Click **Add Search Application** and, in the **Search application name** field, type the enterprise search administrator ID that was specified when WebSphere II OmniFind Edition was installed.
 - e. Ensure that **All collections and external sources** is selected, and click **OK**.
2. Edit the config.properties file:
 - a. If you are using UNIX, open a console window. If you are using Microsoft Windows, open a command prompt window.
 - b. Change to the WEB-INF directory for the sample search application. The following examples are shown on two lines for readability; specify the command on a single line:
UNIX:

```
cd $ES_INSTALL_ROOT/installedApps/ESSearchApplication.ear/  
ESSearchApplication.war/WEB-INF
```


Windows:

```
cd %ES_INSTALL_ROOT%\installedApps\ESSearchApplication.ear\  
ESSearchApplication.war\WEB-INF
```
 - c. Use a text editor to edit the config.properties file.
 - d. Change the username property to the name of a valid WebSphere Application Server user.
 - e. Change the password property to the password for the specified user.
 - f. Save and exit the file.

3. Restart the ESSearchApplication application in WebSphere Application Server:
 - a. On the search server, start the WebSphere Application Server Administrative Console.

You can open the Administrative Console in the following ways:

 - Use the Windows **Start** menu to select the program.
 - For WebSphere Application Server version 5, open a Web browser and go to `http://hostname:port/admin`, where *hostname* is the host name of the search server and *port* is the port number for the WebSphere Application Server Administrative Console. Typically, the Administrative Console port is 9090.
 - For WebSphere Application Server version 6, open a Web browser and go to `http://hostname:port/ibm/console`, where *hostname* is the host name of the search server and *port* is the port number for the WebSphere Application Server Administrative Console. Typically, the Administrative Console port is 9060.
 - b. When you are prompted for a user ID and password, enter the administrator ID and password that were specified when global security was enabled in WebSphere Application Server.
 - c. After you log in to the Administrative Console, click **Applications** and then click **Enterprise Applications**.
 - d. Select the check box for ESSearchApplication and click **Stop**.
 - e. After the application stops, select the check box for ESSearchApplication again, and click **Start**.

Enterprise search external sources

An *external source* is a data source that you enable for searching with an enterprise search application without the need to crawl, parse, or index documents in the data source.

You can search the following types of data sources as external sources:

- Java database connectivity (JDBC) database tables (IBM DB2 Universal Database (DB2 UDB) and Oracle databases only). A separate external source is created for each table in a JDBC database.
- Lightweight Directory Access Protocol (LDAP) servers. One external source is created for each LDAP server.

After you configure information about an external source, you must associate it with at least one search application. Users can then search the external source at the same time that they query collections that were created by crawling, parsing, and indexing data for enterprise search.

Related concepts

"Search and index API federators" in "Programming Guide and API Reference for Enterprise Search"

Adding external sources to the system

When you add an external source to the enterprise search system, you specify the type of source that you want to add. A wizard helps you specify information about the data source and how it can be searched.

Before you begin

To add an external source to the system, you must be a member of the enterprise search administrator role.

Restrictions

To search an Oracle database as an external source, the Oracle client program must be installed on the search servers for enterprise search.

About this task

When you add information about an external source to the system, you enable users to query the source with an enterprise search application. You can enable Lightweight Directory Access Protocol (LDAP) servers and Java database connectivity (JDBC) database tables to be searched.

When you configure an LDAP server, a wizard helps you specify information that enables the system to connect to the server and specify options for how the server is to be searched.

When you configure a JDBC database, a wizard helps you specify information that enables the system to connect to the database, select the tables that you want to

enable for searching, and specify options for how data in the tables is to be searched. A separately searchable external source is created for each table that you add to the system.

Procedure

To add an external source to the system:

1. Click **External Sources** to open the External Sources view.
2. Click **Add External Source**.
3. Select the type of external source that you want to add, either LDAP server or JDBC database.
4. Click **Next** to begin configuring the external source.

A wizard for the type of source that you are creating opens. Follow the wizard prompts to configure the external source. Click **Help** on any page in the wizard to learn more about the options that you can specify.

The following default JDBC driver names and locations might help you when you configure connection information for DB2 Universal Database (DB2 UDB) and Oracle databases:

DB2: Legacy JDBC Driver

Driver name: `COM.ibm.db2.jdbc.app.DB2Driver`

Location: `db2_install_root/java/db2java.zip`

DB2: Universal JDBC Driver

Driver name: `com.ibm.db2.jcc.DB2Driver`

Location:

`db2_install_root/java/db2jcc.jar`

`db2_install_root/java/db2jcc_license_cu.jar`

Oracle

Driver name: `oracle.jdbc.driver.OracleDriver`

Location:

`oracle_home/jdbc/lib/classes12.zip`

`oracle_home/jdbc/lib/nls_charset12.zip`

5. After you specify options for searching the external source, click **Finish**.
Your new external source is listed on the External Sources view with other external sources that were added to the system.

Related concepts

"Search and index API federators" in "Programming Guide and API Reference for Enterprise Search"

Associating search applications with external sources

Before you can search an external source, you must associate at least one search application with it.

Before you begin

To associate search applications with the external sources that they can search, you must be a member of the enterprise search administrator role.

Procedure

To associate a search application with one or more external sources:

1. Click **Security** in the toolbar of the administration console.

2. On the Search Applications page, click **Configure search applications**.
3. On the Configure Search Applications page, click **Add Search Application**.
4. Type the name of the search application.
5. Select the external sources that the application can search:
 - Click **All collections and external sources** if you want the search application to access all of the external sources that you add to the system.
 - Click **Specific collections and external sources** if you want the search application to access only the external sources that you specify.
When you select this option, a list of collection names and external source names is displayed. Select the **Select** check box for each external source that the application can search.
6. Click **OK**.

Related concepts

"Search and index API federators" in "Programming Guide and API Reference for Enterprise Search"

Enterprise search security

Security mechanisms in enterprise search enable you to protect sources from unauthorized searching and restrict administrative functions to specific users.

With enterprise search, users can search a wide range of data sources. To ensure that only users who are authorized to access content do so, and to ensure that only authorized users are able to access the administration console, enterprise search coordinates and enforces security at several levels.

Web server

The first level of security is the Web server. If you enable global security in WebSphere Application Server, you can assign users to administrative roles and authenticate users who attempt to administer the system. When a user logs in to the administration console, only the functions and collections that the user is authorized to administer are available to that user.

Search applications can also use the authentication support in WebSphere Application Server to authenticate users who use the search application to search collections.

Collection-level security

When you create a collection, you can enable security at the collection level. You cannot change this setting after the collection is created. If you do not enable collection-level security, you cannot later specify document-level security controls.

When collection-level security is enabled, the global analysis processes apply the following special rules:

- To ensure that the security controls for each document are evaluated, documents with duplicate (or near duplicate) content are indexed independently instead of having their content indexed jointly in a canonical representation.
- The anchor text processing phase of global analysis normally associates text that appears in one document (the source document) with another document (the target document) in which that text does not necessarily appear. This enables the target document to be retrieved by queries that specify text that appears in the source document. This type of anchor text processing presents a security risk if users are allowed to view the target document but not the source document. When collection security is enabled, anchor text in the links of forbidden documents is excluded from the index. A document is returned in the search results only if its own content or metadata matches query.

There is a trade-off between enabling collection security and search quality. Enabling collection security reduces the information that is indexed for each document. A side effect is that fewer results will be found for some queries.

Collection-level security is also available to your search applications through an application ID. To search collections, an enterprise search administrator must associate your search application with the specific collections that it can search. You can then use standard access control mechanisms to permit or deny users access to search applications.

Document-level security

When you configure crawlers for a collection, you can enable document-level security. If you choose this option, the crawler can associate security tokens with each document that it crawls. The security tokens are stored with the documents in the index. For some crawlers, you can also specify that user credentials are to be validated with current access control data (as configured on the native data source) during query processing.

Your search applications can use the security tokens and user credentials to enforce access controls. To ensure that users search and retrieve only the documents that they have permission to access, a search application can include credentials from the logged in user on the queries that it passes to the search servers.

Security for your collections extends beyond the authentication and access control mechanisms that enterprise search can use to protect indexed content. Safeguards also exist to prevent a malicious and unauthorized user from gaining access to data while it is in transit. For example, the search servers use protocols such as the Secure Sockets Layer (SSL), the Secure Shell (SSH), and the Secure Hypertext Transfer Protocol (HTTPS) to communicate with the index server and the search application.

Additional security is provided through encryption. For example, the password for the enterprise search administrator, which is specified when the product is installed, is stored in an encrypted format. Passwords that users specify in user profiles are also stored in an encrypted format.

For increased security, you need to ensure that the server hardware is appropriately isolated and secure from unauthorized intrusion. By installing a firewall, you can protect the enterprise search servers from intrusion through another part of your network. Also ensure that there are no spare open ports on the enterprise search servers. Configure the system so that it listens for requests only on ports that are explicitly assigned to enterprise search activities and applications.

Administrative roles

Enterprise search uses the concept of roles to control access to various functions in the administration console.

During the installation of WebSphere Information Integrator OmniFind Edition (WebSphere II OmniFind Edition), the installer configures a user ID and password for the enterprise search administrator. The first time that you access the administration console, you must log in as this user. If you do not enable global security in WebSphere Application Server, this user ID is the only user ID that you can use to access the enterprise search administration console.

If you enable global security in WebSphere Application Server, you can enroll additional users as enterprise search administrative users. By assigning users to roles, you can restrict access to specific collections and control the functions that each administrative user can do. The user IDs that you assign to administrative roles in enterprise search must exist in a WebSphere Application Server user registry.

When an administrative user logs in, enterprise search authenticates the user ID. Only the collections and functions that the user is allowed to administer are available in the console.

You can enroll users in the following administrative roles:

Enterprise search administrator

These users create collections and have the authority to administer all aspects of your enterprise search system. When you install WebSphere II OmniFind Edition, you specify the user ID and password for the first enterprise search administrative user. After logging in the first time, this user can assign other users to the enterprise search administrator role.

Collection administrator

These users can edit, monitor, and control the operation of specific collections or all collections. These users cannot create collections. Collection administrators can monitor and operate system-level activities only if that authority is granted to them by an enterprise search administrator.

Operator

These users can monitor and control the operation of specific collections or all collections. These users can start and stop collection activities, for example, but they cannot create collections or edit collections. An operator can monitor and operate system-level activities only if that authority is granted to the operator by an enterprise search administrator.

Monitor

These users can monitor specific collections or all collections. These users cannot control operations (such as starting and stopping servers), create collections, or edit collections. A monitor can observe, but not operate, system-level activities only if that authority is granted to the monitor by an enterprise search administrator.

Related tasks

“Logging in to the administration console” on page 18

To administer an enterprise search system, you specify a URL in a Web browser and then log in to the administration console.

“Starting the enterprise search servers” on page 213

To enable users to search a collection, you must start the system processes and then start the servers that crawl, parse, index, and search the collection.

Configuring administrative users

By configuring administrative roles, you can restrict access to collections and control the functions that each administrative user can do.

Before you begin

Before you assign a user to an administrative role, ensure that security is enabled in WebSphere Application Server. Also ensure that the user ID exists in a WebSphere Application Server user registry.

To configure administrative users, you must be a member of the enterprise search administrator role.

Procedure

To assign users to administrative roles:

1. Click **Security** to open the Security view.
2. On the Administrative Roles page, click **Add User**.
3. Type the user ID of the user that you want to enroll and select an appropriate administrative role.
4. If you are not enrolling this user as an enterprise search administrator, specify whether this user can access pages from the **System** toolbar.
For example, you might want to allow some operators or collection administrators to monitor system-level log files.
5. If you are not enrolling this user as an enterprise search administrator, select the collections and external sources that this user can administer.
You can select the check boxes for individual collections and external sources or enable the user to administer all collections and external sources.

Authentication versus access control

To protect content from unauthorized users, and to control access to administrative functions, enterprise search supports user authentication and access control.

Authentication

Authentication is the process by which a system verifies that users are who or what they declare themselves to be. Because access is typically based on the identity of the user who requests the resource, authentication is essential to effective security.

To authenticate users who attempt to access the administration console, enterprise search leverages the authentication support that is provided with WebSphere Application Server.

To authenticate users who search enterprise search collections, your search applications can leverage security in WebSphere Application Server and implement your preferred methods for authenticating user credentials. Typically, user credentials consist of a user ID and a password that are passed to the search application when a user logs in or attempts to access the search application.

User authentication can be implemented in other ways, depending on the resources and protocols that are available in your enterprise. For example, you might force users to identify themselves by using smart cards, by managing digital certificates and a public key infrastructure, or by assigning tickets when users log in to track their authentication state.

Access control

Access control refers to limiting what users can do after they identify themselves and are authenticated. An access control list (ACL) is the most common way in which access to resources is limited. An ACL is a list of user identifications (user names, group names, user roles, and so on). Each user identification is associated with a set of permissions that define the user's rights and privileges.

For example, access controls can allow or deny access to files on a file server and control whether a user who is allowed access can read, create, edit, or delete files on that server.

In enterprise search, all access control depends on whether a user has permission to read data in the index. Depending on how you enable access controls in your search applications and the rules that you specify for the collection and for crawlers when you administer enterprise search, you can:

- Allow all users to search all of the documents in a collection.
- Allow all users to search all of the documents that were crawled by certain crawlers, and restrict access to documents that were crawled by other crawlers.
- Allow specific users to search specific documents. For example, when you specify the databases that you want to crawl with a Notes crawler, you can specify options that enable some users to search the documents in certain views and folders, and prevent other users from searching the documents.

Disabling security for an enterprise application in WebSphere Application Server

To control which WebSphere II OmniFind Edition activities require user authentication, you can disable global security for individual enterprise applications in WebSphere Application Server.

About this task

The WebSphere II OmniFind Edition installation program deploys three enterprise applications to WebSphere Application Server:

- The ESAdmin application contains the interface for the enterprise search administration console.
- The ESSearchApplication application contains the interface for the sample search application.
- The ESSearchServer application provides all remote communication for the WebSphere II OmniFind Edition SI-API implementation and enables the SI-API interfaces to communicate with the search servers.

By default, all three enterprise applications support WebSphere Application Server global security. When these applications detect that global security is enabled, they begin authenticating all requests that they receive.

Some organizations might want to enable or disable security for specific WebSphere II OmniFind Edition enterprise applications. For example, you might want to authenticate all users who access the enterprise search administration console, but not authenticate users who use the WebSphere II OmniFind Edition SI-API interfaces or the sample search application.

Procedure

To disable security for a particular enterprise application:

1. On the search server, start the WebSphere Application Server Administrative Console.

You can open the Administrative Console in the following ways:

- Use the Windows **Start** menu to select the program.
- For WebSphere Application Server version 5, open a Web browser and go to `http://hostname:port/admin`, where *hostname* is the host name of the search server and *port* is the port number for the WebSphere Application Server Administrative Console. Typically, the Administrative Console port is 9090.

- For WebSphere Application Server version 6, open a Web browser and go to `http://hostname:port/ibm/console`, where *hostname* is the host name of the search server and *port* is the port number for the WebSphere Application Server Administrative Console. Typically, the Administrative Console port is 9060.
- 2. When you are prompted for a user ID and password, enter the administrator ID and password that were specified when global security was enabled in WebSphere Application Server.
- 3. After you log in to the Administrative Console, click **Applications** and then click **Enterprise Applications**.
- 4. Select the check box next to the name of the enterprise application for which you want to disable security.
- 5. Scroll down and click the **Map security roles to users/groups** link.
- 6. Locate the **AllAuthenticated** role and select the check box under the **Everyone?** column.
- 7. Click **OK**.
- 8. Click the **Save** link to save your changes.
- 9. If you are using WebSphere Network Deployment, select the **Synchronize changes with Nodes** check box.
- 10. Click **Save**.
- 11. Click **Applications** again, and then click **Enterprise Applications**.
- 12. Select the check box for application that you just modified and click **Stop**.
- 13. After the application stops, select the check box for application again, and click **Start**.

Collection-level security

To provide collection-level security, you configure options for indexing content and options for allowing search applications to search specific collections.

When you create a collection, you can choose an option to enable collection security. If you choose this option, you can later configure document-level security controls. When collection security is enabled, the enterprise search global analysis processes also apply different rules for indexing duplicate documents and anchor text in documents.

After you create a search application, a search application ID enables you to specify which collections the search application can search, and which users can access the search application.

Duplicate document analysis

If you enable collection security, the global analysis processes do not identify duplicate documents in the collection.

During global analysis, the indexing processes identify documents that are duplicates, or near duplicates, of each other. They then associate all of these documents with one canonical representation of the content. By allowing duplicate documents to be identified, you can ensure that search results do not contain multiple documents with the same (or nearly the same) content.

If you enable collection security when you create a collection, duplicate documents are not identified, and so they are not associated with a common canonical

representation. Instead, each document is indexed independently. This ensures that users search only the documents with security tokens that match their credentials. For example, two documents might be nearly identical in content, but use different access control lists to enforce security.

Disabling duplicate document analysis can enhance the security of documents in a collection, but search quality might be degraded if users receive multiple copies of the same document in the search results.

Anchor text analysis

If you enable collection security, the global analysis processes apply special rules for indexing the anchor text in documents crawled by Web crawlers. If you do not enable collection security, you can specify whether you want to index the anchor text in links to forbidden documents when you configure individual Web crawlers.

Anchor text is the information within a hypertext link that describes the page that the link connects to. For example, in the following link, the text Query Syntax is the anchor text in a link that connects to the syntax.htm page on a Web site:

```
<a href="../doc/syntax.htm">Query Syntax</a>
```

Typically, the Web crawler follows links in documents to crawl additional documents and includes these linked pages in the index. During global analysis, the index processes associate the anchor text not only with the document in which it is embedded (the source document) but also with the target document. In the example above, the anchor text Query Syntax is associated with the target page syntax.htm and with the page that contains the anchor construct.

If you enable collection security when you create a collection, anchor text processing is disabled. The anchor text is no longer indexed with a document unless it actually appears in the document or in the document metadata. This security control ensures that users are not exposed to information in documents that they are not allowed to access because the anchor text to forbidden documents is not associated with the documents that they do have access to.

Enabling collection security can enhance the security of Web documents by enabling users to search only the documents with security tokens that match their credentials. However, by not processing anchor text, the search results might not include all of the documents that are potentially relevant to a query.

If you do not enable collection security, you can select an option to index the anchor text in links to forbidden documents when you configure individual Web crawlers. If you specify that the anchor text is to be indexed, the analysis and indexing processes index the anchor text in all pages that are retrieved by the Web crawler. If you specify that anchor text is not to be indexed, the anchor text in links to forbidden documents is excluded from the index.

Indexing the anchor text in links to forbidden documents

If a document includes links to documents that the Web crawler is forbidden to crawl, you can specify whether you want to retain the anchor text for those links in the index when you configure a Web crawler.

Before you begin

To configure options for indexing anchor text, you must be a member of the enterprise search administrator role or be a collection administrator for the Web crawler that you want to configure.


About this task

Directives in a robots.txt file or in the metadata of Web documents can prevent the Web crawler from accessing documents on a Web site. If a document that the Web crawler is allowed to crawl includes links to forbidden documents, you can specify how you want to handle the anchor text for those links.

You can specify whether you want to index the anchor text to forbidden documents when you configure the Web crawler. For maximum security, specify that you do not want to index the anchor text in links to forbidden documents. By not indexing anchor text, however, the search results might not include all of the documents that are potentially relevant to a query.

Procedure

To enable or disable the indexing of anchor text in links to forbidden documents:

1. Edit a collection, select the Crawl page, locate the Web crawler that you want to configure and click  **Crawler properties**.
2. Click **Edit advanced Web crawler properties**.
3. To index the anchor text in all of the documents that this crawler crawls, select the **Index the anchor text in links to forbidden documents** check box. Users will be able to learn about pages that the Web crawler is not allowed to crawl by searching for text that is in the anchor text of links that point to those pages. To exclude anchor text in links to forbidden documents from the index, clear this check box. Users will not be able to learn about pages that the Web crawler is not allowed to crawl. The anchor text will be excluded from the index in addition to the forbidden documents.
4. Click **OK** and then, on the Web Crawler Properties page, click **OK** again.
5. For the changes to become effective, stop and restart the crawler.

To apply the changes to documents that were previously indexed, you must recrawl the documents so that they can be indexed again. If a previous crawl added information about forbidden documents to the index, that information will then be removed from the index.

Security with search application IDs

To provide collection-level security, you specify which search applications can search each collection and external source.

| All search applications are required to pass an application identifier to the
| enterprise search APIs. An enterprise search administrator and your search
| applications can use this identifier to enforce security for collections and external
| sources.

| Before a search application can access a collection or external source, an enterprise
| search administrator must associate the search application with the specific
| collections and sources that it can search. A search application can search all of the
| collections and external sources in an enterprise search system, or search only the
| collections and external sources that you specify.

To enforce access controls, you can associate a security tokens (such as user IDs, group IDs, or user roles) with your search application and allow only those users to access the application. For example, you can restrict access to the URL that launches your search application.

For more information about search application IDs and how to incorporate security controls into your custom search applications, see the Search and Index API for enterprise search.

Related concepts

"Search applications for enterprise search" on page 161

A search application enables you to search collections and external sources in the enterprise search system. You can create any number of search applications, and a single search application can search any number of collections and external sources.

"Search and index API overview" in "Programming Guide and API Reference for Enterprise Search"

"Search and index API security" in "Programming Guide and API Reference for Enterprise Search"

Document-level security

If security is enabled for a collection when it is created, you can configure document-level security controls. Document-level security ensures that users who search collections are able to access only the documents that they are allowed to see.

To control access to documents in the collection, crawlers can collect security for the index. For some types of data sources, you can also validate user current credentials when a query is submitted. To validate a user's current credentials, you can build support for user profiles into your custom search applications. By storing user profiles, you enable users to access documents without being challenged multiple times to specify their credentials.

Related concepts

"Search applications for enterprise search" on page 161

A search application enables you to search collections and external sources in the enterprise search system. You can create any number of search applications, and a single search application can search any number of collections and external sources.

"Document-level security with the Portal Search Engine" on page 201

You can use the IBM WebSphere Portal Search Engine to enforce document-level security when users search enterprise search collections.

"Search and index API security" in "Programming Guide and API Reference for Enterprise Search"

Validation by stored security tokens

If security is enabled for a collection when it is created, you can configure document-level security controls by storing security data in the index.

By default, each document is considered a public document, which means that it can be searched by all users. For most document types, you can achieve document-level security by associating one or more security tokens with documents and storing these tokens with the documents in the index. When you

configure a crawler, you specify that you want to use security tokens to limit which users can access the documents that are crawled by that crawler.

If a data source type includes fields, you can specify that you want to use the value in one of those fields to enforce access controls. If the data source does not have fields, if you do not want to use a field value for security purposes, or if the field that you specify does not include a value that enables access controls to be enforced, you can define security tokens for the crawler to associate with documents.

The administrator for each collection decides the security tokens that the crawler is to associate with documents. For example, a security token might represent a user ID, a group ID, a user role, or any other value that you determine is valid for the data source. If a data source administrator updates the native access control list, the updated security controls become available the next time that the index is refreshed or reorganized.

Security tokens accompany documents as the documents pass through the stages of parsing, analysis, and indexing. If your search applications enable security, you can use the security tokens to control access to documents. Users who search the collection are able to search only the documents that their credentials permit them to see. If a user's credentials do not pass the security rules, the user cannot search documents that are protected by the security tokens.

You can apply custom business rules to determine the value of the security tokens by encoding the rules in a Java class. When you configure crawler properties, you specify the name of the plug-in that you want the crawler to use when it crawls documents. The security tokens that your plug-in adds are stored in the index and can be used to control access to documents.

Validation of current credentials during query processing

If security is enabled for a collection when it is created, certain types of domains enable you to validate a user's current credentials when the user submits a query.

When you configure the following types of crawlers, you can select an option to validate user credentials by comparing the credentials to current access controls that are managed by the native repository:

- Content Edition crawler (Documentum, FileNet Panagon Content Services, and Portal Document Manager repository types only)
- Domino Document Manager crawler
- Notes crawler
- QuickPlace crawler
- WebSphere Portal crawler
- Windows file system crawler

Before responding to a query, the search servers interface with the native repositories to validate the user's current permissions, and then remove all documents that the user does not have permission to view from the search results.

This approach for enforcing document-level security provides a high level of security because the user's credentials are compared to current security data as opposed to security data that is stored in the index. It also ensures that access is controlled by the security mechanisms of the native repository, regardless of how

complex those mechanisms might be. Because document filtering occurs in real time, the search results reflect the latest access control settings for each document that matches the search criteria.

Another advantage of this approach is that it does not impact the size of the index (extra space is not required to index the security tokens). However, because the validation requires connections to the native repositories, the approach can impact query performance.

For maximum security, and to minimize the impact on query performance, combine the option for storing security tokens in the index with the option to validate current access controls. When a user submits a query, validation occurs in two stages:

- First, the search servers use the indexed security data to quickly determine whether the user has permission to access the server and database from which a document was crawled (the index is optimized for speed and produces sub-second response times).
- Next, the search servers create an interim list that contains only the documents that exist in domains on servers that the user is allowed to access. The search servers use this list to connect to native repositories and determine whether the user is allowed to view the requested document.

If a user has access to a server and domain, there is a high probability that the user has access to the documents. However, this final filtering stage ensures that only the documents that match the user's current permission settings are returned in the search results.

Related concepts

"Enforcement of document-level security for Windows file system documents" on page 193

To enable current credentials to be validated when a user searches documents that were crawled by a Windows file system crawler, you must configure domain account information on both the crawler server and Microsoft Windows server.

"Enforcement of document-level security for Lotus Domino documents" on page 195

If the Lotus Notes server to be crawled uses the Notes remote procedure call (NRPC) protocol, you must configure the crawler server so that document-level access controls can be enforced.

Related tasks

"Configuring Lotus Domino Trusted Servers to validate user credentials" on page 195

To enforce security for documents that were crawled by a Notes crawler that uses the Notes remote procedure call (NRPC) protocol, the Domino servers to be crawled must be configured to be Lotus Domino Trusted Servers.

User profiles and identity management

By creating user profiles for enterprise search, users can store credentials that enable them to search secure domains.

To search a domain that requires user credentials to be validated when a query is submitted, users must provide the search application with the credentials that they use to log in to the domain. With WebSphere Information Integrator OmniFind Edition *identity management*, users can store credentials for any number of domains in a user profile. The credentials are encrypted and stored securely in the enterprise search system.

Users can create a user profile and register their credentials while they use a search application. In the sample search application for enterprise search, this capability is provided by the **My Profile** option (your custom search applications might implement this capability differently).

The user profile stores the various credentials that the user must specify to log in to the domains to be searched. Users can create a user profile if all of the following conditions are true:

- Global security is enabled in WebSphere Application Server.
- WebSphere II OmniFind Edition identity management is enabled in the enterprise search administration console.
- Security is enabled in at least one of the collections that the search application can search.
- At least one secure collection includes documents that were crawled by a crawler that enforces access control by requiring the user's current credentials to be validated when a query is submitted.
- The option to validate current credentials during query processing was selected when document-level security was configured for at least one of the following crawler types:
 - Content Edition (for certain types of repositories)
 - Domino Document Manager
 - Notes
 - QuickPlace
 - WebSphere Portal
 - Windows file system

The profile lists all of the domains that are available to the search application that require user credentials to be validated during query processing. Users can choose which domains they want to store credentials for. If the user does not specify credentials for a domain, documents from the data sources in that domain are excluded from the search results.

If you do not use WebSphere II OmniFind Edition identity management, the search application must supply the user's security context (USC) string when users query domains that require current credentials to be validated.

Configuring identity management

You can use WebSphere II OmniFind Edition identity management to store user profiles. Profiles enable users to search domains that require the user's credentials to be validated during query processing.

Before you begin

To configure identity management options, you must be a member of the enterprise search administrator role.

About this task

If you specify that user credentials are to be validated during query processing when you configure document-level security options for a crawler, users must provide their credentials when they query a domain that requires validation. With WebSphere II OmniFind Edition identity management, users can create a user

profile and register the credentials that they use to log in to the secure domains. The credentials are encrypted in a secure database that is managed by WebSphere II OmniFind Edition.

The search servers use the stored credentials to authenticate the user when the user searches a secure domain. If the credentials are not valid, documents from the secure domain are excluded from the search results.

Procedure

To configure identity management:

1. Click **Security** to open the Security view.
2. On the Search Applications page, click **Configure identity management**.
3. On the Configure Identity Management page, select the check box that enables WebSphere II OmniFind Edition to manage user credentials in user profiles.
4. Click **OK**.

Enforcement of document-level security for Windows file system documents

To enable current credentials to be validated when a user searches documents that were crawled by a Windows file system crawler, you must configure domain account information on both the crawler server and Microsoft Windows server.

When you configure a Windows file system crawler, you specify whether you want to crawl subdirectories on the local computer or subdirectories on a remote computer. If security is enabled for the collection, you can also specify options for controlling access to documents in the crawled subdirectories.

If you choose to enforce access controls by validating the user's current credentials when the user submits a query, you must ensure that domain accounts are correctly configured. Requirements for setting up domain accounts for files that were crawled on the local computer are different from requirements for files that were crawled on a remote Windows server.

Validation with local access control data

To validate current user credentials, the system uses both local user account information and domain account information (if the computer belongs to a Windows domain). To validate credentials during query processing, both user names must be listed in the security information for the documents to be searched.

Local accounts

For a local account, the user name is in the following format:

COMPUTER_NAME\USERNAME

To log in, users specify only the user name, but the properly specified Windows user rights assignment uses the full name. For example, if the local account user name is abcuser, the full account name might be WINSERVER1\abcuser.

When users use a search application and configure a profile for searching secure documents on a local system, they must specify the user name that they use to log in to Windows (for example, abcuser).

Domain accounts

For a domain account, the user name is in the following format:

DOMAIN NAME\USERNAME

To log in, users specify this information in the following format:

USERNAME@DOMAIN NAME

For example, if you configure user rights assignments for a file and select the domain WIN1\abcuser, the account is then displayed as abcuser@win1.company.com.

When users use a search application and configure a profile that enables them to search documents in a secure domain, they must specify the user name that they use to log in to Windows (for example, abcuser@win1.company.com).

To enforce current credential validation on local computers, the user accounts that are used by the crawler server must have the following Windows user rights. (To assign user rights, use the Windows Administrative Tools: **Administrative Tools** → **Local Security Policy** → **Local Policies** → **Local User Rights Assignment**.)

- The user ID that the crawler server is running as must have the **Act as part of the operating system** right. (This right is configured for the enterprise search administrative user on the crawler server when WebSphere Information Integrator OmniFind Edition is installed.)
- Users must have the **Log on Locally** user right.

Validation with remote domain access control data

For the Windows operating system, any directory that starts with `\\servername` is considered a remote directory. For example:

`\\software\utilities\IBM`

To access a remote directory, users specify their user names in the following format:

USERNAME@DOMAIN NAME

When users use a search application and configure a profile that enables them to search secure documents on a remote system, they must specify the user name that they use to access the remote Windows system (for example, abcuser@win1.company.com).

To enforce current credential validation on remote computers, user accounts must have the following Windows user rights. (To assign user rights, use the Windows Administrative Tools: **Administrative Tools** → **Domain Security Policy**.)

- The crawler server and the Windows server to be searched must be members of the same domain.
- The user ID that the crawler server is running as must have the **Act as part of the operating system** right. (This right is configured for the enterprise search administrative user on the crawler server when WebSphere Information Integrator OmniFind Edition is installed.)
- Users must have the **Log on as a batch job** user right.

Related concepts

“Validation of current credentials during query processing” on page 190
If security is enabled for a collection when it is created, certain types of domains enable you to validate a user’s current credentials when the user submits a query.

“Windows file system crawlers” on page 86

To include documents that are stored in Microsoft Windows file systems in an enterprise search collection, you must configure a Windows file system crawler.

Enforcement of document-level security for Lotus Domino documents

If the Lotus Notes server to be crawled uses the Notes remote procedure call (NRPC) protocol, you must configure the crawler server so that document-level access controls can be enforced.

To enforce document-level security with documents that were crawled on a Lotus Notes server that uses the NRPC protocol, you must install a Domino server on the crawler server. This Domino server must be a member of your Domino domain. Follow the instructions in the Lotus Domino documentation to install and configure the Domino server.

You must also complete the following tasks so that the search servers can verify whether a user who searches a secure collection is authorized to view Lotus Notes documents that match the search criteria. Documents that the user is not authorized to view are removed from the search results before the results are returned to the user.

- “Configuring Lotus Domino Trusted Servers to validate user credentials.”
- Enabling global security in WebSphere Application Server and configuring the search application to use security. This step ensures that users will be prompted to specify credentials when they attempt to use the search application. The search servers can then use these credentials to verify each user’s access to Lotus Notes documents.

Related concepts

“Validation of current credentials during query processing” on page 190
If security is enabled for a collection when it is created, certain types of domains enable you to validate a user’s current credentials when the user submits a query.

“Notes crawlers” on page 55

To include IBM Lotus Notes databases in an enterprise search collection, you must configure a Notes crawler.

Configuring Lotus Domino Trusted Servers to validate user credentials

To enforce security for documents that were crawled by a Notes crawler that uses the Notes remote procedure call (NRPC) protocol, the Domino servers to be crawled must be configured to be Lotus Domino Trusted Servers.

Before you begin

This procedure is required if you want to enforce document-level security when searching remote databases. To search databases that are local to the crawler server, this procedure is not necessary.

To configure Trusted Servers, a Domino server must be installed on the crawler. This Domino server must be a member of your Domino domain.

About this task

When you configure document-level security options for a Notes crawler, you specify whether you want to enforce access controls by validating the user's current credentials when the user submits a query. To enforce this type of security, the Domino servers to be crawled must be Lotus Domino Trusted Servers.

When users search a domain that requires their current credentials to be validated, the Trusted Server enables the Domino server ID to switch context to the current user ID. The Domino database is opened as if the current user had opened it, and all of the database access control list information for that user is enforced.

The ability to switch contexts in this manner is typically available only for databases that are stored in the data directory of the local Domino server. Beginning with Lotus Domino version 6.5.1, this ability is provided through the Trusted Server. To configure the Trusted Server, a Domino administrator specifies which Domino servers are to be trusted to perform sensitive operations, such as acting as another user when a database is accessed from a remote computer.

Procedure

To configure a Trusted Server, complete the following steps on all Domino servers that are crawled by a Notes crawler:

1. On a Domino server, use the Domino domain administrator ID file to open the Lotus Domino Administrator client.
2. Select **File** → **Open server**.
3. Type the name of the Domino server for which you want to enable Trusted Server capabilities.
4. Select the **Configuration** tab.
5. Expand the **Server** object, select the **Current Server** document, and click **Edit Server**.
6. Select the **Security** tab, scroll to the bottom of the document, locate the **Trusted Servers** entry, and click the down arrow button.
7. Specify one of the following options:

LocalDomainServers

Select this option if all servers in the Domino domain are to be considered Trusted Servers.

server_name

Specify the name of a Domino server that you want to be able to crawl and search as a Trusted Server.

If the Domino server to be crawled is in a different Domino domain, then you must specify the server name or select the

OtherDomainServers group. You must also follow the Domino procedures for cross-certification of the WebSphere II OmniFind Edition Domino server ID file with the other Domino domain. See the Domino server documentation for information about these procedures.

8. Click **Save and Close** to save your changes.
9. Stop and restart the remote Domino servers that you enabled to act as Trusted Servers.

Related concepts

| “Validation of current credentials during query processing” on page 190
| If security is enabled for a collection when it is created, certain types of
| domains enable you to validate a user’s current credentials when the user
| submits a query.

| “Notes crawlers” on page 55

| To include IBM Lotus Notes databases in an enterprise search collection, you
| must configure a Notes crawler.

Disabling document-level security

You can enable users to search a collection regardless of whether any access controls are associated with the documents in the index. For crawlers that support current credential validation, you can also enable users to search a collection without validating current access controls during query processing.

Before you begin

To configure document-level security options, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

Restrictions

You can specify document-level security options only if security was enabled for the collection when the collection is created.

About this task

| When you configure a crawler, you can associate security tokens with the
| documents that are being crawled. Your search applications can use these tokens,
| which are stored in the index, to enforce access controls when users search the
| collection. For some crawlers, you can also specify that you want to validate
| current access controls that are associated with documents in their native
| repositories when users submit queries.

To remove these security restrictions, you can specify that the search servers are to ignore any security tokens that are passed with a query. You can also enable users to query documents without having their credentials compared to current access controls.

You might want to disable document-level security temporarily if you are testing a new collection or if you need to troubleshoot a problem with a search application.

Procedure

To disable document-level access controls:

1. Edit a collection, select the General page, and click **Configure document-level security**.
2. On the Document-Level Security page, select the **Ignore document-level access controls in the index** check box if you do not want the security tokens that crawlers associated with documents to be used when users query the collection. Crawlers continue to add security tokens to documents, but the search servers ignore the tokens and allow users to search the previously protected documents.

3. Select the **Do not validate current credentials during query processing** check box if you do not want to validate the current access controls that are associated with documents in their native repositories when users submit queries. This check box is available only for documents that were crawled by crawlers that support this capability.

If you select this check box, other document-level security options remain in effect. For example, if you specified options to store access controls in the index when you configured the crawler, those security controls continue to apply unless you also select the **Ignore document-level access controls in the index** check box.

4. Click **OK**.

Enterprise search integration with WebSphere Portal

You can expand the search capabilities of IBM WebSphere Portal by deploying enterprise search portlets in WebSphere Portal and the WebSphere Portal Search Center.

Integration points

The enterprise search portlets integrate with WebSphere Portal in several ways:

WebSphere Portal

WebSphere Portal provides users with a single access point for interacting with applications, content, processes, and people. The WebSphere Portal framework enables new applications, called portlets, to be integrated and deployed without impacting other applications in the portal.

If you deploy the enterprise search portlet into WebSphere Portal, you can search enterprise search collections from the WebSphere Portal interface. Through WebSphere Portal configuration settings, you can ensure that the enterprise search portlet has the same look and feel as other portlets in your WebSphere Portal environment.

Portal Search Engine

The WebSphere Portal Search Engine crawls Web sites. Administration portlets enable administrators to build indexed collections, and search portlets enable users to search those collections.

If you use WebSphere Portal Version 5.0.2 or later, you can migrate configuration information for Portal Search Engine collections and taxonomies to enterprise search.

If you use WebSphere Portal Version 5.1 or later, you can use the Portal Search Engine Document Search portlet to search enterprise search collections. Configuration properties enable administrators to easily switch between the two search capabilities as needed.

WebSphere Portal Search Center

The WebSphere Portal Search Center provides a central starting point for searching all sources that are made available for searching through WebSphere Portal. The Search Center and the Universal search portlet enable you to search WebSphere Portal content and any other collections that an administrator registers with the Search Center.

The Search Center has a paged interface. You can search all available collections through one common page, or you can select a page to search an individual collection. For example, there is a page for Portal Search Engine indexes and a page for Portal Document Management libraries.

To enable enterprise search collections to be searched from the Search Center, WebSphere Information Integrator OmniFind Edition (WebSphere II OmniFind Edition) provides an adapter and a registration portlet. The adapter adds a tab for enterprise search to the Search Center interface, and the registration portlet registers the enterprise search portlet with the Search Center.

You must install the adapter and the registration portlet before you can use the Search Center to search enterprise search collections.

WebSphere Portal crawler

An enterprise application, ESPACServer.ear, is installed on the search servers when WebSphere II OmniFind Edition is installed. After you deploy this enterprise application in WebSphere Portal, you can use the enterprise search administration console to configure a WebSphere Portal crawler and add WebSphere Portal sites to your enterprise search collections.

Benefits of integrating

Enterprise search enhances the WebSphere Portal search environment by providing support for searching a wider range of data source types. With the enterprise search portlet, you can search Web sites plus all of the other data source types that are supported by WebSphere II OmniFind Edition.

Enterprise search also offers benefits in scalability. The Portal Search Engine is useful for small-sized or medium-sized businesses where a single server is sufficient to support the search and retrieval workload. To support enterprise-level capacities, enterprise search distributes the workload over four servers (one for crawling data, one for parsing and indexing data, and two to support search and retrieval processes).

Portlet deployment overview

The portlets that you can use for enterprise search depend on the version of WebSphere Portal that you use:

WebSphere Portal Version 5.0.2

You can deploy the enterprise search portlet and use it to search enterprise search collections. This portlet can coexist with the Portal Search Engine portlets.

WebSphere Portal Version 5.1 or later

- You can deploy the enterprise search portlet and use it to search enterprise search collections. This portlet can coexist with the Portal Search Engine portlets.
- After you deploy the enterprise search portlet, you can configure the Document Search portlet in the Portal Search Engine to search enterprise search collections.
- After the enterprise search portlet is registered with the WebSphere Portal Search Center, you can use the Search Center to search enterprise search collections. You can select a page to search only enterprise search collections, or you can enter a query that searches enterprise search collections and all of the other collections that are available in the Search Center.

Related concepts

“Migration from WebSphere Portal to enterprise search” on page 207
Enterprise search provides a migration wizard that you can use to migrate taxonomies and collections from IBM WebSphere Portal to enterprise search.

Related tasks

“Deploying the enterprise application for the WebSphere Portal crawler” on page 84
Before you create a WebSphere Portal crawler, you must deploy an enterprise application, ESPACServer.ear, in WebSphere Portal.

Document-level security with the Portal Search Engine

You can use the IBM WebSphere Portal Search Engine to enforce document-level security when users search enterprise search collections.

If an enterprise search crawler associates a group ID security token with the documents that it crawls, and if you configure the Document Search portlet for the Portal Search Engine to search enterprise search collections, then the Portal Search Engine can derive the group ID for the logged in user, and pass the security token for that group ID with the query to enterprise search. The security token ensures that only the documents that a user is authorized to see are returned in the search results.

If the crawler associated another type of security token with documents, such as a user ID or a user role, and you want to enforce document-level security when you search enterprise search collections, then you must create a custom search portlet. The Portal Search Engine derives security tokens for group IDs only.

Related concepts

“Search applications for enterprise search” on page 161

A search application enables you to search collections and external sources in the enterprise search system. You can create any number of search applications, and a single search application can search any number of collections and external sources.

“Document-level security” on page 189

If security is enabled for a collection when it is created, you can configure document-level security controls. Document-level security ensures that users who search collections are able to access only the documents that they are allowed to see.

Deploying the Search portlet

The Search portlet, `ESSearchPortlet.war`, enables you to use the WebSphere Portal Search Engine to search enterprise search collections.

Before you begin

You must deploy the Search portlet for enterprise search in WebSphere Portal before you can configure the Portal Search Engine to use the portlet.

About this task

After you deploy the Search portlet, you can continue to use the Portal Search Engine to search indexed data in WebSphere Portal. To enable users to search enterprise search collections, a WebSphere Portal administrator edits properties in the Portal Search Engine configuration. You can switch between these two search capabilities and use the most appropriate search solution for your needs.

The `ESSearchPortlet.war` file is installed in the `ES_INSTALL_ROOT/bin` directory on the search servers when WebSphere II OmniFind Edition is installed. The default installation paths are as follows:

- UNIX systems: `/opt/IBM/es/bin/ESSearchPortlet.war`
- Windows systems: `C:\Program Files\IBM\es\bin\ESSearchPortlet.war`

Procedure

To deploy the Search portlet in WebSphere Portal:

1. Install the portlet:
 - a. Log in to the WebSphere Portal server with the WebSphere Portal server administrator ID.
 - b. Select **Administration** from the toolbar.
 - c. Select **Portlets** in the navigation area on the left, then select **Install** from the **Portlets** menu.
 - d. Click **Browse**, select the ESSearchPortlet.war file from your system, and click **Next**.
 - e. On the next page, click **Install**. After the portlet is installed, the following message is displayed: APIN0005I: Portlets were successfully installed.
2. Modify the portlet parameters:
 - a. Select **Manage Portlets** in the navigation area on the left, then select **IBM Enterprise Search**.
 - b. After new icons are displayed to the right of the selection box, select **Modify Parameters**.
 - c. In the list of portlet parameters, modify the following parameters:
 - port** Set this value to the Web Server port number for the WebSphere II OmniFind Edition Search server. The default value is 80.
 - applicationName**
Set this value to the name of a valid search application for enterprise search. (The names of available search applications are listed on the Search Applications page in the enterprise search administration console.) The default value is Default, which is a search application that is always configured on the search servers.
 - hostname**
Set this value to the fully qualified host name of a WebSphere II OmniFind Edition search server.
 - d. Click **Save** to save your changes, and then click **Cancel** to exit the Modify Parameters page.
3. Create a portal page for the portlet:
 - a. Select **Portal User Interface** in the navigation area on the left, then select **Manage Pages**.
 - b. Select the **My Portal** link.
 - c. Click **New Page**, specify a title for the page, and click **OK**. After the page is created, the following message is displayed: APPR0010I: *page_title* has been created successfully.
 - d. Click **OK**. The page that you created appears in the list of available pages.
4. Add the Search portlet to the page:
 - a. Select the Edit icon (pencil) for the page that you created.
 - b. Click **Add portlets**.
 - c. In the **Search for** text box, type Enterprise and then click **Search**.
 - d. Select the check box next to **IBM Enterprise Search** and click **OK**.
 - e. Click **Done**. This step indicates that you successfully installed the Search portlet and added the portlet to a WebSphere Portal page.
5. Access the portlet:
 - a. Select **My Portal** in the navigation bar at the top of the page.

- b. Select the page that matches the page title that you specified when you added the portlet to the system.

The first time that you access the portlet page, the page might be slow to appear because the system must compile Java Server Pages (JSP files) for the portlet.

Configuring the WebSphere Portal Search and Browse portlet for enterprise search

WebSphere Portal provides a portlet that you can use to search and browse native WebSphere Portal collections. You can configure this portlet to search enterprise search collections.

Procedure

To configure the Search and Browse portlet to search enterprise search collections:

1. Stop the WebSphere_Portal server instance.
2. Copy the following files from the WebSphere II OmniFind Edition ES_INSTALL_ROOT/lib directory to the /WebSphere/PortalServer/shared/app directory on the WebSphere Portal server:

```
esapi.jar  
siapi.jar
```

3. Start the WebSphere_Portal server instance and log in to the WebSphere Portal server with the WebSphere Portal administrator ID.
4. Copy the portlet:
 - a. Click **Administration** in the top right corner, then expand the **Portlet Management** object and select **Portlets**.
 - b. Search for the word search.
 - c. For the Search and Browse portlet, click **Copy Portlet**, provide a unique name for the portlet, and click **OK**.
5. Configure the portlet:
 - a. Click the **Configure Portlet** icon next to the portlet name that you specified.
 - b. Click the right arrow to go to the second page of parameters.
 - c. In **New parameter**, enter ApplicationInfoId, specify the name of a valid WebSphere II OmniFind Edition search application in the **New value** field, then click **Add** to add the new parameter.

The names of available search applications are listed on the Search Applications page in the enterprise search administration console. The default value is Default, which is a search application that is always configured on the search servers.

- d. Delete the following parameters:

```
ApplicationInfoId_EXAMPLE  
QueryFactoryImp  
BrowseFactoryImp  
SearchFactoryImp  
IndexName  
EJB_Example  
IIOP_URL_Example  
SOAP_URL_Example  
EJB  
SOAP_URL  
IIOP_URL
```

- e. Add the following parameters and specify the values that are shown below. Click **Add** to add each parameter name and value. Parameter names are case sensitive.

```
IMPLEMENT = ES
ApplicationInfoId = search_application_name (such as Default)
SearchFactoryImp = com.ibm.es.api.search.RemoteSearchFactory
BrowseFactoryImp = com.ibm.es.api.browse.RemoteBrowseFactory
hostname = search_server_host_name (such as omnifind.ibm.com)
port = search_server_port_number (such as 80)
```

6. Assign the new copy of the Search and Browse portlet to a portal page. See the WebSphere Portal administration documentation for assistance.
7. Access the portlet:
 - a. Select **My Portal** in the navigation bar at the top of the portal page.
 - b. Select the page that matches the name that you specified when you copied the portlet in step 4 on page 203.

Installing the enterprise search adapter for the Search Center

Before you can use the WebSphere Portal Search Center to search enterprise search collections, you must install an adapter for enterprise search.

About this task

The adapter, `ESSearchAdapter.ear`, adds a page for enterprise search to the Search Center interface. After you install this application and the registration portlet, you can add enterprise search collections to the Search Center and search those collections with the Universal Search Portlet.

The `ESSearchAdapter.ear` file is installed in the `ES_INSTALL_ROOT/bin` directory on the search servers when WebSphere II OmniFind Edition is installed. The default installation paths are as follows:

- UNIX systems: `/opt/IBM/es/bin/ESSearchAdapter.ear`
- Windows systems: `C:\Program Files\IBM\es\bin\ESSearchAdapter.ear`

Procedure

To install the adapter for enterprise search in the Search Center:

1. Stop the `WebSphere_Portal` server instance.
2. If it is not already started, start the WebSphere Application Server `server1` server instance.
3. Copy the following files from the WebSphere II OmniFind Edition `ES_INSTALL_ROOT/lib` directory to the `/WebSphere/PortalServer/shared/app` directory on the WebSphere Portal server:

```
esapi.jar
siapi.jar
```

If you are prompted to overwrite the `siapi.jar` file, specify **Yes**.

4. On the WebSphere Portal server, start the WebSphere Application Server Administrative Console. If you are prompted to log in, log in.

You can open the Administrative Console in the following ways:

- Use the Windows **Start** menu to select the program.

- For WebSphere Application Server version 5, open a Web browser and go to `http://hostname:port/admin`, where *hostname* is the host name of the search server and *port* is the port number for the WebSphere Application Server Administrative Console. Typically, the Administrative Console port is 9090.
 - For WebSphere Application Server version 6, open a Web browser and go to `http://hostname:port/ibm/console`, where *hostname* is the host name of the search server and *port* is the port number for the WebSphere Application Server Administrative Console. Typically, the Administrative Console port is 9060.
5. Click **Applications** and then click **Install new application**.
 6. Click **Browse**, select the `ESSearchAdapter.ear` from your system, and click **Next** twice. If you receive a warning about policy files, click **Continue**.
 7. Continue clicking **Next** until you see the Map modules to application servers page, make the following selections, and click **Apply**:
 - In the **Clusters and Servers** field, select **WebSphere:cell=cell_name, node=node_name, server=server1**.
 - Select the check box next to the **ESSearchAdapterEJB** module.
 8. Continue clicking **Next** until you see the Summary page, then click **Finish**.
 9. Click the **Save to Master Configuration** link, then click the **Save** button to save your changes to the WebSphere Application Server configuration.
 10. Restart the WebSphere Portal server.

Installing the enterprise search registration portlet for the Search Center

Before you can use the WebSphere Portal Search Center to search enterprise search collections, you must install a registration portlet for enterprise search.

About this task

The registration portlet, `ESSearchAdapterPortlet.war`, registers the enterprise search portlet with the Search Center. After you install this portlet (and the adapter application for enterprise search), you can add enterprise search collections to the Search Center and search those collections with the Universal Search Portlet.

The `ESSearchAdapterPortlet.war` file is installed in the `ES_INSTALL_ROOT/bin` directory on the search servers when WebSphere II OmniFind Edition is installed. The default installation paths are as follows:

- UNIX systems: `/opt/IBM/es/bin/ESSearchAdapterPortlet.war`
- Windows systems: `C:\Program Files\IBM\es\bin\ESSearchAdapterPortlet.war`

Procedure

To install the registration portlet for enterprise search:

1. Log in to the WebSphere Portal server with the WebSphere Portal administrator ID.
2. Select **Administration** from the toolbar.
3. Select **Portlets** in the navigation area on the left, then select **Install** from the **Portlets** menu.
4. Click **Browse**, select the `ESSearchAdapterPortlet.war` file from your system, and click **Next**.

- |
 - |
 - |
 - |
 - |
 - |
5. On the next page, click **Install**. After the portlet is installed, the following message is displayed: APIN00051: Portlets were successfully installed.
 6. Assign the portlet (which is named IBM WebSphere II OmniFind Edition registration portlet for enterprise search) to a portal page. See the WebSphere Portal administration documentation for assistance.

Migration from WebSphere Portal to enterprise search

Enterprise search provides a migration wizard that you can use to migrate taxonomies and collections from IBM WebSphere Portal to enterprise search.

To migrate taxonomies and collections, you run the migration wizard on the enterprise search index server. After you migrate a taxonomy, you can use it with enterprise search collections. You can also use enterprise search to administer and search collections that you migrate from WebSphere Portal.

In enterprise search, a taxonomy is called a category tree. After you migrate a rule-based taxonomy, you can use the enterprise search administration console to administer the category tree. To use model-based taxonomies with enterprise search, WebSphere Portal must be installed on the index server.

If you want to migrate taxonomies and collections, always migrate the model-based taxonomy files first, before you migrate collections. If you do not do this, model-based categorization will not work with the collections that you migrate from WebSphere Portal.

Related concepts

“Enterprise search integration with WebSphere Portal” on page 199

You can expand the search capabilities of IBM WebSphere Portal by deploying enterprise search portlets in WebSphere Portal and the WebSphere Portal Search Center.

Related tasks

“Configuring categories” on page 103

You can create any number of categories for a collection, and each category can contain any number of rules. The rules determine which documents are associated automatically with the category.

Migrating a model-based taxonomy from WebSphere Portal

You can select which model-based taxonomy you want to use with an enterprise search collection by using the WebSphere Portal Taxonomy Management Portlet. Collections that you already migrated to enterprise search are not affected by a new taxonomy migration.

Before you begin

| Before you run the migration wizard for the first time, stop the enterprise search
| system so that changes to configuration files can be made. When you first run the
| migration wizard, you must specify the path where WebSphere Application Server
| and WebSphere Portal are installed. When you run the migration wizard after the
| first time, the enterprise search system can be active.

About this task

To migrate a model-based taxonomy, you must select and export the taxonomy in WebSphere Portal. Then, use the enterprise search migration wizard to migrate the taxonomy to enterprise search.

Procedure

To migrate a model-based taxonomy from WebSphere Portal to enterprise search:

1. Export your current model-based taxonomy from the WebSphere Portal Taxonomy Management portlet. The taxonomy comprises the following XML files:

```
synonyms.xml  
titles.xml  
treenodes.xml
```

2. Copy these files to the enterprise search index server.
3. On the enterprise search index server, log in as the enterprise search administrator. This user ID was specified when WebSphere II OmniFind Edition was installed.
4. Change to the enterprise search installation directory:

```
UNIX: cd $ES_INSTALL_ROOT/bin  
Windows: cd %ES_INSTALL_ROOT%\bin
```

5. If you are starting the migration wizard for the first time, enter the following command to stop the enterprise search system:

```
esadmin stop
```

6. Enter the following command to start the migration wizard, then click **Next**.

```
UNIX: ./eswpsmigrate.sh  
Windows: eswpsmigrate.bat
```

If this is not the first time that you are running the migration wizard, and the enterprise search system is active, ensure that you do not create an enterprise search collection while the migration wizard is running.

7. Select **Import model-based taxonomy files from WebSphere Portal**, then click **Next**.
8. If you are starting the migration wizard for the first time, specify the paths to where WebSphere Application Server and WebSphere Portal are installed. Click **Next**.
9. Browse for the directory that contains the model-based taxonomy files, select the XML files that you must migrate to use the model-based taxonomy (synonyms.xml, titles.xml, and treenodes.xml), then click **Next**.
10. If this was the first time that you ran the migration wizard, enter the following command after the migration is finished to restart the enterprise search system:

```
esadmin start
```

If errors occur, see the MigrationWizard.log file that is in the directory where the migration wizard is installed.

Related concepts

“Model-based categories” on page 101

If you use model-based categories in your IBM WebSphere Portal system, you can continue to use those categories with enterprise search collections.

Related tasks

“Configuring categories” on page 103

You can create any number of categories for a collection, and each category can contain any number of rules. The rules determine which documents are associated automatically with the category.

Migrating a collection from WebSphere Portal

To migrate collections from WebSphere Portal to enterprise search, prepare the collections in WebSphere Portal, then use the migration wizard to migrate them.

Before you begin

If you plan to migrate model-based taxonomies and collections, you must migrate the model-based taxonomy files before you use this procedure to migrate collections. If you do not do this, model-based categorization will not work with the collections that you migrate from WebSphere Portal.

Procedure

To migrate a collection from WebSphere Portal to enterprise search:

1. In WebSphere Portal Search Engine, stop all of the crawler processes in the collections that you want to migrate, and approve or reject all pending documents. (Enterprise search does not support the concept of pending documents.)
2. For each collection that you want to migrate, use the Portal Search Engine portlets to export the settings to XML files.
3. If the enterprise search index server is installed on a separate server, copy the exported XML files to the index server.
4. On the enterprise search index server, log in as the enterprise search administrator. This user ID was specified when WebSphere II OmniFind Edition was installed.

5. Change to the enterprise search installation directory:

```
UNIX: cd $ES_INSTALL_ROOT/bin  
Windows: cd %ES_INSTALL_ROOT%\bin
```

6. Enter the following command to start the migration wizard, then click **Next**.

```
UNIX: ./eswpsmigrate.sh  
Windows: eswpsmigrate.bat
```

7. Select **Migrate the search settings from the Portal Search Engine in WebSphere Portal**, then click **Next**.
8. Browse to the directory that contains the exported Portal Search Engine configuration files, select the files that you want to migrate, then click **Next**. The selected configuration files are analyzed and validated.
9. Enter the following information for each collection, then click **Next** to start migrating the collections to enterprise search:

- The name of the collection as you want to use it in enterprise search.
- The criterion by which the document importance is determined for the collection. The static ranking factor can be none, based on document dates, or based on links to Web documents from other Web documents.
- The type of categories that you want to use for this collection. You can select either none, rule-based categories, or model-based categories.

If you select rule-based categories, the taxonomy and the rules of the WebSphere Portal collection are migrated to enterprise search.

You can select model-based categories only if you previously migrated the model-based taxonomy from WebSphere Portal to enterprise search.

If errors occur during migration, see the MigrationWizard.log file that is in the directory where the migration wizard is installed.

You can now use the enterprise search administration console to configure additional settings for the migrated collections.

Requirement: When you configure Web crawler properties for a collection that you migrated, you must specify an e-mail address for receiving comments about the crawler and a user agent name (for assistance, click **Help** while you configure Web crawler properties).

10. Start the crawling, parsing, and indexing processes for the migrated collection from the enterprise search administration console.
11. After you determine that the migrated collection is searchable in enterprise search, delete the original collection in the Portal Search Engine.
12. Optional: As a WebSphere Portal administrator, take the following steps if you want to enable users to search the migrated collection from a portal in WebSphere Portal.
 - a. Deploy the enterprise search portlet in your WebSphere Portal installation.
In a WebSphere Portal server cluster, this should be done on the server where the WebSphere Application Server deployment manager is installed. The deployment manager distributes the enterprise search portlet to the other servers in the WebSphere Portal server cluster.
 - b. Add the enterprise search portlet to the appropriate portal pages.
In WebSphere Portal, access control of the search portlet is modeled by accessibility to specific pages and portlets. Although collection settings are migrated, the portlet must be positioned manually by the WebSphere Portal server administrator.

Related concepts

“Enterprise search collections” on page 27

An enterprise search collection contains the entire set of sources that users can search with a single query. Through federation, users can search multiple collections with a single query.

“Rule-based categories” on page 99

You can configure rules to control which documents are associated with categories in an enterprise search collection.

“Model-based categories” on page 101

A category tree enables you to view all of the rule-based categories in a collection. You use the category tree to create categories, delete categories, and edit the rules that associate documents with categories.

“Static ranking” on page 150

For certain types of documents, you can associate a static ranking factor that increases the importance of those documents in the search results.

Migrated collection settings

When you migrate collections from IBM WebSphere Portal, the migration wizard creates default settings for collections and crawlers.

If the same setting exists in Portal Search Engine collections and enterprise search collections, then the wizard uses the Portal Search Engine setting when it migrates the collection to enterprise search. For settings that exist only in enterprise search, the wizard uses the settings that you specify when you migrate the collection or the default settings for collections in enterprise search.

Settings that exist in Portal Search Engine and enterprise search

The migration wizard migrates the following settings for each collection that you migrate:

- The Portal Search Engine sites within the Portal Search Engine collection
- The collection language
- The taxonomy (or category tree) and the rules for the rule-based categories, if the enterprise search collection uses rule-based categorization

Each Portal Search Engine site in a collection is consolidated into an enterprise search Web crawler. The migration wizard migrates the following crawler settings:

- The start URL
- The number of parallel crawling processes
- The crawling depth
- The timeout (in seconds) for retrieving a document
- The default character set
- Include and exclude rules for crawling

Settings that exist only in enterprise search

When you migrate a collection, you specify information about the collection. The migration wizard migrates those settings and uses the default settings for collections in enterprise search to configure each collection that you migrate.

You can modify the collection and Web crawler configurations by using the enterprise search administration console. The values that are shown in parentheses () are the default settings for the migrated data.

- The collection name
- The document static ranking strategy
- The type of categorization that is used, such as rule-based or none
- Whether to use the search cache and how many queries with search results the search cache can hold (yes, 5000)
- Whether to monitor search response times and issue an alert if a limit is exceeded (yes, 5 seconds)
- Whether to use access controls (no)
- A schedule to refresh the index
- A schedule to reorganize the index
- The log detail level (all messages)

The migration wizard also creates the following settings for each Web crawler:

- The crawler name
- The crawler description
- The maximum page length
- The document security settings
- The document multipurpose Internet mail extensions (MIME) types that need to be crawled, if applicable to the data source type

Before you start a newly migrated Web crawler, review all of the crawler properties and crawl space settings and ensure that all required values are specified (required fields are marked with a red asterisk). In particular, ensure that you specify an

e-mail address for receiving comments about the crawler and a user agent name for the crawler. For assistance, click **Help** while you configure Web crawler properties.

Migration wizard log file

The migration wizard writes all messages to the WpsMigratorLog.log file in the directory where the migration wizard is installed.

For each migrated collection, the WpsMigratorLog.log log file contains the values of all of the settings that were read from the WebSphere Portal Search Engine, and specifies where these settings were imported to enterprise search collections.

Starting and stopping the enterprise search servers

After you create a collection, you must start the servers for crawling, parsing, indexing, and searching data. Stop and restart the server after you make changes to the collection.

Most enterprise search servers can run continuously or in accordance with schedules that you specify. For example, you can specify schedules for reorganizing or refreshing the index. After you start the servers for parsing data and searching the index, you typically need to stop and restart them only when you change the configuration settings (such as updating categories or increasing the size of the search cache).

If you make changes to the content of a collection, or if you change the rules for how the crawlers are to collect data from the sources in your enterprise, you typically need to stop and restart the crawlers for the changes to become effective. If you do not change the crawling rules, the crawlers either run continuously (in the case of Web and NNTP crawlers) or according to schedules that you specify.

Starting the enterprise search servers

To enable users to search a collection, you must start the system processes and then start the servers that crawl, parse, index, and search the collection.

Before you begin

Configure the data sources that you want to crawl and specify options for how you want that data to be parsed, indexed, and searched. For example, if you want users to be able to view category details in the search results, configure categories before you start the parser.

To start the enterprise search servers, you must be a member of the enterprise search administrator role, a collection administrator for the collection, or an operator with authority to administer that collection.

You must start the enterprise search servers for a collection in the correct order. For example, you must start a crawler and crawl data before you can parse and index documents.

Procedure

To start the enterprise search servers:

1. To use the enterprise search administration console or search application:
 - a. If it is not already running, start the IBM HTTP Server.
 - b. If they are not already running, use the WebSphere Application Server Administrative Console to start the ESAdmin and ESSearchApplication enterprise applications.
2. If the enterprise search common communication layer (CCL) is not running, start it:
 - a. On the index server, log in with the enterprise search administrator user ID that was specified when WebSphere II OmniFind Edition was installed.

UNIX systems





Enter the following command:


```
startccl.sh -bg
```

Windows systems

Select **Start** → **Programs** → **Administrative Tools** → **Services**, then restart the IBM WebSphere Information Integrator OmniFind Edition service.

3. If the enterprise search system is not running, start it:
 - a. On the index server, log in with the enterprise search administrator user ID that was specified when WebSphere II OmniFind Edition was installed.
 - b. Enter the following command:

```
esadmin start
```
4. Start the enterprise search administration console and log in as the enterprise search administrator. If you use administrative roles, you can log in as a collection administrator or operator who has authority for the collection that you want to start.
5. On the Collections view, locate the collection that you want to administer and click  **Monitor**.
6. On the Crawl page, for each crawler that you want to start, click  **Start**.
 - If you start a Web or NNTP crawler, the crawler begins crawling data immediately. These types of crawlers run continuously to crawl and recrawl documents on Web sites and NNTP news groups.
 - If you start one of the other crawler types, the crawler session starts. The crawler will begin crawling at its scheduled date and time. If you did not schedule the crawler, or if you want to start the crawler sooner, monitor the crawler, and click the start icon for each data source that you want to crawl. After the crawler starts, you can let it run continuously. If you scheduled the crawler, the crawler will run again at the scheduled dates and times.
7. After data is crawled, open the Parse page and click  **Start** to start the parser. You can let the parser run continuously. You typically do not need to stop the parser unless you make changes to how the data is parsed (such as updating categories or XML field mappings).
8. Optional: To force the indexing processes to start, instead of waiting for indexing to begin at the scheduled date and time, open the Index page and, in the **Reorganization** area, click  **Start**.

You can let the indexing processes run continuously. The index will be refreshed and reorganized at the scheduled dates and times.
9. On the Search page, click  **Start**.

You can let the search servers run continuously. You typically do not need to stop the search servers unless you make changes to the search cache or document summary settings.

Related concepts

“Enterprise search administration overview” on page 15

You use the enterprise search administration console to create and administer collections and external sources, start and stop components, monitor system activity and log files, configure administrative users, associate search applications with collections and external sources, and specify information to enforce security.

“Administrative roles” on page 182

Enterprise search uses the concept of roles to control access to various functions in the administration console.

Related tasks

“Logging in to the administration console” on page 18

To administer an enterprise search system, you specify a URL in a Web browser and then log in to the administration console.

Related reference

“Enterprise search commands, return codes, and session IDs” on page 249

You can use commands to diagnose problems, determine the status of the different parts of the system, start and stop sessions, or start and stop the system.

Stopping the enterprise search servers

You might need to stop and restart an enterprise search server if you make changes to its configuration or if you need to troubleshoot problems.

Before you begin

To stop the enterprise search servers, you must be a member of the enterprise search administrator role, a collection administrator for the collection, or an operator with authority to administer that collection.

About this task


You can stop the enterprise search servers independently of each other. For example, if you stop and restart a crawler to incorporate changes that you made to its configuration, you do not need to stop and restart the parser.

If you want to stop the enterprise search system instead of stopping individual servers, you can log in on the index server with the enterprise search administrator ID (this ID was specified when WebSphere II OmniFind Edition was installed). Then, enter the following command:

```
esadmin stop
```

Procedure

To stop enterprise search servers:

1. In the Collections view, locate the collection that you want to administer and click  **Monitor**.
2. On the Crawl page, locate the crawler that you want to administer, and stop or pause it.

If you change the crawl space or crawler properties, stop and restart the crawler to incorporate the changes. If you change the crawl space and want to apply the changes to documents that are already in the collection, you must also recrawl the documents.

Tip: You might see a message about the requested operation timing out even though the process is still running in the background. To determine whether the task has completed, click **Refresh** in the administration console (do not click **Refresh** in the Web browser). The process is finished when the status icon for the crawler indicates that it is stopped.

3. On the Parse page, click **Stop** to stop the parser.
When you change the rules for parsing data, stop and restart the parser to incorporate the changes. The changes apply only to newly crawled documents. If you want to apply the changes to documents that are already in the index, you must start a full crawl to recrawl all of the documents, which enables them to then be parsed and indexed again.
4. On Index page, click **Stop** to stop an index that is being refreshed or reorganized.
You can also stop an index build while you are monitoring the index queue. To do this, select **System** on the toolbar, open the Index page, and then click **Stop** for the index that you want to stop building.
5. On the Search page, click **Stop** to stop the search servers. Typically, you need to stop and restart the search servers only when you change the search cache or document summary settings.

Related concepts

“Enterprise search administration overview” on page 15

You use the enterprise search administration console to create and administer collections and external sources, start and stop components, monitor system activity and log files, configure administrative users, associate search applications with collections and external sources, and specify information to enforce security.

Related reference

“Enterprise search commands, return codes, and session IDs” on page 249

You can use commands to diagnose problems, determine the status of the different parts of the system, start and stop sessions, or start and stop the system.

Monitoring enterprise search activity

When you monitor system and collection activities, you can view the status of various processes, watch for potential problems, or adjust configuration settings to enhance performance.

With the enterprise search administration console, you can monitor the system and adjust operations as needed. You can view detailed statistics for each major activity (crawling, parsing, indexing, and searching). The statistics include average response times and progress information, such as how many documents were crawled or indexed during a crawl or index-building session.

By clicking icons, you can stop and start most activities. These operations enable you to pause an activity, make changes to its configuration or troubleshoot a problem, and restart processing when you are ready to allow the activity to proceed.

Related concepts

“Enterprise search administration overview” on page 15

You use the enterprise search administration console to create and administer collections and external sources, start and stop components, monitor system activity and log files, configure administrative users, associate search applications with collections and external sources, and specify information to enforce security.

Related tasks

“Starting the enterprise search servers” on page 213

To enable users to search a collection, you must start the system processes and then start the servers that crawl, parse, index, and search the collection.

“Stopping the enterprise search servers” on page 215

You might need to stop and restart an enterprise search server if you make changes to its configuration or if you need to troubleshoot problems.

Related reference

“Enterprise search commands, return codes, and session IDs” on page 249

You can use commands to diagnose problems, determine the status of the different parts of the system, start and stop sessions, or start and stop the system.

Estimating the number of documents in a collection

When you create or edit an enterprise search collection, you provide an estimate for how many documents you expect the collection to hold. The Resource Manager uses this number to estimate the memory and the disk resources that are required for the collection, but not to enforce a limit on the size of the collection.

Before you begin

To change the estimated size of a collection, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

About this task

When the collection grows to the size that you estimate, the system does not stop adding documents to the index. The Resource Manager issues warnings when your estimate of memory and disk resource requirements is more than what is currently available in the system. The warnings allow you to prevent future out-of-resource problems.

If you configure alerts for the collection and enable the option to be notified when the number of documents in the index exceeds a limit, the default limit matches the value that you specify for the estimated number of document in the collection. The Monitor uses this number in conjunction with the alert threshold percentage that you specify and sends e-mail when the maximum number of documents configured for the collection is being approached.

Procedure

To provide an estimate for the potential size of a collection:

1. Edit a collection, select the General page, and click **Configure general options**.
2. In the **Estimated number of documents** field, type a number that represents how large you expect the collection to grow. The default value is 1 000 000 documents.

Checking the availability of system resources

If you check system resources after you create a collection or crawler but before you start crawling documents, you can determine whether the available system resources are adequate for running WebSphere II OmniFind Edition at full capacity based on your current configuration settings.

Before you begin

To check system resources, you must be a member of the enterprise search administrator role.

About this task

When you create a collection or crawler, the system automatically checks the availability of resources. You can also select an option to check the availability of system resources at any time.

The system compares how much space you estimated that you would use when you configured collection and crawler properties to the available system space, and then displays a message that informs you about the availability of resources. The message indicates which server might have insufficient resources, how much space your estimate of the size of the collection requires, and how much free space is available.

If resources are insufficient, try one of the following corrective actions:

- Increase the size of the file system identified in the message text.
- Edit general options for the collection and specify a smaller number for the estimated number of documents.
- Edit crawler properties and specify smaller numbers for the maximum number of documents to crawl, the maximum page size, and the maximum number of threads.

- For a Web crawler, edit the crawler properties to specify smaller numbers for the maximum number of active hosts, maximum number of new documents, and maximum number of documents in temporary storage.
- Edit index properties for the system and reduce the number of index builds that are allowed to run concurrently.
- Avoid having too many processes in multiple collections simultaneously active at any given time. For example, resources can be impacted when crawlers and parsers from multiple collections run at the same time.
- Delete crawlers from the system.
- Delete collections from the system.

If no warnings about potential low resources were detected, there are probably sufficient resources to accommodate the data to be crawled and indexed.

Procedure

To determine whether sufficient resources are available for the current configuration of your enterprise search system:

1. Click **System** to open the System view.
2. On the General page, click **Check system resources**.

Monitoring a collection


You can view general information about the status of each component in a collection or select options to view detailed information about individual components and URIs.


Before you begin


All enterprise search administrative users can monitor collections. To start or stop components, or to enable or disable schedules, you must be a member of the enterprise search administrator role, a collection administrator for the collection, or an operator for the collection.

Procedure

To monitor a collection:

1. In the Collections view, locate the collection that you want to monitor to and click  **Monitor**. Information about the current status of each collection component is displayed.

Tip: If you are editing a collection and are already on the General page, you can click  **Monitor** to change to the view for monitoring the collection.

2. To see detailed information about a URI, click  **URI details**.
For example, you might want to see whether a specific URI is in the index, or whether the index that the URI is in was copied to the search servers.
3. To monitor an individual component and see detailed statistics about that component's activity, click the **Status** icon.

Related concepts

"Enterprise search collections" on page 27

An enterprise search collection contains the entire set of sources that users can search with a single query. Through federation, users can search multiple collections with a single query.

Viewing details about a URI

You can view detailed information about a URI. You can see current and historical information about how the document that is represented by this URI is crawled, indexed, and searched.

Before you begin

Before you submit a request to view a URI report or send a report to an e-mail address, ensure that the component that you want to receive information from is active. For example, to view details about how a document is crawled, indexed, and searched, ensure that the Web crawler, index server, and search servers are running. To track a dropped document, ensure that logging options for document tracking are configured.

About this task

Collecting information about a URI is a time-consuming process. You can choose an option to view the information that you request, then wait for it to be displayed. A more efficient option is to send the report to an e-mail address that you specify.


Before you can receive a report, you must ensure that information about your mail server has been configured for enterprise search. You specify this information when you configure e-mail options on the Log page of the System view.


The index server and search servers can provide information about all URIs (such as whether a URI is in the index and whether it has been copied to the search servers). To view information about how a document was crawled, you must specify the URI for a document that was crawled by a Web crawler.

Procedure

To view details about a URI:

1. In the Collections view, locate the collection that you want to monitor to and click  **Monitor**.

Tip: If you are editing a collection and are already on the General page, you can click  **Monitor** to change to the view for monitoring the collection.

2. Click  **URI details**.
3. On the URI Details page, type the URI that you want to view information for.
4. Select the check boxes for the type of information that you want to see:

Crawler details (available for Web crawlers only)

Select this check box to see information about how a document was crawled by a Web crawler, and information about its current status in the crawl space.

Index details

Select this check box to see whether a document was indexed and copied to the search servers.

Search details

Select this check box to see information about how the document can be searched and whether the document is available for searching.

Documents dropped by the parser

Select this check box to see whether the document was dropped from the enterprise search system while it was being parsed and, if so, the reason that it was dropped.

Documents dropped from the index

Select this check box to see whether a document was dropped from the enterprise search system while it was being indexed or analyzed and, if so, the reason that it was dropped.

5. To wait for the report to be displayed, click **View report**.
6. To send the report to an e-mail address so that you can view at a later time, click **Send report**.
 - a. On the Send a Detailed URI Report page, type an e-mail address for receiving the report in the **E-mail address to notify** field.
 - b. Click **Send Report**.

Related concepts

“Document tracking” on page 234

Documents can be dropped from the system at various stages in processing. You can specify options to learn when a document was dropped and what problems caused it to be dropped.

Related tasks

“Viewing reports about dropped documents” on page 235

You can view detailed information about documents that are dropped from an enterprise search system. This information is available only if you enabled document tracking for the collection.

Related reference

“URI formats in an enterprise search index” on page 89

The uniform resource identifier (URI) of each document in an enterprise search index indicates the type of crawler that added the document to the collection.

Monitoring crawlers


You can view general information about the status of each crawler in a collection or select options to view detailed information about a crawler activity.


Before you begin


If your administrative role limits you to monitoring collections, you can view crawler statistics but you cannot change a crawler’s behavior (such as starting or stopping the crawler).

Procedure

To monitor a crawler:

1. In the Collections view, locate the collection that you want to monitor and click  **Monitor**.
2. Open the Crawl page.

Tip: If you are editing a collection and are already on the Crawl page, you can click  **Monitor** to change to the view for monitoring crawlers.

3. If the crawler is running or paused and you want to see detailed status information about the crawler, click  **Details**. The types of statistics that you see vary with the crawler type.

If your administrative role allows you to administer processes for a collection, you can start, stop, and pause the crawler while you view details about crawler activity. If the crawler can be scheduled, you can also enable and disable the crawling schedule.

4. If the crawler is stopped or paused and you want to start a crawler session, click  **Start** or **Resume**.

For Web crawlers:

If the crawler was stopped, the crawler begins crawling again and crawls the entire crawl space. If the crawler was paused, it resumes crawling at the beginning of the target where it was paused.

If you want to force the crawler to start a full crawl immediately, click the **Details** icon, and then click the **Start a full recrawl** icon. The crawler starts crawling the entire crawl space, including pages that did not change since the last time that they were crawled. You might want to recrawl all documents, for example, if you change the rules for parsing documents and want to apply those rules to documents that were previously indexed.



For NNTP crawlers:

If the crawler was stopped, the crawler begins crawling again and crawls the entire crawl space. If the crawler was paused, it resumes crawling at the beginning of the target where it was paused.

For all other crawler types:

If the crawler was stopped, the crawler begins crawling at its scheduled date and time. The first time that the crawler crawls a data source, the crawler does a full crawl. When a scheduled crawl repeats, the crawler crawls either all updates to the data source (document additions, deletions, and modifications), or only document additions and modifications. You configure the type of crawl in the crawler schedule.

If you did not schedule the crawler, or if you want to start the crawler sooner, click the **Details** icon. Then, in the crawl space details area, click the icon for the type of crawl that you want to start: a full crawl, all updates, or new and modified documents only. You must click the appropriate start icon for each data source that you want to crawl (such as a server, database, or subfolder).

5. If the crawler is running and you want to stop it, click  **Stop** or  **Pause**. The crawler stops crawling data until you restart or resume the crawler.

If you resume a paused crawler, the crawler resumes crawling at the beginning of the target where it was paused. For example, the DB2 crawler resumes crawling at the first row in the table that was being crawled when you paused the crawler.

Related concepts

“Enterprise search crawler administration” on page 33

You configure crawlers for the different types of data that you want to include in a collection. A single collection can contain any number of crawlers.

Related reference

“Enterprise search commands, return codes, and session IDs” on page 249

You can use commands to diagnose problems, determine the status of the different parts of the system, start and stop sessions, or start and stop the system.

Viewing details about Web crawler activity

By viewing details about Web crawler activity, you can assess overall performance and adjust the Web crawler properties and crawl space definitions as necessary.


Before you begin


All enterprise search administrative users can monitor crawler activities. To start or stop a crawler, you must be a member of the enterprise search administrator role, a collection administrator for the collection, or an operator for the collection.

Procedure

To view details about a Web crawler's activity:

1. In the Collections view, locate the collection that owns the Web crawler that you want to monitor to and click  **Monitor**.
2. Open the Crawl page.

Tip: If you are editing a collection and are already on the Crawl page, you can click  **Monitor** to change to the view for monitoring crawlers.

3. If the Web crawler that you want to monitor is running or paused, click  **Details**.
4. On the details page for the Web crawler, view or select the following options to see detailed statistics about the crawler's current and past activity.
 - a. Click **Thread details** to see how many threads are actively crawling Web sites and how many are in an inactive state.
 - b. Click **Active sites** to see information about the Web sites that the crawler is actively crawling.
 - c. Click **Recently crawled URLs**. This information shows what the crawler recently crawled. If the items in the list do not change as you refresh the view, then no crawling is occurring.
 - d. Click **Crawler history** to view reports about past crawler activity.
 - e. In the **URL status** area, type the URL for a Web site that you want to see information about, then click **View**.

For example, use this option to see whether a URL is in the crawl space, whether it has been crawled or only discovered, when it should be crawled again, and information about the last attempt to crawl the Web site.

After details about the URL are displayed, you can click **Site history** to see additional statistical information about crawler activity at that URL.

Web crawler thread details

You can monitor the Web crawler to see how many threads are actively crawling Web sites and how many are in an inactive state.

When you view details about a Web crawler while monitoring a collection, you can view the status of the crawler threads. The states that you are most likely to see include:

Waiting

Indicates that the thread does not have a URL to crawl. This condition can occur when a thread finishes a crawl and the crawler cannot find more URLs to crawl fast enough. For example, if the crawler property that

controls how long the crawler must wait before it can retrieve another page from same site is too high, it can prevent URLs from being supplied fast enough.

Fetching

Indicates that the thread is downloading a page from a Web site.

Completed

Indicates that the thread is sending the pages that it crawled to the rest of the crawler, but is not yet ready to crawl another URL.

Suspended

Indicates that the crawler is paused

Ideally, all threads are fetching pages all of the time. If threads are often in a completed state, then the database might be having throughput problems.

If threads are often in a waiting state, review the value specified for the **Maximum number of active hosts** field in the crawler properties. If the value is low, there might not be enough sites in the crawl space to keep the threads busy, or there might not be enough URLs eligible to be crawled. Conditions that can cause low activity include DNS lookup failures and robot lookup failures.

Web crawler active sites

You can monitor the Web crawler to see information about the Web sites that the crawler is actively crawling.

When you view details about a Web crawler while monitoring a collection, you can view statistics about active sites. The statistics show:

- How many URLs the crawler brought from its internal database to memory for crawling at this time
- How many URLs the crawler has attempted to crawl so far
- How much time remains before a site is deactivated and removed from memory for this iteration of the crawler
- How much time a site has been in memory so far

This information changes from moment to moment as the crawler progresses through the crawling rules that are configured for it. Ideally, the number of activated URLs is close to the value that is configured for the **Maximum number of active hosts** field in the crawler properties.

If the number of activated URLs is near zero, then the crawler is not finding eligible URLs. Conditions that can cause such low activity include DNS lookup failures, network connectivity issues, database errors, and crawl space definition problems. For example:

- If many sites have been in memory for a long time, and few URLs have been crawled, look for network connectivity problems.
- If not enough sites are in the list, look for crawl space definition problems or DNS lookup problems.
- If sites are being crawled at a reasonable rate, but are leaving memory with many URLs not being crawled, edit the crawler properties and adjust the timeout value in the **Maximum time that URLs can remain in memory** field to keep the sites in memory longer.

Web crawler crawl rate

You can monitor the Web crawler to see information about how fast the crawler is downloading pages from Web sites.

When you view details about a Web crawler while monitoring a collection, you can view statistics about how fast the crawler is crawling data (the crawl rate). You can also view statistics about how many URLs the crawler crawled since the current session began.

The crawl rate is the number of pages that are being crawled per second. This number correlates to several properties that you can configure for the Web crawler:

- The number of crawler threads
- The number of active sites
- The amount of time that the crawler must wait before it can retrieve another page from the same Web server

If the crawler has one active site per crawler thread, and the crawler must wait two seconds before it can retrieve another page from the same Web server, then the crawler cannot crawl faster than one page per thread per two seconds. For example, if the crawler uses the default number of threads (200), then crawler can crawl 100 pages per second for 200 threads.

If there are twice as many active sites as crawler threads, and the crawler must wait two seconds before it can retrieve another page from the same Web server, then the crawler could reach one page per thread per second. However, network download speeds and database throughput would then become limiting factors. An indication of strong crawler performance is when the crawl rate aligns with the number of crawler threads, active sites, and crawler wait time.

Another factor to review when you monitor Web crawler performance is the number of URLs that the crawler crawled since the start of the current crawler session. Divide that number by the total amount of the time that the crawler has been running to calculate an average of the long-term throughput. If this number is not increasing, the crawler is either finished, or it is unable to proceed. For example, network connectivity errors, database errors, and DNS lookup failures can block the progress of the crawler.

Creating Web crawler reports

By viewing reports about past Web crawler activity, you can assess overall performance and adjust the Web crawler properties and crawl space definitions as necessary.

Before you begin

If your administrative role limits you to monitoring collections, you can view crawler statistics and create reports about crawler activity, but you cannot change the crawler's behavior (such as starting or stopping the crawler).

About this task

Different types of reports can provide you with information about Web crawler activity. For certain types of reports, information is returned as fast as it can be collected from the crawler's internal database. The Site report and HTTP return code reports take time to create. If you create these types of reports, you can


specify an e-mail address for receiving the report instead of waiting for results to be returned to the enterprise search administration console.


For information about how to interpret statistics in the reports, click **Help** while you are monitoring the Web crawler and creating the reports.

Procedure

To create Web crawler reports:

1. In the Collections view, locate the collection that owns the Web crawler that you want to monitor to and click  **Monitor**.
2. Open the Crawl page.

Tip: If you are editing a collection and are already on the Crawl page, you can click  **Monitor** to change to the view for monitoring crawlers.

3. If the Web crawler that you want to create reports for is running or paused, click  **Details**.
4. On the details page for the Web crawler, select an option for the type of report that you want to create:
 - In the **Crawler status summary** area, click **Crawler history** to create reports about the crawler and all of the sites that it discovers or crawls.
 - In the **URL status** area, specify the URL of specific site that you want to create a report for, click **View**, and then click **Site history**.
5. For both crawler history and site history reports, select the check box of each statistic that you want to see in a report, then click **View report**.

For these types of statistics, the crawler returns a report to the administration console as fast as it can retrieve information from its internal database.

6. If you are creating a crawler history report, specify options for creating a Site report, then click **Run Report**.

This report is created with the statistics that you choose to include and saved in a file that you specify (the file name must be absolute). You can specify that you want to receive e-mail after the report is created.

7. If you are creating a crawler history report, specify options for creating an HTTP return code report, then click **Run Report**.

This report provides information about the number of HTTP return codes distributed per site. The report is saved in a file that you specify (the file name must be absolute). You can specify that you want to receive e-mail after the report is created.

Use this report to see which sites return a large number of 4xx return codes (which indicate that pages were not found), 5xx return codes (which indicate a server problem), 6xx return codes (which indicate connectivity problems), and so on.

This report is most useful when the crawler has been active for some time (for example, a crawler that has been active for weeks). It can help you identify vanished sites, newly arrived sites, sites with huge numbers of URLs (which might indicate redundant crawling of a Lotus Notes database), and sites with a recursive file system served by the HTTP server. If the sites with large numbers of HTTP return codes are not contributing to the index, you can improve the performance of the crawler by removing the sites from the crawl space.

Web crawler HTTP return codes

When you monitor a Web crawler, you can view information about the HTTP return codes that the crawler receives from the pages that it attempts to crawl.

Table summary

When you monitor the Web crawler history, or monitor the status of a specific URL, you can see information about the HTTP return codes that were returned to the crawler. You can use this information to manage the crawl space and optimize crawler performance. For example, if the crawler receives a large number of HTTP return codes for a URL, and the return codes indicate that pages at that location cannot be crawled, you can improve performance by removing that URL from the crawl space.

The following table lists the HTTP return codes and how the Web crawler interprets them. Values from 100 to 505 are standard HTTP return codes (see <http://www.w3.org/Protocols/rfc2616/rfc2616.html> for more information). The remaining HTTP return codes are proprietary to enterprise search and the Web crawler.

Table 4. HTTP return codes from the Web crawler

| Code | Description | Code | Description | Code | Description | Code | Description |
|------|-------------------------------|------|-------------------------------|------|----------------------------|------|---|
| NULL | Uncrawled | 401 | Unauthorized | 500 | Internal server error | 700 | Parse error (no header end) |
| 100 | Continue | 402 | Payment required | 501 | Not implemented | 710 | Parse error (header) |
| 200 | Successful | 403 | Forbidden | 502 | Bad gateway | 720 | Parse error (no HTTP code) |
| 201 | Created | 404 | Not found | 503 | Service unavailable | 730 | Parse error (body) |
| 202 | Accepted | 405 | Method not allowed | 504 | Gateway timeout | 740 | Excluded by robots.txt file |
| 203 | Non-authoritative information | 406 | Not acceptable | 505 | HTTP version not supported | 741 | Robots temporarily unavailable |
| 204 | No content | 407 | Proxy authentication required | 611 | Read error | 760 | Excluded by crawl space definition |
| 205 | Reset content | 408 | Request timeout | 612 | Connect error | 770 | Bad protocol or nonstandard system port |
| 206 | Partial content | 409 | Conflict | 613 | Read timeout | 780 | Excluded by file type exclusions |
| 300 | Multiple choices | 410 | Gone | 614 | SSL handshake failed | 2004 | No index META tag |
| 301 | Moved permanently | 411 | Length required | 615 | Other read error | 3020 | Soft redirect |
| 302 | Found | 412 | Precondition failed | 616 | FBA anomaly | 4044 | Excluded by robots.txt file |
| 303 | See other | 413 | Request entity too large | 617 | Encoding error | | |

Table 4. HTTP return codes from the Web crawler (continued)

| Code | Description | Code | Description | Code | Description | Code | Description |
|------|--------------------|------|-------------------------|------|--------------------|------|-------------|
| 304 | Not modified | 414 | Request URI is too long | 680 | DNS lookup failure | | |
| 305 | Use proxy | 415 | Unsupported media type | | | | |
| 306 | (Unused) | 417 | Expectation failed | | | | |
| 307 | Temporary redirect | | | | | | |

Table notes

4xx return codes

You will rarely see a 400 (bad request) code. According the HTTP return code standard, 4xx codes are supposed to be indicate that the client (the crawler) failed. However, the problem is usually at the server or in the URL that the crawler received as a link. For example, some Web servers do not tolerate URLs that try to navigate up from the site root (such as `http://xyz.ibm.com/../../sales`). Others Web servers have no problem with this upward navigation and ignore the parent directory operator (`..`) when the crawler is already at the root.

Some servers treat a request for the site root as an error, and some obsolete links might request operations that are no longer recognized or implemented. When asked for a page that it no longer serves, the application server throws an exception, which causes the Web server to return the HTTP return code 400 because the request is no longer considered valid.

- 615** Indicates that the crawler server that downloads data from Web sites encountered an unexpected exception. A large number of this type of return code might indicate that there is a problem with the crawler.

6xx return codes

Except for 615, the 6xx return codes indicate problems that can be expected in crawling, such as timeouts. The following return codes might require corrective action:

611, 612, and 613

Indicate slow sites or poor network performance.

- 614** Indicates that the crawler is unable to crawl secure (HTTPS) sites. If you believe that these sites should be accessible, verify that the certificates are set up correctly on the crawler server and on the target Web server. For example, if a site is certified by a recognized certificate authorities (CAs), you can add new CAs to the trust store that is used by the crawler.

Also look at how self-signed certificates are configured on the sites that you are trying to crawl. The crawler is configured to accept self-signed certificates. Some sites create a self-signed certificate for a root URL (such as `http://sales.ibm.com/`), and then try to use that certificate on subdomains (such as `http://internal.sales.ibm.com/`). The crawler cannot accept certificates that are used in this manner. It accepts self-signed

certificates only if the domain name of the subject (sales.ibm.com) and the signer of the certificate match the domain name of the page that is being requested.

- 616** Indicates that the login form still appears in the download after reauthentication.
- 617** Indicates the inability to create a String from a document's byte content because the encoding string (charset) is invalid or the document contains invalid bytes.
- 680** Indicates that the crawler was not able to obtain IP addresses for hosts in the crawl space, perhaps because of network access problems. This type of error means that the crawler is not able to crawl entire sites, not just that it was unable to crawl some URLs. A large number of this type of return code greatly reduces throughput.

7xx return codes

The 7xx codes are mostly due to rules in the crawl space:

710 - 730

Indicate that problems prevented the crawler from doing a complete download, or that the crawler encountered invalid HTML data at a site. If you see a large number of these types of return codes, contact your enterprise search support representative for assistance.

740 or 4044

Indicate that the content of a file cannot be indexed because the document is excluded by restrictions in the site's robots.txt file.

740 Indicates that anchor links that point to the excluded document can be included in the index.

4044 Indicates that the anchor links in documents that point to the excluded document are also excluded from the index.

- 741** Indicates that a site has a robots.txt file that allows the crawl, but the download failed. If it is repeatedly unable to crawl the URL, the URL is removed from the crawl space. If you seen a large number of this type of return code, check to see whether the target site is temporarily or permanently unavailable. If the target site is no longer available, remove it from the crawl space.

The remaining 7xx return codes mostly occur when you make changes to the crawl space after the crawler has been running for awhile. These return codes typically do not indicate problems that you need to address.

- 3020** Indicates that a document with return code 200 contains a location header that refers the user agent to another URL.

Monitoring the parser

Monitor the parser when you need to view information about documents that are analyzed by the parser before they are added to the enterprise search index. Options enable you to review statistics and administer parser activity.

Before you begin

If your administrative role limits you to monitoring collections, you can view the status of the parser, but you cannot start or stop the parser.

About this task


When you monitor parser details, you see a snapshot of parser activity that provides statistics about parsing activities at a specific moment in time. The statistics show you the number of documents that were crawled and are being parsed or waiting to be parsed, and the number of documents that were parsed and are waiting to be stored in the index.


When the parser is active, messages provide you with additional information about the state of the parser. For example:


- The parser might be actively parsing documents.
- The parser might be idle. The parser sleeps until more documents are available to parse. If errors occur, the parser waits to be restarted. The parser restarts itself if no parser services are available (for example, an automatic restart occurs when a connection to the parser service cannot be established or if all of the parser Java virtual machines are busy with other collections).
- The parser might be paused (for example, the parser might be paused until an index reorganization is completed).

Procedure


To monitor the parser for a collection:

1. In the Collections view, locate the collection that you want to monitor and click  **Monitor**.
2. Open the Parse page.


Tip: If you are editing a collection and are already on the Parse page, you can click  **Monitor** to change to the view for monitoring the collection.

3. If the parser is running and you want to see detailed status information about parsing activity, click  **Details**.

If your administrative role allows you to administer processes for a collection, you can start and stop the parser while you view details about parsing activities.

4. If the parser is stopped and you want to start it, click  **Start**.

When you first create a collection, start the parser only after the crawler begins crawling data. This ensures that the parser has data to analyze and categorize. Unless you make changes to parsing rules, you can let the parser run continuously.

5. If the parser is running and you want to stop it, click  **Stop**.

You need to stop and restart the parser when you make changes to parsing rules. For example, if you change the parser configuration, you must stop and restart the parser before your changes become effective.

Monitoring index activity for a collection


Monitor the index for a collection when you need to see the progress of an index that is being built, enable or disable the index schedule, or start and stop indexing activity.


Before you begin






All enterprise search administrative users can monitor index activities. To start or stop an index build, or to enable or disable the index schedule, you must be a member of the enterprise search administrator role, a collection administrator for the collection, or an operator for the collection.

Procedure

To monitor the index for a collection:

1. In the Collections view, locate the collection that you want to monitor and click  **Monitor**.
2. Open the Index page.

Tip: If you are editing a collection and are already on the Index page, you can click  **Monitor** to change to the view for monitoring the collection.

3. If an index is scheduled, and you do not want it to be built at the scheduled date and time, click  **Disable schedule**. The index will not be built until you enable the schedule or start the index building process.
4. If an index is scheduled, but the schedule for building it is disabled, click  **Enable schedule**. The index will be queued for building at the date and time that you specified in the index schedule.
5. If an index is stopped and you want to start it, click  **Start**.
Typically, indexing occurs on a regularly scheduled basis. If you stop an index while it is being built, or if you disable the schedule for an index, you can click **Start** to force the index build to begin.
6. If an index build is active and you want to stop it, click  **Stop**.
You might need to stop an index build, for example, to force an index reorganization after you change the change the type of categorization used in the collection.
7. If errors occurred during an index build, click  **Error**.

The Contents of Log File page is displayed so that you can view additional information about the indexing errors. On that page, you can select individual error messages to see details about the problem.

Related concepts

“Enterprise search index administration” on page 125

To ensure that users always have access to the latest information, enterprise search creates an index for each collection and maintains that index by periodically refreshing and reorganizing the content.

Monitoring the enterprise search index queue

You can view the status of all index builds in the index queue, stop an index that is being built, or delete an index from the queue.

Before you begin

To administer the index queue, you must be a member of the enterprise search administrator role.

About this task

Multiple indexes can be built at the same time, but only one index per collection can be in the queue at a time. When you configure index options for the system, you specify how many indexes can share the queue and indexing resources concurrently.

Procedure


To monitor the index queue:

1. Click **System** to open the System view.
2. Select the Index page.


A list of the collections that have indexes in the index queue is displayed. For each index, you can see the type of index that is being built (refresh or reorganization), the time that the index entered the index queue, and the time that a build of the index began (if a build is in progress).

3. To administer an individual index, click the **Status** icon.

For example, you might want to see how close an index is to being completed, see how many documents are in the index, or disable the index schedule.

4. To stop an index that is being built, click  **Stop**.

For example, if you changed category rules, you might want to stop an index refresh so that you can force an index reorganization to start instead.

To start an index build after you stop it, either wait for the index to enter the index queue at its next scheduled start time, or click the **Status** icon to monitor the index, then click  **Start** to refresh or reorganize the index.

5. To remove an index from the index queue, click  **Remove**.

Related concepts

“Enterprise search index administration” on page 125

To ensure that users always have access to the latest information, enterprise search creates an index for each collection and maintains that index by periodically refreshing and reorganizing the content.

Monitoring the search servers


You can view detailed status information about search server activity for a specific collection, or view detailed status information for the search servers throughout your enterprise search system.


Before you begin



All enterprise search administrative users can monitor search servers for the collections that they are authorized to administer. To monitor all of the search servers in your enterprise search system, you must be a member of the enterprise search administrator role.

To start or stop a search server, you must be a member of the enterprise search administrator role, a collection administrator for the collection, or an operator for the collection.

Procedure

1. To monitor the search servers for a single collection:
 - a. In the Collections view, locate the collection that you want to monitor and click  **Monitor**.
 - b. Open the Search page.

Tip: If you are editing a collection and are already on the Search page, you can click  **Monitor** to change to the view for monitoring the collection.

2. To monitor all of the search servers in your enterprise search system:
 - a. Click **System** to open the System view.
 - b. Select the Search page.
3. If a search server is stopped and you want to start it, click  **Start**.
4. If a search server is running and you want to stop it, click  **Stop**.

If you enable or disable the search cache, make changes to the search cache size, or make changes to quick links, you must stop and restart the search servers for your changes to become effective.
5. To see a summary of how much time a search server spends processing search requests, click **Response timehistory**.

The report shows, in milliseconds, the average amount of time that the search server spent responding to search requests on a particular date.

The average response time is an indicator for how well the system is performing, and corresponds to quality of service. An increase in response time might indicate that the system is under heavy load. For example, the number of collections being searched and the collection size might be overwhelming the system.
6. To see a list of the most frequently submitted queries, click **Popular queries**.

The report shows you the keywords in the 50 most frequently submitted queries and how many times users submitted a particular query.

By reviewing the most frequent queries, you can identify candidates for quick links. By creating quick links, you can positively impact the search quality for many users. You can ensure that highly relevant documents are always returned in the search results.

You might also want to create links to the resources that answer those queries from the enterprise portal. For example, if users frequently search for information about expense accounts, include a link to the page that discusses expense account procedures on your intranet home page.
7. To see a list of the most recently submitted queries, click **Recent queries**.

The report shows you the keywords in the 50 most recently submitted queries.

By reviewing the most recent queries, you can identify current trends and urgent situations in the organization. For example, you might see a surge of interest being shown for some topic. That surge in interest might indicate that a quick link for that topic is needed or that you need to make that topic available to users in other ways (such as providing a link on the enterprise portal).

Monitoring the Data Listener


Monitor the Data Listener to see its status and to view details about client Data Listener application activity.

Before you begin

To monitor the Data Listener, you must be a member of the enterprise search administrator role.

Procedure

To monitor the Data Listener:

1. Click **System** to open the System view.
2. On the Data Listener page, view the status icons to see whether the Data Listener is active or stopped.
3. If the Data Listener is running and you want to see detailed status information about client application activity, click  **Details**.

Status icons on the Data Listener Details page indicate whether the Data Listener is running or stopped. The statistics show how many requests are waiting to be processed, the current state of each thread that is working on client application requests, and how many threads are active for a given thread state.

4. If you change the port number for the Data Listener, click  **Restart**.

The Data Listener is started when the enterprise search system is started. You do not need to restart the Data Listener unless you change its port number.

Related tasks

“Configuring support for Data Listener applications” on page 87

You can extend enterprise search by using the Data Listener API to create an external crawler. Your custom Data Listener applications can add data to a collection, remove data from a collection, or instruct a Web crawler to visit and revisit URLs.

Document tracking

Documents can be dropped from the system at various stages in processing. You can specify options to learn when a document was dropped and what problems caused it to be dropped.

If the parser encounters an error that prevents the document from being parsed, a message with a reason code is logged about the dropped document. (This type of error does not cause older versions of the document to be removed from the index.)

Documents can be dropped during the indexing stages, and this information is also logged. For example, URIs and URI patterns can be explicitly deleted. A document might have been crawled by a crawler that was later deleted. The source document might no longer exist (a negative HTTP code is associated with the document), or the HTTP code associated with the document might be unknown. Documents can also be dropped if rank information is missing for a document that requires global analysis.

If you know that a document was crawled, but the document does not appear in the index, you can use the enterprise search administration console to track the flow of the document through the system. Detailed reports can show you when, where, and why the document was dropped. For example, the report might indicate that the document was unexpectedly dropped during global analysis, or the report might indicate that an administrator removed the URI from the index.

Related tasks

“Viewing details about a URI” on page 220

You can view detailed information about a URI. You can see current and historical information about how the document that is represented by this URI is crawled, indexed, and searched.

Configuring log files for document tracking

To determine when, where, and why a document was dropped from the system, you can configure log files to track information about dropped documents.

Before you begin

To configure options for tracking dropped documents, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

About this task

To prevent log files from consuming too much disk space, the system rotates log files, and always starts a new log file whenever the current date changes. If one log file grows to its maximum allowable size, and the date did not change, the system creates a new log file. When the maximum number of log files is reached, the oldest log file is discarded so that a new one can be created.

Procedure

To configure log files for document tracking:

1. Edit a collection, select the Log page, and click **Configure document tracking**.
2. On the Document Tracking page, ensure that the check box for tracking documents is selected.
3. Specify the number of log files that are to be used to log information about documents that are dropped from the system. These log files are shared by all sessions in which documents can be dropped.
4. Click **OK**.

Viewing reports about dropped documents

You can view detailed information about documents that are dropped from an enterprise search system. This information is available only if you enabled document tracking for the collection.

Before you begin

Before you submit a request to view a report about dropped documents or send a report to an e-mail address, ensure that the sessions that you want to receive information from are active. For example, to learn about documents that were dropped during parsing or indexing, ensure that the parser and index sessions for the collection are started.

Before you can receive a report, ensure that information about your mail server is configured for enterprise search. You specify this information when you configure e-mail options on the Log page of the System view.

About this task


Collecting information about dropped documents is a time-consuming process. You can choose an option to view the information and wait for it to be displayed. A more efficient option is to send the report to an e-mail address that you specify.


If a document was dropped, the report shows the date and time that the document was dropped, the severity level of the error, the component and session where the problem occurred, and the error message.

Procedure

To view details about dropped documents:

1. In the Collections view, locate the collection that you want to monitor to and click  **Monitor**.

Tip: If you are editing a collection and are already on the General page, you can click  **Monitor** to change to the view for monitoring the collection.

2. Click  **URI details**.
3. On the URI Details page, type the URI that you want to view information for.
4. Select the check boxes for the type of information that you want to see:

Documents dropped by the parser

Select this check box to see whether the document was dropped while it was being parsed and, if so, the reason that it was dropped.

Documents dropped from the index

Select this check box to see whether a document was dropped while it was being indexed or analyzed and, if so, the reason that it was dropped.

5. Specify how you want to view the report:
 - To wait for the report to be displayed, click **View report**.
 - To send the report to an e-mail address so that you can view at a later time, click **Send report**.

On the Send a Detailed URI Report page, type an e-mail address for receiving the report in the **E-mail address to notify** field, and then click **Send Report**.

Related tasks

“Viewing details about a URI” on page 220

You can view detailed information about a URI. You can see current and historical information about how the document that is represented by this URI is crawled, indexed, and searched.

Related reference

“URI formats in an enterprise search index” on page 89

The uniform resource identifier (URI) of each document in an enterprise search index indicates the type of crawler that added the document to the collection.

Enterprise search log files and alerts

You can choose the types of messages that you want to log for a collection and for the system, specify options for creating and viewing log files, receiving alerts, and receiving e-mail about messages.

During normal operations, the enterprise search components write log messages to a common log file. This log file is in the `ES_NODE_ROOT/logs` directory on the index server. You can use the administration console to view this common log data.

If a problem occurs, such as a network communication failure, the components write log messages to a logs directory on the server where the component is installed. To view these local log files, use a file viewer on that computer, such as the `tail` utility on a UNIX system. You cannot use the administration console to view these types of log files.

When you configure log files, you can choose the types of messages that you want to log (such as error or warning messages), specify how often old log files are to be discarded to make room for new log files, specify a maximum size for the log files, and select the language of the messages. You can also specify options for receiving e-mail whenever certain events occur, or whenever certain messages or types of messages are logged.

When you monitor log files, you can choose which log file you want to open. You can filter the content of the log file so that you view only messages of a specific severity level (such as error messages only) or messages that were produced by a specific enterprise search session. When you view a log file, you can view details about individual messages. For example, you might want to see the name of the function that produced the message and other information that can help you take corrective action, if necessary.

Related concepts

"Messages for enterprise search" in "Messages Reference"

Alerts

You can configure enterprise search to write messages to the log file whenever it detects that certain events occurred.

Messages that are triggered by events, called alerts, inform you about conditions that you might want to address, such as a resource that is running out of free space. When you configure alerts for enterprise search, you specify the conditions that you want the system to monitor. Whenever the condition occurs, the system automatically writes a message to the log file.

If you want to be notified directly about a condition, you can specify options to receive e-mail whenever one of the monitored messages is logged.

You can configure alerts for collection-level events and for events that occur at the system level. At the collection level, the system can:

- Monitor the number of documents that each crawler crawls, and issue an alert message when the maximum number of documents allowed is about to be reached.
- Monitor the number of documents being added to the index for your collections, and issue an alert message when the maximum number of documents allowed is about to be reached.
- Inform you when the time that is required to respond to search requests is exceeding a limit that you specify.

At the system level, the system can monitor the disk space on each enterprise search server and issue an alert message when the amount of free space is low.

Configuring collection-level alerts

By configuring alerts, you can ensure that messages are written to the log file whenever certain collection-level events occur. You can also receive e-mail whenever messages about these events are logged.

Before you begin

To configure alerts for a collection, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

Procedure

To configure collection-level alerts:

1. Edit a collection, select the Log page, and click **Configure alerts**.
2. If you want the system to monitor the number of documents that each crawler is crawling, take the following steps:
 - a. Select the **When the number of documents crawled by any crawler reaches a percentage of the maximum allowed** check box.
 - b. In the **Percentage** field, specify when you want a message to be logged. Specify this number as a percentage of the maximum number of documents that the crawlers can crawl (you specify the **Maximum number of documents to crawl** when you configure the crawler properties). The default value is 90 percent.

Because you can configure different limits for different crawlers, separate messages are logged for each crawler. For example, if you use the default alert threshold, allow a DB2 crawler to crawl 2 000 000 documents, and allow a Notes crawler to crawl 1 000 000 documents, one message will be logged when the DB2 crawler crawls 1 800 000 documents and another message will be logged when the Notes crawler crawls 900 000 documents.
3. If you want the system to monitor the number of documents that are being added to the index, take the following steps:
 - a. Select the **When the number of documents in the collection reaches a percentage of the estimated size** check box.
 - b. In the **Percentage** field, specify when you want a message to be logged. Specify this number as a percentage of the estimated number of documents that you expect the collection to hold. The default value is 85 percent.

The **Limit** field shows the current estimated size of the collection. To change this value, open the General page of the collection, select the option to configure general options, and specify a new value in the **Estimated number of documents** field.

Attention: This limit, and the estimated number of documents that you configure for a collection, are used only for monitoring the growth of the collection. They do not enforce an absolute limit on how large the index can grow.

4. If you want the system to inform you when the time required to respond to search requests is exceeding a limit, take the following steps:
 - a. Select the **When the search response time exceeds a limit** check box.
 - b. In the **Limit** field, type the number of seconds that you consider acceptable as a maximum search response time.

When this number is exceeded, the system writes a log message about the event. For example, if you keep the default value, then the system creates a log message whenever a search server averages five seconds or longer to respond to search requests.

Typical response times are less than a half a second. Averages greater than one second might indicate that your operating system needs tuning for better performance or that a problem exists in the search server configuration settings. For example, you might want to increase the amount of space that you allocate for the search cache.

5. Click **OK**.

If you want to receive e-mail when the system logs messages about these events, open the Log page, then click **Configure e-mail options for messages** so that you can specify an e-mail address. The message IDs for the alerts that you enabled are automatically added to the list of message IDs for which e-mail is to be sent.

Before you can receive e-mail, you must also ensure that information about your mail server is configured. To do this, an enterprise search administrator must select **System** on the toolbar, open the Log page, then click **Configure e-mail options for messages**.

Related tasks

“Receiving e-mail about logged messages” on page 242

You can specify options to receive e-mail whenever certain messages, or certain types of messages, are logged.

Configuring system-level alerts


By configuring alerts, you can ensure that messages are written to the log file whenever certain system-level events occur. You can also receive e-mail whenever messages about these events are logged.

Before you begin

To configure system-level alerts, you must be an enterprise search administrator.

Procedure

To configure system-level alerts:

1. Click **System** to open the System view.
2. Click  **Edit** to change to the system editing view.
3. On the Log page, click **Configure alerts**.
4. If you want the system to monitor the amount of free space that is available on the enterprise search servers, select the **When the amount of available file system space reaches a percentage of the total space** check box.

5. In the **Percentage** field, specify when you want the system to notify you that the amount of free space on a server is low. Specify this number as a percentage of total file system space. The default value is 80 percent.
If your enterprise search system is set up on multiple servers, the system creates a separate log message for each server. For example, a message informs you when the space on the crawler server is low; and separate messages inform you about space constraints on the index and search servers.
6. Click **OK**.

If you want to receive e-mail whenever the system logs a message about this event, open the Log page, then click **Configure e-mail options for messages** so that you can specify an e-mail address and information about your mail server.

Related tasks

“Receiving e-mail about logged messages” on page 242

You can specify options to receive e-mail whenever certain messages, or certain types of messages, are logged.

Configuring log files

You can specify the types of messages that you want to log and specify options for creating log files.

Before you begin

To configure collection-level log files, you must be a member of the enterprise search administrator role or be a collection administrator for the collection. To configure system-level log files, you must be an enterprise search administrator.



About this task

To prevent log files from consuming too much disk space, the system rotates log files, and always starts a new log file whenever the current date changes. If one log file grows to its maximum allowable size, and the date did not change, the system creates a new log file. When the maximum number of log files is reached, the oldest log file is discarded so that a new one can be created.

To receive e-mail about logged messages, you first specify information about how the e-mail is to be delivered. You then specify which messages you want to receive e-mail for.

Procedure

To configure enterprise search log files:

1. If you want to configure options for creating and rotating system-level log files:
 - a. Click **System** to open the System view.
 - b. Click  **Edit** to change to the system editing view.
 - c. On the Log page, click **Configure log file options**. The System-Level Log File Options page is displayed.
2. If you want to configure options for creating and rotating collection-level log files:
 - a. In the Collections view, locate the collection that you want to specify options for and click  **Edit**.

- b. On the Log page, click **Configure log file options**. The Collection-Level Log File Options page is displayed.
3. In the **Type of information to log** field, select the types of messages that you want to log:
 - Error messages only**

Error messages indicate that an undesirable situation or unexpected behavior occurred and that the process cannot continue. You must take action to correct the problem.
 - Error and warning messages**

Warning messages indicate a possible conflict or inconsistency, but they do not cause a process to stop. This option is the default.
 - All messages**

Information messages provide general information about the system or current task and do not require any corrective action.
4. In the **Maximum size of each log file** field, type the maximum number of megabytes for each log file. The default value is 5MB.

When the log file grows to this size, a new log file is created, up to the maximum number of log files that you allow. By keeping log files relatively small, you can view them more efficiently.
5. In the **Maximum number of log files** field, type the maximum number of log files that you want to create. The default value is 10.

If you want to ensure that older log messages are available for review, increase this value. If you are more interested in recent messages and do not need to maintain a long history of activity, decrease this value.
6. In the **Default locale** field, select the language that you want to use to log messages. The default value is English.
7. Click **OK**.

Configuring SMTP server information

Before you can receive e-mail about enterprise search activities, you must configure information about your Simple Mail Transfer Protocol (SMTP) server.

Before you begin

To configure information about your SMTP server, you must be a member of the enterprise search administrator role.


About this task

Several enterprise search administrative functions enable you to receive e-mail. Before you can receive e-mail from any of these functions, you must specify information about your SMTP server:

- If you configure collection-level alerts or system-level alerts, you can receive e-mail whenever those messages are logged. You can also receive e-mail when other messages are logged, not just messages that are triggered by monitored events.
- If you want to see detailed information about a URI in the index or a document that was dropped from the enterprise search system, you can receive the report by e-mail.
- If you monitor a Web crawler, and specify that you want to create Web crawler history reports, you can be notified by e-mail after a report is created.

Procedure

To configure information about your SMTP server:

1. Click **System** to open the System view.
2. Click  **Edit** to change to the system editing view.
3. On the Log page, click **Configure e-mail options for messages**.
4. On the E-mail Options for System Messages page, in the **SMTP mail server to use for delivering e-mail** field, type the fully qualified host name or IP address of the SMTP server that you want to use.

The system uses this server to send e-mail to the addresses that you specify.

5. In the **Frequency to check for e-mail** field, specify how often you want the system to check for eligible messages and send e-mail about them.

The system combines all of the messages for a specific e-mail address into one message, and sends that message at the frequency that you specify.

6. Click **OK**.

Receiving e-mail about logged messages

You can specify options to receive e-mail whenever certain messages, or certain types of messages, are logged.

Before you begin

To configure e-mail options for system-level messages, you must be a member of the enterprise search administrator role. To configure e-mail options for collection-level messages, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.


Before you can receive e-mail, you must first configure information about your Simple Mail Transfer Protocol (SMTP) server so that e-mail can be delivered.

About this task

When you configure alerts, you can choose an option to log messages when certain events occur. If you enable those options, you can then configure options to receive e-mail automatically whenever those messages are logged. You can also specify options to receive e-mail when other messages are logged, not just messages that are triggered by events.


Procedure

To configure e-mail options for messages:

1. If you want to receive e-mail about system messages:
 - a. Click **System** to open the System view.
 - b. Click  **Edit** to change to the system editing view.
 - c. On the Log page, click **Configure e-mail options for messages**.
 - d. On the E-mail Options for System Messages page, select the **Send e-mail about system-level messages** check box.
 - e. In the **E-mail address for receiving e-mail** field, type one or more e-mail addresses. Typically, an enterprise search administrator should receive information about system messages.

Separate each address with a comma. For example:

steinbeck@us.ibm.com, yeats@ireland.ibm.com, dante@it.ibm.com.

- f. If you want to receive e-mail about all error messages that are logged, select the **Send e-mail about all error messages** check box.
 - g. If you want to receive e-mail only when certain system-level messages are logged, type the message IDs for those messages in the **Send e-mail about certain messages** area. Type one message ID per line. For example:
FFQC4819E
FFQ00005E
Several message IDs are listed by default (click **Help** for a description of these messages).
 - h. Click **OK**.
2. If you want to receive e-mail about messages for a collection:
 - a. Click **Collections** to open the Collections view.
 - b. In the list of collections, locate the collection that you want to configure and click  **Edit**.
 - c. On the Log page, click **Configure e-mail options for messages**.
 - d. On the E-mail Options for Collection Messages page, select the **Send e-mail about collection-level messages** check box.
 - e. In the **E-mail address for receiving e-mail** field, type one or more e-mail addresses. Typically, a collection administrator should receive information about collection-level messages.
Separate each address with a comma. For example:
steinbeck@us.ibm.com, yeats@ireland.ibm.com, dante@it.ibm.com.
 - f. If you want to receive e-mail about all error messages that are logged, select the **Send e-mail about all error messages** check box.
 - g. If you want to receive e-mail only when certain collection-level messages are logged, type the message IDs for those messages in the **Send e-mail about certain messages** area. Type one message ID per line. For example:
FFQC4819E
FFQ00005E
Several message IDs are listed by default (click **Help** for a description of these messages).
 - h. Click **OK**.

Related concepts

"Messages for enterprise search" in "Messages Reference"

Related tasks

"Configuring collection-level alerts" on page 238

By configuring alerts, you can ensure that messages are written to the log file whenever certain collection-level events occur. You can also receive e-mail whenever messages about these events are logged.

"Configuring system-level alerts" on page 239

By configuring alerts, you can ensure that messages are written to the log file whenever certain system-level events occur. You can also receive e-mail whenever messages about these events are logged.


Viewing log files


You can view log messages that the system and collection components write to a common log file. You can also specify filters to view messages of a specific severity level and messages from specific enterprise search sessions.

Before you begin

All enterprise search administrative users can view log files for the collections that they are authorized to administer. To view system-level log files, you must be a member of the enterprise search administrator role or have permission to access the **System** toolbar.

Procedure

1. To view the log files for a single collection:
 - a. Click **Collections** to open the Collections view.
 - b. In the list of collections, locate the collection that you want to view, click  **Monitor**, and open the Log page.

Tip: If you are editing a collection and are already on the Log page, you can click  **Monitor** to change to the view for monitoring the collection.

2. To view system-level log files:
 - a. Click **System** to open the System view.
 - b. Select the Log page.
3. In the **Log file** field, select the log file that you want to view.


The name of each log file includes the log file type (such as a system or collection identifier), the date that the file was created, and a numeric suffix that indicates the order in which the file was created on that date. For example:

```
log_file_type_2005-05-26_1.log  
log_file_type_2005-05-26_2.log  
log_file_type_2005-05-25_1.log  
log_file_type_2005-05-25_2.log  
log_file_type_2005-05-25_3.log
```

4. To view only messages of specific severity levels, select the appropriate check boxes in the **Severity** field.
5. To view only messages from specific sessions, select the appropriate check boxes in the **Session** field.
6. Click **View log**.

For each message on the Contents of Log File page, you see the date and time that the message was issued, the message severity level, the name of the session that issued the message, and the message ID and error text.

You can click buttons to go to the first page, last page, previous page, or next page of the log file. You can also specify a page number and go directly to that page.

7. To see more detailed information about a message, click  **Details**.

On the Log Message Details page, you see the host name of the enterprise search server where the message occurred, the name of the file that produced the error, the function name and line number where the error occurred, the process ID, and the thread ID.

Backing up and restoring an enterprise search system

Backup and restore scripts enable you to back up and restore the enterprise search system.

If the system fails because of an irrecoverable error, you must re-install WebSphere Information Integrator OmniFind Edition and then run the restore script. You can also use these scripts to restore essential system files to one or more new servers.

The scripts back up and restore the following files:

- Configuration files from the ES_NODE_ROOT/master_config directory.
- Database files for the crawlers.
- Index files. If the enterprise search indexes are not in the ES_NODE_ROOT/data directory, you cannot use the enterprise search scripts to back up and restore the index files.

The backup script creates the following subdirectories under a directory that you specify when you run the script. (The enterprise search administrator ID must have permission to write to the directory that you specify.)

master_config

Contains the configuration files from the ES_NODE_ROOT/master_config directory

database

Contains the database files from the crawler server

data

Contains the index files from the index server

You must have enough disk space available to back up the enterprise search system files to another directory. The backup and restore scripts do not check the files. When you start a backup, most system sessions are temporarily unavailable. The search processes will continue to run. You should start the backup after you reorganize the index so that you have the most current index.

On a multiple server installation, back up and restore the system from the enterprise search index server. Because the index server creates a database catalog, the index server can access and back up the product database tables on the crawler server.

Related reference

“Enterprise search commands, return codes, and session IDs” on page 249
You can use commands to diagnose problems, determine the status of the different parts of the system, start and stop sessions, or start and stop the system.

Backing up the enterprise search system

You can back up the enterprise search system by using the esbackup.sh script for UNIX or the esbackup.bat script for Microsoft Windows.

Restrictions

All system sessions are stopped while the backup and restore scripts are running. To avoid seeing incorrect or inconsistent system information, do not use the enterprise search administration console while the scripts are running.

The enterprise search administrator ID must have permission to write to the directory that you specify when you run the backup script.

Procedure

To back up the enterprise search system:

1. On the index server, log in as the enterprise search administrator. This user ID was specified when WebSphere II OmniFind Edition was installed.
2. Enter the following command, where *backup_directory* is a directory where you want to back up the files:

UNIX: `esbackup.sh -c -d -i backup_directory`

Windows: `esbackup.bat -c -d -i backup_directory`

You can specify the following options:

- c Backs up the configuration files. This is the default option.
- d Backs up crawled documents in the crawler data store (these documents are not yet parsed or indexed).
- i Backs up the index files. If the index files are not in the `ES_NODE_ROOT/data` directory, you cannot use the `esbackup` script to back up the index files.

Related reference

“Enterprise search commands, return codes, and session IDs” on page 249
You can use commands to diagnose problems, determine the status of the different parts of the system, start and stop sessions, or start and stop the system.

Restoring the enterprise search system

You can restore system configuration files after you re-install WebSphere II OmniFind Edition by using the `esrestore.sh` script for UNIX or the `esrestore.bat` script for Microsoft Windows.

Restrictions

All system sessions are stopped while the backup and restore scripts are running. To avoid seeing incorrect or inconsistent system information, do not use the enterprise search administration console while the scripts are running.

Procedure

To restore the enterprise search system:

1. On the index server, log in as the enterprise search administrator. This user ID was specified when WebSphere II OmniFind Edition was installed.
2. Start the common communication layer (CCL) if it is not already started:

UNIX: `startccl.sh -bg`

Windows: `startccl.bat`

To start the CCL in the background on a Windows system, click **Start** → **Programs** → **Administrative Tools** → **Services**, and restart the WebSphere Information Integrator OmniFind Edition service.

3. Stop the controller if it is not already stopped:

```
esadmin stop
```

4. Enter the following command, where *backup_directory* is the directory where you backed up the files:

```
UNIX: esrestore.sh -c -d -i backup_directory
```

```
Windows: esrestore.bat -c -d -i backup_directory
```

You can specify the following options:

- c Restores the configuration files. This is the default option.
- d Restores crawled documents in the crawler data store. (These documents were not yet parsed or indexed.)
- i Restores the index files. If the index files are not in the ES_NODE_ROOT/data directory, you cannot use the esrestore script to restore the index files.

Related reference

“Enterprise search commands, return codes, and session IDs” on page 249
You can use commands to diagnose problems, determine the status of the different parts of the system, start and stop sessions, or start and stop the system.

Restoring enterprise search system files to new servers

You can back up system files from one enterprise search server and restore the system files to one or more new servers instead of restoring them to the current servers.

Before you begin

You must install WebSphere II OmniFind Edition on the new servers before you run the restore script.

Restrictions

All system sessions are stopped while the backup and restore scripts are running. To avoid seeing incorrect or inconsistent system information, do not use the enterprise search administration console while the scripts are running.

The enterprise search administrator ID must have permission to write to the directory that you specify when you run the backup script.

About this task

The server information that is stored in the ES_NODE_ROOT/master_config/nodes.ini file is not included in the backup files.

Procedure

To restore the enterprise search system files to one or more new servers:

1. On the index server, log in as the enterprise search administrator. This user ID was specified when WebSphere II OmniFind Edition was installed.

2. Run the backup script on the current (old) index server, where *backup_directory* is a directory where you want to back up the files:

UNIX: `esbackup.sh backup_directory`

Windows: `esbackup.bat backup_directory`

3. Use an FTP program to send all of the files in the *backup_directory* to the new index server.
4. Run the restore script on the new index server:

UNIX: `esrestore.sh backup_directory`

Windows: `esrestore.bat backup_directory`

Related reference

“Enterprise search commands, return codes, and session IDs” on page 249
You can use commands to diagnose problems, determine the status of the different parts of the system, start and stop sessions, or start and stop the system.

Enterprise search commands, return codes, and session IDs

You can use commands to diagnose problems, determine the status of the different parts of the system, start and stop sessions, or start and stop the system.

In a multiple server installation, you can run the commands from any server in your system. However, you should run the commands from the index server. The index server, or controller server, can access information from all other servers in the system.

Most commands have the following formats:

```
esadmin command_name arguments
esadmin session_ID action -option
```

For more information about all commands, enter `esadmin help`. For more information about a specific command, enter `esadmin action help`.

Enterprise search esadmin commands

Enter the following commands on one line.

Table 5. Enterprise search esadmin commands

| Command | Description |
|---|--|
| <code>esadmin <i>crawler_session_id</i> start</code> | Starts a crawler session. This command does not start any crawling activity. Sample command: <code>esadmin col1.WEB1.esadmin start</code> Sample messages and return codes: FFQC5310I WEBCrawler1 (sid: col1.WEB1.esadmin) is not running. FFQC5314I Result: 0 |
| <code>esadmin <i>crawler_session_id</i> startCrawl</code> | Starts crawling. Sample command: <code>esadmin col3.DB21.esadmin startCrawl</code> Sample messages and return codes: FFQC5303I DB2Crawler1 (sid: col3.DB21.esadmin) is already running. PID: 23650 FFQC5314I Result: 0 |
| <code>esadmin <i>crawler_session_id</i> pause</code> | Pauses crawling. Sample command: <code>esadmin col3.DB21.esadmin pause</code> Sample messages and return codes: FFQC5303I DB2Crawler1 (sid: col3.DB21.esadmin) is already running. PID: 23650 FFQC5314I Result: 0 |

Table 5. Enterprise search esadmin commands (continued)

| Command | Description |
|--|--|
| esadmin <i>crawler_session_id</i> resume | Resumes crawling. Sample command: esadmin col3.DB21.esadmin resume Sample messages and return codes: FFQC5303I DB2Crawler1 (sid: col3.DB21.esadmin) is already running. PID: 23650 FFQC5314I Result: 0 |
| esadmin <i>crawler_session_id</i> stopCrawl | Stops crawling. Sample command: esadmin col3.DB21.esadmin stopCrawl Sample messages and return codes: FFQC5303I DB2Crawler1 (sid: col3.DB21.esadmin) is already running. PID: 23650 FFQC5314I Result: 0 |
| esadmin <i>crawler_session_id</i> stop | Stops a crawler session. Sample command: esadmin col3.DB21.esadmin stop Sample messages and return codes: FFQC5303I DB2Crawler1 (sid: col3.DB21.esadmin) is already running. PID: 23650 FFQC5314I Result: 0 |
| esadmin <i>crawler_session_id</i> getCrawlerStatus | Gets the status of a crawler. The information that is returned depends on whether the crawler is a Web crawler or a crawler for all other data sources. Example for a Web crawler: esadmin col1.WEB1.esadmin getCrawlerStatus Possible return codes and messages for a Web crawler: FFQC5303I WebCrawler1 (sid: col1.WEB1.esadmin) is already running. PID: 23650 Example for a non-Web crawler: esadmin col3.DB21.esadmin getCrawlerStatus Possible return codes and messages for a non-Web crawler: FFQC5303I db2crawler (sid: db2col.DB2_96945) is already running. PID: 5936 For more information about returned status messages, see "Detailed information for status commands" on page 254. |

Table 5. Enterprise search esadmin commands (continued)

| Command | Description |
|---|---|
| <pre>esadmin dscrawler_session_id getCrawlSpaceStatus</pre> | <p>Gets general crawl space status for any crawler other than the Web crawler.</p> |
| <pre>esadmin web_crawler_session_id getCrawlStatus -selections value</pre> | <p>Sample command:</p> <pre>esadmin col3.DB21.esadmin getCrawlSpaceStatus</pre> <p>Sample messages and return codes:</p> <pre>FFQC5303I DB2Crawler1 (sid: col3.DB21.esadmin) is already running. PID: 23650</pre> <p>Gets general crawl space status for the Web crawler.</p> <p>Sample command:</p> <pre>esadmin col1.WEB1.esadmin getCrawlStatus</pre> <p>For more information about returned status messages, see “Detailed information for status commands” on page 254.</p> |
| <pre>esadmin dscrawler_session_id getCrawlSpaceStatusDetail -ts target_server_id esadmin webcrawler_session_id getCrawlDetailsPerSite -url string selections num -threshold num</pre> | <p>Gets detailed crawl space status for any crawler other than a Web crawler. If you do not specify the target server option, data for all target servers is returned. For example, if the DB2 crawler crawls the FOUNTAIN and SAMPLE databases and you do not specify the target server option, the status of all tables in the FOUNTAIN and SAMPLE databases is returned.</p> <p>Sample command:</p> <pre>esadmin col3.DB21.esadmin getCrawlSpaceStatusDetail -ts FOUNTAIN</pre> <p>Sample messages and return codes:</p> <pre>FFQC5303I DB2Crawler1 (sid: col3.DB21.esadmin) is already running. PID: 23650</pre> <p>Gets detailed crawl space status for the Web crawler.</p> <p>Sample command:</p> <pre>esadmin col1.WEB1.esadmin getCrawlDetailsPerSite</pre> <p>For more information about returned status messages, see “Detailed information for status commands” on page 254.</p> |
| <pre>esadmin startParser -cid collection_ID</pre> | <p>Starts the parser.</p> <p>Sample command:</p> <pre>esadmin startParser -cid col1</pre> <p>Sample messages and return codes:</p> <pre>FFQC5303I Controller (server1) (sid: controller) is already running. PID: 25917 FFQC5314I Result: 0</pre> |

Table 5. Enterprise search esadmin commands (continued)

| Command | Description |
|--|---|
| esadmin stopParser <i>collection_id</i> | <p>Stops the parser.</p> <p>Sample command:</p> <pre>esadmin stopParser -cid coll</pre> <p>Sample messages and return codes:</p> <pre>FFQC5303I Controller (node1) (sid: controller) is already running. PID: 15292 FFQC5312E Error executing action: [stopParser] FFQC4823E Session Parser Driver - Collection coll (node1) [coll.parserdriver] is not running.</pre> |
| esadmin monitor getCollectionParserMonitorStatus -cid <i>collection_ID</i> | <p>Gets the parser status.</p> <p>Sample command:</p> <pre>esadmin monitor getCollectionParserMonitorStatus -cid coll</pre> <p>Sample messages and return codes:</p> <pre>FFQC5303I Monitor (node1) (sid: monitor) is already running. PID: 12543</pre> <p>For more information about returned status messages, see “Detailed information for status commands” on page 254.</p> |
| esadmin startMain -cid <i>collection_id</i> | <p>Starts reorganizing an index.</p> <p>Sample command:</p> <pre>esadmin startMain -cid coll</pre> <p>Sample messages and return codes:</p> <pre>FFQC5303I Controller (node1) (sid: controller) is already running. PID: 25917 FFQC5314I Result: 1117671147056</pre> |
| esadmin startDelta -cid <i>collection_id</i> | <p>Starts refreshing an index.</p> <p>Sample command:</p> <pre>esadmin startDelta -cid coll</pre> <p>Sample messages and return codes:</p> <pre>FFQC5303I Controller (node1) (sid: controller) is already running. PID: 4548 FFQC5314I Result: 1117670603408</pre> |

Table 5. Enterprise search esadmin commands (continued)

| Command | Description |
|--|---|
| <pre>esadmin monitor getCollectionIndexMonitorStatus -cid <i>collection_id</i> -buildType [main delta] -numrecords <i>lastNrecords</i></pre> | <p>Gets the status of an index refresh or reorganization. (A main index refers to a reorganized index; a delta index refers to refreshed index). The option numrecords shows the last N index build status records. If numrecords is omitted, the status for the last 20 index builds are returned.</p> <p>Sample command:</p> <pre>esadmin monitor getCollectionIndexMonitorStatus -cid coll -buildType main -numrecords 4</pre> <p>Sample messages and return codes:</p> <pre>FFQC5303I Monitor (node1) (sid: monitor) is already running. PID: 12649</pre> <p>For more information about returned status messages, see “Detailed information for status commands” on page 254.</p> |
| <pre>esadmin startSearch -cid <i>collection_id</i></pre> | <p>Starts the search server processes.</p> <p>Sample command:</p> <pre>esadmin startSearch -cid coll</pre> <p>Sample messages and return codes:</p> <pre>FFQC5303I Controller (node1) (sid: controller) is already running. PID: 25917 FFQC5314I Result: 0</pre> |
| <pre>esadmin stopSearch -cid <i>collection_id</i></pre> | <p>Stops the search server processes.</p> <p>Sample command:</p> <pre>esadmin stopSearch -cid coll</pre> <p>Sample messages and return codes:</p> <pre>FFQC5303I Controller (node1) (sid: controller) is already running. PID: 15292 FFQC5314I Result: 0</pre> |

Table 5. Enterprise search esadmin commands (continued)

| Command | Description |
|--|--|
| esadmin monitor getCollectionSearchMonitorStatus -cid <i>collection_id</i> | Gets the status of the search server. Sample command: esadmin monitor getCollectionSearchMonitorStatus -cid coll Sample messages and return codes: FFQC5303I Monitor (node1) (sid: monitor) is already running. PID: 12649 |
| esadmin searchmanager_session_id getStatus -cid <i>collection_id</i> | Returns detailed search index status information for a collection on a given search server. There is one search manager session per search server. Each search manager session is responsible for monitoring and operating the search indexes on a specific search server. Sample command: esadmin searchmanager.node1 getStatus -cid coll Sample messages and return codes: FFQC5303I Search Manager (node1) (sid: searchmanager.node1) is already running. PID: 15711 FFQC5314I Result: PID=18390 CacheHits=3 QueryRate=1 Port=44008 SessionId=coll.runtime.node1 CacheHitRate=0.333 ResponseTime=70 Status=1 SessionName=coll.runtime.node1.1 For more information about returned status messages, see "Detailed information for status commands." |

Detailed information for status commands

Some commands can return extensive information. This section describes the information that can be returned for crawler status and the crawl space status. The table from the section "Enterprise search esadmin commands" on page 249 provides possible returned information from each esadmin commands. This section describes returned information from the following commands:

- Web crawler status
- Non-Web crawler status
- Crawl space status for the Web crawler
- Crawl space status for non-Web crawlers
- Detailed crawl space status for the Web crawler
- Detailed crawl space status for non-Web crawlers
- Parser status
- Index build status
- Search server status
- Detailed search server status

Web crawler status: When you run the command to obtain Web crawler status, the command returns information in an XML document format. The following information can be returned by the Web crawler status command:

```
FFQC5314I Result: <?xml version='1.0' encoding='UTF-8'?>
<CrawlerStatus>
<CrawlerRunLevel Value="Running"/>
<CrawlerThreadStateDist Count="4" Total="200">
<CrawlerThreadState State="FETCHING" Count="100"/>
. . .
</CrawlerThreadState State="FETCHING" Count=100>
<ActiveBucketList Count="500">
<ActiveBucket URL="http://w3.ibm.com/"
NumActURLs="355"
NumProcURLs="350"
TimeRem="5" Duration="1195"/>
. . .
</ActiveBucketList>
<CrawlRate Value="75"/>
<RecentlyCrawledURLList Count="40">
<RecentlyCrawledURL URL="http://w3.ibm.com/foo.html"/>
<RecentlyCrawledURL URL="http://w3.ibm.com/foo.html"/>
<NumURLsThisSession Value="160000"/>
</CrawlerStatus>
```

The following table describes each XML element and its possible attributes that are returned by the Web crawler status command:

Table 6. Web crawler status information

| Element | Attributes | Description |
|--------------------------|--|---|
| CrawlerStatus | <ul style="list-style-type: none"> CrawlerThreadStateDist ActiveBucketList CrawlRate RecentlyCrawledURLList NumURLsThisSession | Crawler status. |
| CrawlerRunLevel Value | <ul style="list-style-type: none"> String (English) "Not started": The crawler session exists, but it has not yet received the start message to process documents. "Started": The crawler is starting. "Running": The crawler finished initialization and startup and is actively crawling. "Paused": The crawler was told to suspend active crawling, but not to exit. "Stopping": The crawler received the stop signal and is going to stop. "Error": The crawler is in an unrecoverable state, and it must be stopped and restarted to resume crawling. | Information about what the crawler is doing. |
| CrawlerThreadState State | String (English) | Crawler thread activity. This field shows what the thread or threads are doing. |

Table 6. Web crawler status information (continued)

| Element | Attributes | Description |
|--------------------|--|---|
| ActiveBucket | <ul style="list-style-type: none"> URL: String (URL spec) The protocol, host and port whose URLs are being crawled. NumActURLs: Integer (positive) The number of URLs in bucket when it was made available for crawling (activated). NumProcURLs: Integer (nonnegative) The number of URLs from bucket that have been processed so far, either crawled or rejected. TimeRem: Integer The number of seconds remaining before the bucket times out. Duration: Integer (nonnegative) The number of seconds since the bucket was activated. | The current activity of a specified Web site. |
| CrawlRate | Value: Integer (nonnegative) Pages per second being crawled (all buckets combined). | The crawler throughput measurement. |
| RecentlyCrawledURL | URL: String (URL spec) String specifying a protocol, host, port and file that was crawled. | A page that was crawled recently. |
| NumURLsThisSession | Value: Integer (nonnegative) | The number of URLs that were crawled since this instance of the crawler (process) started crawling. |

Non-Web crawler status: When you run the command to obtain crawler status for a non-Web crawler, the command returns information in an XML document format. The following information can be returned by a non-Web crawler status command:

```
FFQC5314I Result: <?xml version='1.0' encoding='UTF-8'?>
<GeneralStatus>
<Status>0</Status>
<StatusMessage>Idle</StatusMessage>
<NumberOfServers>1</NumberOfServers>
<NumberOfCompletedServers>1</NumberOfCompletedServers>
<NumberOfTargets>3</NumberOfTargets>
<NumberOfCompletedTargets>3</NumberOfCompletedTargets>
<NumberOfCrawledRecords>115</NumberOfCrawledRecords>
<RunningThreads>0</RunningThreads>
</GeneralStatus>
```

The following tables describe the XML elements and attributes for each enterprise search crawler except for the Web crawler. This information is returned with the crawler status command.

Table 7. Elements and attributes for the NNTP crawler, the DB2 crawler, and the Notes crawler for the crawler status command

| Element and attribute name | NNTP crawler | DB2 crawler | Notes crawler |
|----------------------------|----------------------|----------------------|----------------------|
| Status | Status (0, 1, 2, -1) | Status (0, 1, 2, -1) | Status (0, 1, 2, -1) |

Table 7. Elements and attributes for the NNTP crawler, the DB2 crawler, and the Notes crawler for the crawler status command (continued)

| Element and attribute name | NNTP crawler | DB2 crawler | Notes crawler |
|----------------------------|---|---|---|
| StatusMessage | Status: 0 - Idle, 1 - Running, 2 - Paused, -1 - Error | Status: 0 - Idle, 1 - Running, 2 - Paused, -1 - Error | Status: 0 - Idle, 1 - Running, 2 - Paused, -1 - Error |
| NumberOfServers | The number of NNTP servers in the crawl space. | The number of databases in the crawl space. | The number of databases in the crawl space. |
| NumberOfCompletedServers | The number of crawled NNTP servers. | The number of crawled databases. | The number of crawled databases. |
| NumberOfTargets | The number of news groups in the crawl space. | The number of databases in the crawl space. | The number of views and folders in the crawl space. |
| NumberOfCompletedTargets | The number of crawled news groups. | The number of crawled tables. | The number of crawled views and folders. |
| NumberOfCompletedRecords | The number of crawled articles. | The number of crawled records. | The number of crawled documents. |
| RunningThreads | The number of crawler threads. | The number of crawler threads. | The number of crawler threads. |

Table 8. Elements and attributes for the Exchange Server crawler, the DB2 Content Manager crawler, and the Content Edition crawler for the crawler status command

| Element and attribute name | Exchange Server crawler | DB2 Content Manager crawler | Content Edition crawler |
|----------------------------|---|---|---|
| Status | Status (0, 1, 2, -1) | Status (0, 1, 2, -1) | Status (0, 1, 2, -1) |
| StatusMessage | Status: 0 - Idle, 1 - Running, 2 - Paused, -1 - Error | Status: 0 - Idle, 1 - Running, 2 - Paused, -1 - Error | Status: 0 - Idle, 1 - Running, 2 - Paused, -1 - Error |
| NumberOfServers | The number of Exchange Server servers in the crawl space. | The number of Content Manager servers in the crawl space. | The number of repositories in the crawl space. |
| NumberOfCompletedServers | The number of crawled Exchange Server servers. | The number of crawled Content Manager servers. | The number of crawled repositories. |
| NumberOfTargets | The number of subfolders in the crawl space. | The number of item types in the crawl space. | The number of classes in the crawl space. |
| NumberOfCompletedTargets | The number of crawled subfolders. | The number of crawled item types. | The number of crawled item classes. |
| NumberOfCompletedRecords | The number of crawled documents. | The number of crawled documents. | The number of crawled documents. |
| RunningThreads | The number of crawler threads. | The number of crawler threads. | The number of crawler threads. |

Table 9. Elements and attributes for the QuickPlace crawler, the Domino Document Manager crawler, the UNIX file system crawler, and the Windows file system crawler for the crawler status command

| Element and attribute name | QuickPlace crawler | Domino Document Manager crawler | UNIX and Windows file system crawlers |
|----------------------------|---|---|---|
| Status | Status (0, 1, 2, -1) | Status (0, 1, 2, -1) | Status (0, 1, 2, -1) |
| StatusMessage | Status: 0 - Idle, 1 - Running, 2 - Paused, -1 - Error | Status: 0 - Idle, 1 - Running, 2 - Paused, -1 - Error | Status: 0 - Idle, 1 - Running, 2 - Paused, -1 - Error |

Table 9. Elements and attributes for the QuickPlace crawler, the Domino Document Manager crawler, the UNIX file system crawler, and the Windows file system crawler for the crawler status command (continued)

| Element and attribute name | QuickPlace crawler | Domino Document Manager crawler | UNIX and Windows file system crawlers |
|----------------------------|--|---|--|
| NumberOfServers | The number of places in the crawl space. | The number of libraries in the crawl space. | Fixed value of 1. |
| NumberOfCompletedServers | The number of crawled places. | The number of crawled libraries. | 0 or 1 if all subdirectories are crawled. |
| NumberOfTargets | The number of place databases and room databases in the crawl space. | The number of cabinets in the crawl space. | The number of subdirectories in the crawl space. |
| NumberOfCompletedTargets | The number of crawled place databases and room databases. | The number of crawled cabinets. | The number of crawled subdirectories. |
| NumberOfCompletedRecords | The number of crawled documents. | The number of crawled documents. | The number of crawled files. |
| RunningThreads | The number of crawler threads. | The number of crawler threads. | The number of crawler threads. |

Crawl space status for the Web crawler: When you run the command to obtain crawl space status for a Web crawler, the command returns information in an XML document format. The following information can be returned by a Web crawl space status command:

Table 10. Selection mask values for the Web crawler crawl space status command

| Mask bit | Selects |
|----------|------------------------------------|
| 1 | Number of pages in raw data store. |
| 2 | Number of discovered sites. |
| 4 | Number of sites with DNS. |
| 8 | Number of sites without DNS. |
| 16 | Number of discovered URLs. |
| 32 | Number of unique saved pages. |
| 64 | Number of crawled URLs. |
| 128 | Number of URLs that are uncrawled. |
| 256 | Number of URLs that are overdue. |
| 512 | HTTP return code distribution. |

All values represent cumulative totals for all sessions that use the current internal database:

```
<CrawlStatus>
  <NumPagesInRDS Value="5422386"/>
  <NumSitesDiscovered Value="15332"/>
  <NumSitesWithDNS Value="14832"/>
  <NumSitesWithoutDNS Value="500"/>
  <NumURLsDiscovered Value="15222999"/>
  <NumUniquePagesSaved Value="6234789"/>
  <NumURLsCrawled Value="7800422"/>
  <NumURLsUncrawled Value="7422577"/>
  <NumURLsOverdue Value="14000"/>
  <HTTPCodeDist Count="4" Total="1031000"/>
</CrawlStatus>
```

```

|         <HTTPCode Code="200" Count ="1000000"/>
|         <HTTPCode Code="301" Count ="1000"/>
|         <HTTPCode Code="404" Count ="10000"/>
|         <HTTPCode Code="780" Count="20000"/>
|       </HTTPCode Code="780" Count="20000">
|     <?CrawlStatus>

```

The return data contains any or all (possibly none) of the following elements:

Table 11. Information returned from the Web crawler crawl space status command

| Element | Attribute | Description |
|---------------------|--|---|
| CrawlerStatus | <ul style="list-style-type: none"> • NumPagesInRDS • NumSitesDiscovered • NumSitesWithDNS • NumSitesWithoutDNS • NumURLsDiscovered • NumUniquePagesSaved • NumURLsCrawled • NumURLsUncrawled • NumURLsOverdue • HTTPCodeDist | Information that can be quickly obtained about the cumulative state of the crawl (all sessions). |
| NumPagesInRDS | Value: Nonnegative integer How many pages are currently in the raw data store (RDS) staging area (from this crawler only). | How full the raw data store (RDS) is becoming (from this crawler's contributions only). |
| NumSitesDiscovered | Value: Nonnegative integer How many hosts were discovered by crawling (or from seeds). | A measure of the crawler's coverage of the domain to be crawled (host count). |
| NumSitesWithDNS | Value: Nonnegative integer How many hosts have associated IP addresses (resolved by the crawler in background). | A measure of how effectively the crawler is able to get IP addresses for hosts that are discovered by DNS names in URLs. |
| NumSitesWithoutDNS | Value: Nonnegative integer How many hosts do not have associated IP addresses (resolved by the crawler in background). | A measure of how effectively the crawler is able to get IP addresses for hosts that are discovered by DNS names in URLs. |
| NumURLsDiscovered | Value: Nonnegative integer How many unique URLs were visited by the crawler. | A measure of the crawler's coverage of the domain to be crawled (URL count). |
| NumUniquePagesSaved | Value: Nonnegative integer How many unique pages were written to the RDS for further processing by other enterprise search components. | This crawler's contribution to the size of the index. |
| NumURLsCrawled | Value: Nonnegative integer How many unique URLs were crawled by the crawler. | A measure of the crawler's ability to process data, end to end. This number is different from the number of pages written to RDS, because not all crawled pages result in being written to RDS. |

Table 11. Information returned from the Web crawler crawl space status command (continued)

| Element | Attribute | Description |
|----------------|--|---|
| NumURLsOverdue | Value: Nonnegative integer How many unique URLs are eligible to be recrawled. | A measure of the crawler's ability to traverse the Web space. |

Crawl space status for non-Web crawlers: When you run the command to obtain crawl space status for a non-Web crawler, the command returns information in an XML document format. The following information can be returned by a non-Web crawl space status command:

```
FFQC5314I Result: <?xml version='1.0' encoding='UTF-8'?>
<ServerStatus>
  <Server Name ="FOUNTAIN">
    <Status>5</Status>
    <StatusMessage>Scheduled</StatusMessage>
    <NumberOfTargets>1</NumberOfTargets>
    <NumberOfCompletedTargets>1</NumberOfCompletedTargets>
    <NumberOfErrors>0</NumberOfErrors>
    <StartTime>1118354510512</StartTime>
    <EndTime>1118354514386</EndTime>
    <ScheduleConfigured>2</ScheduleConfigured>
    <ScheduleTime>1118393377000</ScheduleTime>
    <TotalTime>3874</TotalTime>
  </Server>
</ServerStatus>
```

The following tables describe the XML elements and attributes for each enterprise search crawler except for the Web crawler. This information is returned with the crawl space status command. For Notes crawlers, when the aggregation level is 0, Server@Name is server name + database name. When the aggregation level is 1, Server@Name is server name + directory name.

Table 12. Elements and attributes for the NNTP crawler, the DB2 crawler, and the Notes crawler for the crawl space status command

| Element and attribute name | NNTP crawler | DB2 crawler | Notes crawler |
|----------------------------|--|--|--|
| Server@Name | News server name | Database name | Database name or directory name |
| Server/Status | Status: (0, 1, 2, 3, 4, -1) <ul style="list-style-type: none"> • 0: Not Crawled • 1: Crawling • 2: Completed (not scheduled) • 3: Waiting • 4: Paused • -1: Error | Status (0, 1, 2, 3, 4, 5, -1) <ul style="list-style-type: none"> • 0: Not Crawled • 1: Crawling • 2: Completed (not scheduled) • 3: Waiting • 4: Paused • 5: Scheduled • -1: Error | Status (0, 1, 2, 3, 4, 5, -1) <ul style="list-style-type: none"> • 0: Not Crawled • 1: Crawling • 2: Completed (not scheduled) • 3: Waiting • 4: Paused • 5: Scheduled • -1: Error |

Table 12. Elements and attributes for the NNTP crawler, the DB2 crawler, and the Notes crawler for the crawl space status command (continued)

| Element and attribute name | NNTP crawler | DB2 crawler | Notes crawler |
|---------------------------------|---|---|---|
| Server/StatusMessage | <ul style="list-style-type: none"> • 0: Not Crawled • 1: Crawling • 2: Completed (not scheduled) • 3: Waiting • 4: Paused • 5: Scheduled • -1: Error | <ul style="list-style-type: none"> • 0: Not Crawled • 1: Crawling • 2: Completed (not scheduled) • 3: Waiting • 4: Paused • 5: Scheduled • -1: Error | <ul style="list-style-type: none"> • 0: Not Crawled • 1: Crawling • 2: Completed (not scheduled) • 3: Waiting • 4: Paused • 5: Scheduled • -1: Error |
| Server/NumberOfTargets | The number of news groups in the crawl space. | The number of databases in the crawl space. | The number of views and folders in the crawl space. |
| Server/NumberOfCompletedTargets | The number of crawled news groups. | The number of crawled tables. | The number of crawled views and folders. |
| Server/NumberOfErrors | Not applicable. | The number of errors. | The number of errors |
| Server/StartTime | The start time if applicable. | The start time if applicable. | The start time if applicable. |
| Server/EndTime | The end time if applicable. | The end time if applicable. | The end time if applicable. |
| Server/ScheduleConfigured | Not applicable. | 0, 1, 2 <ul style="list-style-type: none"> • 0: The crawler is not configured for scheduling according to the crawler configuration files. • 1: The crawler is configured for scheduling, but the scheduling was disabled for the session • 2: The crawler is configured for scheduling, and the scheduling is enabled for the session | 0, 1, 2 <ul style="list-style-type: none"> • 0: The crawler is not configured for scheduling according to the crawler configuration files. • 1: The crawler is configured for scheduling, but the scheduling was disabled for the session • 2: The crawler is configured for scheduling, and the scheduling is enabled for the session |
| Server/ScheduleTime | Not applicable. | Schedule time if applicable. | Schedule time if applicable. |
| Server/TotalTime | The total time if applicable. | The total time if applicable. | The total time if applicable. |
| Server/AggregationLevel | Not applicable. | 0: The Notes crawler crawls documents with normal mode. (The other crawlers except for the Notes crawler always return 0.) | 0, 1: <ul style="list-style-type: none"> • 0: The Notes crawler crawls documents with normal mode. (The other crawlers except for the Notes crawler always return 0.) • The Notes crawler crawls documents with directory mode. |

Table 13. Elements and attributes for the Exchange Server crawler, the DB2 Content Manager crawler, and the Content Edition crawler for the crawl space status command

| Element and attribute name | Exchange Server crawler | DB2 Content Manager crawler | Content Edition crawler |
|---------------------------------|---|---|---|
| Server@Name | Exchange Server server name. | DB2 Content Manager servers. | Repository name. |
| Server/Status | Status (0, 1, 2, 3, 4, -1) <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused -1: Error | Status (0, 1, 2, 3, 4, 5, -1) <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused 5: Scheduled -1: Error | Status (0, 1, 2, 3, 4, 5, -1) <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused 5: Scheduled -1: Error |
| Server/StatusMessage | <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused 5: Scheduled -1: Error | <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused 5: Scheduled -1: Error | <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused 5: Scheduled -1: Error |
| Server/NumberOfTargets | The number of subfolders in the crawl space. | The number of item types in the crawl space. | The number of item classes in the crawl space. |
| Server/NumberOfCompletedTargets | The number of crawled subfolders. | The number of crawled item types. | The number of crawled item classes. |
| Server/NumberOfErrors | The number of errors. | The number of errors. | The number of errors. |
| Server/StartTime | The start time if applicable. | The start time if applicable. | The start time if applicable. |
| Server/EndTime | The end time if applicable. | The end time if applicable. | The end time if applicable. |
| Server/ScheduleConfigured | 0, 1, 2 <ul style="list-style-type: none"> 0: The crawler is not configured for scheduling according to the crawler configuration files. 1: The crawler is configured for scheduling, but the scheduling was disabled for the session 2: The crawler is configured for scheduling, and the scheduling is enabled for the session | 0, 1, 2 <ul style="list-style-type: none"> 0: The crawler is not configured for scheduling according to the crawler configuration files. 1: The crawler is configured for scheduling, but the scheduling was disabled for the session 2: The crawler is configured for scheduling, and the scheduling is enabled for the session | 0, 1, 2 <ul style="list-style-type: none"> 0: The crawler is not configured for scheduling according to the crawler configuration files. 1: The crawler is configured for scheduling, but the scheduling was disabled for the session 2: The crawler is configured for scheduling, and the scheduling is enabled for the session |
| Server/ScheduleTime | Schedule time if applicable. | Schedule time if applicable. | Schedule time if applicable. |
| Server/TotalTime | The total time if applicable. | The total time if applicable. | The total time if applicable. |

Table 13. Elements and attributes for the Exchange Server crawler, the DB2 Content Manager crawler, and the Content Edition crawler for the crawl space status command (continued)

| Element and attribute name | Exchange Server crawler | DB2 Content Manager crawler | Content Edition crawler |
|----------------------------|--|--|--|
| Server/AggregationLevel | 0: The Notes crawler crawls documents with normal mode. (The other crawlers except for the Notes crawler always return 0.) | 0: The Notes crawler crawls documents with normal mode. (The other crawlers except for the Notes crawler always return 0.) | 0: The Notes crawler crawls documents with normal mode. (The other crawlers except for the Notes crawler always return 0.) |

Table 14. Elements and attributes for the QuickPlace crawler, the Domino Document Manager crawler, the UNIX file system crawler, and the Windows file system crawler for the crawl space status command

| Element and attribute name | QuickPlace crawler | Domino Document Manager crawler | UNIX and Windows file system crawlers |
|---------------------------------|--|---|---|
| Server@Name | Place directory | Library database | A fixed value of localhost. |
| Server/Status | Status (0, 1, 2, 3, 4, -1) <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused -1: Error | Status (0, 1, 2, 3, 4, 5, -1) <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused 5: Scheduled -1: Error | Status (0, 1, 2, 3, 4, 5, -1) <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused 5: Scheduled -1: Error |
| Server/StatusMessage | <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused 5: Scheduled -1: Error | <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused 5: Scheduled -1: Error | <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused 5: Scheduled -1: Error |
| Server/NumberOfTargets | The number of place databases and room databases in the crawl space. | The number of cabinets in the crawl space. | The number of subdirectories in the crawl space. |
| Server/NumberOfCompletedTargets | The number of crawled place databases and room databases. | The number of crawled cabinets. | The number of subdirectories in the crawl space. |
| Server/NumberOfErrors | The number of errors. | The number of errors. | The number of errors. |
| Server/StartTime | The start time if applicable. | The start time if applicable. | The start time if applicable. |
| Server/EndTime | The end time if applicable. | The end time if applicable. | The end time if applicable. |

Table 14. Elements and attributes for the QuickPlace crawler, the Domino Document Manager crawler, the UNIX file system crawler, and the Windows file system crawler for the crawl space status command (continued)

| Element and attribute name | QuickPlace crawler | Domino Document Manager crawler | UNIX and Windows file system crawlers |
|----------------------------|--|--|--|
| Server/ScheduleConfigured | 0, 1, 2 <ul style="list-style-type: none"> 0: The crawler is not configured for scheduling according to the crawler configuration files. 1: The crawler is configured for scheduling, but the scheduling was disabled for the session 2: The crawler is configured for scheduling, and the scheduling is enabled for the session | 0, 1, 2 <ul style="list-style-type: none"> 0: The crawler is not configured for scheduling according to the crawler configuration files. 1: The crawler is configured for scheduling, but the scheduling was disabled for the session 2: The crawler is configured for scheduling, and the scheduling is enabled for the session | 0, 1, 2 <ul style="list-style-type: none"> 0: The crawler is not configured for scheduling according to the crawler configuration files. 1: The crawler is configured for scheduling, but the scheduling was disabled for the session 2: The crawler is configured for scheduling, and the scheduling is enabled for the session |
| Server/ScheduleTime | Schedule time if applicable. | Schedule time if applicable. | Schedule time if applicable. |
| Server/TotalTime | The total time if applicable. | The total time if applicable. | The total time if applicable. |
| Server/AggregationLevel | 0, 1 <ul style="list-style-type: none"> 0: The Notes crawler crawls documents with normal mode. (The other crawlers except for the Notes crawler always return 0.) The Notes crawler crawls documents with directory mode. | 0, 1 <ul style="list-style-type: none"> 0: The Notes crawler crawls documents with normal mode. (The other crawlers except for the Notes crawler always return 0.) The Notes crawler crawls documents with directory mode. | 0, 1 <ul style="list-style-type: none"> 0: The Notes crawler crawls documents with normal mode. (The other crawlers except for the Notes crawler always return 0.) The Notes crawler crawls documents with directory mode. |

Detailed crawl space status for the Web crawler: When you run the command to obtain detailed crawl space status for the Web crawler, the command returns information in an XML document format. The following information can be returned by the detailed crawl space status command:

Table 15. Selection mask values for the Web crawler detailed crawl space status command

| Mask bit | Selects |
|----------|------------------------------------|
| 1 | Number of pages in raw data store. |
| 2 | Number of discovered sites. |
| 4 | Number of sites with DNS. |
| 8 | Number of sites without DNS. |
| 16 | Number of discovered URLs. |
| 32 | Number of unique saved pages. |
| 64 | Number of crawled URLs. |
| 128 | Number of URLs that are uncrawled. |
| 256 | Number of URLs that are overdue. |
| 512 | HTTP return code distribution. |

Sample returned information:

```
<CrawlDetailsPerSite>
  <Site URL=http://w3.ibm.com/">
  <NumURLsDiscovered Value="5422386"/>
  <NumURLsOverdue Value="15332"/>
  <NumURLsCrawled Value="15332"/>
  <NumURLsUncrawled Value="15332"/>
  <NumURLsOverdueBy Threshold="604800" Value="14832"/>
  <NumURLsActivated Value="2200"/>
  <LastActivationTime Value="1076227340"/>
  <LastActivationDuration Value="4300"/>
  <IPAddressList Count="1"/>
    <IPAddress Value="9.205.41.33"/>
  </IPAddressList>
  <RobotsContent>
    robots content. . .
  </RobotsContent>
  <HTTPCodeDist Count="4" Total="1031000"/>
    <HTTPCode Code="200" Count ="1000000"/>
    <HTTPCode Code="301" Count ="1000"/>
    <HTTPCode Code="404" Count ="10000"/>
    <HTTPCode Code="780" Count="20000"/>
  </HTTPCodeDist>
</CrawlDetailsPerSite>
```

The following table describes each field that is returned for the Web crawler detailed crawl space status:

Table 16. Information returned from the Web crawler detailed crawl space status command

| Element | Attributes | Description |
|---------------------|---|---|
| CrawlDetailsPerSite | <ul style="list-style-type: none"> LastActivationTime: LastActivationDuration: IPAddressList: RobotsContent: HTTPCodeDist: | Information that can be quickly obtained about the detailed state of one site. |
| Site | URL | URL of the site root page. |
| NumURLsDiscovered | Value | The number of URLs that were discovered from the site. |
| NumURLsOverdue | Value | The number of URLs that are eligible to be recrawled from the site. |
| NumURLsCrawled | Value | The number of URLs that were crawled for the site. |
| NumURLsUncrawled | Value | The number of URLs that are not yet crawled for the site. |
| NumURLsOverdueBy | Threshold, Value: Integer (positive or negative) The threshold for finding recrawls overdue by a specified amount (negative = seconds before now) time or upcoming during a specified interval of time (positive = seconds after now). | The number of URLs that are eligible to be recrawled at least some number of seconds (threshold) ago or that are becoming eligible in the next so many seconds (threshold). |

Table 16. Information returned from the Web crawler detailed crawl space status command (continued)

| Element | Attributes | Description |
|------------------------|--|---|
| NumURLsActivated | Value | Number of URLs brought into memory during the last scan of this site and made available to crawler threads. |
| LastActivationTime | Value | The number of seconds since epoch at which this site's URLs were last brought into memory. |
| LastActivationDuration | Value | The number of seconds that this site's URLs were last in memory and available to crawler threads. |
| IPAddressList | IPAddress | All known IP addresses for this site's server host. |
| IPAddress | Value | IPv4 dot-notation address for the site's server host. |
| RobotsContent | Text | Text from the robots file, if any text exists. |
| HTTPCodeDist | HTTPCode | Distribution of HTTP codes from this site's attempted downloads. |
| HTTPCode | Code: Integer An HTTP return code or another internal code. | How many times a particular HTTP return code occurred during the crawl of this site. |

Detailed crawl space status for non-Web crawlers: When you run the command to obtain detailed crawl space status for non-Web crawlers, the command returns information in an XML document format. The following information can be returned by the detailed crawl space status command for non-Web crawlers:

```
FFQC5314I Result: <?xml version='1.0' encoding='UTF-8'?>
<TargetStatus>
  <Target Name ="escmgr.crawlerinstances">
    <Status>2</Status>
    <StatusMessage>Completed</StatusMessage>
    <NumberOfRecords></NumberOfRecords>
    <NumberOfCrawledRecords>117</NumberOfCrawledRecords>
    <NumberOfInsertedRecords>21</NumberOfInsertedRecords>
    <NumberOfUpdatedRecords>45</NumberOfUpdatedRecords>
    <StartTime>1118354510727</StartTime>
    <EndTime>1118354514386</EndTime>
    <AggregationLevel>0<AggregationLevel>
  </Target>
</TargetStatus>
```

Table 17. Elements and attributes for the NNTP crawler, the DB2 crawler, and the Notes crawler for the detailed crawl space status command

| Element and attribute name | NNTP crawler | DB2 crawler | Notes crawler |
|----------------------------|-----------------|---|---------------------|
| Target@Name | News group name | Table name | View or folder name |
| Target@CrawlType | Not applicable. | 0,1: • 0: Active crawl (Normal) • 1: Passive crawl (DB2 Event Publishing) | 0 |

Table 17. Elements and attributes for the NNTP crawler, the DB2 crawler, and the Notes crawler for the detailed crawl space status command (continued)

| Element and attribute name | NNTP crawler | DB2 crawler | Notes crawler |
|---------------------------------|--|--|--|
| Target/Status | Status: (0, 1, 2, 3, 4, -1) <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused -1: Error | Status: (0, 1, 2, 3, 4, -1) <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused -1: Error | Status: (0, 1, 2, 3, 4, -1) <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused -1: Error |
| Target/StatusMessage | <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused -1: Error | <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused -1: Error | <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused -1: Error |
| Target/NumberOfRecords | The last article number on the server. | The number of crawled records. | The number of crawled documents. |
| Target/NumberOfCompletedRecords | The number of crawled articles. | The number of crawled records. | The number of crawled documents. |
| Target/NumberOfInsertedRecords | The number of newly posted articles. | The number of inserted records. | The number of inserted records. |
| Target/NumberOfUpdatedRecords | Not applicable. | The number of updated records. | The number of updated records. |
| Target/NumberOfDeletedRecords | Not applicable. | The number of deleted records. | The number of deleted records. |
| Target/StartTime | The date and time that the crawler last started. | The date and time that the crawler last started. | The date and time that the crawler last started. |
| Target/EndTime | The date and time that crawling was completed. | The date and time that crawling was completed. | The date and time that crawling was completed. |
| Target/TotalTime | The amount of time that the crawler spent crawling. | The amount of time that the crawler spent crawling. | The amount of time that the crawler spent crawling. |
| Target/AggregationLevel | 0: The Notes crawler crawls documents with normal mode. (The other crawlers except for the Notes crawler always return 0.) | 0: The Notes crawler crawls documents with normal mode. (The other crawlers except for the Notes crawler always return 0.) | 0, 1: <ul style="list-style-type: none"> 0: The Notes crawler crawls documents with normal mode. (The other crawlers except for the Notes crawler always return 0.) The Notes crawler crawls documents with directory mode. |
| Target/LastUpdatedTime | Not applicable. | The last updated time: <ul style="list-style-type: none"> 0: Active crawl (Normal) 1: Passive crawl (DB2 Event Publishing) | Not applicable. |

Table 17. Elements and attributes for the NNTP crawler, the DB2 crawler, and the Notes crawler for the detailed crawl space status command (continued)

| Element and attribute name | NNTP crawler | DB2 crawler | Notes crawler |
|----------------------------|-----------------|--|-----------------|
| Target/LastResetTime | Not applicable. | The last time reset statistics: <ul style="list-style-type: none"> • 0: Active crawl (Normal) • 1: Passive crawl (DB2 Event Publishing) | Not applicable. |

Table 18. Elements and attributes for the Exchange Server crawler, the DB2 Content Manager crawler, and the Content Edition crawler for the detailed crawl space status command

| Element and attribute name | Exchange Server crawler | DB2 Content Manager crawler | Content Edition crawler |
|---------------------------------|---|---|---|
| Target@Name | Subfolder name | Item type name | Item class name |
| Target@CrawlType | 0 | 0 | 0 |
| Target/Status | Status (0, 1, 2, 3, 4, -1) <ul style="list-style-type: none"> • 0: Not Crawled • 1: Crawling • 2: Completed (not scheduled) • 3: Waiting • 4: Paused • -1: Error | Status (0, 1, 2, 3, 4, -1) <ul style="list-style-type: none"> • 0: Not Crawled • 1: Crawling • 2: Completed (not scheduled) • 3: Waiting • 4: Paused • -1: Error | Status (0, 1, 2, 3, 4, -1) <ul style="list-style-type: none"> • 0: Not Crawled • 1: Crawling • 2: Completed (not scheduled) • 3: Waiting • 4: Paused • -1: Error |
| Target/StatusMessage | <ul style="list-style-type: none"> • 0: Not Crawled • 1: Crawling • 2: Completed (not scheduled) • 3: Waiting • 4: Paused • -1: Error | <ul style="list-style-type: none"> • 0: Not Crawled • 1: Crawling • 2: Completed (not scheduled) • 3: Waiting • 4: Paused • -1: Error | <ul style="list-style-type: none"> • 0: Not Crawled • 1: Crawling • 2: Completed (not scheduled) • 3: Waiting • 4: Paused • -1: Error |
| Target/NumberOfRecords | Not applicable. | Not applicable. | Not applicable. |
| Target/NumberOfCompletedRecords | The number of crawled documents. | The number of crawled documents. | The number of crawled documents. |
| Target/NumberOfInsertedRecords | The number of inserted records. | The number of inserted records. | The number of inserted records. |
| Target/NumberOfUpdatedRecords | Not applicable. | The number of updated records. | The number of updated records. |
| Target/NumberOfDeletedRecords | Not applicable. | The number of deleted records. | The number of deleted records. |
| Target/StartTime | The date and time that the crawler last started. | The date and time that the crawler last started. | The date and time that the crawler last started. |
| Target/EndTime | The date and time that crawling was completed. | The date and time that crawling was completed. | The date and time that crawling was completed. |
| Target/TotalTime | The amount of time that the crawler spent crawling. | The amount of time that the crawler spent crawling. | The amount of time that the crawler spent crawling. |

Table 18. Elements and attributes for the Exchange Server crawler, the DB2 Content Manager crawler, and the Content Edition crawler for the detailed crawl space status command (continued)

| Element and attribute name | Exchange Server crawler | DB2 Content Manager crawler | Content Edition crawler |
|----------------------------|--|--|--|
| Target/AggregationLevel | 0: The Notes crawler crawls documents with normal mode. (The other crawlers except for the Notes crawler always return 0.) | 0: The Notes crawler crawls documents with normal mode. (The other crawlers except for the Notes crawler always return 0.) | 0: The Notes crawler crawls documents with normal mode. (The other crawlers except for the Notes crawler always return 0.) |
| Target/LastUpdatedTime | Not applicable. | Not applicable. | Not applicable. |
| Target/LastResetTime | Not applicable. | Not applicable. | Not applicable. |

Table 19. Elements and attributes for the QuickPlace crawler, the Domino Document Manager crawler, the UNIX file system crawler, and the Windows file system crawler for the detailed crawl space status command

| Element and attribute name | QuickPlace crawler | Domino Document Manager crawler | UNIX and Windows file system crawlers |
|---------------------------------|--|--|--|
| Target@Name | Place database name or room database name | Cabinet database name | Subdirectory name |
| Target@CrawlType | 0 | 0 | 0 |
| Target/Status | Status (0, 1, 2, 3, 4, -1) <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused -1: Error | Status (0, 1, 2, 3, 4, -1) <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused -1: Error | Status (0, 1, 2, 3, 4, -1) <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused -1: Error |
| Target/StatusMessage | <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused -1: Error | <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused -1: Error | <ul style="list-style-type: none"> 0: Not Crawled 1: Crawling 2: Completed (not scheduled) 3: Waiting 4: Paused -1: Error |
| Target/NumberOfRecords | Not applicable. | Not applicable. | Not applicable. |
| Target/NumberOfCompletedRecords | The number of crawled documents. | The number of crawled documents. | The number of crawled files. |
| Target/NumberOfInsertedRecords | The number of inserted records. | The number of inserted records. | The number of inserted records. |
| Target/NumberOfUpdatedRecords | The number of crawled place databases and room databases. | The number of crawled cabinets. | The number of subdirectories in the crawl space. |
| Target/NumberOfDeletedRecords | The number of updated records. | The number of updated records. | The number of updated records. |
| Target/StartTime | The date and time that the crawler last started. | The date and time that the crawler last started. | The date and time that the crawler last started. |
| Target/EndTime | The date and time that crawling was completed. | The date and time that crawling was completed. | The date and time that crawling was completed. |

Table 19. Elements and attributes for the QuickPlace crawler, the Domino Document Manager crawler, the UNIX file system crawler, and the Windows file system crawler for the detailed crawl space status command (continued)

| Element and attribute name | QuickPlace crawler | Domino Document Manager crawler | UNIX and Windows file system crawlers |
|----------------------------|--|--|--|
| Target/TotalTime | The amount of time that the crawler spent crawling. | The amount of time that the crawler spent crawling. | The amount of time that the crawler spent crawling. |
| Target/AggregationLevel | 0: The Notes crawler crawls documents with normal mode. (The other crawlers except for the Notes crawler always return 0.) | 0: The Notes crawler crawls documents with normal mode. (The other crawlers except for the Notes crawler always return 0.) | 0: The Notes crawler crawls documents with normal mode. (The other crawlers except for the Notes crawler always return 0.) |
| Target/LastUpdatedTime | Not applicable. | Not applicable. | Not applicable. |
| Target/LastResetTime | Not applicable. | Not applicable. | Not applicable. |

Parser status: When you run the command to obtain parser status, the command returns information in an XML document format. The following information can be returned by the parser status command:

```
FFQC5314I Result: <Monitor Type="Parser">
<ParserStatus>
  <Status>1<Status>
  <State>Parsing<State>
  <SnapshotTimeStamp>1124318637564</SnapshotTimeStamp>
  <NumberOfDocsToBeIndexed>231974</NumberOfDocsToBeIndexed>
  <ParseRate>0</ParseRate>
  <ParseRateMBPerHour>0</ParseRateMBPerHour>
  <NumberOfCpmThreads>3</NumberOfCpmThreads>
  <ParserServiceSession>parserservice.1</ParserServiceSession>
</ParserStatus>
</CrawlerStatus>
<Name>WEBCrawler1</Name>
<Crawlerid>coll.WEB1.esadmin</Crawlerid>
<Type>WEB</Type>
<ParserStatus>1</ParserStatus>
<NumberOfDocsToBeParsed>15881</NumberOfDocsToBeParsed>
<NumberOfDocsAlreadyParsed>29</NumberOfDocsAlreadyParsed>
</CrawlerStatus>
<CrawlerStatus>
<Name>Data Listener (server1)</Name>
<Crawlerid>datalistener</Crawlerid>
<Type>datalistener</Type>
<ParserStatus>0</ParserStatus>
<NumberOfDocsToBeParsed>0</NumberOfDocsToBeParsed>
<NumberOfDocsAlreadyParsed>0</NumberOfDocsAlreadyParsed>
</CrawlerStatus>
</Monitor>
```

The following table describes the XML elements for information that is returned by the parser status command:

Table 20. Elements for the parser status command

| Element | Description |
|---------|---|
| Status | 1 if the parser session for this collection is running, and 0 if the parser session is stopped. |

Table 20. Elements for the parser status command (continued)

| Element | Description |
|---------------------------|--|
| State | The possible states are: Idle, Restart, Parsing, Resuming, Stopped, Initializing, Pause, NoParserServiceIsAvailable. A status of Idle indicates that the parser is sleeping for N minutes waiting for more documents to arrive from the crawlers in this collection. The default sleep time is 300 seconds. A status of Restart indicates that the parser is waiting for the parsing/tokenizing JVM to be restarted. The parsing/tokenizing JVM runs on a separate session and is where documents are ultimately processed. A status of Parsing indicates that the parser is processing documents. A status of Parsing indicates that the parser is processing documents. A status of Pause indicates that the parser was paused by the index build session for this collection. A status of Initializing means that the parser is starting and initializing its state. A status of Resuming indicates that the parser has been resumed from a Pause state to a Parsing state by the index build session for this collection. A status of NoParserServiceIsAvailable indicates that there are no parsing/tokenizing JVMs available to process the documents for this collection. This means that all parsing/tokenizing JVMs are being used by other collections. |
| SnapshotTimeStamp | The next time in seconds since 1970 when the parser will get read documents from the crawler tables. |
| NumberOfDocsToBeIndexed | The number of documents in the store for this collection. This number also includes documents that are marked for deletion from the next index build. |
| ParseRate | The parsing rate in documents per second. |
| ParseRateMBPerHour | The parsing rate in MB per hour. |
| NumberOfCpmThreads | The number of CPM threads that are used by the parsing/tokenizing JVM to process documents for this collection. |
| ParserServiceSession | The name of the parsing/tokenizing JVM that is processing the documents for this collection. This field is available only if the parser is in the Parsing state. |
| Name | The name of the crawler. |
| Type | The type of crawler (Web, NNTP, DB2, and so on.) |
| ParserStatus | <ul style="list-style-type: none"> • 0: The documents from this crawler were not parsed yet. • 1: The documents from this crawler are currently being parsed. • 2: The documents from this crawler were parsed. |
| NumberOfDocsToBeParsed | The number of documents waiting to be parsed from this crawler. |
| NumberofDocsAlreadyParsed | The number of parsed documents from this crawler. |

Index build status: When you run the command to obtain index build status, the command returns information in an XML document format. The following information can be returned by the index build status command:

```
FFQC5314I Result: <Monitor Type="Parser">
<Monitor Type="MainIndexHistory" Count="1">
  <IndexStatus Id="1"/>
  <StartTime>1123101789411<StartTime>
```

```

    <Progress>0</Progress>
    <CurrentPhase>0</CurrentPhase>
    <TotalPhase>0</TotalPhase>
    <IndexCopyProgress>0</IndexCopyProgress>
    <CurrentServer>0</CurrentServer>
    <TotalServer>0</TotalServer>
    <IndexCopyTime>0</IndexCopyTime>
    <IndexBuildTime>0</IndexBuildTime>
    <Status>0</Status>
    <StopTime>1123101789618</StopTime>
    <NumberOfDocuments>0</NumberOfDocuments>
  </IndexStatus>
  <CurrentIndexWildcardSupport/>
  <NextIndexWildcardSupport/>
  <ScheduleStatus>
    <Status>1</Status>
</Monitor>

```

The following table describes each XML element for information that is returned by the index build status command:

Table 21. Elements for the index build status command

| Element | Description |
|-----------------------------|--|
| IndexStatusId | The index status ID. |
| StartTime | The start time in seconds since 1970 when this index build started. |
| Progress | The percentage completion for this index build. |
| CurrentPhase | <ul style="list-style-type: none"> 1: store rewrite phase 2: global analysis phase 3: index build phase |
| TotalPhase | number of phases for this index build. This value is currently 3. |
| IndexCopyProgress | The percentage completion for the index copy. The index copy process copies the built index from the index build server to the search servers. |
| CurrentServer | The search server that the index copy is copying the index to. |
| TotalServer | The number of search servers to copy the index to. |
| IndexCopyTime | The total time for all phases of index build. |
| Progress | 0 for successful index build and copy and a non-zero number for the error message code. |
| StopTime | The end time for index build (all phases) and the index copy. |
| TotalTime | The period between the start time and the stop time. |
| NumberOfDocuments | The number of documents in the index. |
| CurrentIndexWildcardSupport | The wildcard setting to be used for next index build. Possible values are None, QueryExpansion, or IndexExpansion. |
| ScheduleStatus | <ul style="list-style-type: none"> 0 if a schedule is not enabled for this collection and index type. 1 if a schedule is enabled for this collection and index type. |
| ScheduledTimeEnabled | The next time in seconds since 1970 when index build for this collection and index type will be run. |

Search server status: When you run the command to obtain search server status, the command returns information in an XML document format. The following information can be returned by the search server status command:

```

FFQC5314I Result: <?xml version="1.0"?>
<Monitor Type="Search" Count="1">
<SearchStatus Name="Search Manager (node1)" SearchID=
"searchmanager.node1" HostName="myComputer.svl.ibm.com">
<Status>1</Status>
</SearchStatus>
</Monitor>

```

The following table describes the XML elements for information that is returned by the search server status command:

Table 22. Elements for the search server status command

| Element | Description |
|------------------|--|
| SearchStatusName | The name and ID of the search manager session that is monitoring and maintaining the search index for this collection. |
| HostName | The host name of the server where the search index is running. |
| Status | <ul style="list-style-type: none"> • 0 if the search index for this collection is not running. • 1 if the search index for this collection is running. |

Detailed search server status: The command to return to search server status can return the following information:

```

FFQC5303I Search Manager (node1) (sid: searchmanager.node1)
is already running. PID: 15711
FFQC5314I Result: PID=18390
CacheHits=3
QueryRate=1
Port=44008
SessionId=coll.runtime.node1
CacheHitRate=0.333
ResponseTime=70
Status=1
SessionName=coll.runtime.node1.1

```

The following table describes the items in the information that is returned from the detailed search server status command:

Table 23. Items for the detailed search server status command

| Item | Description |
|--------------|--|
| CacheHits | The number of results retrieved from the search cache. |
| QueryRate | The number of queries received in the last time interval. By default, the time interval is five minutes. |
| Port | The port number that is used by the search index to listen or receive queries. |
| SessionId | The session ID for this collection's search index. |
| CacheHitRate | The number of results retrieved from the search cache as a percentage of all search results. |
| ResponseTime | The average response time in milliseconds for the specified time interval. (The default is five minutes.) |
| Status | <ul style="list-style-type: none"> • 0 if the search index for this collection is not running. • 1 if the search index for this collection is running. |
| SessionName | The session name for this collection's search index. |

Return codes for esadmin commands

The following codes can be returned for esadmin commands:

Table 24. Return codes for esadmin commands

| Code | Name | Description |
|------|------------------------------------|--|
| 0 | CODE_ERROR_NONE | The command completed successfully. |
| 102 | CODE_ERROR_INSTANTIATION_EXCEPTION | An error occurred when instantiating a command handler. |
| 103 | CODE_ERROR_ACCESS_EXCEPTION | An illegal access error occurred when instantiating a command handler. |
| 104 | CODE_ERROR_EXECUTE_EXCEPTION | |
| 105 | CODE_ERROR_THROWABLE | |
| 106 | CODE_ERROR_NO_SUCH_METHOD | |
| 107 | CODE_ERROR_INVALID_SESSION | |
| 108 | CODE_ERROR_INVALID_PARAMETER | |
| 109 | CODE_ERROR_SESSION_NOT_RUNNING | |

Obtaining sessions IDs

Use the esadmin check command to show a list of enterprise search components and their corresponding session IDs. The following table shows a list of common sessions, their IDs, the server that they are on, and the state of the session.

Table 25. Examples of sessions names, origin servers, session IDs, and session states

| Session | Server where the session is running | Session ID | Session state |
|---------------------|-------------------------------------|----------------|----------------|
| configmanager | index server | 10433 | Started |
| controller | index server | 10464 | Started |
| customcommunication | index server | Not applicable | Not applicable |
| datalistener | index server | 10582 | Started |
| discovery | index server | 10649 | Started |
| monitor | index server | 10682 | Started |
| parserservice | index server | 10718 | Started |
| resource.node1 | index server | 10759 | Started |
| samplecpp | index server | 10827 | Started |
| sampletest | index server | 10857 | Started |
| scheduler | index server | 10889 | Started |
| searchmanager.node1 | index server | 10927 | Started |
| utilities.node1 | index server | 10384 | Started |

Related concepts

“Backing up and restoring an enterprise search system” on page 245
Backup and restore scripts enable you to back up and restore the enterprise search system.

| “Monitoring enterprise search activity” on page 217

| When you monitor system and collection activities, you can view the status of
| various processes, watch for potential problems, or adjust configuration settings
| to enhance performance.

| **Related tasks**

| “Monitoring crawlers” on page 221

| You can view general information about the status of each crawler in a
| collection or select options to view detailed information about a crawler
| activity.

| “Starting the enterprise search servers” on page 213

| To enable users to search a collection, you must start the system processes and
| then start the servers that crawl, parse, index, and search the collection.

| “Stopping the enterprise search servers” on page 215

| You might need to stop and restart an enterprise search server if you make
| changes to its configuration or if you need to troubleshoot problems.

Enterprise search documentation

You can read the WebSphere Information Integrator OmniFind Edition (enterprise search) documentation in PDF or HTML.

The WebSphere Information Integrator OmniFind Edition installation program can automatically install the information center. The installation program installs the information center on the search server. If you do not install the information center, when you click help, the information center on an IBM Web site opens. To see HTML topics for enterprise search, you start the information center.

To see PDF documents, go to docs/*locale*/pdf. For example, to find documents in English, go to docs/en_US/pdf. You also can view the latest PDF documentation on the WebSphere Information Integrator OmniFind Edition support site.

The following table shows the available documentation, file names, and locations.

Table 26. PDF documentation for enterprise search

| Header | Header | Header |
|---|------------------------|---|
| <i>Installation Guide for Enterprise Search</i> (topics for this document are also available in the information center) | iiysi.pdf | docs/ <i>locale</i> /pdf/ |
| <i>Administering Enterprise Search</i> (Topics for this document are also available in the information center.) | iiysa.pdf | docs/ <i>locale</i> /pdf/ |
| <i>Programming Guide and API Reference for Enterprise Search</i> (Topics for this document are also available in the information center.) | iiysp.pdf | docs/ <i>locale</i> /pdf/ |
| <i>Messages for Enterprise Search</i> (Topics for this document are also available in the information center.) | iiysm.pdf | docs/ <i>locale</i> /pdf/ |
| <i>Installation Requirements for Enterprise Search</i> (Topics for this document are also available in the information center.) | iiysr.txt or iiysr.htm | docs/ <i>locale</i> / (This file can also be launched from the First Steps program.) |
| <i>Release Notes</i> | iiysn.pdf | Available only on the IBM WebSphere Information Integrator OmniFind Edition documentation Web site. |
| WebSphere Information Integrator Information Center | Not applicable | |

WebSphere II OmniFind Edition accessibility

The IBM WebSphere Information Integrator OmniFind Edition user interfaces and documentation are accessible.

Installation program

You can use keyboard shortcuts to move through the WebSphere II OmniFind Edition installation program. The following table describes some keyboard shortcuts.

Table 27. Keyboard shortcuts for the installation program

| Action | Shortcut |
|---|---|
| Highlight a radio button | Arrow key |
| Select a radio button | Tab key |
| Highlight a push button | Tab key |
| Select a push button | Enter key |
| Go to the next or previous window or cancel | Highlight a push button by pressing the Tab key and press Enter |
| Make the active window inactive | Ctrl + Alt + Esc |

Enterprise search administration console and information center

The administration console and the information center are browser-based interfaces that you can view in Microsoft Internet Explorer or Mozilla FireFox. See the online help for Internet Explorer or FireFox for a list of keyboard shortcuts and other accessibility features for your browser.

PDF documentation

You can view all of the enterprise search documentation in PDF. The PDF documents are accessible per Adobe Acrobat Version 6.0. The PDF documents are structured and should be readable by most screen readers.

Glossary of terms for enterprise search

This glossary defines terms that are used in the enterprise search interfaces and documentation.

access control list

A list that identifies the users who can access the associated object and that specifies the user's access rights to that object.

administrative role

A classification of a user that determines the functions that the user can do in the enterprise search administration console. The role also determines which collections the user can administer.

analysis engine

See text analysis engine.

analysis results

The information that is produced by annotators. Analysis results, which correspond to the information that you want to search for, are written to a data structure called a common analysis structure.

annotation

Information about a span of text. For example, an annotation could indicate that a span of text represents a company name. In UIMA, an annotation is a special kind of feature structure.

annotator

A software component that performs specific linguistic analysis tasks and produces and records annotations. An annotator is the analysis logic component in an analysis engine.

boolean search

A search in which one or more search terms are combined by using operators such as AND, NOT, and OR.

boost class

A specification that can influence the relative rank of a document in the search results.

boost word

A word that can influence the relevant rank of a document in the search results. During query processing, the importance of a document that contains a boost word might be raised or lowered, depending on a score that is predefined for the word.

category

A group of documents that have similar properties.

category tree

A hierarchy of categories that is displayed in the enterprise search administration console.

certificate

A digital document that binds a public key to the identity of the certificate owner, thereby enabling the certificate owner to be authenticated. A certificate is issued by a certificate authority.

certificate authority

An organization that issues certificates and authenticates the entities (individuals or organizations) that are involved in electronic transactions. Certificate authorities guarantee that the two parties exchanging information are really who they claim to be.

character normalization

A process in which the variant forms of a character, such as capitalization and diacritical marks, are reduced to a common form.

clitic A word that syntactically functions separately but is phonetically connected to another word. A clitic can be written as connected or separate from the word it is bound to. Common examples of clitics include the last part of a contraction in English (*wouldn't* or *you're*).

collection

A set of data sources and options for crawling, parsing, indexing, and searching those data sources.

common analysis structure

A structure that stores a document that is being analyzed by a text analysis engine. Information is stored in the common analysis structure in the form of annotations and other feature structures.

Common Communication Layer (CCL)

The communication infrastructure that unites the various components (controller, parser, crawler, indexer) of WebSphere Information Integrator OmniFind Edition.

concept extraction

A search function that identifies significant vocabulary items (such as people, places, or products) in text documents and produces a list of those items. See also theme extraction.

crawl space

A set of sources that match specified patterns (such as database names, file system paths, domain names, IP addresses, and URLs) that a crawler reads from to retrieve items for indexing.

crawler

A software program that retrieves documents from data sources and gathers information that can be used to create search indexes.

credential

Detailed information, which is acquired during authentication, that describes the user, any group associations, and other security-related identity attributes. Credentials can be used to perform a multitude of services, such as authorization, auditing, and delegation.

data source

Any repository of data from which documents can be retrieved, such as the Web, relational and nonrelational databases, and content management systems.

data source type

A grouping of data sources according to the protocol that is used to access the data.

dequeue

To remove items from a queue.

diacritics

A mark that is added to a letter to change a word's pronunciation or to distinguish between similar words, such as an accent mark or the German umlaut.

discoverer

A function of a crawler that determines which data sources are available for the crawler to retrieve information from.

distinguished name

The name that uniquely identifies an entry in a directory. A distinguished name consists of attribute:value pairs, separated by commas. Also, a set of name-value pairs (such as CN=person's name and C=country or region) that uniquely identifies an entity in a digital certificate.

Document Object Model

A system in which a structured document, such as an XML file, is viewed as a tree of objects that can be programmatically accessed and updated.

Domino Document Manager cabinet

A Domino Document Manager database that is used to organize documents. Cabinets hold Domino databases.

Domino Document Manager library

A Domino Document Manager database that is the entry point to Domino Document Manager.

Domino Internet Inter-ORB Protocol (DIIOP)

A server task that runs on the server and works with the Domino Object Request Broker to allow communication between Java applets that are created with the Notes Java classes and the Domino server. Browser users and Domino servers use DIIOP to communicate and to exchange object data.

dynamic ranking

A type of ranking in which the terms in the query are analyzed with respect to the documents that are being searched to determine the rank of results. See also text-based scoring. Contrast with static ranking.

dynamic summarization

A type of summarization in which the search terms are highlighted and the search results contain phrases that best represent the concepts of the document that the user is searching for. Contrast with static summarization.

enqueue

To place items in a queue.

enterprise search administrator

An administrative role that enables a user to administer the entire enterprise search system.

escape character

A character that suppresses or selects a special meaning for one or more characters that follow.

external data source

A data source for federation that is not crawled, parsed, or indexed by WebSphere Information Integrator OmniFind Edition. Searches of external data sources are delegated to the query application programming interface of those data sources.

feature path

A path that is used to access the value of a feature in a UIMA feature structure.

feature structure

The underlying data structure that represents the result of text analysis. A feature structure is an attribute-value structure. Each feature structure is of a type, and every type has a specified set of valid features or attributes, much like a Java class.

federated search

A search capability that enables searches across multiple search services and returns a consolidated list of search results.

federation

The process of combining naming systems so that the aggregate system can process composite names that span the naming systems.

field The smallest identifiable part of a record.

fielded search

A query that is restricted to a particular field.

free text search

A search in which the search term is expressed as free-form text.

full text index

A data structure that references data items to enable the search to quickly find documents that contain the query terms.

fuzzy search

A search that returns words with spelling that is similar to that of the search term.

hybrid search

A combined boolean search and free text search.

identity management

The ability to encrypt user credentials in a secure store.

index See full text index.

index queue

A list of requests for index reorganization or requests for index refresh to be processed.

index refresh

The process of adding new information to an existing index in an enterprise search system. Contrast with index reorganization.

index reorganization

The process of building the index in an enterprise search system. Contrast with index refresh.

information extraction

A type of concept extraction that automatically recognizes significant vocabulary items, such as names, terms, and expressions, in text documents.

IP address

The unique 32-bit address that identifies a host on the network.

Java Database Connectivity (JDBC)

An industry standard for database-independent connectivity between the

Java platform and a wide range of databases. The JDBC interface provides a call-level API for SQL-based database access.

JavaScript

A Web scripting language that is used in browsers and Web servers.

JavaServer Pages (JSP)

A server scripting technology that enables Java code to be dynamically embedded within Web pages (HTML files) and executed when the page is served, in order to return dynamic content to a client.

Java virtual machine (JVM)

A software implementation of a processor that runs compiled Java code (applets and applications).

Katakana

A character set that consists of symbols that are used in one of the two common Japanese phonetic alphabets, which is used primarily to write foreign words phonetically.

keystore file

A key database file that contains public keys that are stored as signer certificates and private keys that are stored in personal certificates.

language identification

An enterprise search function that determines the language of a document.

lemma

The canonical form of a word. Lemmas are significant in highly inflected languages such as Czech.

lemmatization

The process of looking up the lemma for a given word in a dictionary. Lemmatization differs from stemming in that stemming is algorithmic and generally does not operate with a dictionary that lists the words of a language.

lexical affinity

The relationship of search words that appear close to each other in the document. Lexical affinity is used to calculate the relevancy of a result.

library

A system object that serves as a directory to other objects. See also Domino Document Manager library.

ligature

Two or more characters that are connected so that they appear as one character, such as joining f and i form the ligature fi.

Lightweight Directory Access Protocol (LDAP)

An open protocol that uses TCP/IP to provide access to directories that support an X.500 model and that does not incur the resource requirements of the more complex X.500 Directory Access Protocol.

linguistic search

A search type that browses, retrieves, and indexes a document with terms that are reduced to their base form (for example, so that *mice* is indexed as *mouse*) or expanded with their base form (as with compound words).

link analysis

A method that is based on the analysis of hyperlinks between documents and used to determine what pages in the collection are important to users.

local federator

A client federator that federates over a set of searchable objects.

Lotus QuickPlace place

A Web venue that is provided by Lotus QuickPlace that enables geographically dispersed participants to collaborate on projects and communicate online in a structured and secure workspace.

Lotus QuickPlace room

A partitioned area of a Lotus QuickPlace place that is restricted to authorized members who share a common interest and a need to work collectively.

masking character

A character that is used to represent optional characters at the front, middle, and end of a search term. Masking characters are normally used for finding variations of a term in an index. See also wildcard character.

MIME type

An Internet standard for identifying the type of object that is being transferred across the Internet.

model-based category

A taxonomy of predefined terms that is used to determine the subject of a document so that the document can be indexed and searched with documents that have similar content.

monitor

An enterprise search user who has the authority to observe collection-level processes.

natural language query

A type of search that analyzes written expressions (such as "Who runs the finance department?") instead of a simple collection of keywords.

newline character

A control character that causes the print or display position to move down one line. Some systems require more than one character.

n-gram segmentation

A method of analysis that considers overlapping sequences of a given number of characters as a single word rather than using blank space to delimit words as in Unicode-based white space segmentation.

no-follow directive

A directive in a Web page that instruct robots (such as the Web crawler) to not follow links found in those pages.

no-index directive

A directive in a Web page that instruct robots (such as the Web crawler) to not include the contents of those pages in the index.

Notes remote procedure call (NRPC)

The architectural layer of Lotus Notes that is used for all Notes-to-Notes communication.

operator

An enterprise search user who has the authority to observe, start, and stop collection-level processes.

parametric search

A type of search that looks for objects that contain a numeric value or attribute, such as dates, integers, or other numeric data types within a specified range.

parser A program that interprets documents that are added to the enterprise search data store. The parser extracts information from the documents and prepares them for indexing, search, and retrieval.

place A virtual location that is visible in the portal where individuals and groups meet to collaborate. In a portal, each user has a personal place for private work, and individuals and groups have access to a variety of shared places, which can be either public places or restricted places. See also Lotus QuickPlace place.

popular ranking

A ranking type that adds to a document's existing ranking based on the document's popularity.

processing engine archive

A .pear zip archive file that includes a UIMA analysis engine and all of the resources required to use it for custom analysis in enterprise search.

proximity search

A search type that looks for certain words in the same sentence, paragraph, or document.

proxy server

A server that acts as an intermediary for HTTP Web requests that are hosted by an application or a Web server. A proxy server acts as a surrogate for the content servers in the enterprise.

quick link

An association between a URI and keywords and phrases.

ranking

The process of assigning an integer value to each document in the search results from a query. The order of the documents in the search results is based on the relevance to the query. A higher rank signifies a closer match. See also dynamic ranking and static ranking.

remote federator

A server federator that federates a set of searchable objects.

Robots Exclusion Protocol

A protocol that allows Web site administrators to indicate to visiting robots which parts of their site should not be visited by the robot.

room A program that allows users to create documents for others to read, respond to comments from others, and review project status and deadlines. Users can also chat with others who are in the same room. See also Lotus QuickPlace room.

rule-based category

Categories that are created by rules that specify which documents are associated with which categories. For example, you can define rules to associate documents that contain or exclude certain words, or that match a URI pattern, with specific categories.

scope A group of related URIs that is used to define the range of a search request.

search application

A program that processes queries, searches the index, returns the search results, and retrieves the source documents for collections in an enterprise search system.

search cache

A buffer that holds the data and results of previous search requests.

search engine

A program that accepts a search request and returns a list of documents to the user.

search index files

The set of files in which an index is stored in the search engine.

search results

A list of documents that match the search request.

Secure Sockets Layer (SSL)

A security protocol that provides communication privacy.

security token

Information about identity and security that is used to authorize access to documents in a collection. Different data source types support different types of security tokens. Examples include user roles, user IDs, group IDs, and other information that can be used to control access to content.

seed URL

The starting point for a crawl.

segmentation

A process by which path control divides basic information units into smaller units, called BIU segments, to accommodate smaller buffer sizes in adjacent servers.

servlet

A Java program that runs on a Web server and extends the server's functionality by generating dynamic content in response to Web client requests. Servlets are commonly used to connect databases to the Web.

soft error page

A special page that explains the problem in detail if an HTTP server cannot return the page that a client requests and configures the HTTP server to return these pages instead of a response that consists of only a header with a return code that indicates what the problem is.

static ranking

A type of ranking in which factors about the documents that are being ranked, such as date, the number of links that point to the document, and so on, augment the rank. Contrast with dynamic ranking.

static summarization

A type of summarization in which the search results contain a specified, stored summary from the document. Contrast with dynamic summarization.

stemming

See word stemming.

stop word

A word that is commonly used, such as *the*, *an*, or *and*, that is ignored by a search application.

stop word removal

The process of removing stop words from the query to ignore common words and return more relevant results.

summarization

The process of including sentences in search results to briefly describe the content of a document. See also dynamic summarization and static summarization.

synonym dictionary

A dictionary that enables users to search for synonyms of their query terms when they search a collection.

taxonomy

A classification of objects into groups based on similarities. In enterprise search, a taxonomy organizes data into categories and subcategories. See also category tree.

text analysis

The process of extracting semantics and other information from text to enhance the retrievability of data in a collection.

text analysis engine

A software component that is responsible for finding and representing context and semantic content in text.

text-based scoring

The process of assigning an integer value to a document that signifies the relevance of the document with respect to the terms in a query. A higher integer value signifies a closer match to the query. See also dynamic ranking.

theme extraction

A type of concept extraction that automatically recognizes significant vocabulary items in text documents to extract the theme or topic of a document. See also concept extraction.

token The basic textual units that are indexed by enterprise search. Tokens can be the words in a language or other units of text that are appropriate for indexing.

tokenizer

A text segmentation program that scans text and determines if and when a series of characters can be recognized as a token.

trailing character

A character that holds the last position in a word.

Unicode-based white space segmentation

A method of tokenization that uses Unicode character properties to distinguish between token and separator characters.

Uniform Resource Identifier (URI)

A compact string of characters that identifies an abstract or physical resource.

Uniform Resource Locator (URL)

A sequence of characters that represents information resources on a computer or in a network such as the Internet. This sequence of characters includes the abbreviated name of the protocol that is used to access the information resource and the information that is used by the protocol to locate the information resource.

Universal Resource Name (URN)

An Internet protocol element that consists of a short string of characters that conform to a certain syntax. The string comprises a name or address that can be used to refer to a resource.

Unstructured Information Management Architecture (UIMA)

An IBM architecture that defines a framework for implementing systems for the analysis of unstructured data.

user agent

An application that browses the Web and leaves information about itself at the sites that it visits. In enterprise search, the Web crawler is a user agent.

Web crawler

A class of robot software that explores the Web by retrieving a Web document and following the links within that document.

weighted term search

A query in which certain terms are given more importance.

wildcard character

A character that is used to represent optional characters at the front, middle, or end of a search term.

word stemming

A process of linguistic normalization in which the variant forms of a word are reduced to a common form. For example, words like *connections*, *connective*, and *connected* are reduced to *connect*.

XML Path Language (XPath)

A language that uniquely identifies or addresses parts of a source XML document. XPath also provides basic facilities for manipulation of strings, numbers, and boolean operators.

Accessing information about WebSphere Information Integration

Information about WebSphere Information Integration products is available by telephone or on the Web.

The phone numbers provided here are valid in the United States:

- To order products or to obtain general information: 1-800-IBM-CALL (1-800-426-2255)
- To order publications: 1-800-879-2755

You can also find information about WebSphere Information Integration on the Web at www.ibm.com/software/data/integration/db2ii/. This site contains the latest information about:

- Product documentation
- Product downloads
- Fix packs
- Release notes and other support documentation
- News about WebSphere Information Integration
- Links to Web resources, such as white papers and IBM Redbooks™
- Links to newsgroups and user groups
- Links to online information centers for WebSphere Information Integration products
- Ordering books

To access product documentation:

1. Visit the Web at www.ibm.com/software/data/integration/db2ii/.
2. Select a product from the drop-down list, for example, WebSphere Information Integrator OmniFind Edition.
3. Click the Support link on the left side of the page.
4. In the Learn section, select the link that you want. If an information center is available for the product that you selected, you can select the link for the information center. See Figure 2 on page 292 for an example.

Learn

- **Product documentation and manuals** (2 items)
- **Redbooks** (1 item)
- **V8.2 Documentation and release notes**

Information Center

Provides fast, online centralized access to product information.

- [1.0](#)

Figure 2. Example of links to product documentation on a WebSphere Information Integration Support Web site

Providing comments on the documentation

Please send any comments that you have about this information or other IBM WebSphere Information Integration documentation.

Your feedback helps IBM to provide quality information. Please send any comments that you have about this information or other WebSphere Information Integration documentation. You can use any of the following methods to provide comments:

1. Send your comments using the online readers' comment form at www.ibm.com/software/awdtools/rcf/ .
2. Send your comments by e-mail to comments@us.ibm.com. Include the name of the product, the version number of the product, and the name and part number of the information (if applicable). If you are commenting on specific text, please include the location of the text (for example, a title, a table number, or a page number).

Contacting IBM

To contact IBM customer service in the United States or Canada, call 1-800-IBM-SERV (1-800-426-7378).

To learn about available service options, call one of the following numbers:

- In the United States: 1-888-426-4343
- In Canada: 1-800-465-9600

To locate an IBM office in your country or region, see the IBM Directory of Worldwide Contacts on the Web at www.ibm.com/planetwide.

Trademarks

This topic lists IBM trademarks and certain non-IBM trademarks.

See <http://www.ibm.com/legal/copytrade.shtml> for information about IBM trademarks.

The following terms are trademarks or registered trademarks of other companies:

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

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

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

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

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

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

Notices

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

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

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country/region or send inquiries, in writing, to: IBM World Trade Asia Corporation Licensing 2-31 Roppongi 3-chome, Minato-ku Tokyo 106-0032, Japan

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

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

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

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

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information that has been exchanged, should contact:

IBM Corporation J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003 U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems, and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

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

COPYRIGHT LICENSE:

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

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:

Outside In (®) Viewer Technology, ©1992-2005 Stellent, Chicago, IL., Inc. All Rights Reserved.

IBM XSLT Processor Licensed Materials - Property of IBM ©Copyright IBM Corp.,
1999-2005. All Rights Reserved.

Index

A

- access controls
 - current validation of 190
 - description 184
 - disabling for a collection 197
 - document-level security 189, 201
 - Portal Search Engine support 201
 - requirements for Lotus Domino 195
 - requirements for Windows file systems 193
 - user profiles 191
- accessibility 279
- active Web sites, monitoring 223, 224
- adapter for enterprise search 204
- address rules for Web crawlers 70
- administration console
 - description 8
 - interface 15
 - logging in 18
 - task summary 15
- administrative roles
 - collection administrator 182, 183
 - configuring 183
 - description 182
 - enterprise search administrator 182, 183
 - monitor 182, 183
 - operator 182, 183
- administrator password
 - changing on a single server 19
 - changing on multiple servers 21
- AIX operating system
 - Content Edition crawler
 - configuration 40
 - DB2 Content Manager crawler
 - configuration 50
 - Domino Document Manager crawler
 - configuration 58
 - event publishing configuration 47
 - Notes crawler configuration 58
 - QuickPlace crawler configuration 58
- alerts
 - collection-level 217, 238
 - description 237
 - documents crawled 238
 - documents indexed 238
 - e-mail options 238, 239
 - free space on servers 239
 - index limits 217
 - receiving e-mail for 242
 - search response times 238
 - SMTP server configuration 241
 - system-level 239
- anchor text analysis
 - collection security 187
 - description 181
 - global analysis 187
 - indexing documents 187
- annotators 108
- APIs
 - Data Listener 10

- APIs (*continued*)
 - description 10
 - Search and Index 10, 161

- application IDs 188
- authentication
 - description 184
 - disabling for enterprise applications 185

B

- backing up enterprise search 245
- backup scripts
 - description 245
 - running 245, 247
- boost classes
 - configuration 156, 157
 - default values 157
 - description 154
 - duplicate document detection 154
 - high recall queries 154, 157
 - low recall queries 154, 157
 - mapping fields to 156
- boost factors
 - boost class configuration 154, 157
 - for boost classes 156, 157
 - for boost word dictionaries 150
 - for URI patterns 152, 153
- boost word dictionaries
 - adding to the system 151
 - associating with a collection 152
 - description 150
- bos.iocp.rte module 62

C

- categories
 - categorization type 102
 - category trees 102
 - creating 103
 - description 98
 - migrating from WebSphere Portal 207, 209
 - model-based 101, 102
 - nesting subcategories 102
 - rule-based 99, 102
 - searching 98
 - URI formats 89
- categorization type
 - model-based 101
 - rule-based 99
 - selecting 28, 102
- category rules
 - configuring 103
 - document content 99, 103
 - URI patterns 99, 103
- category trees
 - description 102
 - migrating from WebSphere Portal 207, 209

- CCLServer_date.log file 24
- checking system resources 218
- Chinese
 - n-gram segmentation 117
 - parsing new line characters 117
- collapsed search results
 - configuring 137
 - description 136
- collapsed URIs
 - configuring 137
 - description 136
- collection administrator
 - description 182
 - role configuration 183
- collection ID, syntax rules 28
- Collection wizard 27
- collection-level security
 - anchor text analysis 187
 - application IDs 188
 - description 181, 186
 - duplicate document detection 186
 - enabling 28
- collections
 - anchor text security 187
 - application ID security 188
 - associating with search applications 162
 - bypassing document-level access controls 197
 - creating with Collection wizard 27
 - creating with Collections view 28
 - default migration settings 210
 - deleting 31
 - description 3
 - draft 27
 - duplicate document security 186
 - editing 30
 - estimating resources 217
 - estimating the size 28
 - federation 27
 - migrating from WebSphere Portal 207, 209
 - MigrationWizard.log file 209
 - monitoring 219
 - parsing 97
 - search servers 139
 - searching 129
 - security 186
 - system status 219
 - ways to create 27
- Collections view
 - creating collections 28
 - description 15
- commands, enterprise search 249
- common analysis structures
 - description 108
 - mapping to JDBC tables 114
 - mapping to the index 113
 - mapping XML elements to 111
- compound terms, parsing 115
- concurrent index builds 128

- config.properties file
 - editing 172
 - property descriptions 163
- Content Edition crawlers
 - configuration 38
 - direct mode 39
 - server mode 39
 - setting up in Solaris operating environment 40
 - setting up on AIX operating system 40
 - setting up on Linux operating system 40
 - setting up on Windows 41
 - URI formats 89
- cookies for Web crawling
 - configuring 80
 - description 79
 - format 79
- cookies.ini file
 - configuring 80
 - description 79
 - format 79
- Crawl page, description 15
- crawl rate, monitoring 225
- crawl space
 - alerts about 238
 - description 3
 - editing 37
 - Web crawler configuration 70
- crawl.rules file 80
- crawler history reports
 - creating 225
 - description 223
 - HTTP return code report 225
 - Site report 225
- crawler plug-ins 88
- crawler properties
 - description 3
 - editing 36
- crawler servers
 - starting 213, 221
 - stopping 215, 221
- crawler types
 - base values for 33
 - combining in a collection 33
- crawlers
 - base values for 33
 - combining crawler types 33
 - configuration overview 33
 - Content Edition 38, 39
 - creating 35
 - Data Listener applications 87
 - DB2 42
 - DB2 Content Manager 49
 - default migration settings 210
 - deleting 37
 - description 3
 - document-level security 189
 - Domino Document Manager 52
 - editing crawl spaces 37
 - editing crawler properties 36
 - enabling document-level security 33
 - Exchange Server 54
 - initial values for 35
 - monitoring 221
 - NNTP 55

- crawlers (*continued*)
 - Notes 56, 57
 - plug-ins 88
 - QuickPlace 64
 - scheduling 33
 - support for external 10, 87
 - system status 221
 - UNIX file system 67
 - URI formats 89
 - Web 68
 - WebSphere Portal 83, 84, 85
 - Windows file system 86
- creating
 - collections 27, 28
 - crawlers 35
 - HTML search fields 108
 - quick links 147
 - rule-based categories 103
 - scopes 135
 - Web crawler reports 225
 - XML search fields 105
- custom text analysis
 - description 108
 - mapping analysis results to JDBC tables 114
 - mapping analysis results to the index 113
 - mapping common analysis structures 113, 114
 - mapping XML elements 111
 - text analysis engines 110, 111

D

- data flow, enterprise search system 11
- Data Listener
 - configuring 87
 - monitoring 233
 - restarting 87, 233
- Data Listener API 10
- data source types
 - Content Edition repositories 38, 39
 - DB2 Content Manager item types 49
 - DB2 databases 42
 - Domino Document Manager databases 52
 - Exchange Server public folders 54
 - NNTP news groups 55
 - Notes databases 56, 57
 - QuickPlace databases 64
 - relational databases 42
 - support for external 2, 10
 - supported by enterprise search 2
 - UNIX file systems 67
 - Web sites 68
 - WebSphere Portal sites 83
 - Windows file systems 86
- DB2 Content Manager crawlers
 - configuration 49
 - setting up in Solaris operating environment 50
 - setting up on AIX operating system 50
 - setting up on Linux operating system 50
 - setting up on Windows 51
 - URI formats 89

- DB2 crawlers
 - configuration 42
 - event publishing 42
 - event publishing configuration 47, 48
 - URI formats 89
 - WebSphere II Event Publisher Edition configuration 43
 - WebSphere MQ configuration 46
 - WebSphere MQ installation 47
 - WebSphere MQ installation on Windows 48
- Default search application 172
- deleting
 - collections 31
 - crawlers 37
 - indexes from the queue 231
- deployment
 - adapter for enterprise search 204
 - ESPACServer.ear file 84
 - ESSearchAdapter.ear file 204
 - ESSearchAdapterPortlet.war file 205
 - ESSearchPortlet.war file 201
 - registration portlet 205
 - Search and Browse portlet 203
 - Search portlet 201
- DIIOF protocol, crawler
 - configuration 61
- direct mode, Content Edition
 - repositories 39
- Directory Assistance configuration 66
- disabling index schedules 127
- discovery 3
- document content options 129
- document importance
 - boost classes 154, 157
 - boost word dictionaries 152
 - enabling for a collection 28
 - in migrated collections 209
 - static 150
 - URI patterns 152, 153
- document summaries
 - customizing 145
 - editing properties for 146
- document tracking
 - description 234
 - disabling 235
 - enabling 235
 - log file configuration 235
 - reports 235
- document types
 - for parser services 118, 120
 - for Stellent sessions 120, 123
 - parsing 118
- document-level security
 - crawler configuration 33
 - crawler plug-ins 88
 - current credential validation 190
 - description 181, 189
 - for Lotus Domino documents 195
 - for Windows file systems 193
 - identity management 192
 - indexed access controls 189
 - Lotus Domino documents 195
 - Portal Search Engine support 201
 - real time validation 190
 - security tokens 189
 - user profiles 191

- documentation 277
- domain rules for Web crawlers 70
- Domino Document Manager crawlers
 - configuration 52
 - DIIO protocol configuration 61
 - IOCP configuration 62
 - NRPC protocol 58, 60
 - setting up in Solaris operating environment 58
 - setting up on AIX operating system 58
 - setting up on Linux operating system 58
 - setting up on Windows 60
 - URI formats 89
- Domino user configuration, QuickPlace crawlers 65
- dropped documents
 - description 234
 - log file configuration 235
 - reports about 235
- duplicate document detection
 - boost class configuration 154
 - description 181
 - enabling security 186
 - global analysis 186
- dynamic ranking 149
- dynamic summarization 145

E

- e-mail notifications
 - for alerts 242
 - for messages 242
 - SMTP server configuration 241
- EAR files
 - ESAdmin application 185
 - ESPACServer.ear 84
 - ESSearchAdapter.ear 204
 - ESSearchApplication application 185
 - ESSearchServer application 185
- editing
 - collections 30
 - crawl spaces 37
 - crawler properties 36
 - Data Listener applications 87
 - search application properties 163, 172
- enabling index schedules 127
- enterprise applications
 - ESAdmin application 185
 - ESPACServer.ear file 84
 - ESSearchAdapter.ear file 204
 - ESSearchApplication application 185
 - ESSearchServer application 185
- enterprise search
 - administration console 8
 - administrative roles 182
 - APIs 10
 - backing up 245
 - backup scripts 245
 - collection-level security 186
 - commands 249
 - components 3
 - crawler servers 3, 33
 - data flow diagram 11
 - document-level security 189
 - enterprise search (*continued*)
 - index servers 6, 125
 - integration with WebSphere Portal 199
 - log files 237
 - monitoring 217
 - overview 1
 - parsers 4, 97
 - port number configuration 24
 - restore scripts 245
 - restoring from a backup 246
 - restoring system files 247
 - return codes 249
 - search applications 11
 - search servers 7, 139
 - security 181
 - session IDs 249
 - starting the servers 213
 - stopping the servers 213, 215
 - URI formats 89
 - enterprise search administrator
 - changing the password on a single server 19
 - changing the password on multiple servers 21
 - description 182
 - role configuration 183
 - error messages
 - receiving e-mail for 240, 242
 - SMTP server configuration 241
 - viewing log files 244
 - ES_INSTALL_ROOT, description 19, 21
 - ES_NODE_ROOT, description 19, 21
 - es_special_field.default_field 157
 - es_special_field.regular_text field 157
 - es.cfg file 19, 21
 - ESAdmin application
 - disabling security 185
 - logging in to 18
 - esadmin command 249
 - esbackup.bat script 245, 247
 - esbackup.sh script 245, 247
 - esexchangepw script 19, 21
 - escrm.sh script 50
 - escrm.vbs script 51
 - esrdb2.sh script 47
 - esrdb2.vbs script 48
 - escrnote.sh script 58
 - escrnote.vbs script 60
 - escrvbr.sh script 40
 - escrvbr.vbs script 41
 - ESPACServer.ear file 84
 - esrestore.bat script 246, 247
 - esrestore.sh script 246, 247
 - ESSearchAdapter.ear file 204
 - ESSearchAdapterPortlet.war file 205
 - ESSearchApplication application
 - config.properties file 163, 172
 - disabling security 185
 - enabling security 174
 - restarting 172, 174
 - ESSearchPortlet.war file 201
 - ESSearchServer application
 - disabling security 185
 - estimating system resources 217
 - event publishing
 - DB2 crawler configuration 43, 46

- event publishing (*continued*)
 - description 42
 - setting up in Solaris operating environment 47
 - setting up on AIX operating system 47
 - setting up on Linux operating system 47
 - setting up on Windows 48
- Exchange Server crawlers
 - configuration 54
 - secure documents 54
 - URI formats 89
- external crawlers
 - configuring 87
 - Data Listener API 10
 - Data Listener applications 87
- external sources
 - application ID security 188
 - associating with search applications 178
 - configuring 177
 - description 177
 - searching 129

F

- federated collections 27
- fielded search 129
- fields, mapping to boost classes 156
- file extensions
 - excluding from Web crawl spaces 70
 - supported by collection parsers 118, 120
 - supported by Stellent sessions 120, 123
- finding enterprise search
 - documentation 277
- firewalls, crawling Exchange Server documents 54
- followindex.rules file
 - configuring 83
 - description 82
- form-based authentication 76, 77
- free space alerts 239
- free text search 129

G

- global analysis
 - anchor text analysis 181, 187
 - description 6
 - duplicate document detection 181, 186
- global Web crawl space 80
- global.rules file 80

H

- high recall queries
 - default boost factors 157
 - description 154
- HTML documents, searching 107
- HTML search fields
 - creating 108
 - description 107

- HTML search fields (*continued*)
 - mapping elements to 107, 108
- HTTP basic authentication 76
- HTTP proxy servers 78
- HTTP return codes
 - received by Web crawlers 225
 - Web crawler report 225

I

- I/O completion port module, crawler
 - configuration 62
- identity management
 - configuration 192
 - user profiles 191
- index builds
 - concurrent 128
 - parallel 128
 - starting 230
 - stopping 230, 231
 - system status 231
- Index page, description 15
- index queue 231
- index refresh
 - description 6, 125
 - scheduling 126, 127
- index reorganization
 - description 6, 125
 - scheduling 126, 127
- index servers
 - starting 213
 - stopping 215
- indexes
 - alerts about 238
 - anchor text 187
 - changing the schedule 127
 - collapsed URIs 129, 136, 137
 - concurrent builds 128
 - deleting from the queue 231
 - description 6, 125
 - disabling the schedule 127, 230
 - enabling the schedule 127, 230
 - monitoring 230, 231
 - parallel builds 128
 - removing URIs 129, 137
 - scheduling 126
 - scopes 129, 134
 - URI formats 89
 - wildcard characters 129, 131, 133
- IOCP, crawler configuration 62
- IP address rules for Web crawlers 70

J

- Japanese
 - n-gram segmentation 117
 - parsing new line characters 117
- Java connector for DB2 Content Manager 50, 51
- JavaScript support in Web crawlers 69
- JDBC external sources
 - configuring 177
 - deleting 177
 - editing 177
 - JDBC drivers 177

K

- keywords in quick links 147
- Korean
 - compound term analysis 115
 - n-gram segmentation 117

L

- LDAP external sources
 - configuring 177
 - deleting 177
 - editing 177
- limiting the Web crawl space 70
- linguistic support
 - boost word dictionaries 150
 - custom text analysis 108
 - native XML search 116
 - semantic search 108, 116
 - stop word dictionaries 143
 - synonym dictionaries 140
- Linux operating system
 - Content Edition crawler
 - configuration 40
 - DB2 Content Manager crawler
 - configuration 50
 - Domino Document Manager crawler
 - configuration 58
 - event publishing configuration 47
 - Notes crawler configuration 58
 - QuickPlace crawler configuration 58
 - Solaris operating environment
 - event publishing configuration 47
- Local User security, QuickPlace crawlers 65
- log files
 - default location 237
 - description 237
 - e-mail options 242
 - filtering 244
 - for document tracking 235
 - maximum size 240
 - migration wizard 212
 - monitoring 244
 - rotating 240
 - severity levels 240
 - SMTP server configuration 241
 - viewing 244

- Log page, description 15
- logging in to the administration
 - console 18
- Lotus Domino domains 195
- Lotus Domino Trusted Servers 195
- low recall queries
 - default boost factors 157
 - description 154

M

- mapping
 - common analysis structures to JDBC tables 114
 - common analysis structures to the index 113
 - fields to boost classes 156
 - HTML search fields 108

- mapping (*continued*)
 - XML elements to common analysis structures 111
 - XML search fields 105
- maximum recrawl interval 74
- migrating
 - collections 209
 - model-based taxonomy 207
 - rule-based taxonomy 209
- migration wizard
 - collections 209
 - default collection settings 210
 - default crawler settings 210
 - description 207
 - log file 212
 - model-based taxonomies 207
 - rule-based taxonomies 209
 - starting 207, 209
- MIME types, including in Web crawl spaces 70
- minimum recrawl interval 74
- model-based categories
 - description 101
 - selecting the categorization type 102
- model-based taxonomy, migrating from WebSphere Portal 207
- monitor
 - description 182
 - role configuration 183
- Monitor view, description 15
- monitoring
 - collections 219
 - crawlers 221
 - Data Listener 233
 - dropped documents 235
 - enterprise search 217
 - log files 244
 - parsers 229
 - popular queries 232
 - recent queries 232
 - response time history 232
 - search servers 232
 - URI details 220
 - Web crawler active sites 224
 - Web crawler crawl rate 225
 - Web crawler thread details 223
 - Web crawlers 223

N

- n-gram segmentation 117
- native XML search 116
- NNTP crawlers, configuring 55
- no-follow directives
 - configuring 83
 - description 82
- no-index directives
 - configuring 83
 - description 82
- Notes crawlers
 - configuration 56
 - DIOP protocol configuration 61
 - document-level security
 - configuration 195
 - field mapping rules 57
 - IOCP configuration 62
 - Lotus Domino Trusted Server 195

Notes crawlers (*continued*)
 NRPC protocol 58, 60
 setting up in Solaris operating environment 58
 setting up on AIX operating system 58
 setting up on Linux operating system 58
 setting up on Windows 60
 tips for using 57
 URI formats 89
 validation of current credentials 195
 NRPC protocol, crawler configuration 58, 60

O

operator
 description 182
 role configuration 183

P

parallel index builds 128
 parametric search 129
 Parse page, description 15
 parser servers
 starting 213
 stopping 215
 thread configuration 114
 parsers
 compound term analysis 115
 data analysis tasks 4
 description 4, 97
 document types for parser services 118, 120
 document types for Stellent sessions 120, 123
 monitoring 229
 n-gram segmentation 117
 native XML search 116
 new line characters 117
 parsing document types 118
 starting 229
 stopping 229
 system status 229
 threads 114
 white space 117
 parserTypes.cfg file 118
 password-protected Web sites 76
 form-based authentication 77
 HTTP basic authentication 76
 password, enterprise search administrator 19, 21
 PDF documentation 277
 plug-ins, for crawlers 88
 popular queries, monitoring 232
 port number, enterprise search 24
 Portal Search Engine
 description 199
 document-level security 201
 passing security tokens 201
 portlets
 description 199
 enterprise search 199
 ESSearchAdapterPortlet.war file 205

portlets (*continued*)
 registration 205
 Search 201
 Search and Browse 203
 supported versions of WebSphere Portal 199
 Taxonomy Management Portlet 207
 prefix rules for Web crawlers 70
 proxy servers 78

Q

query validation 190
 quick links
 creating 147
 description 147
 searching 147
 URI formats 89
 QuickPlace crawlers
 configuration 64
 DIIOP protocol configuration 61
 Directory Assistance configuration 66
 Domino user configuration 65
 IOCP configuration 62
 Local User security 65
 NRPC protocol 58, 60
 setting up in Solaris operating environment 58
 setting up on AIX operating system 58
 setting up on Linux operating system 58
 setting up on Windows 60
 URI formats 89

R

ranking search results
 boost classes 154, 156, 157
 boost word dictionaries 152
 description 149
 dynamic 149
 static 150
 text-based scoring 149
 URI patterns 152, 153
 recent queries, monitoring 232
 recently crawled URLs, monitoring 223
 recrawl intervals for Web crawlers 74
 refreshing indexes 6, 125
 removing URIs from an index 137
 reorganizing indexes 6, 125
 response time history, monitoring 232
 restore scripts
 description 245
 running 246, 247
 restoring
 from a backup 246
 system files to new servers 247
 restoring enterprise search 245
 return codes, enterprise search 249
 revisiting URLs as soon as possible 74
 Robots Exclusion protocol 68
 robots.txt file 68
 rule-based categories
 creating 103
 description 99

rule-based categories (*continued*)
 selecting the categorization type 102
 rule-based taxonomy, migrating from WebSphere Portal 207, 209
 runtime-generic.properties file 146

S

sample search application
 accessing 173
 config.properties file 163, 172
 default deployment 172
 description 11, 162
 disabling security 185
 enabling security 174
 search functions 161, 162
 starting 173
 WebSphere global security 174
 scheduling
 crawlers 33
 index builds 126, 127
 scopes
 creating 135
 description 134
 searching 134
 URI formats 89
 URI patterns 134, 135
 scripts
 esbackup.bat 245, 247
 esbackup.sh 245, 247
 escrcm.sh 50
 escrcm.vbs 51
 escrdb2.sh 47
 escrdb2.vbs 48
 escrnote.sh 58
 escrnote.vbs 60
 escrivbr.sh 40
 escrivbr.vbs 41
 esrestore.bat 246, 247
 esrestore.sh 246, 247
 startccl 246
 Search and Browse portlet
 configuration 203
 Search and Index API 10, 161
 search applications
 application IDs 188
 associating with collections 162
 associating with external sources 178
 collection-level security 188
 custom 161
 description 11
 sample 161, 162
 search cache
 configuring 140
 description 140
 Search Center for WebSphere Portal
 adapter for enterprise search 199, 204
 description 199
 registration portlet 199, 205
 search options
 document content 129
 fielded search 129
 for search results 129
 free text search 129
 parametric search 129
 Search page, description 15

- Search portlet deployment 201
 - search response time
 - alerts about 238
 - monitoring 232
 - search results
 - boost class configuration 154, 156, 157
 - collapsing 136, 137
 - customizing summaries 145, 146
 - description 149
 - dynamic ranking 149
 - dynamic summarization 145
 - grouping 136, 137
 - ranking 153
 - static ranking 150
 - summaries 145, 146
 - text-based scoring 149
 - URI pattern configuration 152
 - wildcard character expansion 133
 - wildcard characters 131
 - search servers
 - associating boost word dictionaries 151, 152
 - associating stop word dictionaries 144
 - associating synonym dictionaries 142
 - boost word dictionaries 150
 - description 7, 139
 - monitoring 232
 - popular queries 232
 - recent queries 232
 - response time history 232
 - search cache 140
 - starting 213, 232
 - stop word dictionaries 143
 - stopping 215, 232
 - synonym dictionaries 140, 142
 - system status 232
 - searching
 - categories 98
 - collections 129
 - external sources 129
 - HTML documents 107, 108
 - quick links 147
 - XML documents 105, 111
 - security
 - access controls 184
 - administrative roles 183
 - anchor text analysis 187
 - authentication 184, 185
 - bypassing document-level access controls 197
 - collection-level 186, 197
 - crawler plug-ins 88
 - description 181
 - disabling for enterprise application 185
 - document-level 189, 190, 192, 197
 - duplicate document detection 186
 - enabling for a collection 28, 181
 - identity management 192
 - Lotus Domino documents 195
 - sample search application 174
 - search application IDs 188
 - user profiles 191
 - WebSphere global security 185
 - Windows domains 193
 - security tokens
 - crawler configuration 189
 - disabling for a collection 197
 - document-level security 189, 201
 - Portal Search Engine processing 201
 - Security view, description 15
 - semantic search 108, 111, 116
 - server mode, Content Edition repositories 39
 - session IDs, enterprise search 249
 - SIAPI (Search and Index API) 10, 161
 - site history reports
 - creating 225
 - description 223
 - SMTP server configuration 241
 - soft error pages, Web crawlers 75
 - Solaris operating environment
 - Content Edition crawler configuration 40
 - DB2 Content Manager crawler configuration 50
 - Domino Document Manager crawler configuration 58
 - Notes crawler configuration 58
 - QuickPlace crawler configuration 58
 - start URLs for Web crawlers 70, 74
 - startcl script 246
 - starting
 - crawler servers 221
 - Data Listener 233
 - enterprise search servers 213
 - index builds 230
 - migration wizard 207, 209
 - parsers 229
 - sample search application 173
 - search servers 232
 - static ranking
 - description 150
 - enabling for a collection 28
 - in migrated collections 209
 - Stellent sessions
 - associating document types 120
 - default document types 123
 - parsing document types 118
 - stellent.properties file 120
 - stellentypes.cfg file 120
 - stop word dictionaries
 - adding to the system 144
 - associating with a collection 144
 - description 143
 - stopping
 - crawler servers 221
 - enterprise search servers 213, 215
 - index builds 230, 231
 - parsers 229
 - search servers 232
 - summaries
 - customizing 145, 146
 - dynamic 145
 - synonym dictionaries
 - adding to the system 142
 - associating with a collection 142
 - description 140
 - synonyms.xml file 207
 - system backup 245
 - system resources
 - checking 218
 - system resources (*continued*)
 - estimating 217
 - system restore 245, 246, 247
 - system status
 - collections 219
 - crawlers 221
 - index builds 231
 - parsers 229
 - search servers 232
 - Web crawlers 223
 - System view, description 15
- ## T
- task summary, administration
 - console 15
 - taxonomies, migrating from WebSphere Portal 207, 209
 - Taxonomy Management Portlet 207
 - text analysis
 - common analysis structures 113, 114
 - mapping XML elements 111
 - text analysis engines 110, 111
 - text analysis engines
 - adding to the system 110
 - associating with collections 111
 - description 108
 - mapping analysis results 113, 114
 - mapping XML elements 111
 - text processing
 - annotators 108
 - common analysis structures 108
 - text analysis engines 108
 - text-based scoring 149
 - thread details, monitoring 223
 - threads
 - parser 114
 - Web crawler 223
 - titles.xml file 207
 - treenodes.xml file 207
 - Trusted Server configuration 195
- ## U
- UIMA
 - adding text analysis engines to the system 110
 - associating with collections 111
 - common analysis structures 113, 114
 - description 108
 - mapping analysis results to JDBC tables 114
 - mapping analysis results to the index 113
 - mapping XML elements 111
 - UNIX file system crawlers
 - configuration 67
 - URI formats 89
 - URI details
 - dropped documents 235
 - monitoring 220
 - URIs
 - category rules 99, 103
 - collapsed in search results 136, 137
 - formats in enterprise search 89
 - influencing static scores 152, 153

- URIs (*continued*)
 - quick links 147
 - removing from an index 137
 - scopes 134, 135
 - viewing details about 220
- URL path depth 70
- user agents 68
- user profiles
 - configuration 192
 - description 191

V

- validation of current credentials 190, 193, 195
- vbr_access_services.jar file 40, 41
- viewing
 - log files 244
 - URI details 220
- visiting URLs as soon as possible 74

W

- Web crawlers
 - active sites 223, 224
 - configuration 68
 - cookie configuration 80
 - cookie format 79
 - cookies 79
 - crawl rate 225
 - crawler history 223
 - crawling rules 70
 - creating reports about 225
 - followindex.rules file 82, 83
 - global crawl space 80
 - JavaScript support 69
 - limiting the crawl space 70
 - monitoring 223
 - no-follow directives 82, 83
 - no-index directives 82, 83
 - password-protected Web sites 76, 77
 - proxy servers 78
 - recently crawled URLs 223
 - recrawl intervals 74
 - site history 223
 - soft error pages 75
 - start URLs 70, 74
 - system status 223
 - thread details 223
 - URL status 223
 - user agents 68
 - visiting URLs as soon as possible 74
- WebSphere global security
 - disabling 185
 - search application properties 172
- WebSphere II Event Publisher Edition, DB2 crawler configuration 43
- WebSphere II OmniFind Edition 279
 - accessibility 279
 - administration console 8
 - APIs 10
 - changing the password on a single server 19
 - changing the password on multiple servers 21
 - commands 249

- WebSphere II OmniFind Edition (*continued*)
 - components 3
 - crawler servers 3
 - data flow diagram 11
 - identity management 192
 - index servers 6
 - integration with WebSphere Portal 199
 - overview 1
 - parsers 4
 - port number configuration 24
 - return codes 249
 - search applications 11
 - search servers 7
 - session IDs 249
- WebSphere MQ, crawler server configuration 47, 48
- WebSphere MQ, DB2 crawler configuration 46
- WebSphere Portal
 - category tree migration 207
 - collection migration 207
 - default migration settings 210
 - integration with enterprise search 199
 - model-based taxonomies 207
 - Search and Browse portlet 203
 - Search portlet deployment 201
 - Taxonomy Management Portlet 207
 - taxonomy migration 207
- WebSphere Portal crawlers
 - configuration 83
 - copying site URLs 85
 - deploying the ESPACServer.ear file 84
 - enterprise application deployment 84
 - URI formats 89
- WebSphere Portal Search Center
 - adapter for enterprise search 199, 204
 - description 199
 - registration portlet 199, 205
- wildcard characters
 - index expansion 131, 133
 - query expansion 131, 133
- Windows domains 193
- Windows file system crawlers
 - configuration 86
 - document-level security configuration 193
 - URI formats 89
- Windows operating system
 - Content Edition crawler configuration 41
 - crawler configuration 60
 - DB2 Content Manager crawler configuration 51
 - event publishing configuration 48
- WpsMigratorLog.log file 212

X

- XML documents
 - native XML search 116
 - searching 105

- XML elements
 - mapping to common analysis structures 111
 - mapping to search fields 105
 - searching 105, 111
- XML fragments, native XML search 116
- XML query syntax, native 116
- XML search fields
 - creating 105
 - description 105, 111
 - mapping elements to 105, 111
- XPath, native XML search 116



Printed in USA



Java[™]
COMPATIBLE

SC18-9283-02



Spine information:



WebSphere II OmniFind Edition

Administering Enterprise Search

Version 8.3