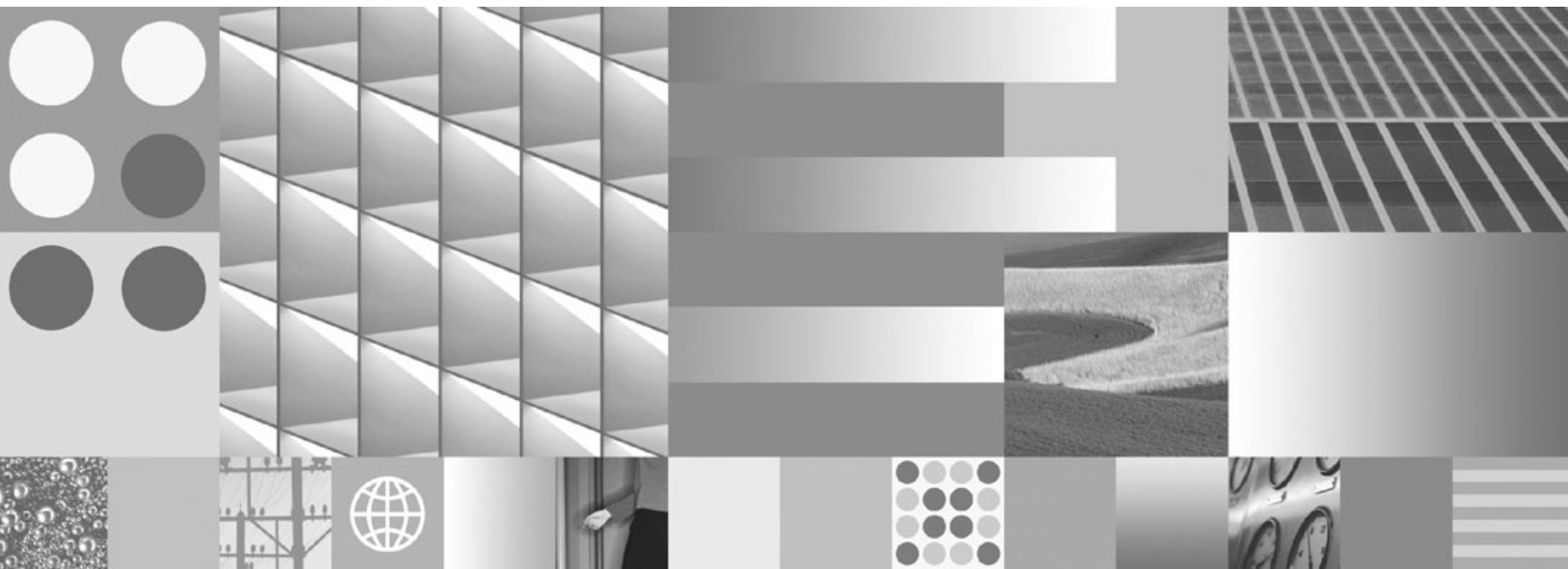


Administering Enterprise Search



Administering Enterprise Search

Note

Before using this information and the product it supports, read the information in "Notices and trademarks" on page 399.

Edition Notice

This edition applies to version 8, release 5, modification 0 of IBM OmniFind Enterprise Edition (product number 5724-C74) and to all subsequent releases and modifications until otherwise indicated in new editions.

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, 2008. All rights reserved.

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

Contents

ibm.com and related resources vii

How to send your comments vii

Contacting IBM. viii

What is enterprise search? 1

Data source types supported by enterprise search 2

Enterprise search component overview 3

Enterprise search crawlers 4

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 system administration 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 20

TCP port numbers used for enterprise search 22

Changing the port number for the enterprise search system. 23

Changing enterprise search server host names or IP addresses 25

Configuring support for dual IP addresses 26

Enabling support for the IPv6 protocol 27

Enterprise search collections 31

Creating a collection by using the Collection wizard 31

Creating a collection by using the Collections view 32

Editing a collection. 34

Deleting a collection 35

Determining the collection ID 36

Crawler administration 37

Creating a crawler 38

Editing crawler properties 39

Editing a crawl space 40

Deleting a crawler 40

Crawler schedules 41

Content Edition crawlers 41

Direct mode access to Content Edition repositories 43

Server mode access to WebSphere II Content Edition repositories. 44

DB2 crawlers 46

Configuring the crawler server on UNIX for DB2 crawlers 48

Configuring the crawler server on Windows for DB2 crawlers 49

Configuring WebSphere Information Integrator Event Publisher Edition for DB2 crawlers 50

Configuring WebSphere MQ for DB2 crawlers. 52

Crawling DB2 databases on a classic data source server 54

DB2 Content Manager crawlers. 54

Configuring the crawler server on UNIX for DB2 Content Manager crawlers 56

Configuring the crawler server on Windows for DB2 Content Manager crawlers. 57

Domino Document Manager crawlers 59

Exchange Server crawlers. 61

JDBC database crawlers 62

Relationship maps for JDBC databases 63

Crawling multiple structured JDBC database tables 65

NNTP crawlers 69

Notes crawlers 70

Tips for crawling Lotus Domino databases 72

Configuring the crawler server on UNIX to crawl Lotus Domino sources. 73

Configuring the crawler server on Windows to crawl Lotus Domino sources. 74

Configuring servers that use the DIIOP protocol 76

Configuring the I/O completion port on AIX to crawl Lotus Domino sources. 77

QuickPlace crawlers 78

Seed list crawlers 81

UNIX file system crawlers 83

Web crawlers. 84

User agent configuration 85

How the Web crawler uses the robots exclusion protocol 85

Support for JavaScript. 87

Rules to limit the Web crawl space 88

Testing URL connections with the Web crawler 92

Recrawl interval settings in the Web crawler 93

Options for visiting URLs with the Web crawler 93

How the Web crawler handles soft error pages 93

Support for crawling secure Web sites 95

Web sites that are served by proxy servers 97

Cookie administration. 98

Global Web crawl space configuration 99

No-follow and no-index directives 101

Overriding no-follow and no-index directives in Web pages 101

Configuring which date the Web crawler uses for crawled documents 102

Web Content Management crawlers 103

WebSphere Portal crawlers 105

Copying the URL to crawl from WebSphere Portal 106

Windows file system crawlers 107

Configuring support for Data Listener applications 109

Custom crawler plug-ins	110
Support for crawling archive files	112
URI formats in an enterprise search index	113

Parser administration 123

Working with categories	124
Rule-based categories	124
Category trees	126
Selecting the categorization type	127
Configuring categories	127
Working with XML search fields	129
XML search fields	129
Mapping XML elements to search fields	130
Working with HTML search fields	132
HTML search fields	132
Mapping HTML metadata elements to search fields	133
Custom text processing	134
Adding text analysis engines to the system	136
Associating a text analysis engine with a collection	137
Mapping XML elements to the common analysis structure	137
Mapping the common analysis structure to the index	139
Mapping the common analysis structure to a relational database	140
Configuring threads for the parser service	141
Enabling advanced analysis for compound terms	141
Enabling support for native XML search	142

Document format detection 145

Default supported document types	146
Document types associated with collection parsers and Stellent parsers	146
Associating document types with a collection parser	147
Default collection parser service rules	149
Parsing unknown document types	149
Changing the replacement rules for some HTML tags	150
Default HTML replacement rules	151
Associating document types with a Stellent parser	152
Default parsing rules for Stellent parsers	154

Language and code page support . . . 159

Automatic language detection	160
Automatic code page detection	161
Linguistic analysis of Chinese, Japanese, and Korean documents	162
N-gram segmentation	162
Removing white space from text	162

Index administration 165

Scheduling index builds	166
Changing the index schedule	167
Enabling and disabling the index schedules	167
Configuring concurrent index builds	168
Building indexes only when changes are detected	169
Stopping index builds	170

Options that influence the searchable view of the index	171
Indexed options for searching documents	171
Duplicate document detection	175
Wildcard characters in queries	176
Scopes	179
Configuring scopes	180
Collapsed URIs	181
Collapsing URIs in the search results	182
Removing URIs from the index	183

Search server administration 185

Search caches	185
Configuring a search cache	186
Custom synonym dictionaries	186
Adding synonym dictionaries to the system	188
Associating a synonym dictionary with a collection	188
Custom stop word dictionaries	189
Adding stop word dictionaries to the system	190
Associating a stop word dictionary with a collection	190
Redeploying custom dictionaries	191
Dynamic summarization	192
Customizing document summaries in the administration console	192
Customizing document summaries by editing properties	193
Working with quick links	194
Quick links	194
Configuring quick links	195

Document ranking 197

Text-based scoring	197
Static ranking	198
Restoring default values for static document ranking	199
Custom boost word dictionaries	200
Adding boost word dictionaries to the system	202
Associating a boost word dictionary with a collection	202
Document ranking that is based on URI patterns	203
Influencing the scores of documents that match URI patterns	203
Document ranking that is based on boost classes	204
Mapping fields to boost classes	206
Configuring boost factors for boost classes	207
Default boost class values	207

Search applications for enterprise search 211

Associating search applications with collections	212
Sample search application functions	212
Search application properties	213
Editing the sample search application properties	230
Customizing search applications	231
Cloning the sample search application	233
Analyzing top results	234
Accessing search applications	237

Configuring the search servers to accept only secure (SSL) requests	238
Configuring the search servers to accept requests through a proxy server	239
Support for external sources.	241
Adding external sources to the system	241
Associating search applications with external sources	243
Enterprise search security.	245
Installation security	246
Authentication versus access control.	247
Administrative roles	248
Configuring administrative users.	249
Collection-level security	249
Duplicate document analysis and collection security	249
Search application identifiers	250
Document-level security	251
Pre- and post-filtering of search results	251
Validation by stored security tokens	252
Validation of current credentials during query processing	253
Anchor text analysis	260
Enabling security for enterprise search	261
Configuring global security and an LDAP user registry in WebSphere Application Server	262
Enabling security for a single server enterprise search system	264
Enabling security for a multiple server enterprise search system.	265
Crawler setup requirements to support security	266
Verifying access to secure Exchange Server documents	269
Enforcement of document-level security for Lotus Domino documents	269
Enforcement of document-level security for Windows file system documents	272
Disabling security for enterprise search.	275
Disabling security for an enterprise application in WebSphere Application Server	275
Disabling document-level security	277
Disabling security for collapsed search results	278
Starting and stopping an enterprise search system	279
Starting an enterprise search system	279
Stopping an enterprise search system	281
Controlling which components are started or stopped	282
Administering the search servers in stand-alone mode	283
Monitoring enterprise search activity	285
Estimating the number of documents in a collection	285
Monitoring a collection	286
Viewing details about a URI	286
Monitoring crawlers	288

Viewing details about Web crawler activity	289
Web crawler thread details	290
Web crawler active sites	290
Web crawler crawl rate	291
Creating Web crawler reports	292
HTTP status codes returned to the Web crawler	293
Monitoring the parser	296
Monitoring index activity for a collection	297
Monitoring the enterprise search index queue	298
Monitoring the search servers	299
Changing how query statistics are calculated	300
Monitoring the Data Listener	301
Document tracking	302
Configuring log files for document tracking	302
Viewing reports about dropped documents	303
Viewing log files about dropped documents	304
Log files and alerts	305
Alerts	305
Configuring collection-level alerts	306
Configuring system-level alerts	307
Configuring log files	308
Configuring SMTP server information	309
Receiving e-mail about logged messages	310
Changing the size of the query log	312
Viewing log files	312
Backing up and restoring an enterprise search system	315
Backing up the enterprise search system	316
Restoring the enterprise search system	317
Exporting and importing collection configurations	318
Integration with Lotus Notes Version 8	323
Creating the enterprise search plug-in update site	323
Installing the enterprise search plug-in in the Lotus Notes version 8 client	323
Integration with WebSphere Portal	325
Setup scripts for integrating enterprise search with WebSphere Portal	326
Setting up enterprise search in WebSphere Portal version 5.1	327
Configuring the WebSphere Portal version 5.1 Search bar to use enterprise search	329
Removing enterprise search from WebSphere Portal version 5.1	331
Setting up enterprise search in WebSphere Portal version 6	332
Configuring the WebSphere Portal version 6 Search Center for enterprise search	334
Configuring the WebSphere Portal version 6 Search bar to use enterprise search	336
Setting up the enterprise search portlet for Lotus Quickr	337
Removing enterprise search from WebSphere Portal version 6	339
Enterprise search integration with WebSphere Portal clustered systems	340

Setting up enterprise search in a WebSphere Portal clustered system	340	Enterprise search documentation. . . .	383
Removing enterprise search from a WebSphere Portal clustered system	343	Accessibility features	385
Migration from WebSphere Portal to enterprise search.	347	Glossary of terms for enterprise search	387
Migrating a collection from WebSphere Portal	347	Notices and trademarks	399
Migrated collection settings.	348	Notices	399
Migration wizard log file	350	Trademarks	401
Enterprise search commands, return codes, and session IDs	351	Index	403
Case sensitivity in enterprise search	381		

ibm.com and related resources

Product support and documentation are available from [ibm.com](http://www.ibm.com)[®].

Support and assistance

Product support is available on the Web.

IBM[®] OmniFind[™] Enterprise Edition

<http://www.ibm.com/software/data/enterprise-search/omnifind-enterprise/support.html>

IBM OmniFind Discovery Edition

<http://www.ibm.com/software/data/enterprise-search/omnifind-discovery/support.html>

IBM OmniFind Yahoo! Edition

<http://www.ibm.com/software/data/enterprise-search/omnifind-yahoo/support.html>

Information center

You can view the product documentation in an Eclipse-based information center with a Web browser. See the information center at <http://publib.boulder.ibm.com/infocenter/discover/v8r5m0/>.

PDF publications

You can view the PDF files online using the Adobe[®] Acrobat Reader for your operating system. If you do not have the Acrobat Reader installed, you can download it from the Adobe Web site at <http://www.adobe.com>.

See the following PDF publications Web sites:

Product	Web site address
OmniFind Enterprise Edition, Version 8.5	http://www.ibm.com/support/docview.wss?rs=63&uid=swg27010938
OmniFind Discovery Edition, Version 8.4	http://www.ibm.com/support/docview.wss?rs=3035&uid=swg27008552
OmniFind Yahoo! Edition, Version 8.4	http://www.ibm.com/support/docview.wss?rs=3193&uid=swg27008932

How to send your comments

Your feedback is important in helping to provide the most accurate and highest quality information.

Send your comments by using the online reader comment form at https://www14.software.ibm.com/webapp/iwm/web/signup.do?lang=en_US&source=swg-rcf.

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

For more information about how to contact IBM, see the Contact IBM Web site at <http://www.ibm.com/contact/us/>.

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. Fast query response times and a consolidated, ranked result set that is based on extensive text analysis enable you to not just locate documents of interest, but extract meaning from document content.

The enterprise search components, which are installed with IBM OmniFind Enterprise Edition, collect information from throughout your enterprise. 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, OmniFind Enterprise 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 semantic 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 change after a document and its security tokens are indexed.

Product tutorial

An online tutorial is available at <http://www.ibm.com/developerworks/edu/dm-dw-dm-0503buehler-i.html>. The tutorial describes installation and configuration steps, shows you how to search different types of data sources, and describes how you can use the product application programming interfaces to extend enterprise search. The tutorial addresses an older version of OmniFind Enterprise Edition, but many of the concepts and procedures are still applicable.

Related concepts

Enterprise search security



Custom text analysis integration



Basic concepts used in text analysis processing

Data source types supported by enterprise search

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

After you install IBM OmniFind Enterprise 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 databases
- IBM Domino[®] Document Manager (formerly Domino.Doc[®]) databases
- IBM Lotus Notes[®] databases
- IBM Lotus[®] QuickPlace[®] databases
- IBM Lotus Quickr[™] content libraries
- IBM WebSphere Information Integrator Content Edition repositories, including Documentum, FileNet[®] Panagon Content Services, FileNet P8 Content Manager, Hummingbird[®] Document Management (DM), Microsoft[®] SharePoint, OpenText Livelink Enterprise Server, and WebSphere Portal Document Manager (PDM)
- IBM WebSphere Information Integrator nickname tables for many database system types, including IBM DB2 for z/OS[®], IBM Informix[®], Microsoft SQL Server, Oracle, and Sybase
- IBM WebSphere Portal sites
- IBM Workplace Web Content Management[™] sites
- Microsoft Exchange Server public folders
- Microsoft SQL Server databases
- Microsoft Windows[®] file systems
- Network news transfer protocol (NNTP) news groups
- Oracle databases
- UNIX[®] file systems
- Web sites on the Internet or on 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:

- Databases that support the Java™ database connectivity (JDBC) protocol (DB2 and Oracle database systems only). A separate external source is created for each table that you enable for searching.
- Lightweight Directory Access Protocol (LDAP) servers. One external source is created for each LDAP server.

For the latest information about supported data source types and the supported product versions, see the system requirements page on the OmniFind Enterprise Edition Support Web site.

Related concepts

Support for external sources

“The enterprise search data flow” on page 11

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 OmniFind Enterprise 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

Crawler administration

Parser administration

Index administration

Search server administration

“The enterprise search data flow” on page 11

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 OmniFind Enterprise 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.

After you start the Web crawler, it runs continuously. You specify which uniform resource locators (URLs) you want to crawl, and the crawler returns periodically to check for data that is new or 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 initially run and how often it must revisit 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

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 OmniFind Enterprise 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 format a document is in. For example, the parser extracts text from the tags in XML and HTML documents. By using Oracle (formerly Stellent) Outside In Content Access for IBM OmniFind Enterprise Edition, 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.
- 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.
- Applies parsing rules that you specify for the collection. When you configure the parser, you can configure the following parsing activities:

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 whether you want to use categorization. With *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.

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. For example, a user might need 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.

Related concepts

Working with categories

XML search fields

HTML search fields



Custom text analysis integration



Text analysis included in enterprise search

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:

Building the main index

During a main index build, 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.

Building delta indexes

During a delta index build, information that was crawled since the last time the main index was built is added to the index.

When you configure index options for a collection, you can specify schedules for building main and delta indexes. The frequency with which you build 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 delta index builds to occur frequently. Periodically schedule a main index build 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 a main index build after the data is recrawled and parsed instead of waiting for the index build 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 build of a very large main index does not block delta index builds for other collections. Index building can be a resource-intensive process, so for large systems, you must monitor system loads to adjust the main and delta index build schedules.

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

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 OmniFind Enterprise 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 OmniFind Enterprise 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 system administration

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 OmniFind Enterprise 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 crawling 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 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 an enterprise search system

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

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 OmniFind Enterprise 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.

Crawler plug-ins

Use plug-in APIs to add metadata to documents when the documents are 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 are included in the search results.
- If a query includes stop words, the stop words are 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 are 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](#)

 [Crawler plug-ins](#)

Sample search application for enterprise search

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

A sample search application is installed when you install OmniFind Enterprise 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.

For certain crawler types, you can use the identity management component for enterprise search to validate current credentials when users access the search application. If the domain to be searched is protected by single sign-on (SSO) security, SSO mechanisms can be used to validate the user throughout the search session. Otherwise, the identity management component can encrypt and store user credentials in a profile and use the credentials to exclude forbidden documents from the search results.

To customize the sample search application, you can use the Search Application Customizer, which is a graphical user interface that enables you to see the effects of your changes as you make them. You can also customize the search application by editing the configuration file for 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](#)

Related tasks

[Accessing search applications](#)

[Editing the sample search application properties](#)

[Customizing search applications](#)

The enterprise search data flow

The enterprise search components that you install with OmniFind Enterprise 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.

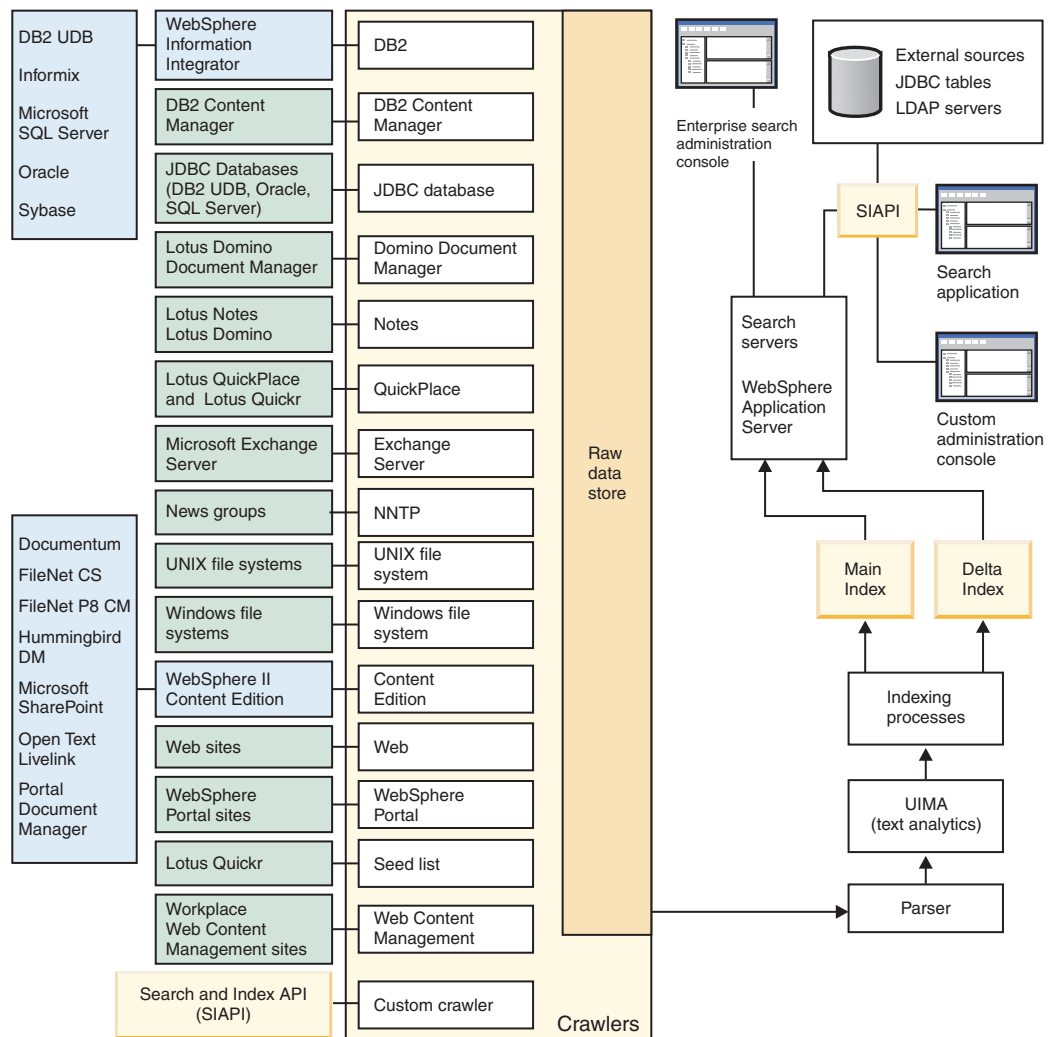


Figure 1. How data flows through an enterprise search system

Related concepts

“Enterprise search component overview” on page 3

Related reference

“Data source types supported by enterprise search” on page 2

Enterprise search system administration

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.


Tip: An online tutorial is available at <http://www.ibm.com/developerworks/edu/dm-dw-dm-0503buehler-i.html>. The tutorial describes installation and configuration steps, shows you how to search different types of data sources, and describes how you can use the product application programming interfaces to extend enterprise search. The tutorial addresses an older version of OmniFind Enterprise Edition, but many of the concepts and procedures are still applicable.

For detailed examples of how to configure crawlers and enable security in small, medium, and large organizations, see the IBM Redbook, IBM OmniFind Enterprise Edition Configuration and Implementation Scenarios.

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: To move to the previously displayed page or to refresh information in the administration console, click **Previous** and **Refresh** in the enterprise search administration console. If you click **Back** or **Refresh** in the Web browser, inconsistent results and a potential loss of data can occur.

- 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 and support semantic search.
 - 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 building 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 documents from the same source are collapsed 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 the data can be parsed and indexed, and the main index must be built before the search servers can 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.

- 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 main index builds for a single collection can run concurrently with delta index builds.
- 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 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.

You also use the Security view to configure identity management options. For example, you can specify options for storing user credentials in profiles that can be used to validate the user's current credentials during query processing. If the source to be searched is protected by single sign-on (SSO) security, you can also specify options for using SSO authentication methods to validate the user's current credentials during query processing.

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

"Crawler administration" on page 37

"Monitoring enterprise search activity" on page 285

Related tasks

"Starting an enterprise search system" on page 279

"Administering the search servers in stand-alone mode" on page 283

"Stopping an enterprise search system" on page 281

"Creating a collection by using the Collection wizard" on page 31

"Creating a collection by using the Collections view" on page 32

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 OmniFind Enterprise 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 **Log Out** 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 248

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.

Restrictions

Passwords can include the following special characters:

```
! @ # $ % ^ & * ( ) - _ = + , . / < > ?
```

On AIX®, Linux®, and Solaris systems, if you specify a password that includes special characters, you must enclose the entire password in single quotation marks.

For example: 'mypwd@\$%'

On a Windows system, if you specify a password that includes special characters, you must enclose the entire password in double quotation marks.

For example: "my?+!pwd"

About this task

The password for the initial enterprise search administrator ID is specified when OmniFind Enterprise 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.

Procedure

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

1. Log in as the enterprise search administrator and stop the enterprise search system:

```
esadmin system stopall
```

Important: When the system is stopped, users cannot submit search requests.

2. Change the system password for the enterprise search administrator user ID by using operating system commands (on AIX, Linux, or Solaris) or by using the change password facility (on Windows).
3. Run the following script, where *newValue* is the password that you specified in step 2:

AIX, Linux, or Solaris

```
eschangepw.sh newValue
```

Windows

```
eschangepw newValue
```

4. Restart the enterprise search system:

```
esadmin system startall
```

Related reference

“Enterprise search commands, return codes, and session IDs” on page 351

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.

Restrictions

Passwords can include the following special characters:

```
! @ # $ % ^ & * ( ) - _ = + , . / < > ?
```

On AIX, Linux, and Solaris systems, if you specify a password that includes special characters, you must enclose the entire password in single quotation marks.

For example: 'mypwd@\$%'

On a Windows system, if you specify a password that includes special characters, you must enclose the entire password in double quotation marks.

For example: "my?+!pwd"

About this task

The password for the enterprise search administrator ID, which is specified initially when OmniFind Enterprise Edition is installed, must be the same on all enterprise search servers.

To change the password and to disseminate the change throughout the enterprise search system, you must run the **eschangepw** script on each computer that you use for enterprise search. The procedure below suggests an order for changing the password on all servers. You are not required to follow this order, but you do need to complete the steps required for each server type.

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.

Procedure

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

1. On the enterprise search index server, log in as the enterprise search administrator and stop the enterprise search system:

```
esadmin system stopall
```

Important: When the system is stopped, users cannot submit search requests.

- a. Change the system password for the enterprise search administrator user ID by using operating system commands (on AIX, Linux, or Solaris) or by using the change password facility (on Windows).
- b. Run the following script, where *newValue* is the password that you specified in step 1a:

AIX, Linux, or Solaris

```
eschangepw.sh newValue
```

Windows

```
eschangepw newValue
```

2. Do the following steps on the other computers in your enterprise search system:

- a. Log in as the enterprise search administrator.
- b. Stop the common communication layer (CCL) for enterprise search:

AIX, Linux, or Solaris

```
stopccl.sh
```

Windows command prompt

```
stopccl
```

Windows Services administrative tool

- 1) Launch Windows Services: **Start** → **Programs** → **Administrative Tools** → **Services**.
 - 2) Right-click **IBM OmniFind Enterprise Edition** and click **Stop**.
- c. Change the system password for the enterprise search administrator user ID by using operating system commands (on AIX, Linux, or Solaris) or by using the change password facility (on Windows). This password must match the password that you specified in step 1a on page 21.
- d. Run the following script, where *newValue* is the password that you specified in step 1a on page 21:

AIX, Linux, or Solaris

```
eschangepw.sh newValue
```

Windows

```
eschangepw newValue
```

- e. Restart the CCL:

AIX, Linux, or Solaris

```
startccl.sh -bg
```

Windows command prompt

```
startccl
```


Windows Services administrative tool

To start the CCL in the background:

- 1) Launch Windows Services: **Start** → **Programs** → **Administrative Tools** → **Services**.
 - 2) Right-click **IBM OmniFind Enterprise Edition** and click **Properties**.
 - 3) Click the **Log On** tab.
 - 4) Change the password by specifying the new password value, and then and click **OK**.
 - 5) Right-click **IBM OmniFind Enterprise Edition** and click **Start**.
3. On the enterprise search index server, log in as the enterprise search administrator and restart the enterprise search system:

```
esadmin system startall
```

Related reference

 Setting the encrypted administrator password to be the same on all servers
“Enterprise search commands, return codes, and session IDs” on page 351

TCP port numbers used for enterprise search

Review the default port numbers that are used in an enterprise search system so that you can avoid port conflicts when you configure resources or assign port numbers to other applications.

If you configure a firewall, you must explicitly enable access to particular port numbers. You must also ensure that all enterprise search servers are inside the firewall.

Table 1. Port number configurations for enterprise search

Port name	Port numbers	Where configured
Common communication layer (CCL)	6002	ES_NODE_ROOT/nodeinfo/es.cfg and ES_NODE_ROOT/master_config/nodes.ini on all enterprise search servers
HTTP on the search servers	80	HTTP_SERVER_ROOT/conf/http.conf on the search servers
WebSphere Application Server Version 6 administration console	9060	On the search servers
WebSphere Application Server Version 5.1 administration console	9090	On the search servers
DB2 crawler	6000, 6001, 6002, 60003, 50000	On the crawler server
Information Center	8889	On the search servers
Anonymous or dynamic ports for CCL, file transfers (ESFTP), and index copy	49152 to 65535	On all enterprise search servers
Apache Derby Network Server	1527	On the crawler server
Custom communication	8890	On the crawler server
Remote client connections to the DB2 server (used only with releases that precede OmniFind Enterprise Edition Version 8.4)	50000	On the crawler server
WebSphere Information Integrator Content Edition FastObjects database	6001 (6002 as an alternate)	In the WebSphere Information Integrator Content Edition administration console
WebSphere Information Integrator Content Edition remote method invocation (RMI) proxy connector	1251 (RMI port)	In the WebSphere Information Integrator Content Edition administration console

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 OmniFind Enterprise Edition is installed. In a multiple 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 An internal warning occurred - Exception Message: {0}
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. Go to the computer where the port number needs to be changed, log in as the enterprise search administrator, and stop the enterprise search system:

```
esadmin system stopall
```

Important: When the system is stopped, users cannot submit search requests.

2. Edit the `ES_NODE_ROOT/nodeinfo/es.cfg` file, locate the following property, specify a new port number value, and then save and close the file:

```
CCLPort=new_port_number
```

3. Restart the common communication layer (CCL) for enterprise search:

AIX, Linux, or Solaris

```
startccl.sh
```

Windows command prompt

```
startccl
```

Windows Services administrative tool

To start CCL in the background:

- a. Launch Windows Services: **Start** → **Programs** → **Administrative Tools** → **Services**.
 - b. Right-click **IBM OmniFind Enterprise Edition** and click **Start**.
4. Go to the index server, log in as the enterprise search administrator, and stop the CCL:

AIX, Linux, or Solaris

```
stopccl.sh
```

Windows command prompt

```
stopccl
```

Windows Services administrative tool

- a. Launch Windows Services: **Start** → **Programs** → **Administrative Tools** → **Services**.
 - b. Right-click **IBM OmniFind Enterprise Edition** and click **Stop**.
5. Edit the `ES_NODE_ROOT/nodeinfo/es.cfg` file:

- a. Locate the following property, where *computer_name* is the name of the computer where you modified the port number in step 2. The *N* in the `nodeN` property is a number that identifies the enterprise search server.

```
nodeN.destination=computer_name
```

- b. Locate the following subproperty, specify the same port number here that you specified for the server in step 2, and then save and close the file:

```
nodeN.port=new_port_number
```

6. Restart the enterprise search system:

```
esadmin system startall
```

Related reference

Changing enterprise search server host names or IP addresses

You can change the host names and IP addresses that the enterprise search servers are configured to use.

For example, you might want to change the IP address if you have several network interface cards (NIC) on each enterprise search server, and you discover that the index server is configured to use a slow network. You can change settings in the configuration files to allow the index server to use a faster network.

Tip: If you do not want to edit configuration files, you can re-install OmniFind Enterprise Edition and specify the new host names or IP addresses when you run the installation program.

Procedure:

To change host names or IP addresses:

1. Log in as the enterprise search administrator. If you have a multiple server configuration, log in on any enterprise search server.
2. Stop the enterprise search sessions:
`esadmin system stopall`
3. Stop the common communication layer (CCL). If you have a multiple server configuration, use one of the following methods to stop the CCL on each enterprise search server:

AIX, Linux, or Solaris

```
stopccl.sh
```

Windows command prompt

```
stopccl
```

Windows Services administrative tool

- a. Start Windows Services: **Start** → **Programs** → **Administrative Tools** → **Services**.
 - b. Right-click **IBM OmniFind Enterprise Edition** and click **Stop**.
4. Do the following steps on each enterprise search server:
 - a. Edit the `ES_INSTALL_ROOT/configurations/ccl.properties` file and specify the host name or IP address that you want to use for this server in the `es_server_hostName` parameter.
 - b. Edit the `ES_NODE_ROOT/nodeinfo/es.cfg` file and specify the host name or IP address that you want to use for this server in the `LocalHostName` parameter.
 - c. Edit the `ES_NODE_ROOT/master_config/nodes.ini` file and replace all occurrences of the `nodeN.destination` parameter with the host name or IP address that you want to use for this server.
 - d. Edit the `ES_NODE_ROOT/config/nodes.ini` file and replace all occurrences of the `nodeN.destination` parameter with the host name or IP address that you want to use for this server.
 5. Restart the CCL. If you have a multiple server configuration, use one of the following methods to restart the CCL on each enterprise search server:

AIX, Linux, or Solaris

```
startccl.sh -bg
```

Windows command prompt

```
startccl
```

Windows Services administrative tool

To start the CCL in the background:

- a. Start Windows Services: **Start** → **Programs** → **Administrative Tools** → **Services**.
 - b. Right-click **IBM OmniFind Enterprise Edition** and click **Start**.
6. Restart the enterprise search sessions:
esadmin system startall

Configuring support for dual IP addresses

If the servers where you install enterprise search are configured to support dual IP addresses, you must manually configure the enterprise search servers to run in that environment.

For example, a desktop administrator might install a Microsoft Loopback Adapter to create a virtual network that supports the networking requirements of certain products, such as Microsoft SQL Server.

Procedure:

To configure an enterprise search system so that it can run on servers that support dual IP addresses:

1. Log in as the enterprise search administrator. If you have a multiple server configuration, log in on any enterprise search server.
2. Stop the enterprise search sessions:
esadmin system stopall
3. Stop the common communication layer (CCL). If you have a multiple server configuration, use one of the following methods to stop the CCL on each enterprise search server:

AIX, Linux, or Solaris

```
stopccl.sh
```

Windows command prompt

```
stopccl
```

Windows Services administrative tool

- a. Start Windows Services: **Start** → **Programs** → **Administrative Tools** → **Services**.
 - b. Right-click **IBM OmniFind Enterprise Edition** and click **Stop**.
4. On each enterprise search server that supports dual IP addresses, edit the ES_NODE_ROOT/nodeinfo/es.cfg file and add a parameter called **LocalIPAddress**. For the value, assign an IP address that can be resolved by DNS.
5. Restart the CCL. If you have a multiple server configuration, use one of the following methods to restart the CCL on each enterprise search server:

AIX, Linux, or Solaris

```
startccl.sh -bg
```

Windows command prompt

```
startccl
```

Windows Services administrative tool

To start the CCL in the background:

- a. Start Windows Services: **Start** → **Programs** → **Administrative Tools** → **Services**.
 - b. Right-click **IBM OmniFind Enterprise Edition** and click **Start**.
6. Restart the enterprise search sessions:
esadmin system startall

Enabling support for the IPv6 protocol

To enable support for addresses and URLs that adhere to the IP version 6 (IPv6) protocol, you must define an environment variable that instructs the enterprise search system to use only IPv6 socket addresses.

Before you begin

1. Verify that the values for the **LocalHostName** property and **LocalIPAddress** property (if given) in %ES_CFG% are either a host name or a valid IPv6 address for the local machine. You can change these values manually, if necessary, and then save %ES_CFG%.
2. Verify that the value for the **DerbyServerHostName** property in %ES_CFG% is a host name. This value cannot be an IPv4 address or an IPv6 address. If you change this property, save the %ES_CFG% file.
3. Verify that the values for the **destination** and **serverhost** properties in the ES_NODE_ROOT/master_config/nodes.ini file on the index server are either a valid IPv6 address or a valid host name. These values cannot be an IPv4 address. If you change this file, save the changes.

Restrictions

Support for the IPv6 protocol is available only in enterprise search systems that you install on Windows 2003 Servers. After you enable support for IPv6 addresses, enterprise search will no longer use IPv4 addresses for any socket communications.

Supported browsers

The Internet Explorer and Mozilla Firefox browsers handle IPv6 addresses differently.

Mozilla Firefox

To run the enterprise search administration console or search application, you can specify the IPv6 address or host name in the URL. For example:

```
http://[2001::db8]/ESAdmin  
http://SearchServer.com/ESSearchApplication/
```

Internet Explorer

To run the enterprise search administration console or search application, you cannot specify the IPv6 address in the URL. You must use the following format and ensure that the host name is mapped to the IPv6 address in DNS or the c:\windows\system32\etc\hosts file:

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

If you specify an IPv6 address in the URL, the message **Invalid syntax error** is displayed. For additional information about this restriction, see <http://support.microsoft.com/kb/325414>.

Support for IPv4 data sources

Data sources that run on an IPv4 server are supported by the enterprise search crawlers. When you configure the Web crawler and specify the start URLs in IPv6 address format, ensure that the URLs are enclosed in brackets. For example:

```
http://[2001:db8:0:1:0:0:0:1]
http://[2001:db8:0:1::1]
```

Procedure

To enable support for the IPv6 protocol:

1. On the enterprise search index server, log in as the enterprise search administrator and stop the enterprise search system:

```
esadmin system stopall
```

Important: When the system is stopped, users cannot submit search requests.

2. Stop the common communication layer (CCL) server on all enterprise search servers:
 - a. Launch Windows Services: **Start** → **Programs** → **Administrative Tools** → **Services**.
 - b. Right-click **IBM OmniFind Enterprise Edition** and click **Stop**.
3. Define **ES_IPV6=1** in the system environment variables. For a multiple server installation of enterprise search, do this step on the search servers.
4. Update the Java Virtual machine (JVM) custom properties in WebSphere Application Server. For a multiple server installation of enterprise search, do these steps on the search servers.
 - a. If it is running, stop the ESSearchServer application server in WebSphere Application Server.
 - b. Open the WebSphere Application Server administrative console and navigate to the Java virtual machine custom properties panel. Select **Servers** → **Application Servers** → **ESSearchServer** → **Java and Process Management** → **Process Definition** → **Java Virtual Machine** → **Custom Properties**.
 - c. Configure the following properties:

```
java.net.preferIPv4Stack=false
java.net.preferIPv6Addresses=true
```

If the custom property is not already listed, create a new property, and enter the property name in the **Name** field and a valid value in the **Value** field.

- d. If you are running WebSphere Application Server Version 6.1, add the **LocalIPAddress** property to the ES_NODE_ROOT/nodeinfo/es.cfg file to contain the IPv6 address for that search server.
 - e. Restart the ESSearchServer application.
5. Do these steps if you are running WebSphere Application Server Version 6.1. For a multiple server installation of enterprise search, do these steps on the search servers.
 - a. Edit the httpd.conf file for the IBM HTTP Server.
 - b. Remove the # character from the start of the following line to uncomment the instruction, which allows the IBM HTTP Server to listen for IPv6 connections on port 80:

```
# Listen [::]:80
```
 - c. Restart the IBM HTTP Server.
 6. Restart the CCL on all enterprise search servers:

- a. Launch Windows Services: **Start** → **Programs** → **Administrative Tools** → **Services**.
 - b. Right-click **IBM OmniFind Enterprise Edition** and click **Start**.
7. Restart the enterprise search system:
- ```
esadmin system startall
```
8. Check the ES\_NODE\_ROOT/node/logs/cc10.log file and verify that the CCL server started without errors. If IPv6 is correctly enabled, messages similar to the following are logged:
- ```
INFO: CCL server ready for business. Now, waiting for external requests.
      CCL host name is fe80::250:56ff:feb4:27d1
      CCL host dot.ip address is fe80:0:0:0:250:56ff:feb4:27d1
      CCL server port is 6002
      Total JVM Runtime memory is 33022Kb
      Current free memory is 21346Kb
      java.net.preferIPv6Addresses=true
      java.net.preferIPv4Stack=false
INFO: Session "TraceDaemonSession" was attached PID=4992
```
9. Do the following steps to verify that the search and administration applications are operating correctly:
- a. Start the search servers for any collection, and then open the sample search application and submit a query. This step verifies that the query submission processing is working, even if there are no documents in the index to search.
 - b. Open the administration console and verify that you are able to log in without problems.

Disabling support for the IPv6 protocol

1. See the **Before you begin** section and, for every instance of an IPv6 address, replace the value with a host name or IPv4 address.
2. Follow the procedure for enabling support for the IPv6 protocol, with these exceptions:
 - In step 3 on page 28, remove the **ES_IPV6** environment variable.
 - In step 4 on page 28, remove the custom JVM properties that were set for the ESSearchServer application in WebSphere Application Server.
3. Remove the **LocalIPAddress** property from the ES_NODE_ROOT/nodeinfo/es.cfg file if it contains an IPv6 address.
4. Follow the procedure in step 9 to verify the search and administration applications.

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 286

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.

Restrictions

You can use the Collection wizard to create the following crawler types:

- Content Edition
- DB2
- DB2 Content Manager
- Exchange Server
- Notes
- UNIX file system
- Web
- Windows file system

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.

For more information about how to configure support for full n-gram parsing and tokenization in enterprise search collections, see <http://www.ibm.com/support/docview.wss?rs=63&uid=swg27011088>.

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 use your custom search applications.

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, building 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:

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 how documents are ranked in the search results.

Index Specify schedules for building the entire index and updating 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.

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.

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**.

Determining the collection ID

For many administrative tasks, you need to know the collection ID.

Before you begin

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

About this task

When you create a collection, you can either specify a value for the ID or allow the system to assign an ID for you. To determine the collection ID after a collection is created, you can use the administration console or view a configuration file.

Procedure

1. To determine the collection ID by using the administration console:
 - a. Click **Collections** to open the Collections view.
 - b. On the General page, click **View collection settings**.

The Collection Settings page shows the collection ID, the fully qualified path where collection data is stored, and the static ranking model that is used with the documents in this collection.
2. To determine the collection ID by viewing a configuration file:
 - a. Open the `ES_NODE_ROOT/master_config/collections.ini` file. For easier viewing, sort this file. In the following sample output, `coll` is the collection ID:

```
% sort $ES_NODE_ROOT/master_config/collections.ini | more
collection1.configfile=coll_config.ini
collection1.datadir=/home/eseach/node/data/coll
collection1.description=
collection1.displayname=Collection1
collection1.flags=0
collection1.id=coll
collection1.sectiontype=collection
collection1.type=1
...
```

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.

Tip: An online tutorial is available at <http://www.ibm.com/developerworks/edu/dm-dw-dm-0503buehler-i.html>. The tutorial describes installation and configuration steps, shows you how to search different types of data sources, and describes how you can use the product application programming interfaces to extend enterprise search. The tutorial addresses an older version of OmniFind Enterprise Edition, but many of the concepts and procedures are still applicable.

For detailed examples of how to configure crawlers and enable security in small, medium, and large organizations, see the IBM Redbook, IBM OmniFind Enterprise Edition Configuration and Implementation Scenarios.

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.

Configuring 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.

Related concepts

"Document-level security" on page 251

Related tasks

"Monitoring crawlers" on page 288

Related reference

"Crawler setup requirements to support security" on page 266

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 the next main index build occurs.

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**.

Crawler schedules

Crawlers that you create for Web sources run continuously. After you start a Web crawler, you typically do not need to stop it 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.

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 the following repository types:

- Documentum
- FileNet P8 Content Manager
- FileNet Panagon Content Services

- Hummingbird Document Management (DM)
- Microsoft SharePoint
- OpenText Livelink
- Portal Document Manager (PDM)

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.

Tip:

For detailed examples of how to configure connectors and a secure Content Edition crawler, see the scenario for a medium organization in the IBM Redbook, IBM OmniFind Enterprise Edition Configuration and Implementation Scenarios.

Crawler server configuration

How you prepare for crawling repositories depends on whether you plan to use direct mode or server mode to connect to the data to be crawled. If you use direct mode, you must configure a connector in WebSphere Information Integrator Content Edition. If you use server mode, you must run a script on the crawler server. This script, which is provided with OmniFind Enterprise Edition, enables the Content Edition crawler to communicate with WebSphere Information Integrator Content Edition servers.

If you use server mode, complete the task that is appropriate for your environment before you create a Content Edition crawler:

- “Configuring the crawler server on UNIX for WebSphere II Content Edition” on page 44.
- “Configuring the crawler server on Windows for WebSphere II Content Edition” on page 45.

For detailed instructions about how to configure your enterprise search system to search WebSphere Information Integrator Content Edition repositories, see the IBM developerWorks article, *Search WebSphere Portal Document Manager using WebSphere Information Integrator OmniFind Edition*, at URL <http://www-128.ibm.com/developerworks/db2/library/techarticle/dm-0606lee/>.

Configuration overview

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, Hummingbird DM, Portal Document Manager, and SharePoint 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.

Direct mode access to Content Edition repositories

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

About this task

In direct mode, the crawler uses a WebSphere Information Integrator Content Edition connector that is installed on the crawler server when OmniFind Enterprise Edition is installed. The crawler uses content integration APIs to connect directly to the repositories to be crawled. Not all content integration server functionality is available when the content integration server runs in direct mode. See the WebSphere Information Integrator Content Edition documentation for information about running the content integration server in direct mode and how the functionality differs from a content integration server that runs in server mode.

This procedure summarizes the steps required to set up direct mode access. For detailed instructions, see the IBM developerWorks article, *Search WebSphere Portal Document Manager using WebSphere Information Integrator OmniFind Edition*, at URL <http://www-128.ibm.com/developerworks/db2/library/techarticle/dm-0606lee/>.

Procedure

To configure the system so that the crawler can access repositories 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 **-Dvbr.as.operationMode=direct** Java system property to the *iice_install_root/bin/Admin.sh* file (on UNIX) or *iice_install_root\bin\Admin.bat* file (on Windows). For example:

Admin.sh file

```
java -classpath \  
"$VBR_CLASSPATH" \  
-Dvbr.home="$VBR_HOME" \  
-Dvbr.as.operationMode=direct \  
-Dlog4j.category.com.venetica.vbr.tools.admin=WARN \  
com.venetica.vbr.tools.admin.AdminFrame $1 $2 $3 $4
```

Admin.bat file

```
java -classpath "%VBR_CLASSPATH%" ^  
-Dvbr.home="%VBR_HOME%" ^  
-Dvbr.as.operationMode=direct ^  
-Dlog4j.category.com.venetica.vbr.tools.admin=WARN ^  
com.venetica.vbr.tools.admin.AdminFrame %*
```

3. Start the WebSphere Information Integrator Content Edition administration console in direct mode and configure the connector for the OmniFind Enterprise Edition crawler server.
4. Select the direct mode option when you use the enterprise search administration console to configure the Content Edition crawler.

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 OmniFind Enterprise Edition, enables the Content Edition crawler to access repositories on the server.

Complete the task that is appropriate for your environment:

- “Configuring the crawler server on UNIX for WebSphere II Content Edition.”
- “Configuring the crawler server on Windows for WebSphere II Content Edition” on page 45.

Configuring the crawler server on UNIX for WebSphere II Content Edition

If you install OmniFind Enterprise 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 OmniFind Enterprise Edition provides on the crawler server after you install WebSphere Application Server.

WebSphere Information Integrator Content Edition is installed on the crawler server when OmniFind Enterprise 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 OmniFind Enterprise Edition is installed in a multiple server configuration, install and bind the WebSphere Application Server Java libraries.
2. On the crawler server, run the setup script for the Content Edition crawler:
 - 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. On the crawler server, stop and restart the enterprise search system:

```
esadmin system stopall  
esadmin system startall
```

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 OmniFind Enterprise 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 OmniFind Enterprise Edition provides on the crawler server after you install WebSphere Application Server.

WebSphere Information Integrator Content Edition is installed on the crawler server when OmniFind Enterprise 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 OmniFind Enterprise Edition is installed in a multiple server configuration, install and bind the WebSphere Application Server Java libraries.
2. On the crawler server, run the setup script for the Content Edition crawler:
 - a. Log in with the enterprise search administrator ID (this user ID was specified when OmniFind Enterprise 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. On the crawler server, stop and restart the enterprise search system:

```
esadmin system stopall  
esadmin system startall
```

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 databases in a collection.

If you use IBM WebSphere Information Integrator to federate and create nickname tables for the following database system types, you can use the DB2 crawler to crawl the tables through the nicknames:

- CA-Datacom
- IBM DB2 for z/OS
- DB2 for iSeries™

- IBM Informix
- IMS™
- Oracle
- Microsoft SQL Server
- Software AG Adabas
- Sybase
- VSAM

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 must be database tables, nickname tables, or views. The DB2 crawler does not support joined tables.

Tip:

For detailed examples of how to configure a secure DB2 crawler, see the scenario for a large organization in the IBM Redbook, *IBM OmniFind Enterprise Edition Configuration and Implementation Scenarios*.

Crawler server configuration

Before you can crawl database tables, you must install the DB2 Administration Client on the crawler server. You must then run a script on the crawler server. This script, which is provided with OmniFind Enterprise Edition, enables the DB2 crawler to communicate with database servers.

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

- “Configuring the crawler server on UNIX for DB2 crawlers” on page 48.
- “Configuring the crawler server on Windows for DB2 crawlers” on page 49.

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 52.
- “Configuring WebSphere Information Integrator Event Publisher Edition for DB2 crawlers” on page 50.

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.
- Specify the databases 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 the crawler server on UNIX for DB2 crawlers

If you install OmniFind Enterprise 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 crawler to communicate with database

servers. If you use event publishing, the script also enables the crawler to access WebSphere MQ queue managers and queues.

About this task

To ensure that the DB2 crawler can crawl database tables, you run a setup script, `escrdb2.sh`, that OmniFind Enterprise Edition provides on the crawler server.

Before you run the script, you must ensure that the DB2 Administration Client is installed on the crawler server.

If you use event publishing, you must install the WebSphere MQ 5.3 modules for Java Messaging on the crawler server so that the DB2 crawler can access WebSphere MQ queue managers and queues. You must run the `escrdb2.sh` setup script after you install the WebSphere MQ modules.

Procedure

To configure the crawler server to support crawling by DB2 crawlers:

1. Optional: If you plan to use event publishing, 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. On the crawler server, run the setup script for the DB2 crawler:
 - a. Log in as the enterprise search administrator (this user ID was specified when OmniFind Enterprise 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. On the crawler server, stop and restart the enterprise search system:

```
esadmin system stopall  
esadmin system startall
```

Configuring the crawler server on Windows for DB2 crawlers

If you install OmniFind Enterprise Edition on a Microsoft Windows computer, you must run a script to configure the crawler server. The script enables the DB2 crawler to communicate with database servers. If you use event publishing, the script also enables the crawler to access WebSphere MQ queue managers and queues.

About this task

To ensure that the DB2 crawler can crawl database tables, you run a setup script, `escrdb2.vbs`, that OmniFind Enterprise Edition provides on the crawler server.

Before you run the script, you must ensure that the DB2 Administration Client is installed on the crawler server.

If you use event publishing, you must install the WebSphere MQ 5.3 modules for Java Messaging on the crawler server so that the DB2 crawler can access WebSphere MQ queue managers and queues. You must run the `escrdb2.vbs` setup script after you install the WebSphere MQ modules.

Procedure

To configure the crawler server to support crawling by DB2 crawlers:

1. Optional: If you plan to use event publishing, 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. On the crawler server, run the setup script for the DB2 crawler:
 - a. Log in with the enterprise search administrator ID (this user ID was specified when OmniFind Enterprise 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. On the crawler server, stop and restart the enterprise search system:

```
esadmin system stopall
esadmin system startall
```

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:
 - AIX, Linux, or Solaris**
db2rc
 - Windows command prompt**
Click **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, click the option 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 and complete 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:
`asncap 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, 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 users, WebSphere MQ users, and OmniFind Enterprise 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.

DB2 crawlers that use event publishing connect to WebSphere MQ queues with a client connection. To enable client connections, log in as a WebSphere MQ administrator and run the following command to set the queue manager CCSID to 819:

```
runmqsc queue_manager_name  
ALTER QMGR CCSID(819)  
END
```

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 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
```

Crawling DB2 databases on a classic data source server

The DB2 crawler can crawl a DB2 database on the classic data source server through WebSphere Information Integrator Classic Federation.

About this task

To crawl a DB2 database on the classic data source server, the database must be federated with a DB2 database on the non-classic data source server by using the ODBC wrapper of WebSphere Information Integrator Classic Federation.

Procedure

To federate the database:

1. Install WebSphere Information Integrator Classic Federation on the classic data source server.
2. Install the WebSphere Information Integrator Classic Federation client module on the non-classic data source server that has the database that will federate with the database on the classic data source server.
3. Configure the ODBC driver of WebSphere Information Integrator Classic Federation to connect to the classic data source server.
4. Connect to the federating database and create the ODBC wrapper to federate with the database on the classic data source server.
5. Configure the DB2 crawler to crawl the federating database on the non-classic data source server. This enables the database on the classic data source server to be crawled through WebSphere Information Integrator Classic Federation.

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 OmniFind Enterprise 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 crawlers” on page 56.
- “Configuring the crawler server on Windows for DB2 Content Manager crawlers” on page 57.

Tip:

For detailed examples of how to configure a secure DB2 Content Manager crawler, see the scenario for a large organization in the IBM Redbook, IBM OmniFind Enterprise Edition Configuration and Implementation Scenarios.

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.

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 crawler server on UNIX for DB2 Content Manager crawlers

If you install OmniFind Enterprise 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 one of the following products on the crawler server:

- IBM DB2 Information Integrator for Content, Version 8.3 for AIX, Solaris, or Linux
- IBM DB2 Information Integrator for Content, Version 8.2 for AIX or Solaris
- IBM DB2 Content Manager Toolkit, Version 8.2 for Linux

To ensure that the DB2 Content Manager crawler can work with DB2 Content Manager, you run a setup script that OmniFind Enterprise 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 DB2 Information Integrator for Content Version 8.3, you can see the Component Selection window with the 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
```

Where:

node_name

Is the short host name of the DB2 Content Manager server (such as ibmes).

hostname

Is the fully qualified host name of the DB2 Content Manager server (such as ibmes.ibm.com).

port

Is the DB2 Content Manager server port number.

database_name

Is the name of the DB2 Content Manager database (such as ICMNLSDB).

alias

Is the alias of the DB2 Content Manager database (such as CMSVR)

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 SConnectDisconnectICM ICMdatabase_name CMadmin_ID CMadmin_password
```

5. On the crawler server, run the setup script for the DB2 Content Manager crawler:

- 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. On the crawler server, stop and restart the enterprise search system:

```
esadmin system stopall
esadmin system startall
```

Configuring the crawler server on Windows for DB2 Content Manager crawlers

If you install OmniFind Enterprise 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 Version 8.3 for Windows on the crawler server. To ensure that the DB2 Content

Manager crawler can work with DB2 Content Manager, you run a setup script that OmniFind Enterprise 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 DB2 Information Integrator for Content Version 8.3, you can see the Component Selection window with the 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
```

Where:

node_name

Is the short host name of the DB2 Content Manager server (such as *ibmes*).

hostname

Is the fully qualified host name of the DB2 Content Manager server (such as *ibmes.ibm.com*).

port

Is the DB2 Content Manager server port number.

database_name

Is the name of the DB2 Content Manager database (such as *ICMNLSDDB*).

alias

Is the alias of the DB2 Content Manager database (such as *CMSVR*).

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 SConnectDisconnectICM ICMdatabase_name CMadmin_ID CMadmin_password
```

4. On the crawler server, run the setup script for the DB2 Content Manager crawler:
 - 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. On the crawler server, stop and restart the enterprise search system:

```
esadmin system stopall  
esadmin system startall
```

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 setup script on the crawler server. This script, which is provided with OmniFind Enterprise 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.

Important: If the Domino Document Manager server uses DIIOP, and you configure the crawler to use HTTPS or DIIOP over SSL so that transmissions between the crawler and the server are encrypted, you must copy the `TrustedCerts.class` file (for example, `c:\certs` or `/data/certs`) from the Domino Document Manager server to the crawler server. In a two server or four server configuration, you must also copy the `TrustedCerts.class` file to the servers where the search component is installed. You must ensure that the file is in the same location on the crawler server and search servers. You specify the directory path for the `TrustedCerts.class` file when you configure the crawler.

If OmniFind Enterprise 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 73.
- “Configuring the crawler server on Windows to crawl Lotus Domino sources” on page 74.
- “Configuring servers that use the DIIOP protocol” on page 76.
- “Configuring the I/O completion port on AIX to crawl Lotus Domino sources” on page 77.

Document-level security

If collection security is enabled, and a 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:

- Configure Lotus Domino Trusted Servers to validate user credentials.
- Enable global security in WebSphere Application Server and configure the search application to use security. This step ensures that login credentials are validated when users attempt to use the search application. The search servers use the credentials to verify each user's authority to access to Lotus Domino documents.

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.
- If the server is configured to use the DIIOP protocol, you can specify how the crawler is to connect to Domino objects. For example, you can specify options for using HTTPS or the Secure Sockets Layer (SSL) to encrypt communications.
- 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.

Related concepts

“Tips for crawling Lotus Domino databases” on page 72

“Enforcement of document-level security for Lotus Domino documents” on page 269

Related tasks

“Configuring the crawler server on UNIX to crawl Lotus Domino sources” on page 73

“Configuring the crawler server on Windows to crawl Lotus Domino sources” on page 74

“Configuring servers that use the DIIOP protocol” on page 76

“Configuring the I/O completion port on AIX to crawl Lotus Domino sources” on page 77

“Configuring Lotus Domino Trusted Servers to validate user credentials” on page 270

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 of the 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.

Related tasks

“Verifying access to secure Exchange Server documents” on page 269

JDBC database crawlers

You use the JDBC database crawler to include databases that can be accessed with a Java Database Connectivity (JDBC) protocol in an enterprise search collection.

You must configure a separate crawler for each type of database system that you want to crawl. When you create a crawler, you specify options for crawling one database. Later, you can add databases of the same type to the crawl space.

Each row in a database table is treated as a document and the values of the database columns are parsed and indexed as searchable fields. You can configure the crawler to crawl multiple structured tables by associating a plug-in with the crawler when you configure crawler properties. With this plug-in, rows from multiple tables in a relational database that have the same key fields can be joined and treated as a single document. When a user searches the database, data from the joined tables appears as additional fields when the document is displayed in the search results.

Supported database systems and drivers

To use the JDBC protocol to crawl tables in a database, the appropriate JDBC driver must exist on the crawler server. The JDBC database crawler supports the following database systems and type 4 JDBC drivers:

Database system	JDBC type 4 driver name	Standard JDBC driver class paths
IBM DB2 Universal Database Version 8.2 and IBM DB2 Enterprise Server Edition Version 9.1 for Linux, UNIX, and Windows	com.ibm.db2.jcc.DB2Driver	<i>db2_install_root</i> /java/db2jcc.jar <i>db2_install_root</i> /java/db2jcc_license_cu.jar
Oracle 9i and 10g	Oracle.jdbc.driver.OracleDriver	<i>oracle_home</i> /jdbc/lib/ojdbc14.jar
Microsoft SQL Server 2000	com.microsoft.jdbc.sqlserver.SQLServerDriver	<i>mssql_jdbc_home</i> /lib/mssqlserver.jar <i>mssql_jdbc_home</i> /lib/msbase.jar <i>mssql_jdbc_home</i> /lib/msutil.jar
Microsoft SQL Server 2005	com.microsoft.sqlserver.jdbc.SQLServerDriver (The JDBC driver for SQL Server 2005 is not supported on AIX systems.)	<i>install_dir</i> /sqljdbc_1.0/ <i>loc</i> /sqljdbc.jar where <i>loc</i> represents your locale, such as <i>install_dir</i> /sqljdbc_1.0/enu/sqljdbc.jar

JDBC database crawlers versus DB2 crawlers

If you are currently using the DB2 crawler, you might want to continue using it. You cannot migrate data stored for a DB2 crawler to a JDBC database crawler.

Use the DB2 crawler instead of the JDBC database crawler in the following situations:

- You want to crawl DB2 databases with a JDBC type 2 driver.
- You want to crawl Oracle and SQL Server databases that are federated with a DB2 database. With the DB2 crawler, you can access all of these database types through a nickname.
- You want to crawl DB2 for z/OS, DB2 for iSeries, Informix, Sybase, VSAM, IMS, CA-Datcom, or Software AG Adabas databases. You should federate these types of databases with a DB2 database and access them with the DB2 crawler through a nickname.
- You want to use event publishing to update the enterprise search index when updates to a database are published.

Configuration overview

To create or change a JDBC database 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 in the crawl space.
- Specify the type of database that you want to crawl.
- Select the database that you want to crawl and, if necessary, specify a user ID and password that enables the crawler to access the database.
- Set up a schedule for crawling the database.
- Select the tables that you want to crawl.
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.
- 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.

Relationship maps for JDBC databases

When you create rules for a plug-in that crawls multiple structured JDBC database tables, you specify information about the root table and how the parent and child tables are joined.

A plug-in provided for enterprise search enables the JDBC database crawler to join multiple structured tables. You create the plug-in by specifying rules in the

ES_INSTALL_ROOT/default_config/crawler_rdb_plugin.xml file. After you configure the crawler to use the plug-in, rows from tables that have the same key fields are joined and treated as a single document. When a user searches the database, data from the joined tables appears as additional fields when the document is displayed in the search results.

Joining tables through key columns

The following figure shows how the relationship map for multiple tables is built. The JDBC database crawler scans a root table in a database. Some of the columns in the table are key fields that can be used to join the table with other tables. Columns in the joined tables can then be used as keys to join additional tables. Rows in the multiple tables are treated as a single document in the crawl space. The column values are treated as document metadata. The root table is *the parent* in the relation and a table joined at the first level is *a child*. Child tables at the first level can also be parents of tables that are joined at a secondary level.

In this example, the Key 1 and Key 2 columns in the root (parent) table are key fields that enable the table to be joined with subsidiary tables that also have Key 1 and Key 2 columns. One table joined at this first level has key fields, Key 3 and Key 4, that enable the table to be joined with additional tables.

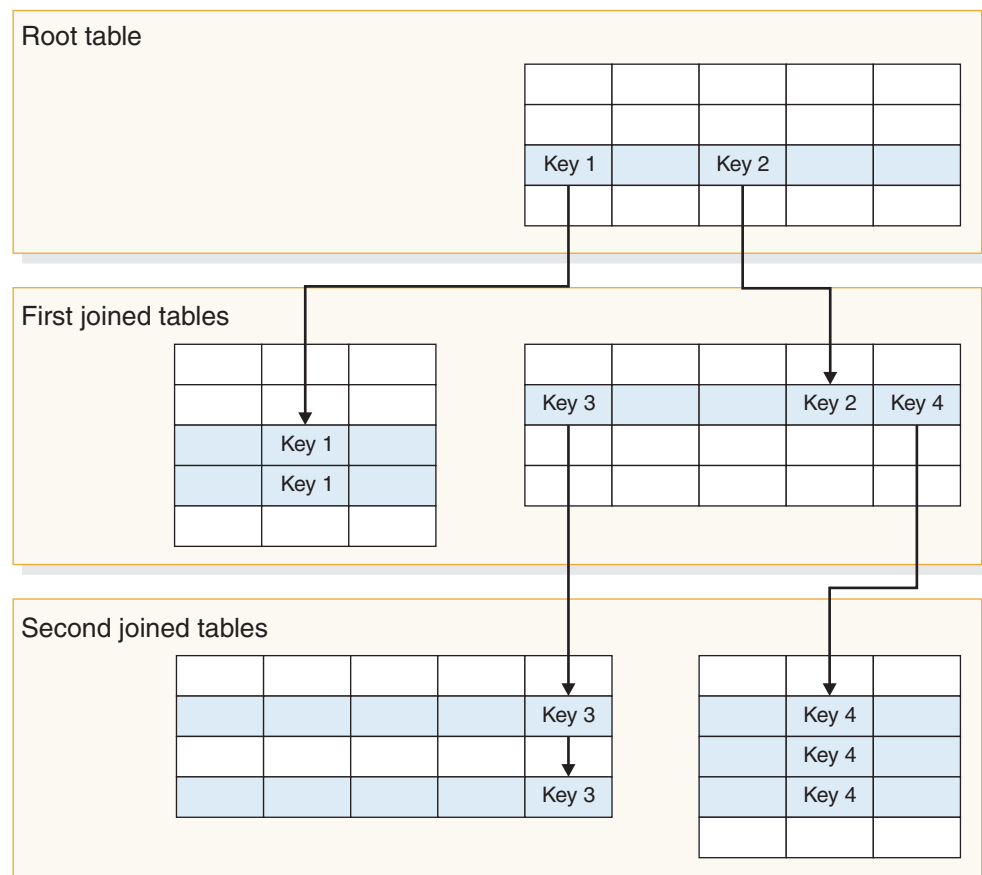


Figure 2. JDBC database tables joined through key fields

Viewing search results

The following figure shows how data from multiple structured tables are presented as a single document in the search results. Without the plug-in, a user who

searched the EMPLOYEE table might see a row from the root table displayed in the search results and see only the values for the EMPLOYEE table columns (ID, Name, and Office).

With the plug-in, the crawler is able to use the Office column as a key to join the EMPLOYEE table with the OFFICE table. The Country column in the OFFICE table serves as a key to join that table with the COUNTRY table. After the tables are joined, users who search the EMPLOYEE table can see values from columns in the OFFICE and COUNTRY tables as additional fields in the search results.

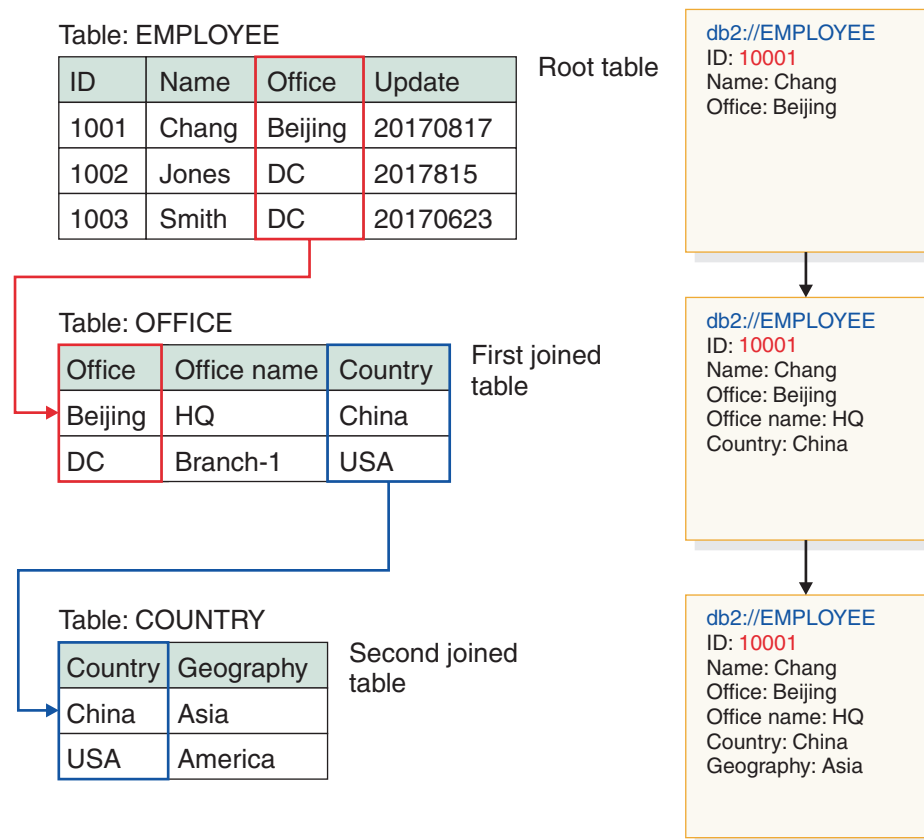


Figure 3. Values from joined JDBC tables are displayed in the search results

Crawling multiple structured JDBC database tables

You can configure the JDBC database crawler to join multiple structured tables that have the same key fields.

Before you begin

To do this task, you must have authority to log in as the enterprise search administrator.

About this task

When you configure crawler properties for a JDBC database crawler, you can specify a plug-in for crawling multiple structured tables that are related to each other through key fields. Without this plug-in, rows in a database table are treated like individual documents and the values of the database columns are searchable as individual fields. With this plug-in, rows from multiple tables in a relational

database that have the same key fields are joined and treated as a single document. The crawler adds data that it retrieves from the joined tables to the metadata for the original row of a database table. When a user searches the database, this additional data appears as additional fields when the document is displayed in the search results.

Restrictions

Data types that cannot be crawled

The crawler cannot crawl fields in the tables that you join that contain these binary data types:

BLOB
CHARACTER FOR BIT DATA
VARCHAR FOR BIT DATA
LONG VARCHAR FOR BIT DATA

Limitations on the scope of the crawl space

The tables to be joined must be in the same relational database. You cannot join tables across databases.

If a table in a database is configured to be joined with other tables, this setting is universal for all crawlers in a collection that are enabled to use the plug-in. However, you can create multiple collections and configure separate crawlers to crawl different root tables and join different tables.

Restrictions on the use of other plug-ins

If you configure the crawler to use the plug-in for crawling multiple structured tables, you cannot associate another plug-in with the crawler. For example, you cannot specify a custom plug-in for applying business and security rules. You cannot associate more than one plug-in with a crawler.

Restrictions on number of tables, rows, fields, and keys

The maximum number of joined tables per database is five, and the sum of the rows in those tables should be less than one million. The maximum number of fields that can be read from a table is 10. To join tables, a key pair is used. That means it is not possible to join tables by using multiple keys.

Ensuring that changes in joined tables are crawled

If the rows in a root table do not change between crawls, and the crawler is not configured to do a full crawl, the crawler ignores the unchanged rows. If rows in a table that is joined to the root table change, even though the root table does not, you need to do one of the following actions to ensure that the changes are detected and crawled:

- A root table in the target database should have a timestamp field. Configure the target database to have a timestamp field that gets updated when a row in the root table changes or when rows in any of the joined subsidiary tables change. When you set up the JDBC database crawler, be sure to specify this timestamp field as the field that the crawler should use to determine whether changes in the tables occurred.
- Specify that the crawler is to do a full crawl when you configure the crawler schedule. This option ensures that all of the tables are crawled each time regardless of whether any changes occurred.

Procedure

To set up the JDBC database crawler to crawl multiple structured tables:


1. Log in as the enterprise search administrator on the crawler server and copy the ES_INSTALL_ROOT/default_config/crawler_rdb_plugin.xml file to create the ES_NODE_ROOT/master_config/crawler_rdb_plugin.xml file.
2. Edit the ES_NODE_ROOT/master_config/crawler_rdb_plugin.xml file with a text editor that supports UTF-8 encoding.
 - a. Edit the <Server DBURL="jdbc:db2://db_server_url:50000/SAMPLE"> element and replace jdbc:db2://db_server_url:50000/SAMPLE with the URL of the JDBC database to be crawled. When you configure the crawler, be sure to specify this same URL for the database to be crawled.
 - b. If the database to be crawled is not a DB2 database, edit the <JDBCdriver>com.ibm.db2.jcc.DB2Driver</JDBCdriver> element and replace com.ibm.db2.jcc.DB2Driver with the appropriate JDBC driver. When you configure the crawler, be sure to specify this same driver for the database to be crawled.
 - c. Edit the <User>username</User> element and replace username with a user ID that has authority to access the database to be crawled.
 - d. Edit the <Password Encryption="True">encrypted_password</Password> element and replace encrypted_password with an encrypted password for the specified user ID. You can copy the encrypted password from the ES_NODE_ROOT/master_config/col_collection_name.JDBC_crawler_name/jdbccrawler.xml file and paste it here. If the password does not need to be encrypted, replace Encryption="True" with Encryption="False", and replace encrypted_password with a plain text password.
 - e. If you leave the <Delimiters Use="True"> element as is, multiple terms in a column are separated by characters (,) defined in the <Delimiter> element. Sets of terms per table are separated by characters (;) defined in the <SecondDelimiter> element. If you set <Delimiters Use="True"> to <Delimiters Use="False">, delimiter characters are not used and multiple metadata fields with the same field name are added as document metadata.
 - f. If you use the <Delimiters Use="True"> element, edit the <Delimiter>,</Delimiter> and <SecondDelimiter>;</SecondDelimiter> elements to specify the characters to use as value separators.
 - g. Edit the <RelationMap Root="DB2INST1.TABLE_0"> element and replace DB2INST1.TABLE_0 with the name of a root table that is to be crawled.
 - h. Edit the <Relation Parent="DB2INST1.TABLE_0" ParentAlias="T0" ParentKey="ID" Child="DB2INST1.TABLE_1" ChildAlias="T1" ChildKey="ID"/> element.
 - Replace Parent="DB2INST1.TABLE_0" with the name of a table that is a parent in the relation.
 - Replace ParentKey="T0" with an alias of the parent table. This alias should be unique and not duplicated in the crawler_rdb_plugin.xml file.
 - Replace ParentKey="ID" with the name of a column that is used as a key field in the relation.
 - Replace Child="DB2INST1.TABLE_1" ChildAlias="T1" ChildKey="ID" with information about a child table to be crawled.

This structure defines how the tables are to be joined. For example, the following relationship map specifies that a root table named DB2INST1.TABLE_A is to be crawled. Tables DB2INST1.TABLE_B and DB2INST1.TABLE_C are joined under the condition DB2INST1.TABLE_A.ID=DB2INST1.TABLE_B.ID AND DB2INST1.TABLE_B.ID=DB2INST1.TABLE_C.ID.

```

<RelationMap Root="DB2INST1.TABLE_A">
<Relation Parent="DB2INST1.TABLE_A" ParentAlias="TA" ParentKey="ID"
  Child="DB2INST1.TABLE_B" ChildAlias="TB" ChildKey="ID"/>
<Relation Parent="DB2INST1.TABLE_B" ParentAlias="TB" ParentKey="ID"
  Child="DB2INST1.TABLE_C" ChildAlias="TC" ChildKey="ID"/>

```

- i. Repeat step 2h on page 67 to create <Relation> elements for all relations that join tables from a root table.
 - j. Edit the <Target TableAlias="T1"> element and replace TableAlias="T1" with a ChildAlias value that you defined in step 2h on page 67.
 - k. Edit the <Field Name="ID" FieldName="ID_1" Enabling="True" Searchable="True" FieldSearchable="True" IsContent="True"/> element.
 - Replace Name="ID" with the name of a column in the documents to be crawled.
 - Replace FieldName="ID_1" with the name of a metadata field in the documents to be crawled. This value is used as the display name for the column in the enterprise search administration console and the search results.
 - Replace Enabling="True" with "False" if this column should not be included in the document metadata.
 - Replace Searchable="True" with "False" to prevent users from searching this column with a free text query.
 - Replace FieldSearchable="True" with "False" to prevent users from searching this column by the column name.
 - Replace IsContent="True" with "False" to indicate that the column does not contain searchable content. If you specify Searchable="True" and IsContent="True", then the value of the column is used to detect duplicate documents and becomes part of the dynamic document summary in the search results.
 - l. Repeat step 2k to create <Field> elements for all of the columns that are to be crawled.
 - m. Repeat steps 2j and 2k to create <Target> and <Field> elements for all of the child tables that are referenced in the relationship map (<RelationMap>).
 - n. Repeat steps 2g on page 67 through 2m to create multiple relationship maps for multiple root tables.
 - o. Repeat steps 2a on page 67 through 2n to configure a relationship map for another database.
3. Configure the crawler to use the plug-in:
 - a. Open the enterprise search administration console, edit a collection, and select the Crawl page.
 - b. Create a JDBC database crawler, or locate a crawler that you want to change and click  **Crawler properties**.
 - c. In the **Plug-in class name** field, type the name of the plug-in for crawling multiple structured tables:
com.ibm.es.plugin.rdb.RDBPlugin
 - d. In the **Plug-in class path** field, type the fully qualified paths for the plug-in and the JDBC drivers used by the plug-in. For example, the path for the JDBC driver for a DB2 database on a Windows system might be:
C:\Program Files\IBM\es\lib\plugin_rdb.jar;C:\Program Files\IBM\SQLLIB\java\db2jcc.jar;C:\Program Files\IBM\SQLLIB\java\db2jcc_license_cu.jar;

- e. Either click **Next** to continue creating the crawler, or click **OK** to save your changes.
4. To deploy the `crawler_rdb_plugin.xml` file to the system configuration, restart the enterprise search system:

```
esadmin system stop
esadmin system start
```

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 to crawl from one NNTP server. Later, when you edit the crawl space, you can add other NNTP servers that you want the crawler to crawl.

When you identify the news groups to crawl, you can select groups to include and select groups to exclude from the crawl space. 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.
- Identify the NNTP server to crawl. If the server is password-protected, you must provide a user ID and password for the crawler to use to access news groups on the server.
- Set up a schedule for crawling the server.
- Specify patterns to include news groups, and specify patterns to exclude certain news groups from the crawl space.
- Specify whether the crawler should automatically detect the language and code page of the articles to be crawled, or whether the crawler should use a specific language and code page.
- 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.

Tip:

For detailed examples of how to configure a secure Notes crawler, see the scenario for a large organization in the IBM Redbook, *IBM OmniFind Enterprise Edition Configuration and Implementation Scenarios*.

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 setup script on the crawler server. This script, which is provided with OmniFind Enterprise 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.

Important: If the Lotus Notes server uses DIIOP, and you configure the crawler to use HTTPS or DIIOP over SSL so that transmissions between the crawler and the server are encrypted, you must copy the `TrustedCerts.class` file (for example, `c:\certs` or `/data/certs`) from the Lotus Notes server to the crawler server. In a two server or four server configuration, you must also copy the `TrustedCerts.class` file to the servers where the search component is installed. You must ensure that the file is in the same location on the crawler server and search servers. You specify the directory path for the `TrustedCerts.class` file when you configure the crawler.

If OmniFind Enterprise 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 73.
- “Configuring the crawler server on Windows to crawl Lotus Domino sources” on page 74.
- “Configuring servers that use the DIIOP protocol” on page 76.
- “Configuring the I/O completion port on AIX to crawl Lotus Domino sources” on page 77.

Document-level security

If collection security is enabled, and a 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:

- Configure Lotus Domino Trusted Servers to validate user credentials.

- Enable global security in WebSphere Application Server and configure the search application to use security. This step ensures that login credentials are validated when users attempt to use the search application. The search servers use the credentials to verify each user's authority to access to Lotus Domino 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.
- If the server is configured to use the DIIOP protocol, you can specify how the crawler is to connect to Domino objects. For example, you can specify options for using HTTPS or the Secure Sockets Layer (SSL) to encrypt communications.
- Select the databases or directories that you want to crawl. When you crawl directories, you can specify patterns to include or exclude databases, which can help you divide the task of crawling large directories across multiple crawlers.
- 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 269

Related tasks

“Configuring Lotus Domino Trusted Servers to validate user credentials” on page 270

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 2. 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 **All** check box under **Crawl** 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.

Related concepts

“Domino Document Manager crawlers” on page 59

“QuickPlace crawlers” on page 78

Configuring the crawler server on UNIX to crawl Lotus Domino sources

If you install OmniFind Enterprise 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 on the crawler server. To ensure that the crawlers can work with the Domino libraries, you run a setup script that OmniFind Enterprise 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 server and the group notes on the crawler server:

- a. Log in as the root user:

```
su - root
```

- b. Add a user:

```
useradd server
```

- c. Add a password for this user:

```
passwd server
```

You will be prompted to change the password.

2. Install Lotus Domino Server on the crawler server:

- a. Insert the Domino Server CD 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.

- e. Ensure that the enterprise search administrator ID has permission to access the home/server directory. This administrator ID specified when OmniFind Enterprise Edition is installed.
3. On the crawler server, run the setup script provided by OmniFind Enterprise Edition:
 - a. Log in as the enterprise search administrator (this user ID was specified when OmniFind Enterprise 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. On the crawler server, stop and restart the enterprise search system:

```
esadmin system stopall
esadmin system startall
```

Related concepts

“Domino Document Manager crawlers” on page 59

“QuickPlace crawlers” on page 78

Configuring the crawler server on Windows to crawl Lotus Domino sources

If you install OmniFind Enterprise 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 Domino client libraries. You install these libraries by installing Lotus Domino Server on the crawler server. To ensure that the crawlers can work with the Domino libraries, you run a setup script that OmniFind Enterprise 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. 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 Domino Server CD. (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 Domino documentation if you need assistance.
 - d. Ensure that the enterprise search administrator ID has permission to access the Domino data directory. This administrator ID specified when OmniFind Enterprise Edition is installed.
3. On the crawler server, run the setup script that is provided by OmniFind Enterprise Edition:
 - a. Log in with the enterprise search administrator ID (this user ID was specified when OmniFind Enterprise 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. On the crawler server, stop and restart the enterprise search system:

```
esadmin system stopall  
esadmin system startall
```

Related concepts

“Domino Document Manager crawlers” on page 59

“QuickPlace crawlers” on page 78

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.

Related concepts

“Domino Document Manager crawlers” on page 59

“QuickPlace crawlers” on page 78

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**.

Related concepts

“Domino Document Manager crawlers” on page 59

“QuickPlace crawlers”

QuickPlace crawlers

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

You can also use the QuickPlace crawler to crawl places that you manage with Lotus Quickr services for Lotus Domino. If you use Lotus Quickr services for

WebSphere Portal, use the Seed list crawler to add documents that are stored in Lotus Quickr libraries to an enterprise search collection.

Tip:

For detailed examples of how to configure a secure QuickPlace crawler, see the scenario for a small organization in the IBM Redbook, IBM OmniFind Enterprise Edition Configuration and Implementation Scenarios.

Crawler server configuration

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

If a 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 target server so that the QuickPlace crawler can access the server.

If a server that you plan to crawl uses a Lightweight Directory Access Protocol (LDAP) server, then the target 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 target server to use the LDAP server as a secondary Domino server.

Important: If the target server uses DIIOP, and you configure the crawler to use HTTPS or DIIOP over SSL so that transmissions between the crawler and the server are encrypted, you must copy the `TrustedCerts.class` file (for example, `c:\certs` or `/data/certs`) from the target server to the crawler server. In a two server or four server configuration, you must also copy the `TrustedCerts.class` file to the servers where the search component is installed. You must ensure that the file is in the same location on the crawler server and search servers. You specify the directory path for the `TrustedCerts.class` file when you configure the crawler.

When you configure the crawler and specify a user ID for the crawler to use, you must specify an ID that has sufficient authority to access all of the QuickPlace places in a Domino domain. To ensure this, assign the user ID to the reserved group named `QuickPlaceAdministratorsSUGroup`.

If OmniFind Enterprise 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 73.
- “Configuring the crawler server on Windows to crawl Lotus Domino sources” on page 74.
- “Configuring servers that use the DIIOP protocol” on page 76.
- “Configuring the QuickPlace server to use Local User security” on page 271.
- “Configuring Directory Assistance on a QuickPlace server” on page 272.

- “Configuring the I/O completion port on AIX to crawl Lotus Domino sources” on page 77.

Document-level security

If collection security is enabled, and a 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:

- Configure Lotus Domino Trusted Servers to validate user credentials.
- Enable global security in WebSphere Application Server and configure the search application to use security. This step ensures that login credentials are validated when users attempt to use the search application. The search servers use the credentials to verify each user’s authority to access to Lotus Domino documents.

Attachment crawling

In Lotus QuickPlace, you can import and publish Microsoft Office documents (the options that you select when importing include Imported Page, Microsoft Word Page, Microsoft Excel Page, Microsoft PowerPoint Page, and Multiple Imported Pages). The QuickPlace crawler can crawl these types of imported documents as attachments only under the following conditions:

- The server uses the DIIOP protocol.
- You enable attachment crawling when you configure crawling options for the crawler.
- You configure the crawler to crawl the "\$FILE" field or all fields.

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 places on the server or only the places that you specify, and whether you want to crawl all of the rooms in the places to be crawled or only the rooms that you specify.

Restriction: When you specify a user ID for the crawler to use, be sure to specify an ID that has sufficient authority to access all of the QuickPlace places in the Domino domain. You can do this by configuring the QuickPlace server and assigning a user ID to the reserved group named QuickPlaceAdministratorsSUGroup. Note that the group name contains no embedded spaces and is case sensitive.

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.
- If the server is configured to use the DIIOP protocol, you can specify how the crawler is to connect to Domino objects. For example, you can specify options for using HTTPS or the Secure Sockets Layer (SSL) to encrypt communications.
- 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.

Related concepts

"Tips for crawling Lotus Domino databases" on page 72

"Enforcement of document-level security for Lotus Domino documents" on page 269

Related tasks

"Configuring the crawler server on UNIX to crawl Lotus Domino sources" on page 73

"Configuring the crawler server on Windows to crawl Lotus Domino sources" on page 74

"Configuring servers that use the DIIOP protocol" on page 76

"Configuring the I/O completion port on AIX to crawl Lotus Domino sources" on page 77

"Configuring Lotus Domino Trusted Servers to validate user credentials" on page 270

"Configuring the QuickPlace server to use Local User security" on page 271

"Configuring Directory Assistance on a QuickPlace server" on page 272

Seed list crawlers

If you use IBM Lotus Quickr services for WebSphere Portal, you can use the Seed list crawler to add documents in Lotus Quickr libraries to an enterprise search collection.

A Lotus Quickr library is a container for document files. The Seed list crawler does not support crawling Web-based content, such as wikis and blogs.

If you use Lotus Quickr services for Lotus Domino, use the QuickPlace crawler to add Lotus Quickr documents to a collection.

WebSphere Portal server configuration

If you install Lotus Quickr on a WebSphere Portal version 6 server, you can use the Seed list crawler to crawl Lotus Quickr library documents. You can configure options for crawling these documents separately from options that you specify for portal sites that are crawled by a WebSphere Portal crawler.

Before you create a Seed list crawler, you must follow the procedures to set up enterprise search in WebSphere Portal. To set up the enterprise search environment, you run a script (`wp6_install.sh` on AIX, Linux, or Solaris, or `wp6_install.bat` on Windows) that is provided with OmniFind Enterprise Edition on the search servers.

A user agent string identifies which browser or robot is accessing a server. When crawling a Lotus Quickr server, the Seed list crawler uses the user agent string `OmniFind SeedlistCrawler/1.0`.

Configuration overview

You can use the Seed list crawler to crawl any number of Lotus Quickr documents. When you configure the crawler, you specify the server to be crawled. The crawler then crawls all documents in the Lotus Quickr libraries on that server.

The documents to be crawled must be accessible by the same Lotus Quickr administrator ID and password. To crawl sites that use different credentials, you must configure a separate Seed list crawler.

To create or change a Seed list 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 the Lotus Quickr documents.
- Specify the server to be crawled and information that enables the crawler to connect to the server.

When you create or edit the crawler, you can test the crawler's ability to connect to the documents to be crawled. Messages tell you whether the crawler can access the documents to be crawled before you start the crawler.

- Specify 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 access controls to be enforced 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.

Important: To search secure Lotus Quickr documents, you must submit searches by using the Search portlet for enterprise search from within WebSphere Portal.

Searches submitted from the sample search application, ESSearchApplication, do not have the proper credentials and cannot verify the user's authority to access documents.

- Specify information that enables the crawler to communicate with a proxy server, if a proxy server is used to serve pages.
- If you use another product to protect your WebSphere Portal server and Lotus Quickr documents (such as IBM Tivoli® Access Manager WebSEAL or CA SiteMinder SSO Agent for PeopleSoft), specify single sign-on credentials that enable the crawler to access documents on the server.
- Specify information about a keystore file so that the crawler can use the Secure Sockets Layer (SSL) protocol to connect to the server.
- Specify the language and code page of the documents to be crawled.
- Specify options for crawling and searching metadata in Lotus Quickr documents.
- Specify schedules for crawling Lotus Quickr documents.

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

"Integration with WebSphere Portal" on page 325

Related tasks

"Setting up enterprise search in WebSphere Portal version 6" on page 332

"Setting up the enterprise search portlet for Lotus Quickr" on page 337

UNIX file system crawlers

To include documents that are stored in AIX, Linux, or Solaris 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 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 AIX, Linux, or Solaris file system sources (the UNIX file system crawler does not appear in the list of available crawler types).

The UNIX file system crawler crawls documents according to read permissions that are specified for the enterprise search administrator.

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.

Tip:

For detailed examples of how to configure a Web crawler, see the scenario for a medium organization in the IBM Redbook, *IBM OmniFind Enterprise Edition Configuration and Implementation Scenarios*.

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

To crawl a Web site that uses the Robots Exclusion 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 tries to comply 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.

How the Web crawler uses the robots exclusion protocol

The Web crawler tries to comply with the Robots Exclusion protocol and not crawl Web sites if rules in the server's `robots.txt` file disallow crawling.

A successful download is when the crawler can retrieve the robots.txt file from a Web server or confirm that a robots.txt file does not exist. The download is considered a failure when the crawler cannot obtain the rules or cannot confirm that a robots.txt file exists.

A successful download does not mean that the crawler has permission to crawl because rules in the robots.txt file can disallow crawling. A download failure temporarily prohibits crawling because the crawler cannot determine what the rules are.

These are the steps that the crawler takes when attempting to download the robots.txt file:

1. When the crawler discovers a new site, it tries to obtain the server's IP address. If this attempt fails, crawling is not possible.
2. When at least one IP address is available, the crawler tries to download the robots.txt file by using **HTTP (or HTTPS) GET**.
3. If the socket connection times out, is broken, or another low-level error occurs (such as an SSL certificate problem), the crawler logs the problem, and repeats the attempt on every IP address known for the target server.
4. If no connection is made after the crawler tries all addresses, the crawler waits two seconds, then tries all the addresses one more time.
5. If a connection is made, and HTTP headers are exchanged, the return status is examined. If the status code is 500 or higher, the crawler interprets this as a bad connection and continues trying other IP addresses. For any other status, the crawler stops trying alternative IP addresses and proceeds according to the status code.

After the crawler receives an HTTP status code below 500, or after the crawler tries all IP addresses twice, the crawler proceeds as follows:

1. If no HTTP status below 500 was received, the site is disqualified for the time being.
2. If an HTTP status of 400, 404 or 410 was received, the site is qualified for crawling with no rules.
3. If an HTTP status of 200 through 299 was received, the following conditions direct the next action:
 - If the content was truncated, the site is disqualified for the time being.
 - If the content parsed without errors, the site is qualified for crawling with the rules that were found.
 - If the content parsed with errors, the site is qualified for crawling with no rules.
4. If any other HTTP status was returned, the site is disqualified for the time being.

When the crawler attempts to download the robots.txt file for a site, it updates a persistent timestamp for that site called the robots date. If a site is disqualified because the robots.txt information is not available, the persistent robots failure count is incremented.

When the retry interval is reached, the crawler tries again to retrieve robots.txt information for the failed site. If the number of successive failures reaches the maximum number of failures allowed, the crawler stops trying to retrieve the robots.txt file for the site and disqualifies the site for crawling.

After a site is qualified for crawling (the check for robots.txt file rules succeeds), the failure count is set to zero. The crawler uses the results of the download until the interval for checking rules elapses. At that time, the site must be qualified again.

Tip:

- If a server returns content but it contains syntax errors, or if the server uses a robots protocol other than the 1994 version, or if the content contains something other than robots rules (such as a soft error page), the crawler acts as though the server does not have an applicable rules file and crawls the site. This action is usually correct because collection administrators do not control site content or default server behavior. If a Web server administrator does not want a site to be crawled, and does not want to install a conforming rules file, the collection administrator can block the site from the Web crawler by specifying the site's domain, IP address, or HTTP prefix in the crawler's rules.
- If a server returns a 302 status code or other redirection codes, the crawler interprets the code to mean that the site has a robots.txt file that should be used, but the file is not at the conforming location (the site root). The Web server administrator must move the file to the correct location so that the Web crawler can abide by the rules in the file.
- If there are certificate problems (for example, the certificate might be out of date, the certificate authority might not be trusted, or the certificate might be self-signed and the crawler is not configured to accept self-signed certificates), the crawler interprets the problem as a failure to connect with the site and disqualifies the site. The same problems would probably prevent crawling other pages from the site, anyway. To enable the site to be crawled, the collection administrator must enable self-signed certificates, add the site's authority to the trusted keystore file, or ask the Web server administrator to obtain an up-to-date certificate.
- The Web crawler might be configured to use HTTP basic authentication (including HTTP basic proxy authentication). If properly configured, authentication is required for downloads of robots.txt files, too. A status code of 403, 407, or other authentication related responses indicates authorization problems, and the crawler disqualifies the site. (Only HTTP basic authentication is supported.)
- If the robots.txt file for a site exceeds the maximum length for a robots page, the collection administrator can raise the configured maximum (the default value of one million bytes should be sufficient).

To help troubleshoot problems, you can request a site report when you monitor the Web crawler. Select options for viewing the contents of the robots.txt file (to see whether rules forbid the Web crawler from accessing the site), seeing the date and time that the crawler last attempted to download the robots.txt file (the crawler will not attempt again until the retry interval elapses), and seeing how many consecutive attempts the crawler made to download the robots.txt file but failed to do so. Click **Help** while you monitor the Web crawler to learn more about these site report options and how to interpret the results.

For more information about the Robots Exclusion protocol, see the following URL: <http://www.robotstxt.org/wc/exclusion.html>.

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, cannot be detected by the Web crawler, and thus will not be indexed.

Because the Web crawler does not parse JavaScript in HTML files, URLs in JavaScript are not crawled. To enable the Web crawler to crawl URLs in JavaScript, you can do either of following actions:

- In the enterprise search administration console, edit the Web crawler and, on the Web Crawl Space page, add the URLs to the list of URLs that the crawler is to use as a starting point for adding URLs to the collection (**Start URLs**). For the changes to become effective, restart the Web crawler (you do not need to start a full crawl).
- Use the anchor tag (``) to specify the URLs as hypertext links in the HTML file.

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
```

You can specify an asterisk as a wildcard character, which causes the rule to apply to any host name that matches the rest of the pattern. For example, you can specify that no domains that begin with `server` and end in `ibm.com` are to be crawled:

```
forbid domain server*.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 and netmask as the target. For example:

IPv4 allow address 9.0.0.0 255.0.0.0

IPv6 If you run enterprise search on a Windows 2003 server, and enabled the enterprise search system to use the IP version 6 (IPv6) protocol, you must enclose the address in brackets.

allow address [2001:db8:0:1:0:0:0:1]

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.

The following rule is a useful rule to include as the final address in your list of address rules. 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).

IPv4 forbid address 0.0.0.0 0.0.0.0

IPv6

forbid address :: ::

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 status 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.

See the preceding discussion about address rules for examples of how to specify the final rule in your list of address rules to prevent the crawler from crawling Web sites that are outside the corporate network.

- 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.

Support for IPv6 addresses: If you run enterprise search on a Windows 2003 server, and enabled the enterprise search system to use the IP version 6 (IPv6) protocol, you must enclose the start URLs in brackets. For example:

```
http://[2001:db8:0:1:0:0:0:1]  
http://[2001:db8:0:1::1]
```

Related tasks

“Enabling support for the IPv6 protocol” on page 27

Testing 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.

You can click **Test** when you specify the domains, HTTP prefixes, or IP addresses to be crawled, or you can select the Test URLs page to test the crawler’s ability to connect to the start URLs in addition to URLs that you specify.

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 status 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 main index build occurs.

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 status code. The status 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 status 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 status 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 (status 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 status code 404 (the page was not found).

Table 3. Soft error page example

URL pattern	Title pattern	Content pattern	HTTP status 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 status codes. Each status 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*

Affect on performance

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 status 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 status code, which indicates that the requester is not authorized to access the requested page.
- The Web server also challenges the requester 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, which enables the authorized user to access additional restricted content without specifying additional credentials.

After 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 asked 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.

Using a plug-in to crawl secure WebSphere Portal sites

If global security is enabled in WebSphere Application Server, and you want to crawl secure WebSphere Portal sites with the Web crawler, you must create a crawler plug-in to handle the form-based authentication requests. For a discussion about form-based authentication and a sample program that you can adapt for your custom Web crawler plug-in, see <http://www.ibm.com/developerworks/db2/library/techarticle/dm-0707nishitani>.

The plug-in is required if you use the Web crawler to crawl any sites through WebSphere Portal, including Workplace Web Content Management sites and Lotus Quickr sites.

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 uses when crawling pages in the proxy server's domain. Before you add a proxy server to the crawl space, obtain the names of the domains that are served by the proxy server, the proxy server host name or IP address, and the port number that the proxy server uses.

If the proxy server requires authentication, also obtain a user name and password that the Web crawler can use to access the pages served by the proxy server. The Web crawler supports only HTTP basic proxy authentication, as described in RFC2616 (<http://rfc.net/rfc2616.html>). Other types of authorization, including Windows NT LAN Manager (NTLM), are not supported.

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 OmniFind Enterprise 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 status 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 status 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 status 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 status 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 status 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 OmniFind Enterprise 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.

Configuring which date the Web crawler uses for crawled documents

You can specify an option in a configuration file to control which date the Web crawler uses as the date of a crawled document.

Before you begin

By default, the Web crawler sets the value of the Date field in crawled documents to the date that a document is crawled. If you prefer, you can configure the Web crawler to set this date to the Last-Modified date and time that is returned by the Web server.

Last-Modified data might not be available for all documents. If you configure the crawler to use this value, and the Web server does not return Last-Modified data for a document, then the crawler sets the value of the Date field for the crawled document to the date and time that the document is crawled.

Table 4. How the Web crawler configures the Date field for crawled documents

Crawler configuration	Data returned by a Web server	Date metadata field for crawled documents
Default configuration	A Web server returns a Last-Modified value for a crawled document	Crawled date and time
	A Web server does not return a Last-Modified value for a crawled document	Crawled date and time
Configured to use the Last-Modified date	A Web server returns a Last-Modified value for a crawled document	Last-Modified value returned by the Web server
	A Web server does not return a Last-Modified value for a crawled document	Crawled date and time

To specify which date the Web crawler is to use for crawled documents, you must be an enterprise search administrator.

Procedure

To configure a Web crawler so that the crawled document date is the Last-Modified date and time returned by the Web servers:

1. Log in as the enterprise search administrator on the crawler server. This user ID was specified when OmniFind Enterprise Edition was installed.
2. Edit the `crawl.properties` file for the Web 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.

```
ES_NODE_ROOT/master_config/crawler_session_ID/crawl.properties
```

For example:

```
/home/esadmin/master_config/col_00112.WEB_23344/crawl.properties  
C:\Program Files\IBM\esadmin\master_config\col_55667.WEB_78899\crawl.properties
```

3. Add the following line, and save the file:

```
which_date=LastModified
```

4. From the enterprise search administration console, restart the Web crawler.

If a Web server returns Last-Modified data for documents that are crawled by this Web crawler, the crawler uses the returned date and time as the Date of the crawled documents.

Web Content Management crawlers

To include IBM Workplace Web Content Management documents in an enterprise search collection, you must configure a Web Content Management crawler.

WebSphere Portal server configuration

If you install Workplace Web Content Management on a WebSphere Portal version 6 server, you can use the Web Content Management crawler to crawl Web Content Management sites. You can configure options for crawling these sites separately from options that you specify for portal sites that are crawled by a WebSphere Portal crawler.

Before you create a Web Content Management crawler, you must follow the procedures to set up enterprise search in WebSphere Portal. To set up the enterprise search environment, you run a script (`wp6_install.sh` on AIX, Linux, or Solaris, or `wp6_install.bat` on Windows) that is provided with OmniFind Enterprise Edition on the search servers.

URL format

When you specify the URLs to crawl, you must use the following format:

```
http_protocol://portal_hostname:port_number/portal_prefix  
/WCM_search_seed_servlet_path/searchseed?site=WCM_site_name&lib=WCM_library_name
```

The following example shows a the URL for a site at the default installation path of Workplace Web Content Management on WebSphere Portal:

```
http://portal.server.ibm.com:80/wps/wcmsearchseed/  
searchseed?site=SiteTest01&lib=Web+Content
```

If the site name or library name contains spaces, you must replace the space with a plus sign (+) character. For example, replace Web Content with Web+Content.

Configuration overview

You can use the Web Content Management crawler to crawl any number of Web Content Management sites. When you configure the crawler, you specify the URLs for the sites to be crawled. The crawler then downloads the pages that are linked from the specified sites.

The sites to be crawled must be accessible by the same WebSphere Portal administrator ID and password. To crawl sites that use different credentials, you must configure a separate Web Content Management crawler.

To create or change a Web Content Management 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 documents in the crawl space.
- Specify the URLs for the sites to be crawled and information that enables the crawler to connect to the sites.

When you create or edit the crawler, you can test the crawler's ability to connect to the URLs to be crawled. Messages tell you whether the crawler can access the documents to be crawled before you start the crawler.

- Specify 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 access controls to be enforced 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.

Important: To search secure Web Content Management pages, you must submit searches by using the Search portlet for enterprise search from within WebSphere Portal. Searches submitted from the sample search application, ESSearchApplication, will not have the proper credentials and cannot verify the user's authority to access documents.

- Specify information that enables the crawler to communicate with a proxy server, if the Web Content Management sites use a proxy server to serve documents.
- If you use another product to protect your WebSphere Portal server and Web sites (such as IBMTivoli Access Manager WebSEAL or CA SiteMinder SSO Agent for PeopleSoft), specify single sign-on credentials that enable the crawler to access documents on the server.
- Specify information about a keystore file so that the crawler can use the Secure Sockets Layer (SSL) protocol to connect to the Web Content Management sites.
- Specify the language and code page of the documents to be crawled.

- Specify options for crawling and searching metadata in Web Content Management documents.
- Specify schedules for crawling the Web Content Management sites.

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

“Integration with WebSphere Portal” on page 325

Related tasks

“Setting up enterprise search in WebSphere Portal version 6” on page 332

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 run a script to set up the enterprise search environment in WebSphere Portal. Different scripts are required for different versions of WebSphere Portal. The scripts are installed on the search servers when OmniFind Enterprise Edition is installed.

- For WebSphere Portal version 5.1.0 or later, you run the `wp5_install.sh` script on AIX, Linux, and Solaris systems or the `wp5_install.bat` script on a Windows system.
- For WebSphere Portal version 6, you run the `wp6_install.sh` script on AIX, Linux, and Solaris systems or the `wp6_install.bat` script on a Windows system.

Tip:

For detailed examples of how to configure a secure WebSphere Portal crawler, see the scenario for a medium organization in the IBM Redbook, IBM OmniFind Enterprise Edition Configuration and Implementation Scenarios.

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 and pages that are linked from the specified portal URL. To crawl another portal site, create another crawler.

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.

When you create or edit the crawler, you can test the crawler's ability to connect to the URL to be crawled. Messages tell you whether the crawler can access the documents to be crawled before you start the crawler.

- Specify 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 access controls to be enforced 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.

Important: To search secure WebSphere Portal pages, you must submit searches by using the Search portlet for enterprise search from within WebSphere Portal. Searches submitted from the sample search application, ESSearchApplication, do not have the proper credentials and cannot verify the user's authority to access documents.

- Specify information that enables the crawler to communicate with a proxy server, if the WebSphere Portal site uses a proxy server to serve pages.
- If you use another product to protect your WebSphere Portal server and Web sites (such as IBMTivoli Access Manager WebSEAL or CA SiteMinder SSO Agent for PeopleSoft), specify single sign-on credentials that enable the crawler to access documents on the server.
- Specify information about a keystore file so that the crawler can use the Secure Sockets Layer (SSL) protocol to connect to the WebSphere Portal site.
- Specify the language and code page of the documents to be crawled.
- Specify options for crawling and searching metadata in WebSphere Portal documents.
- Specify schedules for crawling WebSphere Portal documents.

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

"Integration with WebSphere Portal" on page 325

Related tasks

"Setting up enterprise search in WebSphere Portal version 6" on page 332

"Setting up enterprise search in WebSphere Portal version 5.1" on page 327

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 site that you want to crawl into the appropriate field when you configure a WebSphere Portal or Web Content Management crawler.

About this task

When you create a WebSphere Portal or Web Content Management crawler, you specify the URL of the site 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 crawler to crawl:

1. When you are ready to specify the URL or URLs to crawl in the enterprise search administration console, ensure that WebSphere Portal server is started and then log in to WebSphere Portal as an administrator.
2. If you use WebSphere Portal version 5.1, complete the following steps on the WebSphere Portal server:
 - a. Click **Administration** in the upper right corner.
 - b. Click **Portal Settings** in the navigation area on the left, and then click **Search Administration**.
 - c. On the Manage Search Collections page, click **PortalCollection** in the Search Collections area. You can select another collection, if other collections are available.
 - d. In the Content Sources in the Collection area, click **Add Content Source**.
 - e. For **Crawl source type**, click **Portal site**. The site URL is displayed in the **Collect documents linked from this URL** field.
 - f. Copy the URL to the clipboard. For example, highlight the URL and then hold the Ctrl key while you press the Insert key.
3. If you use WebSphere Portal version 6, complete the following steps on the WebSphere Portal server:
 - a. Click **Administration** in the lower left corner.
 - b. Click **Manage Search** from the navigation area on the left.
 - c. On the Manage Search page, click the **Search Collections** link.
 - d. On the Manage Search page, click **Default Portal Search Service** from the Search service options.
 - e. In the Search Collections table, click the **Portal Content** collection.
 - f. In the Content Sources table, click the Edit icon on the far right (the pencil icon) next to **Portal Content Source**.
 - g. For **Content source type**, click **Portal Site**. The site URL is displayed in the **Collect documents linked from this URL** field.
 - h. Copy the URL to the clipboard. For example, highlight the URL and then hold the Ctrl key while you press the Insert key.
4. Return to the enterprise search administration console and paste the URL that you copied into the 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 AIX, Linux, or Solaris systems, 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).

Tip:

For detailed examples of how to configure a secure Windows file system crawler, see the scenario for a small organization in the IBM Redbook, *IBM OmniFind Enterprise Edition Configuration and Implementation Scenarios*.

Crawling shared network directories

The Windows file system crawler crawls documents according to read permissions that are specified for the enterprise search administrator. The administrator is the OmniFind Enterprise Edition services account.

You can specify a user ID and password for the directories to be crawled. However, the user ID and password are used only to connect to shared network directories. The crawler crawls files according to the read permissions that are set for this user for the shared network directories, not for local drives.

Connections to network directories are not disconnected until you restart the OmniFind Enterprise Edition service. After a connection is established, it is possible to access the directory with an incorrect user ID and password. However, this connection is allowed only for the Windows file system discovery and crawler sessions that are under the control of the enterprise search system. To prevent possible security risks, ensure that authorizations for the enterprise search administrator's account (under which the OmniFind Enterprise Edition service runs) are set properly.

To avoid problems with connecting to a network directory in the future, specify the same user ID and password for the same network directory. If you specify the wrong user ID and password and restart the OmniFind Enterprise Edition service, the Windows file system crawler might fail to crawl because it is attempting to connect to the directory with incorrect credentials. The crawl can succeed if the network connection is established by another Windows file system crawler that is using the correct user ID and password.

Configuration overview

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 272

"Secure search of Windows trusted domains" on page 274

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.

Important: The Data Listener will not be supported in future releases. Use the search and index (SI-API) APIs instead of the Data Listener APIs to develop client applications for enterprise search. The following information is provided for users who previously created Data Listener applications.

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. Also specify the maximum number of documents, per collection, that can be held in temporary storage until the parser starts parsing them.
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 or the number of documents that can be held in temporary storage, restart the Data Listener:
 - a. Click  **Monitor** to change to the system monitoring view.
 - b. On the Data Listener page, click  **Restart**.

Related tasks

“Monitoring the Data Listener” on page 301

Custom crawler plug-ins

When you configure properties for crawlers, you can specify a Java class to use to enforce document-level access controls. You can also use the Java class to update the index by adding, modifying, or removing metadata and document content. By writing a plug-in, you can also extend the crawler’s ability to crawl archive files.

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, metadata, and document content. The class can return new or modified security tokens, metadata, and content, or the class can remove security tokens, metadata, and content.

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, metadata, or content of 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 rebuild the main index.

Plug-ins 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.

Plug-ins to add, modify, or remove metadata

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.

Plug-ins to add, modify, or remove document content

Document content comprises the parts of a document that contain searchable content and content that can become part of the dynamic document summary in the search results. The crawler plug-in allows you to apply your own business rules to determine the content that is to be indexed for each document.

Web crawler plug-ins

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.

For a discussion about form-based authentication and a sample program that you can adapt for your custom Web crawler plug-in, see <http://www.ibm.com/developerworks/db2/library/techarticle/dm-0707nishitani>.

Archive file plug-ins

By writing a plug-in, you can extend the crawlers and enable support for crawling archive file formats other than ZIP and TAR. For example, you can write a plug-in to support the crawling of documents in LZH format.

Unfenced mode

When you configure a non-Web crawler, you can select an option to run the plug-in in unfenced mode. In this mode, the plug-in process runs inside the crawler process, which can improve the plug-in performance.

Important: If the plug-in encounters a problem that is not recoverable when it runs in this mode, the crawler process might be terminated.

Related concepts

Support for crawling archive files

The enterprise search crawlers can extract files from an archive file (such as a ZIP or TAR file) so that individual files in the archive can be indexed and searched.

Supported archive file formats

The following archive file formats are supported:

Table 5. Archive file formats supported by enterprise search crawlers

File extension	MIME type	Data type	Notes
.zip, .ZIP	application/zip	zip	<ul style="list-style-type: none">• Depends on capabilities of the java.util.zip package• Supports deflated (method 8) compression:<ul style="list-style-type: none">– No support for encrypted files– No support for zip64
.tar	application/tar	tar	Supported tar formats: <ul style="list-style-type: none">• GNU tar 1.13• POSIX 1003.1-1998 (ustar)• POSIX 1003.1-2001 (pax)
.tar, .gz, .tgz	not applicable	tgz	Depends on capabilities of the java.util.zip package

Restrictions and guidelines

Automatic code page detection is not available for files that are extracted from an archive file. When extracting the files, the crawler uses the code page setting that it is configured to use with plain text and unknown document types. When you use the enterprise search administration console to configure language and code page settings for a crawler, you specify the code page that the crawler should use for plain text documents and for documents whose code page cannot be detected automatically.

To determine when files in an archive file need to be recrawled, the crawler uses the modified date in the archive entry header data for each file. When you monitor a crawler, the statistics that are shown for crawled documents, including statistics for inserted, updated, and deleted documents, include information about files that were extracted from archive files.

To enable crawlers to crawl archive files in other archive file formats, such as LZH files, you must write a crawler plug-in and then configure the crawler to use the plug-in.

Migration

To enable crawlers that existed prior to the installation of OmniFind Enterprise Edition Version 8.4 to crawl archive files, you need to edit the crawler's crawl space. Ensure that the settings for MIME types to exclude and file extensions to exclude do not contain references to .zip, .tar, .tgz, or .gz files.

Important: If you change these settings for a Content Edition crawler or DB2 Content Manager crawler, you must recrawl all documents so that the changes can be applied.

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.

Archive files

The URI format for documents that are extracted from an archive file (such as a .zip or .tar file) and then crawled is:

```
Original_URI(?|&)ArchiveEntry=Entry_Name(&ArchiveEntry=Entry_Name)
```

Parameters

Original_URI

The location of the archive file on the data source.

Entry_Name

The URL-encoded name of the archive entry in the archive file.

Examples

```
file:///d:/Archive1.zip  
file:///d:/Archive1.zip?ArchiveEntry=Folder1/PowerPoint.ppt  
file:///d:/Archive1.zip?ArchiveEntry=Folder2/Text.txt
```

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

URL encoding is applied to the *PID* parameter.

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+icmnlsdb12+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
```

JDBC database crawlers

The URI format for documents that are crawled by a JDBC database crawler is:

```
jdbc://DB_URL/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.

DB_URL The URL 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:

DB2 database:

```
jdbc:db2://host01.svl.ibm.com:50000/SAMPLE/DB2INST1.ORG/DEPTNUMB/51
```

Oracle database:

```
jdbc:oracle:thin:@/host01.svl.ibm.com:1521:ora/SCOTT.EMP/EMPNO/7934
```

MS SQL Server 2000 database:

```
jdbc:microsoft:sqlserver://host01.svl.ibm.com:1433;  
DatabaseName=Northwind/dbo.Region/RegionID/100
```

MS SQL Server 2005 database:

```
jdbc:sqlserver://host01.svl.ibm.com:1433;  
DatabaseName=Northwind/dbo.Region/RegionID/100
```

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
```

Seed list crawlers

The URI format for documents that are crawled by a Seed list crawler is:

`seedlist://Page_URL?pageID=Page_ID[&useSSL;=true]`

Parameters

URL encoding is applied to all of the fields.

Page_URL

The URL for the document (unique for each document).

Page_ID

The object identifier for the document.

useSSL When the protocol is HTTPS, `&useSSL;=true` is added to the URI. Otherwise, `useSSL` is omitted.

Example

HTTPS protocol:

```
seedlist://quickrserver.ibm.com:10035/1otus/my poc?uri=dm:bec6090046f1cd52bc5cfcb06e9f4550&verb;=view&pageID;=N1FSZUR1MkJQNjZSMZQMUMwM1FPNjZCQzY2SUw2SuhPNk1RQ0M2Uk80Nk9PNjVCRUM2UUs2TDFDMA==&useSSL;=true
```

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
```

Web Content Management crawlers

The URI format for WebSphere Content Management documents that are crawled by a Web Content Management crawler is:

`wcm://Page_URL?pageID=Page_ID[&useSSL=true]`

Parameters

URL encoding is applied to all of the fields.

Page_URL

The URL for the document (unique for each document).

Page_ID

The page identifier.

useSSL When the protocol is HTTPS, useSSL=true is added to the URI. Otherwise, useSSL is omitted.

Examples

HTTP protocol:

```
wcm://wp6server.ibm.com:9081/wps/wcm/myconnect/Web+Content
/Site01/SiteArea01/ContentTest01?pageID=
6QReDeJ9DI3R0663E03Q06L1E2MR47MHOC3Q862RD6J0863B0GJS86J9E0
```

HTTPS protocol:

```
wcm://wp6server.ibm.com:9444/wps/wcm/myconnect/Web+Content/Site01
/SiteArea01/ContentTest01?pageID=
6QReDeJ9DI3R0663E03Q06L1E2MR47MHOC3Q862RD6J0863B0GJS86J9E0&useSSL=true
```

WebSphere Portal crawlers: WebSphere Portal version 5

The URI format for WebSphere Portal version 5 documents that are crawled by a WebSphere Portal crawler is:

```
wps://Page_URL?portletDefID=Portlet_Def_ID&portletID=Portlet_ID
&pageID=Page_ID[&useSSL=true]
```

Parameters

URL encoding is applied to all of the fields.

Page_URL

The URL for the document (unique for each document).

Portlet_Def_ID

The portlet definition identifier.

Portlet_ID

The portlet identifier.

Page_ID

The page identifier.

useSSL When the protocol is HTTPS, useSSL=true is added to the URI. Otherwise, useSSL is omitted.

Examples

Document body:

```
wps://wpsserver.ibm.com:9081/wps/myportal!/ut/p/kcxml/04_Sj9SPykssy0x+
LKnPy1vM0Y_QjzKCN4g3cQbJgQio-pFQAW99X4_83FT9AP2C5IhyR0dFRQD8qHRj/delta
/base64xml/L01DU1kvd0NrQUpORUEvNFBVR0VoQSEvN18wXzZPLzZfMF80RA!!
?portletDefID=3_0_3S&pageID=6_0_6J
```

Examples

Enabled for SSL:

```
wps://wpsserver.ibm.com:9081/wps/myportal!/ut/p/kcxml/04_Sj9SPykssy0x+
LKnPy1vM0Y_QjzKCN4g3cQbJgQio-pFQAW99X4_83FT9AP2C5IhyR0dFRQD8qHRj/delta
/base64xml/L01DU1kvd0NrQUpORUEvNFBVR0VoQSEvN18wXzZPLzZfMF80RA!!
?portletDefID=7_0_A4&pageID=6_0_6J&useSSL=true
```

WebSphere Portal crawlers: WebSphere Portal version 6

The URI format for WebSphere Portal version 6 documents that are crawled by a WebSphere Portal crawler is:

```
wp6://Page_URL?portletURL=Portlet_URL?portletDefID=Portlet_Def_ID
&pageID=Page_ID[&useSSL=true]
```

Parameters

URL encoding is applied to all of the fields.

Page_URL

The URL for the document (unique for each document).

Portlet_URL

The unique URL for the document.

Portlet_Def_ID

The portlet definition identifier.

Page_ID

The page identifier.

useSSL When the protocol is HTTPS, useSSL=true is added to the URI. Otherwise, useSSL is omitted.

Examples

HTTP protocol:

```
wp6://wp6server.ibm.com:9081/wps/myportal!/ut/p/c1/04_SB8K8xLLM9MSSzPy
8xBz9CP0os3gjE59gQwMLQ0P_IDMnAyNHA3f3UESTD1NjA6B8pFm8AQ7gaEBAAdjIPrz6_
Tzyc1P1C3IjDHQdFRUBTu-saA!!/d12/d0/Y2BkbGBgY1rDwMDEJ1XAwMggYxZvZOITbGh
gYWjobuhmaGDka0Bu5uHqFRpkaAAAEisaBQ!!
?portletUrl=/wps/myportal!/ut/p/c1/04_SB8K8xLLM9MSSzPy8xBz9CP0os3gjE59
gQwMLQ0P_IDMnAyNHA3f3UESTD1NjA6B8pFm8AQ7gaEBAAdjIPrz6_Tzyc1P1C3IjDHQdF
RUBTu-saA!!/d12/d0/Y2BiUZnBwMqsyBykYGBmS2tcDoDe40MebyRiU-woYGFoaG7oZu
hgZGjgbuZh6uXY4ihAYOMGQ7Z0CDBAwDcXPkM
&portletDefID=6_24LS10811G1F102A0G6HEJUR10
&pageID=3_24LS108110R6B02A0GGU94LN00
```

HTTPS protocol:

```
wp6://wp6server.ibm.com:9444/wps/myportal!/ut/p/c1/04_SB8K8xLLM9MSSzPy
8xBz9CP0os3gjE59gQwMLQ0P_IDMnAyNHA3f3UESTD1NjA6B8pFm8AQ7gaEBAAdjIPrz6_
Tzyc1P1C3IjDHQdFRUBTu-saA!!/d12/d0/Y2BkbGBgY1rDwMDEJ1XAwMggYxZvZOITbGh
gYWjobuhmaGDka0Bu5uHq5RtqaAAA50L41Q!!
?portletUrl=/wps/myportal!/ut/p/c1/04_SB8K8xLLM9MSSzPy8xBz9CP0os3gjE59
gQwMLQ0P_IDMnAyNHA3f3UESTD1NjA6B8pFm8AQ7gaEBAAdjIPrz6_Tzyc1P1C3IjDHQdF
RUBTu-saA!!/d12/d0/Y2BiUZnBwMqsyBykYGBmS2tcDoDe40MebyRiU-woYGFoaG7oZu
hgZGjgbuZh6uXWZihAYOMGQ5Z31BDAAwAk73P2
&portletDefID=6_24LS10811G1F102A0G6HEJMU10
&pageID=3_24LS108110R6B02A0GGU94T410&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

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](#)
-  [Text analysis included in enterprise search](#)
-  [Basic concepts used in text analysis processing](#)
-  [Semantic search applications](#)

 Semantic search query terms

“Language and code page support” on page 159

“Document format detection” on page 145

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.

When you configure a category, you specify rules that instruct the crawler to associate documents with the category. You can group documents that share a URI pattern or group documents that contain specific content (for example, documents that contain or exclude specific words and phrases).

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 or use rule-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 is degraded until you recrawl all documents and rebuild the main 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 is degraded until you recrawl all documents and rebuild the main index.

Rule-based categories

You can configure rules to control which documents are associated with categories in an enterprise search collection.

You can create 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 options 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 rules

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 rules

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 the title of a document contains the word `"health"` but not the phrase `"employee benefits"`, the document passes the rule:

```
+title:health -title:"employee 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 in content rules:

- + 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.

field_name:

Precede a term or phrase with a field name to indicate that the term or phrase must (or must not) occur in the specified document field.

All content and metadata fields that are configured to be searchable fields in the collection are supported. The following SI-API field keywords and field types are not supported:

- site:
- url:
- link:
- docID:
- samegroupas:
- parametric fields
- security tokens
- attributes (such as \$source, \$language, \$doctype, and so on)

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.

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 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 then rebuild the main 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 categorize documents according to rules that you configure specifically for this collection.

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 and rebuild the main 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**.

- To use the URI of a document to determine 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`

- To determine whether a document belongs to the category by querying searchable content, click **Document content**, select the language of the documents, and then specify the words and phrases that must or must not appear in the document content.

You express the rule in the same format as a query, but only the include (+), exclude (-), phrase (" "), and field name (*field_name*:) query operators are allowed. N-gram segmentation is not supported with content rules.

If a document includes or excludes the words that you specify, the document is associated with the category.

For example: `+finance -accounting +title:"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 reference

“URI formats in an enterprise search index” on page 113

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.

Typically, all text data in an XML document is indexed. By mapping XML elements to search fields, you can use the structure information of XML documents to support more specific queries. For example, you can make the data in an XML element searchable by field name and returnable in the search results by mapping the XML element name to a field name. You can make the mapping more precise by specifying attributes of the XML element (and the values of those attributes) as criteria for the element to become a search field.

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.

With enterprise search, you can associate, or map, XML elements to 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 the values of those elements by specifying the field names in queries. Queries that search named 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 directly query these elements. A search for author:Smith finds XML documents that have Smith in elements that are mapped to a field named author.

For another example, an XML element named <summary> might contain information that is useful to show in the search results. If you map the <summary> element to a search field, and specify that the value of this element is to be shown in the search results, the content of the element is part of the result document.

When you map an XML element to a field name, text within the element is searchable under the field name that you specify. If the XML element includes attributes, however, the attribute values are not indexed and not directly searchable. To query the attribute values, you must configure parsing options in

the enterprise search administration console and enable native XML search. After you enable native XML search, the structure of the XML document is available in the index and you can query the document by specifying XPath query constraints. For example:

```
@xml:elementName[@attributeName="attributeValue"]
```

Related concepts

 Semantic search query term

Related tasks

“Enabling support for native XML search” on page 142

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.

Restrictions

Multiple XML field mappings can exist per collection, but only one XML root element mapping. The root element of an XML document applies the mappings accordingly.

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 the parser is restarted. The new and changed mappings apply to new data that is parsed after the parser is restarted, and have no effect on data that is already in the index. To update documents that are already in the index, you must crawl and index the documents again.

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`.

Tip: Click **Help** for information about ASCII characters and metacharacters that are not supported in field names.
 - b. In the **XML element name** field, type `jobrole`.
 - c. To enable users to query the `jobrole` field, select the **Fielded search** check box.
 - d. To ensure that a match occurs only when the search terms match the entire value of the `jobrole` field (that is, no other words exist in the field), select the **Complete match** check box.
 - e. To enable users to sort the search results by the values in the `jobrole` field, select the **Sortable** check box.
The field is shown as a sortable field only if the search application supports this capability. The sample search application for enterprise search does not include fields that are mapped from XML elements in the list of fields that you can select for sorting search results.
 - f. To enable users to view the values of the `jobrole` field in the search results, select the **Search results** check box.
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 `jobrole`.
 - c. In the **XML element name** field, type `jobroles`.

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, specify that the search terms must completely match the field value, use the field to sort search results, and view the field in the search results, select the appropriate check boxes.
7. Map the XML element `section`, which has an attribute named `ID` that contains the value `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 **XML element 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, select the **Fielded search** check box.
- g. To ensure that a match occurs only when the search terms match the entire value of the expertise field (that is, no other words exist in the field), select the **Complete match** check box.
- h. To enable users to sort the search results by the values in the expertise field, if the search application supports this capability, select the **Sortable** check box.
- i. To enable users to view the expertise field values in the search results, select the **Search results** check box.

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 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.

By mapping HTML metadata elements to search fields, you enable users to search HTML documents with more precise queries.

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.

When you map HTML metadata elements to search fields, you specify whether you want to map all HTML metadata elements, only the elements that belong to the Dublin Core set of metadata elements, or only the HTML metadata elements that you specify. For a description of the elements in the Dublin Core metadata element set, see the Dublin Core Metadata Initiative web site:

<http://dublincore.org/documents/dcmi-terms/#H2>

If you choose to create mappings for all HTML metadata elements or all Dublin Core metadata elements, the default search field name matches the metadata element name. You can override the default search field name and specify different search options for specific elements by adding the elements to the list of elements that you want to configure individually.

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 the parser is restarted. The new and changed mappings apply to new data that is parsed after the parser is restarted, and have no effect on data that is already in the index. To update documents that are already in the index, you must crawl and index the documents again.

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, specify which HTML metadata elements you want to map to search fields:
 - If you map all HTML elements or all elements that belong to the Dublin Core element set, you specify whether you want users to be able to search the fields by field name and whether the fields should be displayed in the search results. You also specify whether the user's search terms must completely match the entire metadata element value (that is, the only words in the field are words that match the user's search terms), and whether the users can sort the search results by this field.

The system automatically uses the metadata element names as the field names. If you want to override the default field names for specific elements,

or use different search options with specific elements, you can map an element name to a unique field name and then specify the search options that you want to use with that element.

- To map specific metadata elements to search fields:
 - a. Click **Add Field** to add a blank line to the list of field mapping rules.
 - b. 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.

Tip: Click **Help** for information about ASCII characters and metacharacters that are not supported in field names.

- c. Type the name of the metadata element that you want to map.
- d. To enable users to query this field, select the **Fielded search** check box.

Tip: If this check box is clear, the field cannot be searched with a fielded query or with a free text query. This action might be useful, for example, if you selected the option to include all HTML metadata elements or the Dublin Core metadata elements, but want to prevent certain fields from being searched.

- e. To enable users to query this field and specify that a document matches only when the query terms match the entire field value (that is, no other words exist in the field), select the **Complete match** check box.
- f. To enable users to sort search results alphabetically (string sort) by this field, select the **Sortable** check box.

The field is shown as a sortable field only if the search application supports this capability. The sample search application for enterprise search does not include fields that are mapped from HTML metadata elements in the list of fields that you can select for sorting search results.

- g. 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 or to sort results numerically according to the value of this field, select the **Parametric search** check box.
- h. To enable users to view this field in the search results, select the **Search results** check box.

Example:

Users can 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 HTML documents that discuss 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.

OmniFind Enterprise 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 mapping files with the collection and map the output of the XML mapping to the 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 the 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 name Alex and a phone number).
- Map the common analysis structure to a relational database. You can map data to IBM DB2 tables 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](#)
-  [Basic concepts used in text analysis processing](#)
-  [Workflow for custom analysis integration](#)
-  [Text analysis algorithms](#)
-  [Semantic search applications](#)

 Semantic search query term

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 local computer and you can browse to locate the file. If the file is larger than 8 MB, the file must be on the index server and you must type the fully qualified path for the file.
7. Click **OK**. Your text analysis engine is listed on the Text Analysis Engines page.

Related concepts

 Workflow for custom analysis integration

 Custom text analysis integration

 Basic concepts used in text analysis processing

 XML markup in analysis and search

Related tasks

 Creating an XML elements to the common analysis structure mapping file

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.

Related concepts

 Workflow for custom analysis integration

 Custom text analysis integration

 Basic concepts used in text analysis processing

Mapping XML elements to the 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 relational database tables with semantic search, you can map the XML elements to the common analysis structure.

Before you begin

To map XML elements to the common analysis structure, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

Restrictions

The maximum size of a mapping file is 8 MB.

About this task

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 mapping files. The mapping files must adhere to the UIMA framework for text analysis.

When you add 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 map 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 configure any mapping files. Instead, you can configure parsing options and enable native XML search for the collection.

Procedure


To map XML elements to the common analysis structure:


1. Edit a collection, select the Parse page, and click **Configure text processing options**.
2. In the **Map XML elements to the common analysis structure** area, click **Add Mapping**.
3. On the Map XML Elements to the Common Analysis Structure page, type a descriptive display name for the mapping file.
4. Specify the location of the file. If the mapping file is on your local system, you can browse to locate the path. If the mapping file is on the index server, you must type the fully qualified path.
5. Click **OK**. Your new mapping file is added to the Text Processing Options page.

Related concepts

 [Workflow for custom analysis integration](#)

 [Custom text analysis integration](#)

 [Basic concepts used in text analysis processing](#)

 XML markup in analysis and search

Related tasks

 Creating an XML elements to the common analysis structure mapping file

Mapping the common analysis structure to the index

You can specify which text analysis results from a common analysis structure are to be mapped to the index and be available to users who query a collection by using semantic search.

Before you begin

To map the 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.

Restrictions

The maximum size of a mapping file is 8 MB.

About this task

By mapping the 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 name Alex and a phone number).

Procedure

To map the common analysis structure to the index:


1. Edit a collection, select the Parse page, and click **Configure text processing options**.
2. In the **Map the common analysis structure to the index** area, click **Select a mapping file**.
3. On the Select a Mapping File for this Collection page, select the mapping file that you want to use with the enterprise search index:
 - To use the default mapping rules with the enterprise search index, select **Default**.
 - To map a custom common analysis structure to the index, specify the location of the mapping file. If the file is on your local system you can browse to locate the file. If the file is on the index server, type the fully qualified path.
4. Click **OK**. The mapping file that you specified is displayed on the Text Processing Options page.

Related concepts


 Workflow for custom analysis integration

 Custom text analysis integration

 Basic concepts used in text analysis processing

 Index mapping for custom analysis results

Related tasks

 Creating the common analysis structure to index mapping file

Mapping the common analysis structure to a relational database

You can specify which text analysis results from a common analysis structure are to be mapped to a relational database for use in database applications.

Before you begin

To map the common analysis structure to a relational database, you must be a member of the enterprise search administrator role or be a collection administrator for the collection.

Restrictions

The maximum size of a mapping file is 8 MB.

About this task

By mapping the common analysis structure to relational database 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 text analysis. You can also use the information for further text processing, such as using the information in data mining applications.

You can use one mapping file at a time to map a common analysis structure to a relational database. If you upload a new mapping file, the current mapping file is overwritten.




Procedure

To map the common analysis structure to a relational database:


1. Edit a collection, select the Parse page, and click **Configure text processing options**.
2. In the **Map the common analysis structure to a relational database** area, click **Add Mapping**.
3. On the Map the Common Analysis Structure to a Relational Database page, type a descriptive display name for the mapping file that you want to use to map information to a relational database.
4. Specify the location of the mapping file. If the file is on your local system you can browse to locate the file. If the file is on the index server, type the fully qualified path.
5. Click **OK**. The display name for the mapping file is shown on the Text Processing Options page.

Related concepts

 Workflow for custom analysis integration

-  Custom text analysis integration
-  Basic concepts used in text analysis processing
-  Database mapping for selected analysis results

Related tasks

-  Creating the common analysis structure to database mapping file

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 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.

For another example, the compound `Mustermann` is split into two tokens (`muster` and `mann`) that are stored separately in the index. When the wildcard query `Musterma*` is entered, the search processes cannot identify `Musterma` as a prefix of a decomposed word. As a result, documents with the term `Mustermann` are not found. If you want users to be able to enter wildcard queries for compound terms, do not enable advanced analysis of compound terms.

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

“Wildcard characters in queries” on page 176

 Linguistic support for semantic search

 Text analysis included in enterprise search

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. 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
-  Text analysis included in enterprise search
-  Semantic search applications
-  Semantic search query term
“XML search fields” on page 129

Document format detection

For enterprise search, a default mapping of URL extension and MIME type is used to determine document types and the parser type to use with each document.

By editing the `parserTypes.cfg` configuration file, you can override and extend the default mapping of URL extensions and MIME types to parser types. The `parserTypes.cfg` file defines rules for mapping file extensions or MIME types to parser types. For example, you can map a file extension such as `.content` and specify that documents of that type are to be parsed by the HTML parser.

Different document formats have different internal representations. An enterprise search system uses internal and third party filters for parsing documents, and many documents are parsed with parser services that are specialized for a specific format.

Document format detection and parser assignment occurs in the following way:

1. The algorithm for detecting the document format checks the extension of the URL of the processed document.
2. The system checks the MIME type of the document, which is part of the metadata that is set by the crawler.
3. The system tries to assign the appropriate parser type to each document. For HTML, text (TXT), and XML documents, the system assigns a parser type that is specific to each document format.

For all other document formats, the system uses the Stellant parser. Note that the Stellant document filtering technology is now owned by Oracle. References to Stellant in this documentation are synonymous with references to Oracle Outside In Content Access technology.

The Stellant parser supports several hundred document formats, but only a subset of the document filters are enabled for enterprise search. You can edit configuration files, however, to allow other document types to be parsed by the Stellant parser.

Important: Document filters that you add that do not belong to the subset of document filters that are enabled for enterprise search in the default system configuration have not been tested and are not supported.

4. If a `parserTypes.cfg` file is not available, the default mapping is used to determine the document type and which parser to use. To determine the document type, the system does the following steps:
 - a. Compares the URL extension to customer-defined extension rules in the `parserTypes.cfg` file.
 - b. Compares the MIME type to customer-defined MIME type rules in the `parserTypes.cfg` file.
 - c. Compares the URL extension to the default rules for enterprise search.
 - d. Compares the MIME type to default MIME type rules for enterprise search.
5. If the system cannot identify the document format of a document, the document is rejected. You might see an error message that states that the document type is not supported.

If Stellant is assigned as the parser type, you might see an error message if Stellant cannot recognize the document format. The error can occur if:

- The document is corrupted.
- The document is not in a format that Stellent supports. To solve this problem, you need to add the rejected file formats to the `stellentTypes.cfg` file. You also need to update the `parserTypes.cfg` file to specify that the MIME type or extension of the rejected document formats are to be associated with the Stellent parser.

Default supported document types

When detecting the document format, only certain document types are evaluated.

The following document formats are native types that are detected and parsed automatically by built-in, collection parser services:

- HTML
- Plain text
- XML

By default, the following document formats are parsed by the Stellent parser:

- Adobe Portable Document Format (PDF)
- Lotus 1-2-3[®]
- Lotus Freelance Graphics[®]
- Lotus Word Pro[®]
- Just System Ichitaro
- Microsoft Excel (versions through 2007)
- Microsoft PowerPoint (versions through 2007)
- Microsoft Visio
- Microsoft Word (versions through 2007)
- Rich Text Format (RTF)
- StarOffice/OpenOffice Calc
- StarOffice/OpenOffice Impress
- StarOffice/OpenOffice Draw
- StarOffice/OpenOffice Writer

Microsoft Office Open XML file formats and OpenOffice.org OpenDocument formats are handled without the need to make changes to the configuration files.

To parse other types of documents, you must update configuration files (`parserTypes.cfg` and `stellenttypes.cfg`) to specify rules for mapping specific document types to a collection parser service or Stellent filter.

Restriction: Processing bidirectional text in PDF files to match the logical reading order of the text is beyond the scope of the Stellent viewer technology. The Stellent parser makes no guarantee about the order of text that it extracts from PDF files. With bidirectional text in PDF files, the order in which the text is parsed is likely to not match the logical reading order of the text. This limitation causes a problem when processing PDF files that are written in Middle Eastern languages such as Hebrew and Arabic, which are written predominantly right-to-left (bidirectional).

Document types associated with collection parsers and Stellent parsers

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 Stellent document filters.

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 parser in error, and then sent back to one of the built-in parsers, a situation that can affect performance. For other documents, it might not be possible to detect the document type, so the documents are skipped. 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 parser 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 parser. 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 `parserTypes.cfg` 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 URL extension and which MIME type is mapped to which parser type.
- How to parse documents whose type is unknown because of incomplete metadata.

The format of the `parserTypes.cfg` 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 *text*, *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 OmniFind Enterprise 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 parser.

```
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 jttdc stellent
EXTENSION jtddc stellent
EXTENSION jtddx stellent
EXTENSION ps none
EXTENSION xml xml
EXTENSION txt text
EXTENSION htm html
EXTENSION htm1 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 none
DEFAULT none
```

Parsing unknown document types

If a document type is unknown (for example, if a document does not have a file extension or there is no MIME type associated with the document), you can configure rules to prevent the parser from dropping the document.

About this task

If the parser does not recognize a file format, the parser attempts to parse the document with the default HTML parser. If the content is not in HTML format, the parser rejects the document. A record of all rejected documents is written to the following location on the index server:

```
ES_NODE_ROOT/data/collection_ID/dropped_doc_logs/dropped_docs_pd_date.log
```

Procedure

To avoid this situation, configure the parser to use the ASCII parser for unknown document types:

1. On the index server, log in as the enterprise search administrator. This user ID was specified when OmniFind Enterprise Edition was installed.
2. Create or edit the following parser configuration file, where *collection_ID* identifies the collection that you want to configure:
`ES_NODE_ROOT/master_config/collection_ID.parserdriver/parserTypes.cfg`
3. Add the following rule to the file:
`UNKNOWN text`
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.

Changing the replacement rules for some HTML tags

You can change the HTML tag replacement rules that the parser uses when parsing HTML documents. 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

To provide text information for the enterprise search index, the HTML parser replaces the markup information (HTML tags) with other characters that model the meaning of the tags in the same way that a Web browser handles them. For example, a paragraph tag (<p>) results in a paragraph delimiter in the Common Analysis Structure (CAS) and in the index. The replacement rules influence which parts of the text information appear in the same paragraph.

To tie the meaning of certain HTML tags more closely to the HTML standard, you can update the parser configuration file for a collection and change some of the replacement rules.

Tip: To learn about other ways that you can control how HTML tags are handled in HTML documents, see <http://www.ibm.com/support/docview.wss?rs=63&uid=swg27011251>.

Procedure

To change the HTML tag replacement rules:

1. On the index server, log in as the enterprise search administrator. This user ID was specified when OmniFind Enterprise Edition was installed.
2. Edit the `ES_NODE_ROOT/master_config/collection_ID.parserdriver/collection.properties` file, where *collection_ID* identifies the collection that you want to configure.
3. Set the following parameter to true:
`trevi.tokenizer.newHtmlTagReplacement=true`

The replacement rules for the following HTML tags are changed to the values shown below. All other HTML tag replacement rules continue to use the default rules.

```
dfn  EMPTY
div  NEW LINE
q    BLANK
span EMPTY
```

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.

Default HTML replacement rules

The HTML parser interprets and replaces markup information in HTML documents to provide text information for the enterprise search index.

The HTML parser for enterprise search uses the following replacement rules when parsing HTML tags. The first column shows the HTML tag name. The second column shows the replacement value.

```
comment BLANK
a        EMPTY
applet   EMPTY
area     EMPTY
b        EMPTY
base     EMPTY
big      EMPTY
body     EMPTY
br       NEW LINE
caption  EMPTY
center   EMPTY
del      EMPTY
dfn      PARAGRAPH
dir      PARAGRAPH
div      BLANK
dl       PARAGRAPH
em       EMPTY
form     EMPTY
frame    PARAGRAPH
h1       PARAGRAPH
h2       PARAGRAPH
h3       PARAGRAPH
h4       PARAGRAPH
h5       PARAGRAPH
h6       PARAGRAPH
hr       PARAGRAPH
iframe   EMPTY
img      NEW LINE
li       BLANK
meta     EMPTY
object   EMPTY
ol       BLANK
option   EMPTY
p        PARAGRAPH
q        PARAGRAPH
samp     PARAGRAPH
script   EMPTY
select   EMPTY
spacer   BLANK
span     NEW LINE
strike   EMPTY
strong   EMPTY
style    EMPTY
table    PARAGRAPH
td       NEW LINE
```

title	PARAGRAPH
tr	NEW LINE
ul	BLANK
xmp	PARAGRAPH

Associating document types with a Stellent parser

To specify which types of documents are to be parsed by Stellent document filters, you create a `stellenttypes.cfg` configuration file. There is no support for this task in the enterprise search administration console.

Oracle Outside In Technology:

The Stellent document filtering technology is now owned by Oracle. References to Stellent in this documentation are synonymous with references to Oracle Outside In Content Access technology.

Restrictions

OmniFind Enterprise Edition supports Stellent filters for the following document types:

- Adobe Portable Document Format (PDF)
- Lotus 1-2-3
- Lotus Freelance Graphics
- Lotus Word Pro
- Ichitaro
- Microsoft Excel
- Microsoft PowerPoint
- Microsoft Visio
- Microsoft Word
- Rich Text Format (RTF)

If you want to include additional types of documents in an enterprise search index, and the document formats are supported through a Stellent filter, you can configure parsing rules in the `stellenttypes.cfg` configuration file.

Important: Document formats that you add that do not match the document types in the preceding list have not been tested for enterprise search and are not supported.

For a complete list of Stellent document formats, see *Outside In Technology: Supported File Formats* at http://www.oracle.com/technology/products/content-management/oit/ds_oitFiles.pdf.

Before you begin

To complete this task, you must log in as an enterprise search administrator.

About this task

The `stellenttypes.cfg` configuration file specifies:

- Accept rules, for the file types that are to be parsed by the Stellent parser. A file type corresponds to one of the file types recognized by the Stellent library.

- Native rules, for the file types that 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 parser in error, due to misleading metadata.
- Reject rules, for the file types that 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 parsers 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 a rule that matches one of the following formats:

```
accept DEFAULT
accept ALL doc_type
accept stellent_type doc_type
native DEFAULT
native stellent_type doctype
reject stellent_type
```

Where:

doc_type

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.

stellent_type

Is one of the filter type values in the Stellent library, such as `FI_123R1`.

DEFAULT

Means that the list of accepted or native types, depending on the type of the rule, includes all of the default rules. 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:

1. If there is a reject rule for *stellent_type*, the document is not accepted.
2. If there is a native rule for *stellent_type* (including the default parsing rules if native DEFAULT is specified), the document is sent back to the built-in parser in addition to the value for the *doc_type* token that is specified by this rule. The value of *doc_type* must be either `txt`, `htm` or `xml`, indicating plain text, HTML or XML, respectively.
3. If there is an accept rule for *stellent_type* (including the default list if accept DEFAULT is specified), the document is accepted.
4. Else, if accept ALL is specified, the document is accepted.
5. Otherwise, the document is rejected and will not be parsed.

If the document type is accepted, then the *doc_type* 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.

Procedure

To associate document types with the Stellent parser:

1. On the index server, log in as the enterprise search administrator. This user ID was specified when OmniFind Enterprise 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.

Examples

In the following configuration file, the Stellent session accepts documents in Microsoft Visio format in addition to the default list of supported document types.

```
accept DEFAULT
accept FI_VISIO3 visio
accept FI_VISIO4 visio
accept FI_VISIO5 visio
accept FI_VISIO6 visio
```

In the following configuration file, Postscript documents are accepted and searchable with a document type of `ps`; documents in X pixmap format (XPM) are sent back to the built-in text parser; documents in the PNG image format are rejected; and all other file types are accepted and made searchable with a document type of `other`.

```
accept DEFAULT
accept FI_POSTSCRIPT ps
native FI_XPIXMAP txt
reject FI_PNG
accept ALL other
```

Default parsing rules for Stellent parsers

If you do not create a configuration file to map file types to Stellent document filters, the parser uses default rules to parse documents.

The Stellent parser accepts and parses the following document types:

```
ACCEPT FI_123R1          123
ACCEPT FI_123R2          123
ACCEPT FI_123R3          123
ACCEPT FI_123R4          123
ACCEPT FI_123R6          123
ACCEPT FI_123R9          123
ACCEPT FI_EXCEL          xls
ACCEPT FI_EXCEL2000     xls
```

ACCEPT FI_EXCEL2002	xls
ACCEPT FI_EXCEL2003	xls
ACCEPT FI_EXCEL2007	xlsx
ACCEPT FI_EXCEL3	xls
ACCEPT FI_EXCEL4	xls
ACCEPT FI_EXCEL5	xls
ACCEPT FI_EXCEL97	xls
ACCEPT FI_EXTPOWERPOINT4	ppt
ACCEPT FI_EXTPOWERPOINTMAC4	ppt
ACCEPT FI_FREELANCE	prz
ACCEPT FI_FREELANCE3	prz
ACCEPT FI_ICHITAR03	jxw
ACCEPT FI_ICHITAR04	jsw
ACCEPT FI_ICHITAR08	jtd
ACCEPT FI_PDF	pdf
ACCEPT FI_PDFMACBIN	pdf
ACCEPT FI_POWERPOINT2	ppt
ACCEPT FI_POWERPOINT2000	ppt
ACCEPT FI_POWERPOINT2007	pptx
ACCEPT FI_POWERPOINT3	ppt
ACCEPT FI_POWERPOINT4	ppt
ACCEPT FI_POWERPOINT7	ppt
ACCEPT FI_POWERPOINT9597	ppt
ACCEPT FI_POWERPOINT97	ppt
ACCEPT FI_POWERPOINTMAC3	ppt
ACCEPT FI_POWERPOINTMAC4	ppt
ACCEPT FI_POWERPOINTMACB3	ppt
ACCEPT FI_POWERPOINTMACB4	ppt
ACCEPT FI_RTF	rtf
ACCEPT FI_RTFJ	rtf
ACCEPT FI_STAROFFICEWRITER8	odt
ACCEPT FI_STAROFFICEDRAW8	odg
ACCEPT FI_STAROFFICEIMPRESS8	odp
ACCEPT FI_STAROFFICECALC8	ods
ACCEPT FI_STAROFFICECALC6	sxc
ACCEPT FI_STAROFFICEDRAW6	sxd
ACCEPT FI_STAROFFICEIMPRESS6	sxi
ACCEPT FI_STAROFFICEWRITER6	sxw
ACCEPT FI_STAROFFICECALC52	sdc
ACCEPT FI_STAROFFICEIMPRESS52	sdd
ACCEPT FI_STAROFFICEWRITER52	sdw
ACCEPT FI_VISIO3	vsd
ACCEPT FI_VISIO4	vsd
ACCEPT FI_VISIO5	vsd
ACCEPT FI_VISIO6	vsd
ACCEPT FI_VISIO2003	vsd
ACCEPT FI_WINWORD1	doc
ACCEPT FI_WINWORD1COMPLEX	doc
ACCEPT FI_WINWORD1J	doc
ACCEPT FI_WINWORD2	doc
ACCEPT FI_WINWORD2000	doc
ACCEPT FI_WINWORD2002	doc
ACCEPT FI_WINWORD2003	doc
ACCEPT FI_WINWORD2007	docx
ACCEPT FI_WINWORD5J	doc
ACCEPT FI_WINWORD6	doc
ACCEPT FI_WINWORD7	doc
ACCEPT FI_WINWORD97	doc
ACCEPT FI_WORD4	doc
ACCEPT FI_WORD5	doc
ACCEPT FI_WORD6	doc
ACCEPT FI_WORDPRO	lwp
ACCEPT FI_WORDPRO97	lwp

The Stellent parser returns the following document types to the collection parser for processing with one of the built-in parsers:

NATIVE FI_7BITTEXT	txt
NATIVE FI_ANSI	txt
NATIVE FI_ANSI8	txt
NATIVE FI_ARABIC_710	txt
NATIVE FI_ARABIC_720	txt
NATIVE FI_ARABIC_WINDOWS	txt
NATIVE FI_ASCII	txt
NATIVE FI_ASCII8	txt
NATIVE FI_CENTRALEU_1250	txt
NATIVE FI_CHINESEBIG5	txt
NATIVE FI_CHINESEGB	txt
NATIVE FI_CYRILLIC1251	txt
NATIVE FI_CYRILLICKO18	txt
NATIVE FI_EBCDIC_1026	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_37	txt
NATIVE FI_EBCDIC_500	txt
NATIVE FI_EBCDIC_870	txt
NATIVE FI_EBCDIC_871	txt
NATIVE FI_HANGEUL	txt
NATIVE FI_HEBREW_E0	txt
NATIVE FI_HEBREW_OLDCODE	txt
NATIVE FI_HEBREW_PC8	txt
NATIVE FI_HEBREW_WINDOWS	txt
NATIVE FI_HTML	htm
NATIVE FI_HTML_ARABIC_ASMO708	htm
NATIVE FI_HTML_ARABIC_DOS	htm
NATIVE FI_HTML_ARABIC_ISO	htm
NATIVE FI_HTML_ARABIC_MAC	htm
NATIVE FI_HTML_ARABIC_WINDOWS	htm
NATIVE FI_HTML_BALTIC_ISO	htm
NATIVE FI_HTML_BALTIC_WINDOWS	htm
NATIVE FI_HTML_CENTRALEUROPEAN_DOS	htm
NATIVE FI_HTML_CENTRALEUROPEAN_ISO	htm
NATIVE FI_HTML_CENTRALEUROPEAN_MAC	htm
NATIVE FI_HTML_CENTRALEUROPEAN_WINDOWS	htm
NATIVE FI_HTML_CHINESEBIG5	htm
NATIVE FI_HTML_CHINESEEUC	htm
NATIVE FI_HTML_CHINESEGB	htm
NATIVE FI_HTML_CHINESESIMPLIFIED_EUC	htm
NATIVE FI_HTML_CHINESESIMPLIFIED_WINDOWS	htm
NATIVE FI_HTML_CHINESETRADITIONAL_WINDOWS	htm
NATIVE FI_HTML_CYRILLIC_DOS	htm
NATIVE FI_HTML_CYRILLIC_ISO	htm
NATIVE FI_HTML_CYRILLIC_KO18R	htm
NATIVE FI_HTML_CYRILLIC_MAC	htm
NATIVE FI_HTML_CYRILLIC_WINDOWS	htm
NATIVE FI_HTML_CYRILLIC1251	htm
NATIVE FI_HTML_CYRILLICKO18	htm
NATIVE FI_HTML_EBCDIC_1026	htm
NATIVE FI_HTML_EBCDIC_273	htm
NATIVE FI_HTML_EBCDIC_277	htm
NATIVE FI_HTML_EBCDIC_278	htm
NATIVE FI_HTML_EBCDIC_280	htm
NATIVE FI_HTML_EBCDIC_284	htm
NATIVE FI_HTML_EBCDIC_285	htm
NATIVE FI_HTML_EBCDIC_297	htm
NATIVE FI_HTML_EBCDIC_37	htm
NATIVE FI_HTML_EBCDIC_500	htm
NATIVE FI_HTML_EBCDIC_870	htm
NATIVE FI_HTML_EBCDIC_871	htm

NATIVE FI_HTML_GREEK_ISO	htm
NATIVE FI_HTML_GREEK_MAC	htm
NATIVE FI_HTML_GREEK_WINDOWS	htm
NATIVE FI_HTML_HEBREW_DOS	htm
NATIVE FI_HTML_HEBREW_ISO_VISUAL	htm
NATIVE FI_HTML_HEBREW_WINDOWS	htm
NATIVE FI_HTML_JAPANESE_MAC	htm
NATIVE FI_HTML_JAPANESE_SHIFTJIS	htm
NATIVE FI_HTML_JAPANESE_EUC	htm
NATIVE FI_HTML_JAPANESE_JIS	htm
NATIVE FI_HTML_JAPANESE_SJIS	htm
NATIVE FI_HTML_KOREAN_JOHAB	htm
NATIVE FI_HTML_KOREAN_WINDOWS	htm
NATIVE FI_HTML_KOREAN_HANGUL	htm
NATIVE FI_HTML_LATIN2	htm
NATIVE FI_HTML_RUSSIAN_DOS	htm
NATIVE FI_HTML_THAI_WINDOWS	htm
NATIVE FI_HTML_TURKISH_DOS	htm
NATIVE FI_HTML_TURKISH_ISO	htm
NATIVE FI_HTML_TURKISH_MAC	htm
NATIVE FI_HTML_TURKISH_WINDOWS	htm
NATIVE FI_HTML_VIETNAMESE_WINDOWS	htm
NATIVE FI_HTML_WESTERNEUROPEAN_ISO	htm
NATIVE FI_HTML_WESTERNEUROPEAN_MAC	htm
NATIVE FI_HTML_WESTERNEUROPEAN_WINDOWS	htm
NATIVE FI_HTML_UNICODE	htm
NATIVE FI_JAPANESE_EUC	txt
NATIVE FI_JAPANESE_JIS	txt
NATIVE FI_LATIN2	txt
NATIVE FI_MAC	txt
NATIVE FI_MAC8	txt
NATIVE FI_PP2KHTML	htm
NATIVE FI_SHIFTJIS	txt
NATIVE FI_UNICODE	txt
NATIVE FI_UTF8	txt
NATIVE FI_W2KHTML	htm
NATIVE FI_WML	xml
NATIVE FI_WML_CHINESEBIG5	xml
NATIVE FI_WML_CHINESE_EUC	xml
NATIVE FI_WML_CHINESEGB	xml
NATIVE FI_WML_CYRILLIC1251	xml
NATIVE FI_WML_CYRILLIC018	xml
NATIVE FI_WML_JAPANESE_EUC	xml
NATIVE FI_WML_JAPANESE_JIS	xml
NATIVE FI_WML_JAPANESE_SJIS	xml
NATIVE FI_WML_KOREAN_HANGUL	xml
NATIVE FI_WML_LATIN2	xml
NATIVE FI_XHTML	htm
NATIVE FI_XL2KHTML	htm
NATIVE FI_XML	xml
NATIVE FI_XML_DOCTYPE_HTML	htm

Language and code page support

Linguistic processing for enterprise search is handled differently by the parser and search servers.

For linguistic processing purposes, the parser does not distinguish between languages and locales. If a user searches a collection that includes documents in multiple languages, however, the search servers enable the search results to be limited to a specific language or locale.

For example, if the metadata in an English document specifies en_US for the document locale, the document is indexed as both an English language document (en) and as a document that uses the United States locale for English (en_US). This type of indexing enables locale-specific information, such as numbers, dates, and times, to be represented correctly. When users search the collection, the document can be found regardless of whether the user searches for en or en_US documents.

If a document is indexed only by the language code, such as en, the document is indexed only by the language code and not the locale. If users search the collection for en_US documents, for example, the document will not be found.

An enterprise search system provides linguistic support for the following languages and two-character language codes, as documented in the ISO 639 standard:

Simple text languages:

- en=English
- sq=Albanian
- az=Azerbaijani-Latin
- bg=Bulgarian
- be=Belarusian
- ca=Catalan
- hr=Croatian
- cs=Czech
- da=Danish
- nl=Dutch
- et=Estonian
- fi=Finnish
- fr=French
- de=German
- el=Greek
- hu=Hungarian
- is=Icelandic
- id=Indonesian
- in=Indonesian
- it=Italian
- kk=Kazakh
- lv=Latvian
- lt=Lithuanian
- lo=Laotian
- mk=Macedonian
- ms=Malay
- mt=Maltese
- no=Norwegian
- nb=Norwegian (Bokmal)
- pl=Polish
- pt=Portuguese
- ro=Romanian
- ru=Russian

sr=Serbian (Cyrillic)
sh=Serbian (Latin)
sk=Slovak
sl=Slovenian
es=Spanish
sv=Swedish
tr=Turkish
uk=Ukrainian
cy=Welsh

Ideographic languages:

For Simplified and Traditional Chinese, expanded language codes are used instead of two-character codes.

zh-CN=Chinese (Simplified)
zh-TW=Chinese (Traditional)
ja=Japanese
ko=Korean

Complex text languages:

ar=Arabic
as=Assamese
bn=Bengali
gu=Gujarati
iw=Hebrew
he=Hebrew
hi=Hindi
kn=Kannada
ml=Malayalam
mr=Marathi
or=Oriya
pa=Punjabi
ta=Tamil
te=Telugu
th=Thai
ur=Urdu
vi=Vietnamese

The enterprise search system can automatically detect many of these languages, and can automatically detect the code page that is used in plain text documents. When you configure a crawler, you can disable automatic language and code page detection if you want to specify an explicit language or code page to use.

Automatic language detection

An enterprise search system can process documents in virtually any language.

If a document is in one of the following languages, the system can detect the language automatically. If you know the language of your documents, you can specify the language to use when you configure a crawler instead of allowing the system to detect the language automatically.

Arabic
Bulgarian
Czech
Chinese, Simplified
Chinese, Traditional
Danish
Dutch
English
Finnish
French, Canadian
French, National
German, National
German, Swiss

Greek
Hebrew
Hungarian
Icelandic
Italian
Japanese
Korean
Norwegian, Bokmal
Polish
Portuguese, Brazilian
Portuguese, National
Romanian
Russian
Spanish
Swedish
Thai
Turkish

Automatic code page detection

An enterprise search system supports documents in a variety of code pages.

For text files, the system can detect the following code pages automatically. For other document formats, the system uses metadata in the document, such as HTML metadata elements, to detect the code page. If you know the code page of your documents, you can specify the code page to use when you configure a crawler instead of allowing the system to detect the code page automatically.

Unicode encoding forms:

UTF-8
UTF-16BE
UTF-16LE

Multiple-byte encoding forms:

Shift-JIS
ISO-2022-CN
ISO-2022-JP
ISO-2022-KR
GB18030
EUC-JP
EUC-KR

Single-byte encoding forms:

ISO-8859-1: Danish, Dutch, German, English, French, Italian, Norwegian, Portuguese, Spanish, Swedish
ISO-8859-2: Czech, Hungarian, Polish, Romanian
ISO-8859-5: Russian
ISO-8859-6: Arabic
ISO-8859-7: Greek
ISO-8859-8: Hebrew, Hebrew in visual order
ISO-8859-9: Turkish
Windows-1250: Czech, Hungarian, Polish, Romanian
Windows-1251: Russian
Windows-1252: Danish, Dutch, German, English, French, Italian, Norwegian, Portuguese, Spanish, Swedish
Windows-1253: Greek
Windows-1254: Turkish
Windows-1255: Hebrew
Windows-1256: Arabic
KOI8-R: Russian

Character set detection is an imprecise operation. The code page detection process attempts to identify the character set (charset) that best matches the characteristics of the byte data, but the process is partly statistical in nature, and the results cannot be guaranteed to be correct.

For the greatest accuracy, the input data should be primarily in a single language. A minimum of a few hundred bytes of plain text in the language is also needed.

If there is a mismatch between the detected encoding and the supported encodings, the system uses the default code page for the collection.

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 white space to delimit words.

Unicode-based white space segmentation uses white 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 white space to delimit words, 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.

For information about how to configure support for full n-gram parsing and tokenization in enterprise search collections, and to learn about how characters are handled in collections that are configured for full n-gram support, see <http://www.ibm.com/support/docview.wss?rs=63&uid=swg27011088>.

Removing white space from text

You can configure the parser to remove white space characters from text.

Before you begin

To complete this task, you must log in as an enterprise search administrator.

About this task

If you enable this option for a collection, the parser removes sequences of white space characters that separate two letter characters. You might want to remove white space characters, for example, if your documents are in a language that does not use white space to delimit word boundaries, such as Chinese or Japanese.

When you configure the parser to remove white space characters, you can specify whether you only want to remove white space that occurs between double-byte character set (DBCS) characters or whether you want to remove all white space, regardless of the character context. You might want to use this latter option, for example, if you include English text in a Japanese document, and want to remove the white space from the English text, too.

The parser removes the following characters:

- 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 stored in the index, crawl the documents again, and rebuild the main index.

Procedure

To remove white space from text:

1. On the index server, log in as the enterprise search administrator. This user ID was specified when OmniFind Enterprise 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:

```
ES_NODE_ROOT/master_config/collection_ID.parserdriver/collection.properties
```

3. Specify how white space is to be removed:
 - To remove white space between DBCS characters, set the value of the `removeCjNewlineChars` property to `true`:

```
removeCjNewlineChars=true
```

- To remove white space anywhere in the documents, set the value of the `removeCjNewlineChars` property to `true` and set the value of the `removeCjNewlineCharsMode` property to `all`:

```
removeCjNewlineChars=true  
removeCjNewlineCharsMode=all
```

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 updating 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, an update (called a *delta index*) is created for the new content. Eventually, the updates need to be merged into the base index. This merging process is called building the *main index*. Whenever a delta index or the main index is built, 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 update indexes frequently, you enable users to search the most current data. Eventually, an index that is continuously updated must be rebuilt. As an index grows larger, it consumes more system resources. To maintain optimal performance, build the main index regularly.

How often you build the main 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 to crawl
- How often you change category rules (the changes do not become effective until the main index build occurs)
- How often you force a crawler to start instead of running at a scheduled time

For collections with several million documents that are built with mostly Web documents, you should build the main index approximately once a day, and update the index every one or two hours.

To maintain a current, searchable index, you do the following tasks:

- Specify schedules for building 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 297

“Monitoring the enterprise search index queue” on page 298

Scheduling index builds

You can specify schedules for building the main index and updating the 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 built on a regular basis. During the main index build, the entire index is rebuilt. The indexing processes read all of the data that was collected by crawlers and analyzed by the parser. During a delta index build, information that was crawled since the last main index build occurred is made searchable.

By default, the option to schedule index builds is selected. This option tells the scheduler process to schedule tasks to build main and delta indexes when 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.

To conserve system resources and improve performance, the system automatically checks to see whether changes that need to be applied to the index occurred. If no changes need to be applied to the index, the scheduled build request is discarded.

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 updated with new content, specify the following options on the Schedule Index Builds page in the **Specify a schedule to build a delta index** area:
 - a. In the **Start on** area, in the **Year, Month, Day, Hour, and Minute** fields, specify when you want the first delta index to be built.
 - b. In the **Update interval** area, in the **days, hours, and minutes** fields, specify how often you want delta indexes to be built.

Typically, you should build delta indexes 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 re-built, specify the following options in the **Specify a schedule to build the main index** area:
 - a. In the **Start on** area, in the **Year, Month, Day, Hour, and Minute** fields, specify when you want the main index to be built the first time.
 - b. In the **Update interval** area, in the **days, hours, and minutes** fields, specify how often you want the main index to be built.

Typically, you should build the full 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 building the 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 updates are to be made to the index and how often the main index is to be built.
2. Click **Apply**.

Enabling and disabling the index schedules

You can enable and disable the schedules for building 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 updating the index.
 - c. Select or clear the **Enable when system starts** check box to enable or disable the schedule for building the main index.
 - 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 index updates to occur at the same time that the main index is being built.

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 build 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 update an index are to be processed at the same time that the main index for the collection is being built. If you enable this option, the search servers will be refreshed with the latest documents (through the delta index) while the more slow running main index build 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.

If you increase the number of concurrent index builds, index build requests that are already in the queue are not started automatically. Your change affects new index builds that are enqueued after you change this value.

If you decrease the number of concurrent index builds, current index builds are not stopped automatically. Your change becomes effective after the current index builds stop, which enables the enqueued index builds to start.

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 delta index builds to run concurrently with main index builds.

Restriction: If you select this option when an index build for any collection is in progress, the index build might fail. Before enabling concurrent delta and main index builds, disable all scheduled index builds and wait for currently running index builds to stop (or end them before they finish, if appropriate). After you enable this option, enable the scheduled index builds that you disabled.

Building indexes only when changes are detected

For a scheduled index build, the build occurs only when changes that need to be applied to the index are detected. You can use the **startIndexBuild** command to start an index build and choose whether the system should check for changes.

Before you begin

To manually specify that an index build is to occur only when changes are detected, you must log in as the enterprise search administrator.

Restrictions

The system does not check for changes that might affect the index until the main index has been built at least twice. This implementation accommodates changes that cannot be detected until after the main index has been built at least once, such as changes to scope definitions or rules for collapsing search results.

About this task

Index builds consume system resources such as CPU, memory, and I/O bandwidth. To conserve system resources and improve performance, the enterprise search system can detect whether changes that need to be applied to the index occurred and build the index only when it is necessary to apply changes. The system can detect the following index modification events:

Main index builds only

- Removing URIs by pattern, which removes documents from the index.
- Removing a crawler from the system, which removes documents from the index.

Main and delta index builds

- Modifying the definitions of scope or rules for collapsing search results.
- Adding or removing documents directly through crawler settings or the push API.

- Modifying boost factors to influence the static ranking scores of documents in the index.
- Modifying the rules for how wildcard characters can be used in queries.

All scheduled index build requests are subject to change detection. The index build proceeds only if there are pending changes that need to be applied to the index. If no change is detected, the index build request is discarded.

To force an index build regardless of whether changes occur, you can use the enterprise search administration console to start the index build. You can also use the **startIndexBuild** command to manually start the index build and specify whether the system should check for changes before proceeding with the request.

Procedure

To start an index build from the command line, and request that the build is to proceed only if changes are detected:

1. Log in as the enterprise search administrator. In a multiple server configuration, log in on the index server.
2. Enter the following command:

```
esadmin controller startIndexBuild -options
```

Options:

-cid *collection_ID*

The collection ID for the collection that you want to start the index build for.

-buildType *build_type*

Specifies the type of index to build. Valid values are main and delta.

-detectChanges

Optional. Causes the index build request to undergo the change determination tests. The index build proceeds only if changes that need to be applied to the index are detected.

In the following example, a request to start a main index build for the col_1 collection proceeds only if changes that need to be applied to the index are detected:

```
esadmin controller startIndexBuild -cid col_1 -buildType main
-detectChanges
```

Stopping index builds

You can use the **stopIndex** command to stop a main and delta index builds instead of using the enterprise search administration console.

Procedure

To stop an index build by using a command instead of the administration console:

1. Log in as the enterprise search administrator. For a multiple server installation, log in on the index server.
2. Enter this command:

```
esadmin stopIndex -cid collection_id -buildType typeWhere:
```

-cid *collection_id*
Specifies the collection ID for the collection that owns the index.

type
Specifies the type of index build to stop. Allowed values are main or delta.

Example

```
esadmin stopIndex -cid coll -buildType delta
```

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.

For some crawler types, and for collections that do not enable security, duplicate document detection is used to prevent users from seeing multiple documents that are the same, or nearly the same, in the search results.

Indexed options for searching documents

When you configure options for searching crawled data, 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.

The search 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.

Crawler options:

When you configure a crawler to crawl data sources 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
- Complete match
- Sortable
- Parametric search
- Search results

- Document content

XML and HTML field mapping options:

When you configure the parser and specify that you want to map XML elements and HTML metadata elements to searchable fields in the index, you specify the following options:

- Fielded search
- Complete match
- Sortable
- Search results

If you configure search options for specific HTML metadata elements, as opposed to all elements or elements that are in the Dublin Core metadata element set, you can also specify that fields that contain numeric values can be searched with a parametric query.

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.

To enable fields to be searched with a free text query, you select the **Free text search** check box when you configure a crawler. To search title, keyword, and description fields, select this check box and the **Fielded search** check box.

Example 1:

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 2:

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 data fields and metadata fields in a document. For example, you can specify that certain words must exist in the title of a document.

To enable fields to be searched by field name, you select the **Fielded search** check box when you configure the crawler or when you configure field mapping options

for XML and HTML elements. To search Title, Keywords, and Description fields, select this check box and the **Free text search** check box.

Example:

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.

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"
```

Complete match

A complete match search can enhance the quality of the search results by enabling you to specify precise queries. With a complete match search, you can query fields and XML elements, and retrieve only those documents in which the entire field value or XML element value matches the query terms. If the value of the field or element contains less content or additional content, a match does not occur.

Tip: When determining whether a complete match exists, the system converts the query terms to lowercase, removes extra spaces in the query string, and performs wildcard character pattern matching. However, no lemmatization or synonym lookup occurs and stop words are not removed. A complete match for an XML element requires the element name, with no nested elements, and the entire value of the element to match the query terms exactly.

You can search fields for complete matches by using the enterprise search (SI-API) query syntax or XMLFrag2 query syntax (XPath queries are not supported). An equal sign (=) preceding the query terms indicates that a complete match search is to be done.

To enable fields to be searched for complete matches, you select the **Complete match** check box when you configure a crawler or when you configure field mapping options for XML and HTML elements. If you enable users to search XML documents with native XML search when you configure parsing options for a collection, all XML elements can be searched for complete matches of the search terms.

Example 1:

A field named `color` contains the value `dark blue`.

- The following complete match query matches because the query contains no other terms:

```
color:="dark blue"
```

- The following complete match query does not match because the query includes the word `skirt` in addition to `dark blue`:

```
color:="dark blue skirt"
```

- The following complete match query does not match because the `color` field also contains the word `dark`:

```
color:=blue
```

Example 2:

Without complete matching, the following XMLFrag2 query might return documents that specify <diagnosis>intraductal carcinoma comedo type</diagnosis> or other terms that do not precisely match the query terms:
`@xmlf2::'<diagnosis>intraductal carcinoma</diagnosis>'`

With complete matching, the following XMLFrag2 query ensures that the only documents returned are those in which the entire content of the XML element value matches the query terms:

```
@xmlf2::'=<diagnosis>intraductal carcinoma</diagnosis>'
```

Sortable

If a data source includes fields, or if users are searching XML or HTML documents, you might want to enable the results to be sorted by the values in a particular field. In the sample search application for enterprise search, the names of all fields that were configured to be sortable fields are listed. Users can choose to sort results alphabetically (according to a string sort) by selecting one of the listed fields instead of sorting the results by relevance or document date. Users can also choose whether the documents are to be sorted in ascending or descending order.

Result documents that do not contain the sort field are listed at the end of the search results. Result documents that contain the sort field, but were indexed before the field was configured to be sortable, are also listed at the end of the search results.

To enable users to sort search results alphabetically by the values in a field, select the **Sortable** check box when you configure the crawler or when you configure field mapping options for XML and HTML elements. If the field contains numeric values, select the **Parametric search** check box to specify that the field values can be used to sort the search results numerically.

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.

To sort results numerically according to a field's value, you must enable the field for parametric search.

To search a field with a parametric query, or to be able to sort results numerically, you select the **Parametric search** check box when you configure the crawler or when you configure field mapping options for specific HTML metadata elements.

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
```

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.

To enable a field to be shown in the search results, you select the **Search results** check box when you configure the crawler or when you configure field mapping options for XML and HTML elements.

Document content

For certain types of documents, such as Web documents, the entire document is considered content. For other types of documents, such as documents that contain fields, you can specify which fields contain useful content as opposed to metadata.

To specify that a field constitutes document content, you select the **Document content** check box when you configure the crawler. If both the **Document content** and **Free text search** check boxes are selected, then the value of the field is used to detect duplicate documents and it becomes part of the dynamic document summary area of the search results.

Related concepts

 [Query syntax](#)

Duplicate document detection

Duplicate document detection is a technique that is used to prevent search results from containing multiple documents with the same or nearly the same content.

Search quality might be degraded if multiple copies of the same (or nearly the same) documents are listed in the search results. Duplicate document analysis occurs only when both of the following conditions are true:

- The collection uses the link-based ranking model. This model applies to crawlers that crawl Web sites, such as the Web crawler or WebSphere Portal crawler.
- Collection-security is disabled.

During global analysis, the indexing processes detect duplicates by scanning the document content for each document. If two documents have the same document content, they are treated as duplicates.

If you want document metadata to also be considered when duplicate detection analysis occurs, you must select the **Document content** check box when you configure crawlers for the collection and specify options for crawling metadata. In this case, the crawler crawls the metadata fields as document content and includes the metadata when analyzing the content for duplicate documents. Similar analysis occurs when you configure options for parsing HTML and XML documents and select the **Document content** check box.

When you specify that a field or metadata field constitutes document content, the content of those fields is added to the dynamic summary of the document in the search results, which can have an impact on whether the document is displayed in the search results. If near duplicate detection is enabled in the search application (the `NearDuplicateDetection` property in the `setProperty` method is set to `Yes`), documents with similar titles and summaries are suppressed when a user views search results. Users can click a link to view the nearly duplicate, suppressed documents.

In a group of duplicate documents, one document is the master and the others are duplicates. All documents in the group of duplicates have the same canonical representation of the content. During indexing, the content (tokens) of the master document are indexed. For the duplicate documents, only the metadata tokens are indexed. When the master document is deleted from the index, the next duplicate becomes the master. When users search the collection, only the master document is returned.

Related concepts

“Duplicate document analysis and collection security” on page 249

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 the main index build occurs.

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*`.

During an delta or main index build, 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 effect 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.

How wildcard characters affect the index

Support for wildcard characters that is based on index expansion increases the size of the index and the time to build the index.

With index expansion, every prefix of a term and the term itself is indexed. For example, the following terms are indexed for the term `support`:

```
s su sup supp suppo support
```

The number of terms that are stored in the index grows by a factor of the average length of a word. Index compression reduces the size of the index, but not significantly. The time that is required to build the index increases by the average length of a word.

An index of English documents grows by a factor of approximately four because the average length of an English word is five to six characters. An index with n-gram tokenization increases approximately two times because each n-gram contains two characters.

Index expansion is recommended for wildcard character support in the following situations:

- The collection is small enough such that the space and time that is consumed by the expanded index does not cause a performance problem.
- All possible expansions of wildcard characters are included in the search results to satisfy user (or enterprise) requirements.

You can mitigate the effect on the index by specifying a minimum prefix length for wildcard expansion. For example, if the minimum prefix length is set to three, the prefixes `s` and `su` are not indexed for the word `support`, and an English index increases by a factor of three instead of four.

With the query expansion approach for wildcard character support, no prefixes are written to the index. Terms are expanded when the query is submitted, and the index grows only by a small data structure that is required to support that expansion. Typically, an index with query expansion is between 10% and 20% larger than an index without wildcard character support, and it takes less than 10%

more time to build the index. The configuration of the maximum number of expansions has no effect on the size of the index or the time to build the index.

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 main index build occurs.

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. Optional: You can specify that you want to support wildcard characters in queries that search free text. 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 specify that wildcard characters cannot be processed in queries that search fields, select **No fields**.
 - 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. For details, click **Help** in the administration console.

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 build the main index twice before the changes become effective. If you configure scopes before the first main index for the collection is built, users will be able to search the collection, but they will not be able to see the scope data in the search results. Build the main index again to ensure that search results reflect the range of URIs in the scope.

If you configure scopes after the main index is built, the changes become effective when the next main index build occurs.

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 build the main index twice before the changes become effective. If you configure scopes before the first index build occurs, users will be able to search the collection, but they will not be able to see the scope data in the search results. Build the main index again to ensure that search results reflect the range of URIs in the scope.

If you configure scopes after the main index is built, the changes become effective when the next main index build occurs.

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 113

Collapsed URIs

Enterprise search can organize the search results so that documents from sources that have the same URI prefix are 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 are returned (or documents that belong to the same URI group), 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 URI prefix that is specified in the query.

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.

If a Web document cannot be matched to a URI prefix, the index server uses the host name of the URL as the URI prefix. If an NNTP document cannot be matched to a URI prefix, 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 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 the main index build occurs.

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/
db2://LOCALDB/SCHEMA1.TABLE1/
exchange://exchangesvr.ibm.com/public/TeamRoom/Folder1/`
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 a main or delta index build occurs.

If you specify a URI pattern to remove multiple URIs, users continue to see URIs that match the specified pattern in the search results until the next time a main index build occurs.

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*
```

Related reference

“URI formats in an enterprise search index” on page 113

Search server administration

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 affect query performance:

Configuring a search cache

To optimize query performance, you can specify that search results (the responses to queries) are to be stored in a cache, and you can configure the amount of space to allocate for cached search results.

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” on page 197

“Custom boost word dictionaries” on page 200

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 query response, they can quickly return search results to the user. If the search servers do not find the appropriate query response, they search the index.

When the search cache fills, the oldest search results and the results 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 query responses that 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 query responses 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 queries that include an acronym (such as ACL) return documents that discuss the expansion of that acronym (such as ACLs, access control lists, access controls, and so on).

The enterprise search query language supports 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. An enterprise search tool, **essyndictbuilder**, 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.

You can write a script that includes these steps and then use the script to redeploy the dictionary in your enterprise search system.

Related concepts

 [Synonym support in search applications](#)

Related tasks

 [Creating an XML file for synonyms](#)

 [Creating a synonym dictionary](#)

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.

Restrictions

The maximum size of a synonym dictionary is 8 MB.

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 on your local system you can browse to locate the file. If the file is on the index server, type the fully qualified path.
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).

In enterprise search, language-specific stop word recognition is performed 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.

When a query is processed, stop words are removed before spelling suggestions are made. If all of the words in a query are stop words, then no stop words are removed during query processing. To ensure that search results are returned, stop word removal is disabled when all of the query terms are stop words. For example, if the word *car* is a stop word and you search for *car*, then the search results contain documents that match the word *car*. If you search for *car volvo*, the search results contain only documents that match the word *volvo*.

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. An enterprise search tool, **esstopworddictbuilder**, 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.

You can write a script that includes these steps and then use the script to redeploy the dictionary in your enterprise search system.

Related concepts

 Custom stop word dictionaries

Related tasks

 Creating an XML file for stop words

 Creating a stop word dictionary

Adding stop word dictionaries to the system

If you create custom stop word dictionaries for removing words from queries, you must add the dictionaries to 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.

Restrictions

The maximum size of a stop word dictionary is 8 MB.

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 on your local system you can browse to locate the file. If the file is on the index server, type the fully qualified path.
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**.

Redeploying custom dictionaries

You cannot use the enterprise search administration console to make changes to a synonym, stop word, or boost word dictionary. However, you can include steps in a script and then use the script to redeploy the dictionary.

Before you begin

To redeploy a custom dictionary, you must log in as the enterprise search administrator.

Procedure

To redeploy a custom dictionary for enterprise search:

1. On the index server, open one of the following files to find the name of the dictionary that you want to redeploy:
 - To find the name of a synonym dictionary, open `ES_NODE_ROOT/master_config/SynonymConfiguration.xml`.
 - To find the name of a stop word dictionary, open: `ES_NODE_ROOT/master_config/StopWordDictionaryConfiguration.xml`
 - To find the name of a boost word dictionary, open `ES_NODE_ROOT/master_config/BoostingWordDictionaryConfiguration.xml`.

The following example shows a `SynonymConfiguration.xml` file with two synonym dictionaries that were uploaded with the names `hello` and `goodbye`:

```
% cat $ES_NODE_ROOT/master_config/SynonymConfiguration.xml
```

```
<SynonymConfiguration>
<Synonyms>
<Synonym Name="hello" ID="SynonymId_1">
<Filename>synonym_hello1.dic</Filename>
<Timestamp>1169766691776</Timestamp>
</Synonym>
<Synonym Name="goodbye" ID="SynonymId_2">
<Filename>synonym_goodbye2.dic</Filename>
<Timestamp>1169767224839</Timestamp>
</Synonym>
</Synonyms>
</SynonymConfiguration>
```

When dictionaries are uploaded, the system assigns unique file IDs. In the previous example, these IDs are `synonym_hello1.dic` and `synonym_goodbye2.dic`. The absolute path for these files on the index server is as follows:

- The path for a synonym dictionary is `ES_NODE_ROOT/data/custom_dictionary/synonym_*.dic`.
- The path for a stop word dictionary is `ES_NODE_ROOT/data/custom_dictionary/stopwordDictionary_*.dic`.
- The path for a boost word dictionary is `ES_NODE_ROOT/data/custom_dictionary/boostingwordDictionary_*.dic`.

On AIX, Linux, and Solaris, you can use the `ls` command to list the available dictionaries. For example:

```
% ls -l $ES_NODE_ROOT/data/custom_dictionary/synonym_*.dic
-rw-rw-r-- 1 esuser users 9 Jan 25 15:11 /home/esuser/node/data/custom_dictionary/synonym_hello1.dic
-rw-rw-r-- 1 esuser users 9 Jan 25 15:11 /home/esuser/node/data/custom_dictionary/synonym_goodbye2.dic
```

2. After you identify the dictionary that you want to update, stop the enterprise search system.
3. Overwrite the dictionary that you want to update with your new dictionary file. For example, overwrite `synonym_hello1.dic` or `synonym_goodbye2.dic`.
4. If your enterprise search system runs on two servers or four servers, manually overwrite the dictionary file on the search servers, too. The dictionary files are located in the same path as they are on the index server (`ES_NODE_ROOT/data/custom_dictionary/`).
5. Restart the enterprise search system.

Related concepts

“Custom boost word dictionaries” on page 200

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 in each sentence in the summary. Shorter sentences are included in the summary if there are not enough sentences that have more words than the `MinWordsPerSentence` value. The default value is 4.

MaxWordsPerSentence

The maximum number of words in each sentence that will be included in the summary. If a sentence has more words than this limit, only part of the sentence (the part that contains the query terms, up to the `MaxWordsPerSentence` value) is included in the summary. The remainder of the sentence is excluded. The default value is 20.

Sentences are selected for document summaries according to a proprietary, internal algorithm that determines the relevance of all sentences that include the search terms. Selection by relevance occurs before any sentences are filtered by sentence length.

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 OmniFind Enterprise 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 113

Document ranking

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.

Text-based scoring

Enterprise search dynamically calculates a score for each document that matches the terms in a query.

The text score for a document represents the importance of the query terms in a document. To calculate the text score of each document that matches a query, enterprise search considers many factors, such as:

- Whether the terms distinguish a document from other documents. For example, if the query terms appear in one document but not in others, it means that these terms are important for that document and distinguish that document from the other documents. Query terms that appear in most documents contribute less to the document score than query terms that appear in a more selective set of documents.
- The number of occurrences of the query terms within a document. The score of a term is computed on the basis of each occurrence of that term within a document. The more occurrences of the query terms within a document, the higher the document's score is. For example, you search for *thinkpad*, a document that includes this term many times is ranked higher in the search results than other documents that contain fewer occurrences.
- For HTML documents, attributes of the query term (that is, the context of the term) are considered. The score of a term is computed on the basis of attributes of that term (such as location, bold, italic, anchor, and so on). In general, terms that occur in the document title have a higher score than terms that occur in a paragraph. Terms that are emphasized (such as bold text) have a higher score than plain text terms. You can configure the importance of attributes in the parser settings.
- The weight of the query terms. To customize the importance of terms in a document, you can configure boost values for terms. In this case, you associate a boost word dictionary (which contains terms and their boost values) with the collection. This dictionary is used during searching, and the boost values of the terms in the dictionary contribute to the document's score. The higher the boost value, the higher the term's contribution is to the document score.
- The proximity of query terms in a document. If the query terms appear close to each other in a document, then their lexical affinity is used to compute the text score. For example, assume that you have two documents. One document talks

about a car park in the city (car and park are close together). The other document talks about a car showroom in a city near to the park (car and park are not close together). If you search for car park, the proximity of the terms in first document cause that document to be ranked higher than the second document.

- The length of each document and the richness of its vocabulary (such as the number of unique words) are also factors in determining the document 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 **Document importance** options. The type of document importance that you select determines whether a static ranking factor is associated with the documents in the collection. When users search the collection that uses static ranking, the static ranking factor influences how documents returned in the search results are ranked.

For Web content, the static ranking factor is based on links. 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, the static ranking factor is based on the document date. The document date field, which is provided by the crawler, can be the date the document was last modified or the date the document was last crawled, depending on how you set up the crawler configuration.

The date of a document might 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 when you configure the crawler.

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. Document ranking is less accurate when sources with different ranking models are combined in the same collection, and the order of the search results might not be as expected.

You should also ensure that documents in the collection contain fields and values that enable static ranking to be applied. For example, imagine a collection that is configured to use static ranking based on document dates, and a crawler in the collection is configured to use a specific field as the document date. If a document does not contain that field, the importance of the document might not be appropriately ranked, and the order of the search results might not be as expected.

Implications of link-based ranking

Static ranking, along with factors such as assigning a score to boost URI patterns, contributes to the static score of a document and influences the importance of the document. The link-based ranking model is typically applied to Web collections because this model calculates the static rank of a document based on the number of links to the document. A document that is linked to from a high number of other documents is ranked as more relevant.

For this reason, if you configure this model for a non-Web collection or a mixed collection (one that contains Web and non-Web documents), the search quality might be degraded because non-Web documents have no concept of linking.

When link-based ranking is enabled, duplicate document detection is also enabled. Duplicate documents have the same static rank as the master document. If URI pattern boost factors are not configured for any documents in the duplicate group, then all of the duplicate documents have the same static score.

Restoring default values for static document ranking

If you configure a static document ranking option when you create a collection, you can set the properties back to the default values by editing the `runtime.properties` files for the collection.

Before you begin

To restore default document ranking values to a collection, you must be an enterprise search administrator.

About this task

To restore the default document ranking values for a collection, you must update the `runtime.properties` files for that collection and all search servers in your enterprise search system. In a multiple server configuration, the `runtime.properties` file is on the index server in the `ES_NODE_ROOT/master_config/collection_ID.runtime.node_ID` directory, where `collection_ID` is the ID for the collection and `node_ID` is the ID for the search servers.

For example, to update the `coll` collection in a multiple server enterprise search system, you update the `runtime.properties` for that collection and both search servers (`node3` and `node4`):

```
ES_NODE_ROOT/master_config/coll.runtime.node3/runtime.properties
ES_NODE_ROOT/master_config/coll.runtime.node4/runtime.properties
```

Procedure

To restore document ranking values to the default values for a collection:

1. Log in as the enterprise search administrator on the index server.
2. Identify the collection ID for the collection to which you want to restore default ranking values. The collection ID is in the `ES_NODE_ROOT/master_config/collections.ini` file. Sort this file for easier viewing. In the following example, `coll` is the collection ID:

```
% sort $ES_NODE_ROOT/master_config/collections.ini | more
collection1.configfile=coll_config.ini
collection1.datadir=/home/esearch/node/data/coll
collection1.description=
collection1.displayname=Collection1
collection1.flags=0
collection1.id=coll
collection1.sectiontype=collection
collection1.type=1
...
```

3. Edit the `runtime.properties` file for the collection that you want to restore and make the following changes:
 - a. Delete the following properties:

```
trevi.autorank.dfthreshold1
trevi.autorank.dfthreshold2
trevi.autorank.dfthreshold3
trevi.autorank.rc0.*
trevi.autorank.rc1.*
```

- b. If the `runtime.properties` file specifies `trevi.sourcetype=1`, which indicates that documents are ranked by links, edit the `ES_INSTALL_ROOT/default_config/runtime.1/runtime.properties` file and copy and paste the following default properties to the `runtime.properties` file:

```
trevi.autorank.dfthreshold1
trevi.autorank.dfthreshold2
trevi.autorank.dfthreshold3
trevi.autorank.rc0.*
trevi.autorank.rc1.*
```

- c. If the `runtime.properties` file specifies `trevi.sourcetype=2`, which indicates that documents are ranked by date, edit the `ES_INSTALL_ROOT/default_config/runtime.2/runtime.properties` file and copy and paste the following default properties to the `runtime.properties` file:

```
trevi.autorank.dfthreshold1
trevi.autorank.dfthreshold2
trevi.autorank.dfthreshold3
trevi.autorank.rc0.*
trevi.autorank.rc1.*
```

- d. If the `runtime.properties` file specifies `trevi.sourcetype=3`, which indicates that a static ranking factor is not used to rank documents in the collection, edit the `ES_INSTALL_ROOT/default_config/runtime.0/runtime.properties` file and copy and paste the following default properties to the `runtime.properties` file:

```
trevi.autorank.dfthreshold1
trevi.autorank.dfthreshold2
trevi.autorank.dfthreshold3
trevi.autorank.rc0.*
trevi.autorank.rc1.*
```

4. In a multiple server configuration, repeat step 3 on page 199 to update the `runtime.properties` file for same collection and the second search server.
5. In the administration console, monitor the Search page and restart the search processes for this collection.

Repeat these steps as necessary for each collection that you want to restore default document ranking values to.

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.

You can use a boost word dictionary to ensure that certain documents are returned when a user specifies certain query terms. For example, assume that you have a collection that contains many documents that talk about cars. For such documents, you might think that certain key words related to car models, the name of the manufacturer, and so on, are important. To influence the ranking of search results, you can assign importance by associating a boost value with the key words (`model`, `manufacturer`, and so on) in a boost word dictionary. When users search the

collection and specify a query that includes any of the key words, the documents about cars are ranked higher in the search results than other documents.

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. (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. An enterprise search tool, **esboosttermdictbuilder**, 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.

You can write a script that includes these steps and then use the script to redeploy the dictionary in your enterprise search system.

Related concepts

 [Custom boost word dictionaries](#)

Related tasks

 Creating an XML file for boost words

 Creating a boost word dictionary

“Redeploying custom dictionaries” on page 191

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.

Restrictions

The maximum size of a boost word dictionary is 8 MB.

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 on your local system you can browse to locate the file. If the file is on the index server, type the fully qualified path.
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.

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, rebuild the main index.

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 and document summaries

When you map a field to a boost class, you must specify whether the field is used to detect duplicate documents and whether the content of the field can be included in document summaries in the search results.

- 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. The content of these types of fields can be used in dynamic document summaries in the search results.
- 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, and the field is not used in dynamic document summaries.

High and low recall values

When a query is evaluated, the search process estimates the number of result documents that will be returned. Thresholds determine whether a query is considered to have low recall value, high recall value, or a value that falls between the low and high values:

Low recall value

If the estimated number of result documents is below the low threshold, the query is considered a low recall query.

High recall value

If the estimated number of result documents is above the high threshold, the query is considered a high recall query.

Mixed recall value

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.

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 mappings for internal fields and regular text that do not have other defining characteristics. 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 a document with this field is returned in the search results, the content of the field will be displayed in the document summary area.

If you clear the check box, the list of available boost classes contains classes that apply to metadata fields. The content of the field will not be displayed in the document summary area of the search results.

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 and index the documents again.

Related concepts

“Document ranking” on page 197

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” on page 197

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  
es_special_field.default_metadata_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 6. 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	es_special_field.default_field
Metadata class A	Low: 1 High: 1	Low: 1 High: 1	Low: 1 High: 1	es_special_field.default_metadata_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

Table 6. Default boost class values (continued)

Default low and high boost factors				
Boost class name	No static ranking	Document links	Document date	Predefined field mappings
Metadata class H	Low: 1 High: 1	Low: 1 High: 1	Low: 1 High: 1	description

Related concepts

“Document ranking” on page 197

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.

To help you customize search applications, you can use the Search Application Customizer. This application enables you to make selections in a graphical interface and view the effects of your changes as you make them. When you save your changes, you update the configuration file for the search application.

Tip:

For detailed examples of how to use the Search Application Customizer and how to install the search application as a portlet in WebSphere Portal, see the IBM Redbook, *IBM OmniFind Enterprise Edition Configuration and Implementation Scenarios*.

Related concepts

“Indexed options for searching documents” on page 171

“Search application identifiers” on page 250

“Document-level security” on page 251



Search and index API overview



Query syntax

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 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 or XML elements, 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, or by the values in a particular field. When sorting by date or by field, you can specify whether you want to view the results in ascending or 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 and opening the document in a Web browser. If a Notes crawler or a Domino Document Manager crawler is configured to use the DIIOP protocol, then documents that are crawled by these crawlers can be displayed by a Lotus Notes client viewer application instead of a Web browser.
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.

Search application properties

You can edit the configuration file for a 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.

You can also edit properties by using the Search Application Customizer. When you make selections with the Customizer, you can see the effect of your changes immediately. When you are satisfied with the options that you specify for searching collections and viewing search results, you can save the options to update the configuration file for the search application.

Important: If you run the search application as a portlet within WebSphere Portal, you cannot use the Search Application Customizer to make changes to the search application interactively. You must edit properties and configure the Portlet instance with the WebSphere Portal Administration interface.

The configuration file for the sample search application for enterprise search is the `config.properties` file. This topic discusses the properties in this file and describes the default properties. If you create configuration files for your custom search applications, the properties in those files and the values specified for those properties, might be different.

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.

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.

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 local computer host name is `MyMachine` and the Web server host name is `www.ibm.com`, specify `www.ibm.com`.

protocol

Specifies the protocol for communicating with the Web server: `http` or `https`. If blank, the default value is `http`.

port Specifies the port number of the Web server that is configured to support your WebSphere Application Server instance. The default value is 80, which is typical when the protocol is HTTP. The typical port used for the HTTPS protocol is 443.

trustStore

If you use the HTTPS protocol, specify the fully qualified path for the

keystore file (the database file that contains the public keys). Also called a *truststore*, this information enables the Secure Sockets Layer (SSL) protocol to be used for trusted communication. To specify a Windows path, escape the backward slash with a second backward slash. For example:
x:\\Application Server\\webserver.key

trustPassword

If you use the HTTPS protocol, specify a password for the specified keystore file.

username

The search application automatically sets this value to the user name that the user specifies when logging in to the search application. Specify a user name here only if you want to override the default behavior for authenticating users. This field is used only if you enabled global security in WebSphere Application Server.

password

The search application automatically sets this value to the password that the user specifies when logging in to the search application. Specify a password here only if you specified a user name. This field is used only if you enabled global security in WebSphere Application Server.

ssoCookieName

Specifies the name of the cookie that contains the single sign-on (SSO) token string. The default value is LtpaToken.

proxyHost

Specifies the fully qualified host name of a proxy server if a proxy server is required to access the search server.

proxyPort

Specifies the port for the specified proxy server host.

proxyUser

Specifies a user name to use to log in to the proxy server if the proxy server requires basic authentication.

proxyPassword

Specifies the password for the specified proxy server 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.

SEVERE

Messages that indicate a serious failure are logged. This is the default value.

INFO Informational messages are logged.

FINE Trace messages with a low amount of detail are logged. (This option corresponds to the FINE logging level in the Java `java.util.logging.Level` class.)

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 OmniFind Enterprise 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.database.icon

Specifies the path and name of an image file that indicates that the document was crawled by a JDBC database crawler. The default icon is `/images/sourceJDBC.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.seedlist.icon**

Specifies the path and name of an image file that indicates that the document was crawled by a Seed list crawler. The default icon is /images/sourceSeedlist.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.wcm.icon**

Specifies the path and name of an image file that indicates that the document was crawled by a Web Content Management crawler. The default icon is /images/sourceWorkplace.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. Documents that were crawled by Notes crawlers or Domino Document Manager crawlers that are configured to use the DIIOP protocol can also be displayed by a Lotus Notes client viewer application.

To enable documents to be displayed with a client viewer application, ensure that the following property is set to true:

```
clientViewer.show=true
```

You can customize the images that represent the client viewer application. In the following example, the Lotus Notes icon indicates that the document can be displayed with the viewer application:

```
client.notes.icon=/images/notes.gif
client.dominodoc.icon=/images/notes.gif
```

In the search results, the icon and the link to the client viewer 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\://=databasetitle filename creator
fields.dominodoc\://=librarydbtitle documentdbtitle filename author
fields.exchange\://=from creator
fields.file\://=directory filename
fields.https\://=documentID
fields.http\://=documentID
fields.jdbc\://=databasename tablename
fields.news\://=group from
fields.quickplace\://=placetitle roomtitle creator
fields.seedlist\://=author
fields.vbr\://=itemname repositorytype revisionuser
fields.wcm\://=author owner modifier
fields.web\://=
fields.wp6\://=
fields.wps\://=
```

Field icons

For data source types and documents that have fields, you can customize the images that represent fields. All fields above the document summary contain an identifying image. 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 the field contains the name of the directory that the document belongs to. The default icon is `/images/closedFolder.gif`.

**field.icon.filename**

Specifies the path and name of an image file that indicates that the field contains the file name of the document. The default icon is `/images/document.gif`.

field.icon.documentID

Specifies the path and name of an image file that indicates that the field contains the document identifier. You might want to use this blank image with Web documents, for example, to specify an image for the URL but not display an image to the user. The default icon is `/images/dot.gif`.

**field.icon.group**

Specifies the path and name of an image file that indicates that the field contains the document identifier. You might want to use this blank image with Web documents, for example, to specify an image for the URL, but not display an image to the user. The default icon is `/images/document.gif`.

**field.icon.from**

Specifies the path and name of an image file that indicates that the field identifies someone who sent the document. The default icon is `/images/author.gif`.

**field.icon.creator**

Specifies the path and name of an image file that indicates that the field identifies the document creator. The default icon is `/images/author.gif`.

**field.icon.author**

Specifies the path and name of an image file that indicates that the field identifies the document author. The default icon is `/images/author.gif`.

**field.icon.revisionuser**

Specifies the path and name of an image file that indicates that the field identifies someone who revised the document. The default icon is `/images/author.gif`.

**field.icon.owner**

Specifies the path and name of an image file that indicates that the field identifies the document owner. The default icon is `/images/author.gif`.

**field.icon.modifier**

Specifies the path and name of an image file that indicates that the field identifies someone who modified the document. The default icon is `/images/author.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 beginning 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, 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 URIs from the same source are to be collapsed in the search results. Site collapsing is available only when results are sorted by relevance. For Web and NNTP data sources, URIs that match the root site URI (such as `www.ibm.com`) are collapsed automatically. For other data source types and for Web sites with deeper path levels (such as

www.ibm.com/hr), site collapsing rules must be configured in the enterprise search administration console.

preferences.spellCorrections=Yes

Specifies that suggested spelling corrections are to be displayed when a user submits a query that contains a possibly misspelled word. Note that stop words are always removed before spelling suggestions are computed.

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=*\npreferences.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 is displayed on the basic search page. The **Refresh** button is always available for advanced searches. If you set this option to true, users can refresh the list of collections and external sources that are available to search.

If you use the Search Application Customizer, you do not need a **Refresh** button.

If you do not use the Search Application Customizer, you might want to show the **Refresh** button when you test changes that you make to the configuration 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 ESSearchServer 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 application was not started in WebSphere Application Server), then the **Refresh** button is displayed automatically to help when you troubleshoot the problem.

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.

extraQueryData.show=false

Controls the display of additional information about the query. Set this option to true if you want to see information about ACL constraints, the names of the collections and external sources that are searched, and the query language.

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.

sorting.show=true

Controls whether an option for sorting the search results is displayed. Set this option to false to suppress the **Sort by** and **Sort order** options for sorting search results.

sourceTypeFilter.show=true

Controls whether an option for filtering results by source type is displayed in the search results. Set this option to false if you do not want enable users to filter results by source type.

To prevent users from filtering results by document type, delete selected or all document type entries in the configuration file (`documentType.label=document_types`).

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.

clientViewer.show=true

Controls whether the Lotus Notes client viewer application is to be used to display a result document. Set this option to false if you do not want to use the viewer application to view Domino documents.

showDetails.show=true

Controls the display of the Show Details and Hide Details links in the search results. Set this option to false if you do not want users to be able to view additional details about result documents.

showDetailsImage.show=true

Controls the display of details about result documents in a window. Set

this option to false if you do not want users to be able to view additional details about result documents by positioning the cursor over a document URI.

numberSearchResultsReturned.show=true

Controls whether the total number of search results is displayed. Set this option to false if you do not want users to see how many documents were returned in the search results.

showMessage.error=true

Controls the display of error messages. Set this option to false if you do not want error messages to be displayed at the top of the search application.

showMessage.warning=true

Controls the display of warning messages. Set this option to false if you do not want warning messages to be displayed at the top of the search application.

showMessage.info=true

Controls the display of informational messages. Set this option to false if you do not want informational messages to be displayed at the top of the search application.

showMessage.success=true

Controls the display of success messages. Set this option to false if you do not want messages that indicate the successful completion of an action to be displayed at the top of the search application page.

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 OmniFind Enterprise 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 enterprise search 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 (Basic 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 tab 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 (SIAPI). 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
```

If the value that you specify for the document type label matches the name of a property in the `application.properties` file, then the value for the property in the `application.properties` file is displayed, not the value that you specify here. For example, if you specify `documentType.unixfs` as the label for a file type filter, then the value for the `unixfs` property in the `application.properties` file (**UNIX file system**) is displayed as the clickable file type filter name.

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.

If the value that you specify for the custom filter label matches the name of a property in the `application.properties` file, then the value for the property in the `application.properties` file is displayed, not the value that you specify here. For example, if you specify `filterCustom.hostData` as the custom filter label, the value for the `hostData` property in the `application.properties` file (**Server settings**) is displayed as the clickable custom filter name.

Several custom filters are commented out and provided as examples in the `config.properties` file.

Duplicate detection

When documents are added to the enterprise search index, analysis is done to remove duplicates so that users do not see the same document repeated in the search results. To further filter the search results, you can specify an option to suppress documents that are nearly identical to each other and prevent them from being displayed in the search results.

preferences.nearDuplicateDetection=No

Specifies that nearly duplicate documents are not filtered during query processing.

If you specify Yes, documents with similar titles and summaries are suppressed when a user views search results. A message informs users that some documents were omitted because they are similar to other documents in the result set. Users can click a link to disable the suppression and view all of the documents in the result set.

To suppress nearly duplicate documents, the Search and Index API (SI-API) Query object for the search application must specify the `setProperty` method with the `NearDuplicateDetection` string set to Yes (for example, `query.setProperty("NearDuplicateDetection", "Yes");`).

Top result analysis (bar charts for metadata fields)

You can specify options to display bar charts that represent the analysis of the top results. The default is to analyze the top 500 results. Each chart corresponds to a single metadata field, and each bar in a chart corresponds to a field value. The length of the bar indicates how frequently the field value occurs. The longer the bar, the greater the number of occurrences of that field value.

When you configure the crawl space for a crawler, you can specify options for metadata fields. To configure a top result chart for a metadata field, you must specify that the field is field searchable and that the field can be returned in the search results.

The properties that you configure for the bar chart have the following format, where *number* is a number that uniquely identifies the bar chart in the search application, *option* is the bar chart option, and *value* is the value of the option:

`topResultsChartsnumber.option=value`

topResultsCharts*number*.titleKey=*application_key*

Specifies a title for the chart, where *number* is a number that uniquely identifies the chart in the search application, and *application_key* is a label for the chart title. This label can be a key in the `application.properties` file for the search application or the value that you specify here. In the following example, the label for the chart title is specified by the value for the `topResults.mostRecentDocuments` key in the `application.properties`

`file:topResultsCharts3.titleKey=topResults.mostRecentDocuments.`

For another example, the title of the chart is the exact value that you specify here: `topResultsCharts3.titleKey=Organizations`

topResultsCharts*number*.enable=true

Specifies whether this bar chart is to be displayed when users view the search results. If you specify `false`, the chart is not displayed.

topResultsCharts*number*.fieldName=*field_name*

Specifies the name of the metadata field whose values are to be analyzed for this chart. For example, `databasetitle`. You must specify a different field name for each chart that you add (the same field cannot be used in more than one chart).

topResultsCharts*number.maxValues.collapsed=number*

Specifies the number of collapsed items to display in this chart. For example, specify 5 to show the top five most frequently occurring values in this field.

topResultsCharts*number.maxValues.expanded=number*

Specifies the number of items to show in this chart when the display of the chart is fully expanded. For example, specify 10 to show no more than 10 different field values when the chart is expanded.

topResultsCharts*number.fieldValueSeparator=character*

Specifies a character that delimits values in the field to be analyzed. For example, if a field contains multiple values that are separated by a semicolon (such as agent;seller;broker), then you can use this property to identify the semicolon (;) as the field value separator so that each value can be added to the bar chart as a separate item. Without this option, the entire field value is added to the chart as a single item.

topResultsCharts*number.canUserChangeFieldName=true*

Specifies whether the user can select a different field when viewing the search results and see the top results for that field. If you specify false, users cannot select a different field to be analyzed when viewing the search results.

If you specify true, a list of all fields that were found in the initial top 500 results is displayed along with results for the current field. If the user selects a field from this list, the chart label changes to **Dynamic field chart** and the bar chart results for the selected field are displayed until the user selects a different field or closes the browser. The next time the user runs the search application, results for the original field are displayed.

topResultsCharts*number.width=number*

Specifies the display width of the bar chart in pixels. For example, 300.


topResultsCharts*number.barheight=number*

Specifies the height of each bar in the bar chart in pixels. For example, 10.

topResultsCharts*number.color=#color_code*

Specifies the hexadecimal code for the base color of the bar in the bar chart. The default value is blue (#0309C0).

topResultsCharts*number.color.gradient=#color_code*

Specifies the gradient color of the bar in the bar chart. The default value is turquoise (#00FFFF). As the number of results for a particular field value move from low to high frequency, the color of the bar changes hue from the value specified for the color option to the value specified for the color.gradient option. For example: 

topResultsCharts*number.sortKey=frequency*

Specifies how items in the bar chart are to be sorted. Supported values:

none Items in the bar chart are not sorted.

label Items in the bar chart are sorted according to the field value. If you configure custom labels with the `topResultsCharts.number listOfLabels.prefixKey=field_name` property, the items in the bar chart are also sorted by label name.

frequency

Items in the bar chart are sorted according to the number of results returned per field value.

topResultsCharts*number.sortOrder=descending*

For items that are sorted by label or frequency, specifies the sort order.
Supported values:

ascending

Items that are sorted by label are listed in alphabetical order from a to z. For items that are sorted by frequency, field values that occur the greater number of times appear lower in the list than values that occur less frequently.

descending

Items that are sorted by label are listed in reverse alphabetical order from z to a. For items that are sorted by frequency, field values that occur the greater number of times appear higher in the list than values that occur less frequently.

topResultsCharts*number.listOfLabels.prefixKey=field_name*

Optional. Enables you to specify information is to always be displayed for certain field values, where *field_name* identifies the field whose value is analyzed for this chart. The labels that you specify for this property are always displayed in the bar chart, even if there are no occurrences of the field value in the search results.

You can configure any number of labels for a field. For each label, you specify two properties that have the following format:

```
field_namenumber.value=value  
field_namenumber.displayValue=display_value
```

where:

field_name

Is the name of the field that you are configuring labels for.

number

Is a number that uniquely identifies the label.

value

Is a value that you want to show in the analysis results.

display_value

Is the label to display in the bar chart. This label can be a key in the `application.properties` file for the search application or the value that you specify here.

For example:

```
topResultsCharts1.listOfLabels.prefixKey=databasetitle
```

```
databasetitle1.value=JK Enterprises Articles & Papers  
databasetitle1.displayValue=Articles & papers
```

```
databasetitle2.value=JK Enterprises Blank Forms  
databasetitle2.displayValue=Blank forms
```

```
databasetitle3.value=JK Enterprises Bulletins & Guidelines  
databasetitle3.displayValue=Bulletins & Guidelines
```

Top result analysis (custom HTML)

You can specify options to display top result charts by extending the `com.ibm.es.searchui.charts.servlet.AbstractDynamicChart` API with a custom Java class. If you use this approach for displaying results, you can use HTML to format the return of any search results, not just queries that search metadata fields.

Results can be returned for documents in enterprise search collections or from searches of external repositories and Web sites.

You can configure any number of charts for a search application. Use the following property to assign a title to each chart:

topResultsCharts*number.titleKey=application_key*

Specifies a title for the chart, where *number* is a number that uniquely identifies the chart in the search application, and *application_key* is a label for the chart title. This label can be a key in the `application.properties` file for the search application or the value that you specify here. In the following example, the label for the chart title is specified by the value for the `topResults.mostRecentDocuments` key in the `application.properties`

```
file:topResultsCharts3.titleKey=topResults.mostRecentDocuments
```

Each item in the chart corresponds to a single search result value and comprises a set of properties that have the following format, where *number* is a number that uniquely identifies the chart, *option* is the chart option, and *value* is the value of the option:

```
topResultsChartsnumber.option=value
```

topResultsCharts*number.enable=true*

Specifies whether this chart is to be displayed when users view search results. If you specify `false`, the chart is not displayed.

topResultsCharts*number.maxValues.collapsed=number*

Specifies the number of collapsed items to display in this chart. For example, specify 5 to show the top five most frequently occurring results that match the criteria.

topResultsCharts*number.width=number*

Specifies the display width of the chart, in pixels. For example, 400.

topResultsCharts*number.dynamicChartClass=custom_class*

Specifies the name of your custom Java class that extends the `com.ibm.es.searchui.charts.servlet.AbstractDynamicChart` API and defines how the output is to be displayed in the chart. For example:

```
topResultsCharts.3.dynamicChartClass=com.ibm.es.searchui.charts.servlet.  
DynamicMostRecentDocuments
```

The following properties, which are included in the default `config.properties` file for the sample search application for example purposes, are used by the sample `DogearSearchResults` Java class. See the Dogear API documentation for information about query parameters that you might want to include in your custom search application.

You cannot set these properties by using the Search Application Customizer:

topResultsCharts*number.xsl.fileName=style_file*

Specifies the path and name of an XSL style sheet that is to be used to format the display of the top results in the chart. For example, `/styles/dogear.xsl`.

topResultsCharts*number.url=url*

Specifies the URL to be searched.

topResultsCharts*number.url.parameters=ps=number*

Specifies the page size. For example, `ps=3` limits the page size to 3.

This is simply a parameter string as defined in the Dogear REST API.

Top result analysis (maximum number of results)

`topResult.resultSize=number`

The default and maximum value for the number of results to return from top result analysis is 500. You can decrease this value, but you cannot increase it. For example, you might want to specify a lower number if you experience problems with requests timing out during top result analysis. You cannot set this property by using the Search Application Customizer.

Related concepts



Setting query properties



Java classes for showing top results

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 configuration 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 configuration file, `config.properties`, that is deployed with the application.

For your changes to become effective, you must stop and restart the `ESSearchServer` application in WebSphere Application Server.

Procedure

To edit the sample search application properties:

1. Log in to the search server as the enterprise search administrator.
2. 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 OmniFind Enterprise Edition installation directory on the search server:

```
ES_INSTALL_ROOT/installedApps/ESSearchApplication.ear/  
    ESSearchApplication.war/WEB-INF/config.properties
```

3. 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.
4. Stop and restart the `ESSearchServer` application.

AIX, Linux, or Solaris

```
./stopServer.sh ESSearchServer  
./startServer.sh ESSearchServer
```

Windows

```
stopServer ESSearchServer  
startServer ESSearchServer
```


These scripts are located in the WAS_INSTALL_ROOT/AppServer/bin directory:

- For WebSphere Application Server version 5, the default installation path is /usr/WebSphere on AIX systems, /opt/WebSphere on Linux or Solaris systems, or C:\Program Files\WebSphere on Windows systems.
- For WebSphere Application Server version 6, the default installation path is /usr/IBM/WebSphere on AIX systems, /opt/IBM/WebSphere on Linux or Solaris systems, or C:\Program Files\IBM\WebSphere on Windows systems.

Related concepts

 [Setting query properties](#)

 [Java classes for showing top results](#)

Related tasks

[“Configuring the search servers to accept only secure \(SSL\) requests”](#) on page 238

Customizing search applications

The Search Application Customizer is a graphical interface that you can use to customize search applications for enterprise search or your custom search applications.

Restrictions

The Search Application Customizer is available as a stand-alone application. You cannot launch the Search Application Customizer within WebSphere Portal to customize search applications that run as portlets. To customize search applications that run as portlets, you must edit properties and configure the Portlet instance with the WebSphere Portal Administration interface.

About this task

The Search Application Customizer enables you to visualize changes that you want to make and to modify a search application without editing the configuration file. For example, you can change the banner and background images, change the layout of the search interface, and specify options for working with search results.

When you make selections in the Search Application Customizer, the effects of your selections are displayed. When you save the changes, you update the configuration file for the search application.

For your changes to become effective, you must stop and restart the ESSearchServer application in WebSphere Application Server.

Tip:

For detailed examples of how to use the Search Application Customizer, see the IBM Redbook, *IBM OmniFind Enterprise Edition Configuration and Implementation Scenarios*.

Procedure

To customize a search application:

1. To customize the sample search application for enterprise search, type the URL for the Search Application Customizer in a Web browser. For example:

`http://SearchServer.com/ESSearchApplication/palette.do`

`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/palette.do`

Tip: If you are enterprise search administrator, you can also open the Search Application Customizer by selecting the **Search Customizer** option in the enterprise search administration console.

2. To customize a custom search application, type the URL for the Search Application Customizer, and append the name of the configuration file for your search application. For example:

`http://SearchServer.com/ESSearchApplication/palette.do?configFile=/WEB-INF/myConfig.properties`

If the file that you specify does not exist, values in the `config.properties` file for the sample search application are displayed.

Tip: You can also specify the configuration file that you want to use with a search application by clicking **Load** after you start the Search Application Customizer and specifying the name of the file.

3. If global security is enabled in WebSphere Application Server, log in with a valid user ID and password.
4. Select the options that you want to customize, such as information about the search server, the types of messages to be displayed, query and search result options, and the names of images that you want to use to identify different types of documents in the search results.
For help with specifying options, click **Help for the customizer**. To see the effect of some changes, such as how search results are presented, type a query and click **Search**.
5. When you are satisfied with the selections that you made, click **Save** to update the configuration file. If you click **Reset**, options displayed in the Search Application Customizer are restored to values in the last saved version of the configuration file.
6. On the search server, log in as the enterprise search administrator and stop and restart the `ESSearchServer` application.

AIX, Linux, or Solaris

```
./stopServer.sh ESSearchServer  
./startServer.sh ESSearchServer
```

Windows

```
stopServer ESSearchServer  
startServer ESSearchServer
```

These scripts are located in the `WAS_INSTALL_ROOT/AppServer/bin` directory:

- For WebSphere Application Server version 5, the default installation path is `/usr/WebSphere` on AIX systems, `/opt/WebSphere` on Linux or Solaris systems, or `C:\Program Files\WebSphere` on Windows systems.
- For WebSphere Application Server version 6, the default installation path is `/usr/IBM/WebSphere` on AIX systems, `/opt/IBM/WebSphere` on Linux or Solaris systems, or `C:\Program Files\IBM\WebSphere` on Windows systems.

Related tasks

“Configuring the search servers to accept only secure (SSL) requests” on page 238

Cloning the sample search application

To use the sample search application as a model for creating your own search applications, you can copy the `config.properties` file or use the Search Application Customizer.

About this task

To facilitate the creation of custom search applications, you can clone configuration options that you specify for the sample search application, and then customize the options that you want to change.

After you clone the sample search application, you specify the name of your configuration file to start the new search application. You also specify the name of your configuration file to customize the new search application with the Search Application Customizer.

By cloning the sample search application, you can quickly create search applications for specific purposes or audiences. For example, you might create one search application for employees in the human resources department, and another search application for salespeople.

For your changes to become effective, you must stop and restart the `ESSearchServer` application in WebSphere Application Server.

Procedure

To clone the sample search application:

1. If you want to edit a configuration file:
 - a. Copy the `config.properties` file for the sample search application and rename it.

The `config.properties` file is installed in the following location, where `ES_INSTALL_ROOT` is the OmniFind Enterprise Edition installation directory on the search server:

```
ES_INSTALL_ROOT/installedApps/ESSearchApplication.ear/  
ESSearchApplication.war/WEB-INF/config.properties
```

The file that you create must exist in the `WEB-INF` subdirectory.
 - b. Edit the properties that you want to use in your custom search application and save the file. At a minimum, you must change the `applicationName` property to specify the name of your search application.
2. If you want to clone the sample search application by using the Search Application Customizer:
 - a. Start the Search Application Customizer by appending the name of the configuration file that you want to create. In the following example, a file named `myNewFile.properties` is created:

```
http://ESServer.com/ESSearchApplication/palette.do?configFile=/WEB-INF/myNewFile.properties
```

Because the file does not yet exist, the values specified in the sample search application configuration file, `config.properties`, are used.

Tip: You can also create a configuration file for a search application by clicking **Load** after you start the Search Application Customizer and specifying the name of a file. The file is created when you click **Save** to save your customization options.

- b. If global security is enabled in WebSphere Application Server, log in with a valid user ID and password.
 - c. Specify a name for your search application, specify the options that you want to use for searching collections, and click **Save**. The changes that you specify are saved to your new configuration file in the WEB-INF subdirectory. For help with specifying options, click **Help for the customizer**. To see the effect of some changes, such as how search results are presented, type a query and click **Search**.
3. On the search server, log in as the enterprise search administrator and stop and restart the ESSearchServer application.

AIX, Linux, or Solaris

```
./stopServer.sh ESSearchServer  
./startServer.sh ESSearchServer
```

Windows

```
stopServer ESSearchServer  
startServer ESSearchServer
```

These scripts are located in the WAS_INSTALL_ROOT/AppServer/bin directory:

- For WebSphere Application Server version 5, the default installation path is /usr/WebSphere on AIX systems, /opt/WebSphere on Linux or Solaris systems, or C:\Program Files\WebSphere on Windows systems.
- For WebSphere Application Server version 6, the default installation path is /usr/IBM/WebSphere on AIX systems, /opt/IBM/WebSphere on Linux or Solaris systems, or C:\Program Files\IBM\WebSphere on Windows systems.

Analyzing top results

You can help users refine a set of search results by configuring options to analyze the top results.

Top result analysis essentially classifies the current set of search results according to the frequency with which the analyzed values occur. Users can filter the search results by selecting one of the analyzed values, which automatically adds the selected value as a new keyword in the search criteria. The value of top result analysis is that it enables users to fine tune the search results without having to use complex or advanced search syntax.

In an enterprise search application, you can use bar charts to graphically show which metadata field values occur most frequently in the search results. You can also create a custom Java class to show the top results, including results from non-enterprise search sources, in any HTML format.

Restrictions

Ensure that the appropriate fonts for your language are installed on the computer where you run WebSphere Application Server and the search application. This step is necessary to ensure that when the bar chart is generated, that the font is set to a font that supports the characters in the chart label. This issue is especially critical

for Asian languages. If you install the fonts after you install WebSphere Application Server, you must restart WebSphere Application Server for the changes to become effective.

If you run the search application as a stand-alone application, you can configure top result charts by using the Search Application Customizer or by editing the configuration file for the search application. If you run the search application as a portlet within WebSphere Portal, you must edit the properties and configure the Portlet instance with the WebSphere Portal Administration interface. You cannot use the Search Application Customizer to configure options for top result analysis.

About this task

You can graphically represent the top results by showing the most frequently occurring metadata field values in a bar chart. You can also use a Java class to extend the search application and show the top results in a different format, such as using HTML to present the top results in an unordered list.

Bar charts for metadata fields

You can specify options to analyze metadata fields and show the results of that analysis in bar charts. The charts are displayed in addition to the results of the user query. Each chart corresponds to a single metadata field (such as the document size, author, date, and so on) and each bar in a chart corresponds to a specific field value.

The length of the bar indicates the number of documents that contain a particular field value relative to other documents that contain different values in that field. The longer the bar, the greater the number of occurrences of that field value. Users can fine tune the results by clicking a bar in the bar chart. The field value that is represented by the selected bar is added as an additional query term, and the new search results are narrowed by the additional search criteria.

When you configure the crawl space for a crawler, you can select an option to specify search options for metadata fields. For example, you can specify whether a metadata field can be searched as free text, searched by field name, shown in the search results, searched as parametric data, and so on. To show charts for metadata fields in the search results, you must configure metadata field options for the crawler. At a minimum, you must specify that the field is field-searchable and that it can be shown in the search results.

If you select the **Complete match** option when you configure options for a metadata field, a complete match query is run when the user selects a bar from the bar chart. A complete match search specifies that results are to be returned only when the query term matches the entire field value. If the field contains less content or additional content, a match does not occur.

If you do not select the **Complete match** option when you configure options for a metadata field, a fielded query is run when the user selects a bar from the bar chart. In this case, the additional query term must occur in the field, but it does not have to match the entire value of the field.

Results formatted with HTML

You can extend the `com.ibm.es.searchui.charts.servlet.AbstractDynamicChart` API with a custom Java class. If you use this approach for displaying results, you can use HTML to format the display of any search results, not just queries that

search metadata fields. Results can be returned for documents in enterprise search collections or from searches of external repositories and Web sites.

Each chart corresponds to a single Java class, and the class specifies how the top results are to be presented in the chart. For example, `com.ibm.es.searchui.charts.servlet.DynamicMostRecentDocuments`, which is a sample class provided with the enterprise search sample code, presents the top results as an unordered list. The documents are sorted by date, and only the document titles and dates are shown.

Another example class provided with enterprise search, `com.ibm.es.searchui.charts.servlet.DogearSearchResults`, shows how you might provide users with a list of bookmarks from Lotus Connections Dogear that are related to the user's query.

When users click a search result in your custom-formatted output, the document is displayed in a new browser window.

Procedure

This procedure shows you how to use the Search Application Customizer to specify that metadata fields are to be analyzed. The most frequently occurring metadata field values are graphically presented in a bar chart.

1. Open the Search Application Customizer. If it is not already displayed, load the configuration file for the search application that you want to customize and click **Apply**.
2. Scroll down to **Top result charts** and click **Add Chart**.
3. When the list of chart options is displayed, leave the **Custom chart** check box clear (select this option only if you created a custom Java class to analyze and return top results), and select the **Enable chart** check box to ensure that users see the bar chart when they view search results.
4. In the **Chart title** field, specify a descriptive label for the chart. The value that you specify here replaces the **New chart row** placeholder text.
5. Decide whether you want to select the **Enable dynamic field selection** check box. If you enable this option, users can select different fields when viewing the search results and see the top result analysis for that field.

You might want to use this option to allow users to fine-tune a set of search results. For example, a user might search a database to find information about female employees. After entering a query to search a field that specifies employee gender (such as `sex:F`), the values for the top results are displayed as bars in the bar chart. Next, the user selects `job` from the list of fields that exist in that initial result set. The bar charts now show results for the top jobs that females perform. Finally, the user selects `designer` from the list of available fields. The results now provide information about female employees who work as designers.

6. In the **Metadata field name** field, type the name of the metadata field whose values are to be analyzed for this chart.
7. If a field contains multiple values that are separated by a delimiting character, specify the character in the **Field value separator** field. For example, if a field contains two values, such as a customer's first name and last name separated by a semicolon, you can specify the semicolon here to add each value to the bar chart separately. If you do not identify the separator character, the entire field value is analyzed as a single item.

8. Specify options for the display of the bar chart, such as the size and color of the bars and how many bars are to be displayed when the chart is collapsed or expanded.
9. Specify options for sorting the results of the analysis. For example, you can sort by field values or by how frequently the values occur.
10. If you want to ensure that information is always displayed for certain field values, even if there are no occurrences of that field value in the search results, click **Add Row** in the **Custom labels** area. Specify the field value that is to be shown in the bar chart even if no occurrences of that value are returned in the search results, and specify a label for this bar in the bar chart. For example, if you always want to see whether a competitor, such as JK Enterprises, occurs in the search results, even if the frequency does not qualify to be shown as a top result, specify JK Enterprises as the source value to be analyzed, and then specify a descriptive label to identify this bar in the bar chart.
11. Click **Apply** next to the chart title to apply the options that you specified for this chart.
12. If search results are already displayed in the search application area, top result analysis is applied to the current search and your new chart is displayed. Otherwise, enter a query to test the display of the bar chart.
13. If you are satisfied with the chart, click **Save** to update the configuration file for the search application.

Related concepts

 [Setting query properties](#)

 [Java classes for showing top results](#)

Accessing search applications

You access a search application by specifying a URL in a Web browser.

Before you begin

You must configure the 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 a 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. To start a custom search application, type the URL for the sample search application and append the name of the configuration file for your search application. For example:
`http://SearchServer.com/ESSearchApplication/search.do?configFile=/WEB-INF/myConfig.properties`
If the file that you specify does not exist, the sample search application for enterprise search is displayed.
3. If global 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 logging in to them.
If the crawler supports single sign-on (SSO) security, you can search secure domains without creating a user profile.
4. On the Search page, submit a query. All collections and external sources that are selected to be searched on the Preferences page will be searched.

Configuring the search servers to accept only secure (SSL) requests

You can disable the HTTP interface on the search servers, and configure the servers to accept search requests only through SSL and the secure HTTPS interface.

About this task

To configure the search servers to use only the Secure Sockets Layer (SSL) protocol when processing search requests, you should disable the HTTP interface. You must also ensure that the same keystore file is stored on both search servers and on any client computers, such as the WebSphere Portal server where the Search portlet for enterprise search is installed. The keystore file, which is also called a *truststore*, contains public keys that enable SSL to be used for trusted communication.

Procedure

To configure the search servers to accept only secure requests:

1. Log in as the enterprise search administrator. For a multiple server installation, do the following steps on the index server:
 - a. Stop the enterprise search system:
`esadmin system stopall`
 - b. Edit the `ES_NODE_ROOT/master_config/nodes.ini` file.
 - c. Change the `node_ID.searchserverport` value from the HTTP port (usually 80) to the HTTPS port (usually 443), and save the file. For a multiple server installation, update both `node_ID.searchserverport` values (one for each search server).
2. Do the following steps to update the search server. For a multiple server installation, do the following steps on both search servers:
 - a. For a multiple server installation, log in to the search server as the enterprise search administrator.
 - b. Edit the `ES_NODE_ROOT/nodeinfo/es.cfg` file.
 - c. Update the `TrustStore` property to specify to the fully qualified path for the SSL keystore file.

- d. Update the HTTPProtocol property to specify HTTPS, and then save the file.
- e. Enter the following command, where *trustStore_password* is the password for the keystore file. This command encrypts the password value and updates the TrustStorePassword value in the *es.cfg* file.

AIX, Linux, or Solaris

```
eschangetrustpw.sh trustStore_password
```

Windows

```
eschangetrustpw trustStore_password
```

- f. Ensure that the *trustStore* and *trustPassword* properties in the *config.properties* file for the search application specify the correct fully qualified path and password for the keystore file. You can verify or change this information by editing the *config.properties* file or by using the Search Application Customizer.
 - g. If you use the Search portlet for enterprise search, ensure that the *trustStore* and *trustPassword* portlet parameters specify the correct fully qualified path and password for the keystore file. Use the portlet management options in the WebSphere Portal administration interface to verify or change this information.
3. Restart the enterprise search system:

```
esadmin system startall
```

Related tasks

“Editing the sample search application properties” on page 230

“Customizing search applications” on page 231

“Setting up enterprise search in WebSphere Portal version 5.1” on page 327

“Setting up enterprise search in WebSphere Portal version 6” on page 332

Configuring the search servers to accept requests through a proxy server

You can configure the search server to accept requests through a through a proxy server.

Procedure

To be able to submit requests to the search servers through a proxy server:

1. Log in as the enterprise search administrator and stop the enterprise search system. For a multiple server installation, log in on the index server.


```
esadmin system stopall
```
2. Edit the *ES_NODE_ROOT/master_config/nodes.ini* file.
 - a. Change the *ProxyServer* property to specify the fully qualified host name for the proxy server.
 - b. Change the *ProxyServerPort* property to specify the port for the proxy server.
 - c. Optional: If the proxy server requires all requests to be authenticated, then update the *ProxyServerUserName* property to specify a valid user name for the proxy server.
 - d. Optional: If the proxy server requires all requests to be authenticated, then enter the following command, where *proxyServer_password* is the password

for the specified proxy server user name. This command encrypts the password value and updates the ProxyServerUserPassword value in the es.cfg file.

AIX, Linux, or Solaris

```
exchangeproxypw.sh proxyServer_password
```

Windows

```
exchangeproxypw proxyServer_password
```

3. Ensure that the proxyHost and proxyPort properties in the configuration file (.properties file) for the search application specify the correct fully qualified host name and port number for the proxy server.

If the proxy server requires authentication, ensure that the proxyUser and proxyPassword properties specify a valid user name and password for the proxy server. You can verify or change this information by editing the configuration file or by using the Search Application Customizer.

4. If you use the Search portlet for enterprise search, ensure that the proxyHost and proxyPort portlet parameters specify the correct fully qualified host name and port number for the proxy server.

If the proxy server requires authentication, ensure that the proxyUser and proxyPassword properties specify a valid user name and password for the proxy server. Use the portlet management options in the WebSphere Portal administration interface to verify or change this information.

5. Restart the enterprise search system:

```
esadmin system startall
```

Support for 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:

- Databases that support the Java database connectivity (JDBC) protocol. Only IBM DB2, Oracle, Microsoft SQL Server 2000, and Microsoft SQL Server 2005 databases are supported. A separate external source is created for each table that you enable for searching.

Restriction: Support for SQL Server 2005 databases is limited to tables that do not contain a Variant data type. The JDBC driver for SQL Server 2005 is not supported on AIX systems.

- 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

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.

The JDBC driver for Microsoft SQL Server 2005 is not supported on AIX systems.

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.

For information about SQL Server 2000 drivers, see <http://www.microsoft.com/downloads/details.aspx?familyid=07287B11-0502-461A-B138-2AA54BFDC03A&displaylang=en>. For information about SQL Server 2005 drivers, see <http://www.microsoft.com/downloads/details.aspx?familyid=e22bc83b-32ff-4474-a44a-22b6ae2c4e17&displaylang=en>.

Procedure

To add an external source to the system:

1. To include JDBC databases in an enterprise search system, do the following steps before you add an external source. You need to do this step, which enables the system to locate the appropriate JDBC drivers, one time.
 - a. On the crawler server, log in as the enterprise search administrator.
 - b. Edit the `ES_INSTALL_ROOT/configurations/interfaces/discovery__interface.ini` file, and specify the `CLASSPATH` and `LD_LIBPATH` environment variables to include the classpath to the JDBC drivers and the path to the library files.
 - c. Edit the `ES_INSTALL_ROOT/configurations/interfaces/customcommunication__interface.ini` file, and specify the `CLASSPATH` and `LD_LIBPATH` environment variables to include the classpath to the JDBC drivers and the path to the library files.
 - d. Optional: To use an Oracle JDBC driver for local or cataloged databases, add the Oracle library path to the `LD_LIBPATH` environment variable (for example, `LD_LIBPATH=.../home/oracle/OraHome1/lib32` and edit `escrset.sh` file to specify the library path and export the Oracle installation directory variable. For example:

```
ORACLE_HOME=/home/oracle/OraHome1
export ORACLE_HOME
```
 - e. Restart the enterprise search system, including the common communication layer (CCL):

AIX, Linux, or Solaris

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

Windows command prompt

```
esadmin stop
stopccl
startccl
esadmin start
```

Windows Services administrative tool

To start CCL in the background:

- 1) Enter `esadmin stop`.

- 2) Launch Windows Services: **Start** → **Programs** → **Administrative Tools** → **Services**.
 - 3) Right-click **IBM OmniFind Enterprise Edition** and click **Stop**. After the service stops, click **Start**.
 - 4) Enter `esadmin start`.
2. Click **External Sources** to open the External Sources view.
 3. Click **Add External Source**.
 4. Select the type of external source that you want to add, either LDAP server or JDBC database.
 5. 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 and Oracle databases:

DB2: Legacy JDBC Driver

Driver name: `COM.ibm.db2.jdbc.app.DB2Driver`

Sample location: `db2_install_root/java/db2java.zip`

DB2: Universal JDBC Driver

Driver name: `com.ibm.db2.jcc.DB2Driver`

Sample locations:

`db2_install_root/java/db2jcc.jar`

`db2_install_root/java/db2jcc_license_cu.jar`

Oracle Driver name: `oracle.jdbc.driver.OracleDriver`

Sample location: `oracle_home/jdbc/lib/ojdbc14.jar`

Microsoft SQL Server 2000

Driver name: `com.microsoft.jdbc.sqlserver.SQLServerDriver`

Sample locations:

`mssql_jdbc_home/lib/mssqlserver.jar`

`mssql_jdbc_home/lib/msbase.jar`

`mssql_jdbc_home/lib/msutil.jar`

Microsoft SQL Server 2005

Driver name: `com.microsoft.sqlserver.jdbc.SQLServerDriver`

Sample location: `install_dir/sqljdbc_1.0/locale/sqljdbc.jar`

For example: `install_dir/sqljdbc_1.0/enu/sqljdbc.jar`

6. 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](#)

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](#)

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.

Tip:

For detailed examples of how to configure security for enterprise search, see the IBM Redbook, *IBM OmniFind Enterprise Edition Configuration and Implementation Scenarios*. Scenarios show you how to enable global security in WebSphere Application Server with an LDAP repository, set up the identity management component, and configure various crawlers to ensure that document-level security is enforced.

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 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 WebSphere Application Server security mechanisms to authenticate users who 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 enterprise search global analysis processes apply different rules for indexing duplicate documents.
- You can configure options to enforce document-level security, such as associating security tokens with documents as they are crawled, requiring current credentials to be validated during query processing, and specifying whether anchor text in Web documents is to be indexed.
- You can enforce security by mapping search applications (not individual users) to the collections and external sources that they can search. You then use standard access control mechanisms to permit or deny users access to search applications.

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.

Document-level security

When you configure crawlers for a collection, you can enable document-level security. For example, you can specify options to associate

security tokens with data as the data is collected by crawlers. Your search applications 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 capability can protect against instances in which a user's credentials change after a document and its security tokens are indexed.

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. When you configure a Web crawler, you can specify whether you want to exclude the anchor text from the index if the text links to a document that the Web crawler is not allowed to crawl.

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 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.

Installation security

The installation program for OmniFind Enterprise Edition establishes an environment for enforcing security when users administer or search enterprise search collections.

Enterprise search administrator ID

During the installation process, the installer is prompted for a user ID and password to use for the enterprise search administrator. The system uses the specified credentials to:

- Authenticate the enterprise search administrator when administrative tasks are performed.
- Create the enterprise search internal database.
- Start all enterprise search sessions or processes.

The user ID that is specified during installation must be a valid operating system user ID with system administrative privileges. The installation program stores the

credentials, appropriately encrypted, into a properties file on each enterprise search server.

Encryption

To protect sensitive data, encryption is used to encode the authentication data portion of all messages that are transmitted through the enterprise search system. This process incurs little overhead because only the authentication IDs and passwords are encrypted. All passwords that are stored by the system (in configuration files, the enterprise search database, and so on) are also encrypted.

WebSphere Application Server security

If WebSphere Application Server was not previously installed on the search server, then the installation program silently installs the product with global security disabled. If global security is later enabled in WebSphere Application Server, then the WebSphere Application Server is responsible for authenticating the enterprise search administrator.

If you enable global security, you must add the enterprise search administrator ID and password that were specified during installation to the WebSphere Application Server user registry, such as a Lightweight Directory Access Protocol (LDAP) directory.

If you enable global security after you install OmniFind Enterprise Edition, you must update configuration values and run a command, **eschangewaspw**, to encrypt and store the WebSphere Application Server credentials in an enterprise search properties file.

Authentication versus access control

To protect content from unauthorized users, and to control access to administrative functions, enterprise search supports user authentication and authorization (access controls).

Authentication

Authentication is any process by which a system verifies the identity of a user who wishes to access the system. Because access control is typically based on the identity of the user who requests access to a resource, authentication is essential to effective security.

The authentication of enterprise search users is implemented through credentials which, at a minimum, consist of a user ID and password.

To authenticate users who access the administration console, enterprise search leverages the authentication support that is provided with WebSphere Application Server.

Authorization (access control)

Authorization is any mechanism by which a system grants or revokes the right to access some data or perform some action. Often, a user must log in to a system by using some form of authentication. Access control mechanisms determine which operations the user can or cannot do by comparing the user's identity to an access control list (ACL). Access controls encompass:

- File permissions, such as the right to create, read, edit or delete a file.
- Program permissions, such as the right to execute a program.
- Data permissions, such as the right to retrieve or update information in a database.

Administrative roles

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

When OmniFind Enterprise Edition (OmniFind Enterprise Edition) is installed, 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 OmniFind Enterprise 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 collections that they are authorized to administer. 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 collections that they are authorized to administer. 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 collections that they are authorized to administer. 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.

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.

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.

After you create a search application, a search application ID enables you to specify which collections and external sources the search application can search, and which users can access the search application.

Duplicate document analysis and collection security

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 the security controls for each document are evaluated so that users search only the documents with security tokens that match their credentials. Two documents might be nearly identical in content, but use different access control lists to enforce security.

For example, for two duplicate documents, document_A and document_B, assume that a user has access rights only to document_B. If document_B is eliminated by duplicate detection, then the user cannot see the document in the search results because of the access constraints that are in place for document_A.

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.

Search application identifiers

The ability to search different collections is controlled by mapping search applications to the collections and external sources that they can search. An application named Default enables the sample search application to be used as provided to search all collections and external sources.

All search applications are required to pass a valid application name (APPID) to the enterprise search application programming interface (API). Only the collections and external sources associated with this APPID can be searched by the search application.

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.

The sample search application (ESSearchApplication) has a properties file that specifies the application name to use. The default location for this properties file is `ES_INSTALL_ROOT\installedApps\ESSearchApplication.ear\ESSearchApplication.war\WEB-INF\config.properties`.

The initial value for the application name is Default. If you change this value, you change the list of collections and external sources that the ESSearchApplication application can search.

To control which users can search which collections, you must associate users (or user groups) with the client application by using standard access control features of WebSphere Application Server, similar to how you might use these features to restrict access to a URL. 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 and index API overview](#)

Document-level security

If security is enabled for a collection when it is created, you can configure document-level security controls. Document-level access control ensures that the search results contain only documents that the user who submitted the search request is authorized to see.

An enterprise search system supports many approaches for configuring document-level security controls:

- Documents can be pre-filtered and associated with security tokens before they are added to the index.
- For some data types, search results can be post-filtered to validate the user's login credentials against current access control data. The enterprise search identity management component can encrypt the various credentials that users need to access different repositories, and store the encrypted credentials in profiles. If the sources to be searched are protected by a product that provides single sign-on (SSO) security, the identity management component can control access to documents without requiring users to create profiles.
- For most crawler types, a custom Java class (plug-in) can be used to associate security tokens with documents in the index.
- For documents crawled by a Web crawler, the anchor text in documents that contain links to forbidden documents can be excluded from the index.

Related concepts

 Application security

Pre- and post-filtering of search results

There are two distinct approaches to filtering documents to ensure that search results contain only the documents that the user who submitted the search request is authorized to view.

- The first approach is to replicate the document's native access control lists (ACLs) at crawl time into the index and to rely on the search engine to compare user credentials to the replicated document ACLs. Pre-filtering the documents, and controlling which documents are added to the index, results in the best performance. However, it is difficult to model all of the security policies of the various back-end sources in the index and implement comparison logic in a uniform way. This approach is also not as responsive to any changes that might occur in the source ACLs.
- The second approach is to post-filter documents in the result set by consulting the back-end sources for current security data. This approach allows the contributing back-end sources to be the final arbiters of the documents returned to the user, and ensures that the result set reflects current access controls. However, this approach results in degraded search performance because it requires that connections exist with all of the back-end sources. If a source is not accessible, then links to documents must be filtered out of the result set along with documents that the user is not authorized to view.

Important: In a multiple server configuration, post-filtering is done at the crawler server for some source types. If the crawler server is brought down for maintenance, users experience no results when they query enterprise search

collections. In addition, no results are returned if the back-end servers that are required to control access are not accessible.

For enterprise search, support for enforcing access controls relies on a combination of these two approaches. The design provides optimum performance while maintaining the precise security policies of the originating document repositories. By storing high-level access control data in the index, the system can provide an interim (potentially smaller) result set which can then be post-filtered to verify current access controls. The assumption is that if the user has access to the repository that owns the document, then chances are that the user also has access to the document.

The access control data that is stored in the index varies with the crawler type. For example, the Notes crawler can store database- and server-level access controls, and the QuickPlace crawler can store access controls for servers, places, and rooms.

All data source types in an enterprise search system support the ability to index native access control lists during crawl time. Some data source types also support the ability to post-filter the result set and verify the user's current credentials (this type of support is provided through native security mechanisms or the enterprise search identity management component).

This two-pronged security design encompasses the following tasks:

- Extracting native ACL information during crawl time.
- Storing server and database ACL information in the index.
- Creating the user's security context when the user logs in or when the session is initialized. This task must account for the different identifiers that a single user must use to access the various back-end sources.
- Processing the search with the user's security context and producing an interim result set that contains only those documents that the user has access to at the repository level.
- Post-filtering the interim result set by consulting the back-end sources that contributed documents to the result set for current native ACL information.

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 assigned a public token that makes the document available to everyone. If security is enabled for the collection, the public token can be replaced with a value that is provided by the administrator or with a value that is extracted from a field in the crawled document. 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.

When a collection administrator configures a crawler, the administrator can specify security options for individual tables, file systems, and so on (that is, different security rules can be configured for different data sources in the crawl space). The administrator can:

- Specify that the documents are public (all users can search the documents)
- Assign user-defined security tokens to each document
- Extract security tokens from a field in the crawled data, and assign the extracted token to each document

Security tokens (with the exception of the default, public token) are completely user-defined. 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.

For example, an administrator might specify that the hrDeptName field is to be used to control access to documents that are crawled by a Notes crawler. The administrator might also specify that if that field does not exist in a document, or if that field does not contain security data, then two user-defined tokens, hrgroup1 and hrgroup2, are to be used to control access to documents.

The security tokens are made available to the crawler through the crawler's configuration file. For each document, the crawler provides the security token value as metadata. The indexing component reads the security token and applies it to the posting information for the document in the index. If the administrator for the native data source updates the access control list, the updated security controls become available the next time a main or delta index build occurs.

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.

How search applications use security tokens

It is the responsibility of the client search application to provide the security tokens at search time so that the documents can be appropriately filtered. If no security token is supplied, then the default public token is automatically applied during search processing.

The sample search application for enterprise search demonstrates how you might implement document-level security. For this example, it is assumed that the administrator assigned a security token value to a group of documents (as opposed to extracting the security token from a field in a crawled document). The search application uses the user's login ID to determine which documents the user can access. Instead of using the actual user ID, the search application relies on the group ID that the user belongs. By using a group ID as a security token, users can be added to and removed from the group without requiring the index to be rebuilt.

The security token assigned by the administrator to a set of documents represents a valid operating system group ID. Different group IDs are assigned to different documents in the crawl space. For example:

Document1-5: Security token = Group1
Document6-10: Security token = Group2

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.

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.

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. After documents are crawled and indexed, the enterprise search identity management component is used to validate users who attempt to search secure collections.

- Content Edition crawler (Documentum, FileNet Panagon Content Services, Hummingbird DM, Portal Document Manager, and SharePoint repository types only)
- DB2 Content Manager crawler
- Domino Document Manager crawler
- Notes crawler
- QuickPlace crawler
- Windows file system crawler

For the following types of crawlers, current user credentials can be validated when users use the Search portlet in WebSphere Portal to search enterprise search collections.

- Web Content Management crawler
- WebSphere Portal crawler

Related concepts

“Enforcement of document-level security for Lotus Domino documents” on page 269

“Enforcement of document-level security for Windows file system documents” on page 272

Related tasks

“Configuring Lotus Domino Trusted Servers to validate user credentials” on page 270

Enterprise search identity management

The management of multiple user credentials is a common problem for an enterprise. An enterprise search system solves the problem by providing an optional identity management component.

Tip:

For detailed examples of how to set up the identity management component for enterprise search, see the IBM Redbook, *IBM OmniFind Enterprise Edition Configuration and Implementation Scenarios*.

The information found in an enterprise can exist in many shapes and forms. It can be distributed throughout the enterprise and managed by the most appropriate software for the task at hand. For example, enterprise users might use an SQL application to access relational databases or a document management system to access documents relevant to their work.

Controlling access to sensitive information in these repositories is typically enforced by the managing software. Users identify themselves to the host system through a user ID and password combination. After being authenticated by the system, the managing software controls which documents the user is allowed to see and act upon based on the user’s defined access rights.

It is common for users to have different user IDs and passwords associated with each repository. Similar to how users are asked to identify themselves to the original enterprise repositories, users must provide credentials before viewing

documents in an enterprise search collection that require current credentials to be validated. Users who have multiple identities must present the corresponding credentials for each identity.

If you specify that you want to use enterprise search for identity management in the administration console, the search servers can use the following approaches to validate a user's current credentials during query processing:

- The search application can prompt the user to register the credentials that they need to access various domains in a user profile. The profile, which is encrypted and stored in a secure data store, enables the user to search the secure domains. If credentials are not specified for a domain that requires current credentials to be validated, documents from that domain are excluded from the search results.
- If documents in a collection were crawled by a crawler that provides support for single sign-on (SSO) security, and you specify that you want to use SSO security to control access to documents, the system will use SSO security methods to authenticate users for the duration of a search session. The user does not need to create a profile that specifies credentials or provide a user ID and password when searching secure domains.

When users search collections that require current credentials to be validated when a query is submitted, the system can use the profile or SSO security methods to deny or permit access to documents.

Obtaining the user's group information

To validate a user's credentials, the identity management component must obtain the user's group information for each of the user's identities and add this information to a user security context (USC) string. This group information is used to filter results in accordance with access control data that is stored in the enterprise search index or in accordance with SSO authentication data. The identity management component does this by using SSO tokens or by using the user's credentials to connect to the back-end system and request the groups that the user is a member of.

When you configure identity management options in the administration console, you can specify how often this group information is to be refreshed. You can extract new group data each time that the user logs in to the search application, or you can extract the group data on a regular basis, such as every three days.

Security without the identity management component

Not all enterprises want to manage the multiple identities of their user communities with the enterprise search identity management component. If you disable the identity management component in the enterprise search administration console, then it is the responsibility of your search application to generate the user security context string. After it is generated, the USC string is used to set the ACL constraints value on each query. For example:

```
Query q = factory.createQuery("IBM");  
q.setACLConstraints("User's Security Context in XML");
```

Tip: To help you write your own identity management functionality, an extension to the Search and Index API (SI-API) provides programmatic control over the identity management database. This API allows you to generate the USC with Java objects, and the XML string is then automatically built.

The XML query string must be of the following form, where ... contains the fully formed XML string:

```
@SecurityContext::'...'
```

The format of the XML string is as follows:

```
<identities id="login_UserName">
  <ssoToken>token_value</ssoToken>
  <identity id="security_domain">
    <type>Notes</type>
    <username>domain_UserName</username>
    <password encrypt="no">domain_userPW</password>
    <groups>
      <group id="g1" />
      <group id="g2" />
    </groups>
    <properties>
      <property name="property_name">property_value</property>
      ...
    </properties>
  </identity>
  ...
</identities>
```

identities

The value of the id attribute is the user ID that the user provides when logging in to the system.

ssoToken

Optional: Specifies the Lightweight Third-Party Authentication (LTPA) token that is created for the user for the duration of the browser session. This parameter is used only if the target domain is enabled for SSO and the crawler is configured to use SSO security.

identity

Contains the user's credentials for a particular data source. The value of the id attribute is the domain that stores the user's credential information (in the case of Domino, this is the Domino domain name).

type

Identifies the type of data and corresponds to the crawler type (Notes, DB2, Exchange Server, and so on).

username

Specifies the user name that is to be used to search the domain.

password

Specifies the password for the specified user name. The encrypt attribute must be set to no (enterprise search does not provide an encryption method outside of the identity management component).

groups

Specifies the group names that the user belongs to. A separate group element is used for each group name.

properties

Specifies a list of connection-specific properties, such as the administrator ID and encrypted password that were used to create the crawler, or whether SSO is enabled for the source.

property_name

The name of the property.

property_value

The value of the property.

User validation with user profiles

Search applications can prompt users to register the credentials that they need to access various domains in a user profile.

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 enterprise search 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.

If credentials are not specified for a domain that requires current credentials to be validated, documents from that domain are excluded from the search results.

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.

Collections can contain documents from many different types of sources. For example, a collection can contain documents that were crawled from a Windows file system and several Lotus Notes databases. The identity management component differentiates between the different types of sources and prompts only for credentials that are needed to access domains that require validation.

By default, each credential is enabled for search and thus requires the user to provide the user ID and password that corresponds to the secure domains. If the user has forgotten the user ID or password for a particular domain, the domain can be disabled for searching by clearing the check box. Disabling a domain prevents secure documents in those domains from being returned in a result set.

After creating a profile, the user can submit a search request. The identity management component has the information necessary to build the user's security context (USC) string to be used on subsequent search requests. If you do not use the identity management component, the search application must supply the USC string when users query domains that require current credentials to be validated.

The next time the user attempts to search enterprise search collections, the identity management component repeats the credential verification process, but this time is able to locate the user's profile. If nothing changed, then the user is positioned automatically where search requests can be submitted and is not prompted to create a profile.

If the identity management component detects a change in any of the user's credentials, the user is automatically presented with the profile page when the search application is accessed. This occurs, for example, when a password for any of the domains that are enabled for search is changed or when a domain that requires authentication is added to a collection.

Users can ignore the recommendation to update the profile, but doing so results in excluding those documents from the search results.

In the sample search application provided with enterprise search, users can update profiles at any time by selecting **My Profile** on the toolbar.

User validation with SSO security

If documents in a collection were crawled by a crawler that provides support for single sign-on (SSO) security, you can specify that you want to use SSO security to control access to documents when you configure identity management options.

SSO enablement

Single sign-on authentication enables a user to be authenticated one time and gain access to many resources without being prompted to present credentials again. In an enterprise search system, SSO authentication eases the burden of managing the many user names and passwords that users must specify to access documents in secure collections.

IBM WebSphere Application Server and Lotus Domino support a form of SSO that is known as Lightweight Third-Party Authentication (LTPA). When a user attempts to access either product, the user is asked to authenticate with a user name and password. This user name and password are verified against an LDAP repository that both products share. After the user is authenticated, a session cookie is created to contain the LTPA token. The user can then access other resources on any server that has the same authentication configuration without being prompted to specify credentials again. This token persists as long as the browser session is valid.

To enable SSO support for use with enterprise search collections:

- Ensure that WebSphere Application Server global security and a valid LDAP registry are enabled on the search servers for enterprise search. The LDAP registry can be any valid LDAP product supported by WebSphere Application Server.
- Ensure that the WebSphere authentication mechanism is configured to use an active authentication mechanism of LTPA. When you configure LTPA, specify a valid but flexible domain name, such as your.server.com.
- Ensure that the LTPA key was exported from WebSphere Application Server and imported into other products in the same domain on which you want to enable support for LTPA.

After you use a browser to verify that the above security configuration is working properly, you can use the enterprise search administration console to configure crawlers that support SSO authentication.

SSO and identity management

When users search collections that require current credentials to be validated, the system can use SSO security methods to deny or permit access to documents. Users are not prompted for credentials when they search sources that support SSO authentication. The identity management component is used if all of the following conditions are true:

- SSO is properly enabled in WebSphere Application Server and the target domains.
- Security is enabled in at least one of the collections that the search application can search.
- The options to use the identity management component and SSO security are enabled in the enterprise search administration console.
- The option to use SSO security and options to enforce document-level security (such as indexing access controls or validating current credentials during query processing) were selected when the following crawler types were configured:

- Content Edition (available for Portal Document Manager repositories only)
- Domino Document Manager (available for crawlers that use the DIIOP protocol only)
- Notes (available for crawlers that use the DIIOP protocol only)
- QuickPlace (available for crawlers that use the DIIOP protocol only)

Search portlet security

When users use the Search portlet for enterprise search to search collections from within WebSphere Portal, security is also provided for documents crawled by the Seed list, Web Content Management, and WebSphere Portal crawlers.

Secure search is supported for these types of sources only when you use the portlet, not a search application that runs outside of WebSphere Portal. Within WebSphere Portal, user credentials are obtained through the search portlet. After a user logs in to WebSphere Portal, all search requests include the user's security data (user name, group membership, and so on). Because this information is always available, LTPA token-based SSO is not required.

If you use another product to protect sites and documents on a WebSphere Portal server, however, you must specify SSO options when you configure the crawler. For example, if you use a product like IBM Tivoli Access Manager WebSEAL or CA SiteMinder SSO Agent for PeopleSoft, you must specify credentials that enable the crawler to access documents on the server through single sign-on. In this case, you are enabling SSO for crawler access to secured content, not enabling SSO for secure search.

Configuring identity management

You can use the identity management component for enterprise search to specify how user credentials are 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

When users search collections that require current credentials to be validated during query processing, the identity management component for enterprise search can use a user profile or single sign-on (SSO) security methods to deny or permit access to documents.

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 for using the enterprise search identity management component to control how user credentials are validated during query processing. If this check box is clear, the search application must supply the user security context (USC) string when users query domains that require current credentials to be validated during query processing.

4. Specify how often the identity management component is to extract user credentials from group records in the WebSphere Application Server user registry. You can refresh the credential data every time the user accesses the search application or after a specified number of days has elapsed.
5. For the crawler types that support SSO authentication, specify whether you want the identity management component to use SSO security tokens instead of user profiles to validate users. You can select the check box to use SSO authentication with all crawler types, or select the check boxes for individual crawler types.

Important: The identity management component uses SSO security mechanisms only if SSO security is properly configured in WebSphere Application Server and the target domains.

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:

```
<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 source page that contains the anchor construct. This association enables the target document to be retrieved by queries that specify text that appears in the source document. The association presents a security risk, however, if users are allowed to view the target document but not the source document.

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; a document is returned in the search results only if its own content or metadata matches the query.

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 specify whether you want to index the anchor text in links to forbidden documents when you configure advanced Web crawler properties.

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 and, on 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, the documents must be recrawled 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.

Enabling security for enterprise search

If you plan to enforce security when users administer or search an enterprise search system, you must configure global security in WebSphere Application Server. You must also configure security information in the enterprise search configuration files, administration console, and search applications.

Tip:

For detailed examples of how to enable global security in WebSphere Application Server with an LDAP repository, including examples of how to configure various crawlers ensure that document-level security is enforced, see the IBM Redbook, IBM OmniFind Enterprise Edition Configuration and Implementation Scenarios.

Procedure

To enable security for an enterprise search system:

1. Decide which type of user registry you want to use for authenticating users. For example, many WebSphere Application Server administrators choose to use the Lightweight Directory Access Protocol (LDAP) user registry.
2. In the enterprise search administration console, select **Security** and assign at least one of the users in the WebSphere Application Server user registry to the **Enterprise search administrator** administrative role.

Alternatively, add the enterprise search administrator ID that is specified when OmniFind Enterprise Edition is installed to the WebSphere Application Server user registry.

Important: After global security is enabled, only user IDs that are in the user registry and that have been assigned an enterprise search administrative role can access the administration console and administer enterprise search.

3. Follow the procedures in “Configuring global security and an LDAP user registry in WebSphere Application Server” to enable global security and configure the user registry.
4. If you enable global security after OmniFind Enterprise Edition is installed, you must provide the enterprise search system with the WebSphere Application Server user ID and password. To provide this information, you use the **eschangewaspw** command:
 - If you installed OmniFind Enterprise Edition on a single server, follow the procedure in “Enabling security for a single server enterprise search system” on page 264.
 - If you installed OmniFind Enterprise Edition on more than one server, follow the procedure in “Enabling security for a multiple server enterprise search system” on page 265.
5. Complete the tasks appropriate for the types of documents that you plan to crawl and search. See “Crawler setup requirements to support security” on page 266 for details.

Configuring global security and an LDAP user registry in WebSphere Application Server

To enable security in OmniFind Enterprise Edition, you must first enable global security in WebSphere Application Server.

About this task

As part of enabling global security, you must configure a user registry to authenticate user IDs. This task discusses how to configure a Lightweight Directory Access Protocol (LDAP) user registry in WebSphere Application Server at the same time that you enable global security. Although other types of user registries are supported by WebSphere Application Server, you cannot use the local operating system registry to authenticate enterprise search users. To use the local operating system registry, operating system user accounts for every user in your enterprise must exist on the search servers for enterprise search.

This task is based on WebSphere Application Server version 6. If you use an earlier version of WebSphere Application Server, the default paths and some user interface labels might be different. This task also uses the IBM Tivoli Directory Server for the LDAP registry. If you use a different registry type or a custom registry, you must provide information appropriate for your registry.

This task summarizes the steps required to configure global security for use with an enterprise search system. For detailed instructions, see the *WebSphere Application Server, Version 6.0.x* Information Center at the following URL: <http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/index.jsp>

Procedure

To enable global security in WebSphere Application Server:

1. On the search server for enterprise search, access the following URL to open the WebSphere Application Server Administrative Console, where *localhost* is *localhost* or the server name, such as *omnifind.search.xyz.com*.
`http://localhost:9060/ibm/console`
2. Click **Security** and then click **Global Security**.
3. Set up WebSphere to use an LDAP registry:
 - a. Under **User registries**, click **LDAP**.
 - b. Specify the server user ID and password that are used to run the application server.
 - c. For the registry type, select the IBM Tivoli Directory Server.
 - d. Specify the LDAP server host name, either an IP address or a domain name service (DNS) host name. The default port number is 389.
 - e. Specify the base distinguished name (DN) that is the starting point for searching the registry, such as `ou=sales,o=ibm,c=us`.
 - f. Because some LDAP servers do not support anonymous binding when the registry is searched, specify the DN for the application server, such as `cn=searchuser,o=ibm,c=us`, and then specify the password for the application server. The application server uses this DN and password to bind to the registry.
 - g. To use Secure Sockets Layer (SSL) communication between WebSphere and LDAP, select the **SSL enabled** check box.
 - h. Click **Apply** and then click **OK**.
4. Under **General Properties**, select the **Enable global security** and **Enforce Java 2 security** check boxes.
5. For the active authentication mechanism, select Simple WebSphere Authentication Mechanism (SWAM).
6. For the active user registry, select Lightweight Directory Access Protocol (LDAP) user registry.
7. Click **OK**.
8. Click the **Save** link at the top of the page. When you are prompted to save your changes, click the **Save** button.
9. On the tool bar, click **Logout**.
10. Stop and restart the ESSearchServer application.

AIX, Linux, or Solaris

```
./stopServer.sh ESSearchServer  
./startServer.sh ESSearchServer
```

Windows

```
stopServer ESSearchServer  
startServer ESSearchServer
```

These scripts are located in the `WAS_INSTALL_ROOT/AppServer/bin` directory:

- For WebSphere Application Server version 5, the default installation path is `/usr/WebSphere` on AIX systems, `/opt/WebSphere` on Linux or Solaris systems, or `C:\Program Files\WebSphere` on Windows systems.
- For WebSphere Application Server version 6, the default installation path is `/usr/IBM/WebSphere` on AIX systems, `/opt/IBM/WebSphere` on Linux or Solaris systems, or `C:\Program Files\IBM\WebSphere` on Windows systems.

11. Restart the WebSphere Application Server Administrative Console.
12. Because the server is now starting in secure mode, type the server user ID and password that you specified when you configured the LDAP user registry to log in to the console (see step 3b on page 263).

Enabling security for a single server enterprise search system

If you enable WebSphere Application Server global security after you install OmniFind Enterprise Edition, you must use the `eschangewaspw` command to update the enterprise search configuration file, `es.cfg`, with the password for the WebSphere Application Server user.

Before you begin

Ensure that the `config.properties` file for the `ESSearchApplication` application specifies a valid WebSphere Application Server user name and password. The default location of this file is `ES_INSTALL_ROOT/installedApps/ESSearchApplication.ear/ESSearchApplication.war/WEB-INF`.

About this task

The `eschangewaspw` command encrypts the password before storing it in the `es.cfg` file.

Procedure

To enable an existing single-server enterprise search system to use global security:

1. On the enterprise search server, log in as the enterprise search administrator and stop the enterprise search system:

```
esadmin system stopall
```
2. Ensure that the `WASUser` entry in the `ES_NODE_ROOT/nodeinfo/es.cfg` file specifies a valid WebSphere Application Server user name.
3. Run the following script, where `WAS_password` is the password for the WebSphere Application Server user name that is specified in the `ES_NODE_ROOT/nodeinfo/es.cfg` file (see step 2).

AIX, Linux, or Solaris

```
eschangewaspw.sh WAS_password
```

Windows

```
eschangewaspw WAS_password
```

4. On Windows, select **Control Panel** → **Administrative Tools** → **Services** and add the same WebSphere Application Server user name and password to the WebSphere Application Server and `ESSearchServer` services.

- Restart the enterprise search system:

```
esadmin system startall
```

Enabling security for a multiple server enterprise search system

If you enable WebSphere Application Server global security after you install OmniFind Enterprise Edition, you must use the **eschangewaspw** command to update the enterprise search configuration file, `es.cfg`, with the password for the WebSphere Application Server user.

Before you begin

Ensure that the `config.properties` file for the `ESSearchApplication` application specifies a valid WebSphere Application Server user name and password. The default location of this file is `ES_INSTALL_ROOT/installedApps/ESSearchApplication.ear/ESSearchApplication.war/WEB-INF` on the search servers.

About this task

The **eschangewaspw** command encrypts the password before storing it in the `es.cfg` file.

Procedure

To enable an existing multiple-server enterprise search system to use global security:

- Do the following steps on the enterprise search index server:
 - Log in as the enterprise search administrator and stop the enterprise search system:

```
esadmin system stopall
```
 - Ensure that the `WASUser` entry in the `ES_NODE_ROOT/nodeinfo/es.cfg` file specifies a valid WebSphere Application Server user name.
 - Run the following script, where `WAS_password` is the password for the WebSphere Application Server user that is specified in the `ES_NODE_ROOT/nodeinfo/es.cfg` file (see step 1b).

AIX, Linux, or Solaris

```
eschangewaspw.sh WAS_password
```

Windows

```
eschangewaspw WAS_password
```

- Do the following steps on the second search server (for a two server configuration), or on the crawler server and both search servers (for a four server configuration):
 - Log in as the enterprise search administrator.
 - Run the following script, where `WAS_password` is the password for the WebSphere Application Server user that is specified in the `ES_NODE_ROOT/nodeinfo/es.cfg` file (see step 1b).

AIX, Linux, or Solaris

```
eschangewaspw.sh WAS_password
```

Windows command prompt

```
eschangewaspw WAS_password
```

3. On Windows, select **Control Panel** → **Administrative Tools** → **Services** and add the same WebSphere Application Server user name and password to the WebSphere Application Server and ESSearchServer services.
4. On the enterprise search index server, restart the enterprise search system:

```
esadmin system startall
```

Crawler setup requirements to support security

To gather information that enables document-level security to be enforced, the crawlers must have permission to access the native security data. For some data types, additional steps must be taken to configure a secure environment.

Table 7. Crawler setup requirements to support security

Content Edition crawlers

Before you create a crawler to access repositories in direct mode, configure the WebSphere Information Integrator Content Edition system to run in direct mode and configure a connector for the crawler server.

Before you create a crawler to access repositories in server mode, run a script (`escrvbr.sh` on AIX, Linux, or Solaris, or `escrvbr.vbs` on Windows) to configure the crawler server.

When you configure the crawler, specify a user ID and password that enables the crawler to access each repository to be crawled. You can specify a different user ID and password, as necessary, for each repository in the crawl space.

Related topics:

- “Direct mode access to Content Edition repositories” on page 43
- “Server mode access to WebSphere II Content Edition repositories” on page 44
- “Configuring the crawler server on UNIX for WebSphere II Content Edition” on page 44
- “Configuring the crawler server on Windows for WebSphere II Content Edition” on page 45

DB2 crawlers

Before you create the crawler, run a script (`escrdb2.sh` on AIX, Linux, or Solaris, or `escrdb2.vbs` on Windows) to configure the crawler server.

When you configure the crawler to crawl remote, uncataloged databases, specify a user ID and password that enables each database on the target database server to be crawled. You can specify a different user ID and password, as necessary, for each database in the crawl space.

Related topics:

- “Configuring the crawler server on UNIX for DB2 crawlers” on page 48
- “Configuring the crawler server on Windows for DB2 crawlers” on page 49

DB2 Content Manager crawlers

Before you create the crawler, run a script (`escrcm.sh` on AIX, Linux, or Solaris, or `escrcm.vbs` on Windows) to configure the crawler server.

When you configure the crawler, specify a user ID and password that enables the crawler to access each server to be crawled. You can specify a different user ID and password, as necessary, for each server in the crawl space.

Related topics:

- “Configuring the crawler server on UNIX for DB2 Content Manager crawlers” on page 56
- “Configuring the crawler server on Windows for DB2 Content Manager crawlers” on page 57

Domino Document Manager, Notes, and QuickPlace crawlers

Table 7. Crawler setup requirements to support security (continued)

<p>To crawl Lotus Domino servers that use the Notes remote procedure call (NRPC) protocol:</p>	<p>Related topics:</p> <ul style="list-style-type: none">• “Configuring the I/O completion port on AIX to crawl Lotus Domino sources” on page 77• “Configuring the crawler server on UNIX to crawl Lotus Domino sources” on page 73• “Configuring the crawler server on Windows to crawl Lotus Domino sources” on page 74• “Configuring Lotus Domino Trusted Servers to validate user credentials” on page 270• “Configuring servers that use the DIIOP protocol” on page 76• “Configuring the QuickPlace server to use Local User security” on page 271• “Configuring Directory Assistance on a QuickPlace server” on page 272
<ul style="list-style-type: none">• On an AIX system, ensure that the I/O Completion Port module is installed and available on the crawler server.• Before you create the crawler, run a script (escrnote.sh on AIX, Linux, or Solaris, or escrnote.vbs on Windows) to configure the crawler server.• A Domino server must be installed on the enterprise search crawler server, and this Domino server must be a member of the Domino domain to be crawled.• To validate current user credentials when a user submits a search request, the Domino server to be crawled must be configured as a Lotus Domino Trusted Server.• When you configure the crawler, specify the path for a Lotus Notes user ID file that is authorized to access the server, such as c:\Program Files\lotus\notes\data\name.id or /local/notesdata/name.id, and the password for this ID file.	
<p>To crawl Lotus Domino servers that use the Domino Internet Inter-ORB Protocol (DIIOP):</p>	
<ul style="list-style-type: none">• On an AIX system, ensure that the I/O Completion Port module is installed and available on the crawler server.• Configure the crawler server so that it can use the protocol.• When you configure the crawler, specify a fully qualified Lotus Notes user ID that is authorized to access the server, such as User Name/Any Town/My Company, and the password for this user ID.	
<p>To crawl QuickPlace servers, you must configure the QuickPlace server to support Local User security or Directory Assistance, depending on the type of security you want to use.</p>	
<hr/> Exchange Server crawlers	
<p>When you configure the crawler, specify a user ID that is authorized to access public folders on the Exchange Server to be crawled and the password for this user ID.</p>	<p>Related topic:</p> <ul style="list-style-type: none">• “Verifying access to secure Exchange Server documents” on page 269
<p>For the crawler to use Exchange Server key management and the Secure Sockets Layer (SSL) protocol when crawling data, also specify the fully qualified path to the keystore file and a password that enables the crawler to access this file. The keystore file must exist on the enterprise search crawler server.</p>	
<hr/> JDBC database crawlers	

Table 7. Crawler setup requirements to support security (continued)

When you configure the crawler, you can specify a user ID and password that enables tables in the target database to be crawled. You can specify a different user ID and password, as necessary, for each database in the crawl space.

NNTP crawlers

The NNTP servers to be crawled must allow the crawler server to read data.

UNIX file system crawlers

The AIX, Linux, and Solaris subdirectories to be crawled must allow the crawler server to read data.

Web crawlers

The Web crawler abides by the robots exclusion protocol. If a Web server includes a robots.txt file in the top level of the server directory, the crawler analyzes the file, and crawls Web sites on that server only if it is allowed to do so. For information about this protocol, see <http://www.robotstxt.org/wc/exclusion.html>.

Related topics:

- “Web sites protected by HTTP basic authentication” on page 95
- “Web sites protected by form-based authentication” on page 96

When you configure the Web crawler:

- You must specify a user agent name for the crawler. Rules in the robots.txt files of the servers to be crawled can specify this name to permit or refuse access.
 - Optional: 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.
 - Optional: 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.
-

Seed list, Web Content Management, and WebSphere Portal crawlers

Before you create a crawler, you must run a setup script to integrate enterprise search with a WebSphere Portal server. Different scripts are provided for different versions of WebSphere Portal.

Related topic:

- “Setup scripts for integrating enterprise search with WebSphere Portal” on page 326

When you configure the crawler, specify a fully qualified distinguished name (DN) that enables the crawler to retrieve pages from the server to be crawled, such as `uid=admin,cn=RegularEmployees,ou=Software Group,o=IBM,c=US`, and specify the password for this DN. The DN must match a DN that is configured in WebSphere Portal.

Ensure that permissions for the user DN that you specify are defined in the Portal Access Control (PAC) component of WebSphere Portal. The crawler uses the PAC to obtain access control data for the documents that it crawls.

Windows file system crawlers

Table 7. Crawler setup requirements to support security (continued)

The subdirectories to be crawled must allow the crawler server to read data. When you configure the crawler to crawl remote file systems, specify a user ID that enables the crawler to access the remote data and specify a password for this user ID.

Related topics:

- “Enforcement of document-level security for Windows file system documents” on page 272
- “Secure search of Windows trusted domains” on page 274

To validate current user credentials when a user submits a search request, 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.

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.

Enforcement of document-level security for Lotus Domino documents

If the Domino 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 for documents on a Domino 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 documents that match the search criteria:

- “Configuring Lotus Domino Trusted Servers to validate user credentials” on page 270.

- “Configuring global security and an LDAP user registry in WebSphere Application Server” on page 262.

Related concepts

“Validation of current credentials during query processing” on page 253

“Notes crawlers” on page 70

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. Click **File** and then select **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.

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 enterprise search 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 253

“Notes crawlers” on page 70

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.

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.

Important: User credentials cannot be validated during query processing if both of the following conditions are true:

- The Windows server to be crawled is not a member of a domain.
- The directory to be crawled is a remote directory, such as \\servername\hostname.

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 OmniFind Enterprise 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 OmniFind Enterprise 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 253

“Windows file system crawlers” on page 107

Secure search of Windows trusted domains

To enforce document-level security for remote Windows file systems, the enterprise search system supports access control list (ACL) verification across trusted domains.

Configuring the crawler

To configure the Windows file system crawler to support trusted domains, you must specify options in a new configuration file. There is no support for configuring this capability in the enterprise search administration console.

1. To support the document-level security across trusted Windows domains, edit the following file:

```
ES_NODE_ROOT/master_config/session_ID/winfscrawler_ext.xml
```

Tip: To determine the session ID for the Windows file system crawler that you want to configure, you can monitor the crawler in the enterprise search administration console or use the **esadmin report collections** command.

2. Specify the Windows domain name and the NETBIOS name of the Active Directory. For example:

```
<ExtendedProperties>
  <SetAttribute XPath="/Crawler/DataSources/Server/Target"
    Name="Domain">jk.enterprises.com
  </SetAttribute>
  <SetAttribute XPath="/Crawler/DataSources/Server/Target"
    Name="NetBIOSDomain">JKE1
  </SetAttribute>
</ExtendedProperties>
```

3. Stop and restart the crawler for the changes to become effective.

Restrictions

- Documents cannot include ACLs from multiple Windows domains. Domain users and groups must belong to one Windows domain per collection.
- To support remote file system access verification, the Windows servers must run in the same Windows domain or in trusted Windows domains.
- The Windows file system crawler reads the NETBIOS name of the Active Directory associated with the Windows server to be crawled and uses the NETBIOS name to filter the file ACL. The Active Directory that the crawler server joins trusts the other Active Directory that defines user accounts and group accounts.
- The user account that you specify for the crawler to use to access a remote Windows server must belong to the Windows domain where you want to enforce and verify access control.
- The Windows operating system allows only one account to connect network folders on one file server. Other accounts cannot connect to the same file server at the same time. Therefore, you cannot configure different accounts for different crawlers to crawl the same Windows server, even if the crawlers are in different collections.

Disabling security for enterprise search

You can disable security for an enterprise search application in WebSphere Application Server. If you previously configured document-level security controls, you can specify that the controls are to be ignored. Security settings also affect how collapsed results are displayed in the search results.

Disabling security for an enterprise application in WebSphere Application Server

To control which enterprise search activities require user authentication, you can disable global security for individual enterprise applications in WebSphere Application Server.

About this task

The OmniFind Enterprise 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 enterprise search 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 enterprise application. For example, you might want to authenticate all users who access the enterprise search administration console, but not authenticate users who use the SI-API interfaces or the sample search application.

Procedure

To disable security for an 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. Stop and restart the ESSearchServer application.

AIX, Linux, or Solaris

```
./stopServer.sh ESSearchServer
./startServer.sh ESSearchServer
```

Windows

```
stopServer ESSearchServer
startServer ESSearchServer
```

These scripts are located in the WAS_INSTALL_ROOT/AppServer/bin directory:

- For WebSphere Application Server version 5, the default installation path is /usr/WebSphere on AIX systems, /opt/WebSphere on Linux or Solaris systems, or C:\Program Files\WebSphere on Windows systems.
- For WebSphere Application Server version 6, the default installation path is /usr/IBM/WebSphere on AIX systems, /opt/IBM/WebSphere on Linux or Solaris systems, or C:\Program Files\IBM\WebSphere on Windows systems.

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 enable or disable document-level security for all documents in a collection, 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

You can configure crawlers to associate security tokens with documents as they are 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 **Enable or disable document-level security**.
2. On the Document-Level Security for All Documents 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.

Disabling security for collapsed search results

If collection security is enabled, search results from the same site cannot be collapsed in the search results unless you specify that you do not want to validate user credentials during query processing.

Before you begin

To enable or disable current credential validation for all documents in a collection, 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 current credentials are validated, the source of each document is inspected and possibly routed for validation. You must disable current credential validation if you want documents that have the same URI prefix, or that belong to a previously configured collapsed URI group, to be collapsed in the search results.

Procedure

To disable current credential validation so that documents can be collapsed in the search results:

1. Edit a collection, select the General page, and click **Enable or disable document-level security**.
2. On the Document-Level Security for All Documents page, select the **Do not validate current credentials during query processing** check box.
3. Monitor the collection, select the Search page, and stop and restart the search server processes.

When users query the collection, documents that have the same URI prefix, or that belong to sites that are configured to be collapsed, are collapsed in the search results. In the sample search application, users can view the collapsed results by clicking the **More results from the same source** link.

Starting and stopping an enterprise search system

After you create a collection, you must start the servers for crawling, parsing, and indexing data (the search servers are started automatically). Stop and restart the servers 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 building main and delta indexes. After you start the enterprise search system, you typically need to stop and restart the server processes 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 collect data from the sources in your enterprise, you must stop and restart the crawlers for the changes to become effective. If you do not change the crawling rules, the Web crawler runs continuously and other crawlers run according to schedules that you specify.

To enhance the availability of the search servers when the index server and administration console are not available, you can specify commands to start the search servers for a collection in stand-alone mode.

Starting an enterprise search system

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

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 index the crawled data.

Restrictions

To start an enterprise search system, you must use a user account that can be authenticated with local authentication. If you attempt to start the system with an Andrew File System (AFS[®]) account, errors occur.

Procedure

To start an enterprise search system:

1. If you use enterprise search in a two-server or four-server configuration, log in as the enterprise search administrator and start the common communication layer (CCL) on each server:

AIX, Linux, or Solaris

```
startccl.sh -bg
```

Windows command prompt




```
startccl
```

Windows Services administrative tool


To start CCL in the background:

- a. Launch Windows Services: **Start** → **Programs** → **Administrative Tools** → **Services**.
 - b. Right-click **IBM OmniFind Enterprise Edition** and click **Start**.
2. Start the enterprise search system components:
 - a. Log in as the enterprise search administrator on any enterprise search server.
 - b. Start all system components: `esadmin system startall`



This command starts the following processes and applications:

 - The Web server (in a multiple server configuration, the Web server is started on both search servers)
 - The ESSearchServer and ESAdmin applications in WebSphere Application Server (in a multiple server configuration, the applications are started on both search servers)
 - The ESAdmin session on the index server
 - The CCL on the computer where you run the command, if the CCL is not already running
 - The database network server for enterprise search
 - The enterprise search information center (in a multiple server configuration, the information center is started on both search servers)
 3. 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.
 4. On the Collections view, locate the collection that you want to administer and click  **Monitor**.
 5. On the Crawl page, for each crawler that you want to start, click  **Start**.
 - If you start a Web crawler, the crawler begins crawling data immediately. These types of crawlers run continuously to crawl and recrawl Web documents.
 - 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.
 6. 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).

7. 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 **Main** area, click  **Start**.

You can let the indexing processes run continuously. The index will be built at the scheduled dates and times.

Tip: The search servers start automatically, and you can let them run continuously. You typically do not need to stop the search servers unless you make changes to the search cache or document summary settings. To restart the search servers, open the Search page, click  **Stop** and then click  **Start**.

To enhance the availability of the search servers when the administration console is not available, you can specify commands to start the search servers for a collection in stand-alone mode. If the index server is not running, the administration console is not available.

Related concepts

“Administrative roles” on page 248

Related tasks

“Logging in to the administration console” on page 18

Stopping an enterprise search system

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 search servers.

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 indexed, 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. 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 built.
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.
6. To stop the enterprise search system instead of individual servers:
 - a. Log in as the enterprise search administrator on any enterprise search server.
 - b. Stop all system components: `esadmin system stopall`
This command stops the following processes and applications:
 - The Web server (in a multiple server configuration, the Web server is stopped on both search servers)
 - The ESSearchServer and ESAdmin applications in WebSphere Application Server (in a multiple server configuration, the applications are stopped on both search servers)
 - The ESAdmin session on the index server
 - The common communication layer (CCL) for enterprise search on the computer where you run the command
 - The database network server for enterprise search
 - The enterprise search information center (in a multiple server configuration, the information center is stopped on both search servers)

Related tasks

“Logging in to the administration console” on page 18

Controlling which components are started or stopped

You can control which components are started or stopped by the **esadmin system startall** and **esadmin system stopall** commands.

About this task

The `ES_INSTALL_ROOT/default_config/AutoRunComponents.properties` file contains a list of the enterprise search components that can be started or stopped by the **esadmin system startall** and **esadmin system stopall** commands. By default, all of the listed components are started and stopped by these commands.

If you want to prevent certain components from being started or stopped, you can edit the properties file.

Procedure

To specify which components are to be started or stopped when you start or stop the enterprise search system:

1. Log in as the enterprise search administrator on the server where you plan to run the **esadmin system startall** or **esadmin system stopall** command.
2. Edit the `ES_INSTALL_ROOT/default_config/AutoRunComponents.properties` file.
3. To prevent a component from being started, add a field for the component called `Component.startable.component_ID=false`, where `component_ID` is the component that you do not want to start.
4. To prevent a component from being stopped, add a field for the component called `Component.stopable.component_ID=false`, where `component_ID` is the component that you do not want to stop.
5. Save and exit the file.

The next time that you use the **esadmin system startall** or **esadmin system stopall** command, the component that you modified will not be started or stopped, according to the changes that you made in the properties file.

Example: In this example, the HTTP server will be started by the **esadmin system startall** command (the default setting), but it will not be stopped by the **esadmin system stopall** command (as controlled by the emphasized line in the example):

```
#####
# Details of component 3.
#####
Component.name.3=IBM HTTP server
Component.impl.class.3=com.ibm.es.control.util.component.impl.HTTPControlImpl
Component.nodes.3=search
# By default all components are startable
Component.stopable.3=false
```

Administering the search servers in stand-alone mode

To ensure high availability of the search servers, you can start the search servers for individual collections even if the index server is not running.

Restrictions

To stop and start the search servers, you must be an enterprise search administrator.

The ability to start and stop the search servers in stand-alone mode is not available from the enterprise search administration console. If the index server is not running, the administration console cannot be accessed.

Before you can start and stop the search servers in stand-alone mode, ensure that the crawler, parser, index, and search servers for the collection have all been started at least once. This is necessary to ensure that required files are synchronized on the search servers.

If document-level security is enabled for the collection, ensure that the crawler server is also started. This is necessary to ensure that document-level security controls can be enforced. If the crawler server is not running, only documents that do not require authentication are returned in the search results.

About this task

If the index server is unavailable, you can ensure that users are able to continue searching the system by starting the search servers in stand-alone mode. You can run the commands for starting and stopping the search servers from any enterprise

search server in a multiple server installation. The commands attempt to start or stop the search servers for the specified collection on all available search servers.

Procedure

To start or stop the search servers in stand-alone mode:

1. To start the search servers for a collection when the index server is not running:

- a. On the crawler server and search servers, log in as the enterprise search administrator and then start the common communication layer (CCL) service:

AIX, Linux, or Solaris

```
startccl.sh -bg
```

Windows command prompt

```
startccl
```

Windows Services administrative tool

To start CCL in the background:

- 1) Launch Windows Services: **Start** → **Programs** → **Administrative Tools** → **Services**.
 - 2) Right-click **IBM OmniFind Enterprise Edition** and click **Start**.
- b. On the search servers, run the **startServer** script, which is located in the *WAS_INSTALL_ROOT/AppServer/bin* directory, to start the ESSearchServer application in WebSphere Application Server:

AIX, Linux, or Solaris

```
./startServer.sh ESSearchServer
```

Windows

```
startServer ESSearchServer
```

- c. Enter the following command, where *collection_id* identifies the collection that owns the search servers that you want to start:

```
esadmin startSearch -cid collection_id
```

2. To stop the search servers for a collection when the index server is not running:

- a. Log in as the enterprise search administrator on any enterprise search server.
- b. Enter the following command, where *collection_id* identifies the collection that owns the search servers that you want to stop:

```
esadmin stopSearch -cid collection_id
```

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 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 tasks

“Starting an enterprise search system” on page 279

“Stopping an enterprise search system” on page 281

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 system 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. If you configure alerts for the collection and select 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 documents in the collection. The system monitors this estimate and the alert threshold percentage that you specify, and sends e-mail when the maximum number of documents configured for the collection is about to be reached.

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.

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.

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 tasks

“Viewing reports about dropped documents” on page 303

Related reference

“URI formats in an enterprise search index” on page 113

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 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.

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.
 - Click **Thread details** to see how many threads are actively crawling Web sites and how many are in an inactive state.
 - Click **Active sites** to see information about the Web sites that the crawler is actively crawling.
 - 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.
 - Click **Crawler history** to view reports about past crawler activity.
 - In the **URL status** area, type a URL that you want to see information about.
 - a. Click **URL details** to see status information for the URL. You can request URL details only for URLs that were previously crawled.
 - b. Click **Site details** to specify information that you want to include in a report about the Web site that the URL belongs to. You can request site details for a previously crawled Web site or for a Web site that has not yet been crawled.

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. You can also ask to see the contents of the robots.txt file for the Web site, which might help you determine why the site is not being crawled.

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 memory 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 memory properties and adjust the timeout value in the **Amount of time that each host can remain active** 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 status 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, and then click **Site details**.
5. For both crawler history and site reports, you can 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, you can 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, you can specify options for creating an HTTP status code report, then click **Run Report**.

This report provides information about the number of HTTP status 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 status codes (which indicate that pages were not found), 5xx status codes (which indicate a server problem), 6xx status 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 status codes are not contributing to the index, you can improve the performance of the crawler by removing the sites from the crawl space.

HTTP status codes returned to the Web crawler

When you monitor a Web crawler, you can view information about the HTTP status 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 status 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 status codes for a URL, and the status 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 status codes and how the Web crawler interprets them. Values from 100 to 505 are standard HTTP status codes (see <http://www.w3.org/Protocols/rfc2616/rfc2616.html> for more information). The remaining HTTP status codes are proprietary to enterprise search and the Web crawler.

Table 8. HTTP status codes from the Web crawler

Code	Description	Code	Description	Code	Description	Code	Description
NULL	Uncrawled	400	Bad Request	500	Internal server error	693	Select fail (URLFetcher)
100	Continue	401	Unauthorized	501	Not implemented	694	Write error (URLFetcher)
101	Switching protocols	402	Payment required	502	Bad gateway	695	Incomplete block header (URLFetcher)
200	Successful	403	Forbidden	503	Service unavailable	699	Unexpected error (URLFetcher)
201	Created	404	Not found	504	Gateway timeout	700	Parse error (no header end)
202	Accepted	405	Method not allowed	505	HTTP version not supported	710	Parse error (header)

Table 8. HTTP status codes from the Web crawler (continued)

Code	Description	Code	Description	Code	Description	Code	Description
203	Non-authoritative information	406	Not acceptable	611	Read error	720	Parse error (no HTTP code)
204	No content	407	Proxy authentication required	612	Connect error	730	Parse error (body)
205	Reset content	408	Request timeout	613	Read timeout	740 or 4044	Excluded by robots.txt file
206	Partial content	409	Conflict	614	SSL handshake failed	741	Robots temporarily unavailable
300	Multiple choices	410	Gone	615	Other read error	760	Excluded by crawl space definition
301	Moved permanently	411	Length required	616	FBA anomaly	761	Disallowed by local crawl space; allowed by global
302	Found	412	Precondition failed	617	Encoding error	770	Bad protocol or nonstandard system port
303	See other	413	Request entity too large	618	Redirect with no redirect URL	780	Excluded by file type exclusions
304	Not modified	414	Request URI is too long	680	DNS lookup failure	786	Invalid URL
305	Use proxy	415	Unsupported media type	690	Malformed URL	2004	No index META tag
306	(Unused)	416	Requested range not satisfiable	691	Lost connection (URLFetcher)	3020	Soft redirect
307	Temporary redirect	417	Expectation failed	692	Write timeout (URLFetcher)		

Table notes

4xx status codes

You will rarely see a 400 (bad request) code. According to the HTTP status code standard, 4xx codes are supposed to 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>). Other 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 status 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 status code might indicate that there is a problem with the crawler.

61x status codes

Except for 615, the 61x status codes indicate problems that can be expected in crawling, such as timing out. The following status codes might require corrective action:

611, 612, and 613

Slow sites or poor network performance might be the cause of these problems.

611 Indicates that an error occurred when the crawler retrieved a document.

612 Indicates that an error occurred when the crawler attempted to connect to a Web server.

613 Indicates that a timeout occurred while the crawler was retrieving a document.

- 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 for form-based authentication (FBA) still appears in the download after reauthentication.

If the information provided in the FBA configuration file (login form, plus authentication data such as the user name, password, and so on) fails to authenticate the crawler, status code 616 is assigned to all pages dependent on the form-based authentication. The administrator should investigate to find out why the FBA configuration is not working.

- 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.

- 618** Indicates that the redirect URL is not valid when the crawler receives the following HTTP status codes. It is possible that the location of the HTTP response header is not valid.

301 Moved Permanently
302 Found

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 status code greatly reduces throughput.

69x status codes

Status codes 690 through 699 are never recorded in the crawler's persistent database. These codes represent outcomes that do not reflect the true outcome of a download from a remote host, but rather a temporary condition inside the crawler, such as one component that shuts down while another is waiting for a result or sending a result. These status codes appear in some logs, but not in the persistent record, and so should not be used as selection-set values.

7xx status 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 status 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 see a large number of this type of status 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 status codes mostly occur when you make changes to the crawl space after the crawler has been running for awhile. These status codes typically do not indicate problems that you need to address.

3020 Indicates that a document with status 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 build 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 a delta index build, for example, to force the main index build after you 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.

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 (delta or full), 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 a delta index build so that you can force the main index build 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 start an index build.
5. To remove an index from the index queue, click  **Remove**.

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 improve 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).

Changing how query statistics are calculated

You can change how the system calculates the number of popular queries and recent queries.

About this task

When you monitor the search servers, you can select options to view a list of the 50 most popular queries and a list of the 50 most recently processed queries. In the default search server configuration, queries that have an equivalent query string and different range settings for results are counted as independent queries. Thus, for example, the query count is incremented when a user clicks an option to view the next page of results.



You can change how the system calculates query statistics by editing the `runtime-generic.properties` file for the search server. If you set the **distinctRecentQueryCheck** parameter to true, the system counts only queries that return the initial page of results as independent queries.

To edit the search server properties, you must log in as the enterprise search administrator. 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

To change how the system calculates query statistics:

1. Log in as the enterprise search administrator. This user ID was specified when OmniFind Enterprise Edition was installed. For a multiple server configuration, log in on the search server.
2. Edit the following file, where *collection_ID* is the ID for the collection that you want to change and *node_ID* identifies the search server where you are making the change:

```
ES_NODE_ROOT/master_config/collection_ID.runtime.node_ID/runtime-generic.properties
```
3. Set the value of the **distinctRecentQueryCheck** parameter to true, and then save and close the file.
4. For a multiple server configuration, repeat the preceding steps on the second search server.
5. If you want to change the query statistic behavior for additional collections, repeat the preceding steps for each collection that you want to change.
6. For the changes to become effective, log in to the enterprise search administration console and restart the search servers:
 - a. Click **System** to open the System view.
 - b. Select the Search page.
 - c. For each search server that you changed, click  **Stop**.
 - d. For each search server that you changed, click  **Start**.

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.

Important: The Data Listener will not be supported in future releases. Use the search and index (SI-API) APIs instead of the Data Listener APIs to develop client applications for enterprise search. The information below is provided for users who previously created Data Listener applications.

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, or change the maximum number of documents that can be held in temporary storage, 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 one or both of these configuration options.

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.

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.

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 document tracking is enabled 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 286

Related reference

“URI formats in an enterprise search index” on page 113

Viewing log files about dropped documents

You can view logged messages about documents that are dropped from an enterprise search system. This information is available only if document tracking is enabled for the collection.


About this task


To view a report about a dropped document, you need to know the URI of the document. By viewing the dropped document log files, you can see the dates and times that any document was dropped, the severity level of the error, the component and session where the problem occurred, and the detailed error message.

Procedure

To view log files for 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  **Dropped document log files**.
3. On the Dropped Document Log Files page, select the log file that you want to view. The name of each log file indicates whether the document was dropped by the parser (pd) or during an index build (in) and includes the date that the file was created. If more than one log file of the same type is created on the same date, a numeric suffix indicates the order in which the file was created on that date. For example:

```
dropped_doc_in_20060525.log  
dropped_doc_pd_20060524.log (contains the most recent entries on this date)  
dropped_doc_pd_20060524.log.1  
dropped_doc_pd_20060524.log.2 (contains the oldest entries on this date)
```

4. 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.

5. 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.

You can click buttons to move to the next and previous messages in the log file.

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](#)

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 310

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 310

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.

Important: Selecting this option can negatively affect system performance. Log all messages only when you need to troubleshoot problems or if you are requested to do so by IBM Software Support.

4. In the **Maximum size of each log file** field, type the maximum number of megabytes for each log file. The default value is 10MB.

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 16.

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**.

8. For your changes to become effective, enter the following commands to stop and restart the enterprise search system.

```
esadmin system stopall  
esadmin system startall
```

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

Related tasks

“Configuring collection-level alerts” on page 306

“Configuring system-level alerts” on page 307

Changing the size of the query log

You can increase or decrease the size of the log files that are created for query processing by editing a configuration file. There is no support for this task in the enterprise search administration console.

About this task

During query processing, log data is written to the `collection_ID_OmniFindQueryLog_date.log` file, where `collection_ID` identifies the collection that you want to configure and `date` is the date that the log file is created. You can increase or decrease the size of this log file, depending on how much data you want to log before a new log file is created.

Procedure

To change the size of the query processing log file:

1. Log in as the enterprise search administrator. On a multiple server enterprise search system, log in on the index server.
2. Open the `ES_ROOT_NODE/master_config/collection_ID.runtime.node1/runtime-generic.properties` file.
3. Search for the property **MaxFileSize**. Increase or decrease its value to increase or decrease the size of the log files, and save your changes.
4. For a single server enterprise search system:
 - a. Use the enterprise search administration console to monitor the collection that you changed and stop the search servers.
 - b. Restart the `ESearchServer` application in WebSphere Application Server.
 - c. In the administration console, restart the search servers that you stopped.
 - d. Open the search application in a new browser.
5. For a multiple server enterprise search system:
 - a. Log in as the enterprise search administrator on the index server.
 - b. Enter the following commands to restart the enterprise search system:

```
esadmin system stopall
esadmin system startall
```

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 system or a collection name) and the date that the file was created. If more than one log file of the same type is created on the same date, a numeric suffix indicates the order in which the file was created. For example:

```
log_file_type_20060526.log (contains the most recent entries on this date)
log_file_type_20060526.log.1
log_file_type_20060526.log.2 (contains the oldest entries on this date)
log_file_type_20060525.log (contains the most recent entries on this date)
log_file_type_20060525.log.1
log_file_type_20060525.log.2
log_file_type_20060525.log.3 (contains oldest entries on this date)
```
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.

You can click buttons to move to the next and previous messages in the log file.

Backing up and restoring an enterprise search system

Backup and restore scripts enable you to back up and restore the enterprise search system.

What the scripts back up

The scripts back up and restore the following files:

- Configuration files from the `ES_NODE_ROOT/master_config` directory
- Database files for the crawlers, including all crawler metadata, such as when data sources were last crawled
- All files in the `ES_NODE_ROOT/data` directory
- Index files for collections that are configured with non-default data directories

Backup directory structure

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

Usage guidelines

- You can back up data from one computer and restore it to another computer. However:
 - You cannot restore files that were backed up from one version of OmniFind Enterprise Edition to a system that is running a different version of OmniFind Enterprise Edition.
 - You must restore data to a system that has the same or a greater number of enterprise search servers. For example, if you back up an enterprise search system that runs on a single server, you can restore data to a system that uses two or four enterprise search servers. You cannot restore data that was backed up from a four server system to a system that uses two servers or a single server.
 - You cannot restore files that were backed up from one operating system to a system that is running a different operating system. For example, if you installed enterprise search on AIX system and now want to run it on Linux, you must install a new enterprise search system on your Linux servers.
- Build the main index before you start the backup so that the most current indexed data is backed up.
- All settings for the installation directory (`ES_INSTALL_ROOT`), the data directory (`ES_NODE_ROOT`), and the enterprise search administrator ID and password must be the same between the backed up system and the system to which data is restored.

- For a multiple server configuration, back up and restore the system from the enterprise search index server. Because all of the crawler data resides in databases on the crawler server, the scripts run remote commands to back up and restore the crawler data.
- 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.
- 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.
- If the system fails because of an irrecoverable error, you must re-install OmniFind Enterprise Edition and then run the restore script.

Backing up the enterprise search system

You back up an enterprise search system by using the `esbackup.sh` script for AIX, Linux, or Solaris, or the `esbackup.bat` script for Microsoft Windows.

Restrictions

The enterprise search administrator ID must have permission to write to the directory that you specify when you run the backup script.

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.

Attention: If you press Ctrl+C to interrupt the backup script, the system enters an inconsistent state. You must enter the following command to start all the service sessions and any running sessions that were stopped during the backup process:

```
esadmin system startall
```

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 OmniFind Enterprise Edition was installed.
2. If the common communication layer (CCL) for enterprise search is not running, start it:

AIX, Linux, or Solaris

```
startccl.sh -bg
```

Windows command prompt

```
startccl
```

Windows Services administrative tool

To start CCL in the background:

- a. Launch Windows Services: **Start** → **Programs** → **Administrative Tools** → **Services**.
 - b. Right-click **IBM OmniFind Enterprise Edition** and click **Start**.
3. For a multiple server configuration, ensure that the CCL is started on each server. Repeat the preceding steps as necessary.
 4. Start the backup, where *backup_directory* is a directory for the backed up data:

AIX, Linux, or Solaris

```
esbackup.sh backup_directory
```

Windows command prompt

```
esbackup.bat backup_directory
```

Related reference

“Enterprise search commands, return codes, and session IDs” on page 351

Restoring the enterprise search system

After you re-install OmniFind Enterprise Edition, you can use the `esrestore.sh` script for AIX, Linux, or Solaris, or the `esrestore.bat` script for Microsoft Windows to restore an enterprise search system.

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.

You cannot restore files that were backed up from one version of OmniFind Enterprise Edition to a system that is running a different version of OmniFind Enterprise Edition. In addition, the system to which you are restoring data must have the same or a greater number of enterprise search servers than the system from which data was backed up.

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 OmniFind Enterprise Edition was installed.
2. If the common communication layer (CCL) for enterprise search is not running, start it:

AIX, Linux, or Solaris

```
startccl.sh -bg
```

Windows command prompt

```
startccl
```

Windows Services administrative tool

To start CCL in the background:

- a. Launch Windows Services: **Start** → **Programs** → **Administrative Tools** → **Services**.
 - b. Right-click **IBM OmniFind Enterprise Edition** and click **Start**.
3. For a multiple server configuration, ensure that the CCL is started on each server. Repeat the preceding steps as necessary.
 4. Stop the controller:

```
esadmin stop
```
 5. Restore enterprise search data, where *backup_directory* is the directory where you backed up the files:

AIX, Linux, or Solaris

```
esrestore.sh backup_directory
```

Windows command prompt
`esrestore.bat backup_directory`

Related reference

“Enterprise search commands, return codes, and session IDs” on page 351

Exporting and importing collection configurations

You can export and import individual collection configurations. Only the collection configuration files are exported and imported, not the collection data.

You can export a collection from one enterprise search system, and then import the collection into a different enterprise search system. However, you can export and import collections only between systems that are running the same version of enterprise search. For example, you can export a collection from a version 8.4 system and then import the collection into a version 8.4 or version 8.4.0.150 system, but you cannot export a collection from a version 8.3 system and import it into a version 8.4 system.

If you export a collection, you can import it only to an enterprise search system that uses the same operating system. For example, you cannot export a collection from an enterprise search system that is installed on Linux and import it to an enterprise search system that is installed on Windows.

Exporting collections

To export a collection, you use the **esadmin export** command to export one collection at a time. There is no support for exporting collections in the enterprise search administration console.

1. Log in as the enterprise search administrator. In a multiple server configuration, you must log in on the index server. (The command fails if you attempt to run it from any other enterprise search server.)
2. Enter the following commands:

```
esadmin export -help  
esadmin export -cid collection_id [-fname export_filename] [-verbose]
```

Where:

-help

Provides help information for the command.

-cid *collection_id*

Specifies the collection ID for the collection to be exported.

Tip: To determine the collection ID for the collection that you want to export, you can use the enterprise search administration console or use the **esadmin report collections** command.

-fname *export_filename*

Specifies the path to the export file. If the file name is not absolute, then the ES_NODE_ROOT/dump directory is assumed. If you omit this option, a file that uses the following naming convention is created in the ES_NODE_ROOT/dump directory:

```
_export_yyyyMMdd_HHmssz.zip
```

where:

yyyymmdd

Is the current year, month, and day when the export command is run.

HHmmss

Is the current hour, minute, and second when the export command is run.

z Is the time zone offset from GMT when the export command is run.
For example, Pacific Standard Time is GMT -0800.

-verbose

Displays information that can help troubleshoot the export operation.

Importing collections

To import a collection, you use the **esadmin import** command to import one collection at a time. There is no support for importing collections in the enterprise search administration console.

1. Log in as the enterprise search administrator. In a multiple server configuration, you must log in on the index server. (The command fails if you attempt to run it from any other enterprise search server.)
2. Enter the following commands:

```
esadmin import -help
esadmin import -fname import_filename
                  [-cid new_collection_id]
                  [-name new_collection_name]
                  [-colDataDir new_collection_data_directory]
                  [-force]
                  [-verbose]
```

Where:

-help

Provides help information for the command.

-fname *import_filename*

Specifies the path to the import file. If file name is not absolute, then the ES_NODE_ROOT/dump directory is assumed.

-cid *new_collection_id*

Specifies a new collection ID if the collection needs to be imported with a different ID.

Tip: To determine the collection ID for the collection that you want to import, you can use the enterprise search administration console or use the **esadmin report collections** command.

-name *new_collection_name*

Specifies a new collection name if the collection needs to be imported with a different name.

-colDataDir *new_collection_data_directory*

Specifies the data directory for the collection. If omitted, a default directory is assigned.

-force

Forces the import of a collection that already exists in the target system. The system uses the collection ID for the imported collection to determine which collection to replace in the target system.

Important:

- After you import a collection, you cannot import a collection with the same collection ID again unless you use this option to force the collection to be imported.
- A collection that is imported by force does not retain the text analysis engines, dictionaries, and crawler plug-ins. These are overwritten with the information contained in the imported collection.
- Any crawlers associated with the collection are removed and replaced with crawlers that are specified in the import file. Because the crawlers are removed, all crawler metadata and documents that are not yet parsed are also removed. After the collection is imported, you must recrawl all documents to regenerate this data for the imported crawlers.

-verbose

Displays information that can help troubleshoot the import operation.

Usage guidelines**Text analysis engines and dictionaries**

Any text analysis engines and dictionaries (such a synonym, stop word, and boost word dictionaries) that are associated with a collection in the export system are not exported with the collection configuration data. Only the name associations to text analysis engines and dictionaries are exported.

On import, if a text analysis engine or dictionary with the same name exists in the target system, then it is associated with the imported collection. Otherwise, a warning message is displayed to indicate that the associations could not be established. For this reason, you should upload the text analysis engines and dictionaries to the target system and use the same names that were used in the imported collection.

If a text analysis engine or dictionary with the same name does not exist in the target system, then the association is broken. If the association is broken, the collection will function properly in the target system, but the collection will not use the corresponding text analysis engines or dictionaries.

Memory models

The memory model configured for the export system is not preserved. The memory model configured for the target system remains as configured. If you attempt to import a collection to a system that is configured for a smaller memory model, a warning message is displayed. The behavior of the collection might not work as expected and could have serious implications on the resource usage of the smaller system. To avoid problems, import the collection to a system that uses the same memory model or a larger memory model than the export system.

Crawlers

You must ensure that the crawler prerequisites are installed and configured in the target system

Imported crawlers do not work unless the data sources that the crawlers crawl are accessible. For example, if the collection includes a file system crawler that crawls a local file system, the crawler cannot crawl unless the same directory structure exists in the target system.

Crawler plug-ins are not exported. When you import a collection, a warning message is displayed, and then the import continues to

completion. After you import the collection, update the crawler properties and associate the crawler plug-ins. You must ensure that the crawler plug-ins are deployed on the target system to ensure the proper functioning of the crawler.

Related reference

“Enterprise search commands, return codes, and session IDs” on page 351

Integration with Lotus Notes Version 8

You can expand the search capabilities of IBM Lotus Notes Version 8 client deployments by deploying the OmniFind Enterprise Edition Lotus Notes search plug-in. This plug-in enables users to search enterprise search collections from the Lotus Notes client search bar.

To integrate enterprise search capabilities with Lotus Notes, you must create the plug-in update site. Users can then install the plug-in in their local Lotus Notes Version 8 client systems.

Creating the enterprise search plug-in update site

To integrate enterprise search with Lotus Notes Version 8, you must deploy the enterprise search Eclipse update site to a Web server in your organization. The update site allows users to deploy the enterprise search plug-in to their local Lotus Notes client installations.

About this task

The files required to create the enterprise search update site are provided in the `com.ibm.es.notes.search.plugin_8.4.0.150.zip` file.

Procedure

To create the enterprise search Eclipse update site:

1. Copy the `com.ibm.es.notes.search.plugin_8.4.0.150.zip` file to a Web server in your organization.
2. Unpack the contents of the zip file in a subdirectory of a shared directory in your Web server installation path.

For example, to deploy the update site to the Web server that you use for enterprise search, unpack the contents of the zip file in the `C:\Program Files\IBM\HTTP Server\htdocs\en_US\omnifind` directory.

Installing the enterprise search plug-in in the Lotus Notes version 8 client

To use enterprise search functions for query processing, you can add the enterprise search plug-in to the Lotus Notes search bar.

About this task

The top-right corner of the Lotus Notes client includes a search bar. The search bar has several search plug-ins that are provided with the basic Lotus Notes client installation. After you install the enterprise search plug-in, you can search enterprise search collections from the search bar.

Procedure

To install the enterprise search plug-in to the Lotus Notes client:

1. Open the NOTES_INSTALL_ROOT\notes.ini file, where NOTES_INSTALL_ROOT is typically c:\lotus\notes.
2. At the end of the file, add a property named OMNIFIND_ENTERPRISE_EDITION_SEARCH_SERVER_URL and specify http://hostname:port/ESearchApplication/search.do?q= for the value, where *hostname* is the host name of the search server for enterprise search and *port* is the Web server port.
3. Save and close the file.
4. Open the Lotus Notes client application.
5. Select **File** → **Application** → **Install**.
6. Select the **Search for new features to install** radio button.
7. Click **Add Remote Location**.
8. Provide a name for this new location, such as enterprise search. In the URL field, enter the root URL of the enterprise search update site. For example, if you unpacked the com.ibm.es.notes.search.plugin_8.4.0.150.zip file into the omnifind subdirectory in the root IBM HTTP Server English directory, then the URL that you specify is http://hostname:port/omnifind/.
9. Click **Finish** to save the new remote site. .
10. Select the check box next to the site name and click **Next**.
11. Review the license agreement, select **I accept the terms in the license agreement**, and click **Next**.
12. Click **Finish** to install the plug-in. If you receive a warning that you are installing an unsigned plug-in, select the **Install this plug-in** radio button and click **OK**.
13. When prompted, click **Yes** to restart the Lotus Notes client.

You can now click the menu next to the Lotus Notes search bar and select **OmniFind Enterprise Edition** as a search option. If you enter a query term and click the **Search** button, the request is directed to the enterprise search server and the results are rendered in the Lotus Notes Web browser window.

Integration with WebSphere Portal

You can expand the search capabilities of IBM WebSphere Portal by deploying the Search portlet for enterprise search portlets in WebSphere Portal and by configuring WebSphere Portal to use the Search portlet as the default search engine.

Integration points

The OmniFind Enterprise Edition installation program provides setup scripts for integrating enterprise search with WebSphere Portal. After you run these scripts, your enterprise search system can integrate with WebSphere Portal in several ways:

Search portlet for enterprise search

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 affecting other applications in the portal.

If you deploy the Search portlet for enterprise search in WebSphere Portal, you can use the WebSphere Portal interface to search enterprise search collections and work with the search results. 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.

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 are registered with the Search Center.

If you run the setup scripts to integrate enterprise search with WebSphere Portal version 5.1, an Enterprise Search page is added to a page in the Search Center interface. You can select this page to search only enterprise search collections, or you can enter a query that searches enterprise search collections and other collections that are available in the Search Center.

If you run the setup scripts to integrate enterprise search with WebSphere Portal version 6, enterprise search functionality is integrated as a federated service that you can use to search enterprise search collections and other collections that are available in the Search Center.

WebSphere Portal Search bar

The top-right corner of all WebSphere Portal interface themes includes a Search bar. The default behavior of this bar is to direct all search requests to the default Search Center search engine. To use the more powerful enterprise search functions for query processing, you can change this default behavior so that all search requests are redirected to the Search portlet for enterprise search instead.

WebSphere Portal and Web Content Management crawlers

To include WebSphere Portal sites and IBM Workplace Web Content Management sites in an enterprise search index, you can use the enterprise

search administration console to configure WebSphere Portal and Web Content Management crawlers. You can then use the enterprise search portlet or a search application to search the indexed content.

The WebSphere Portal crawler can crawl WebSphere Portal version 5.1 and WebSphere Portal version 6 sites. The Web Content Management crawler can crawl sites on a WebSphere Portal version 6 server.

IBM Lotus Quickr documents

To include IBM Lotus Quickr documents in an enterprise search index, you can use the enterprise search administration console to configure a Seed list crawler. You can then use the enterprise search portlet in WebSphere Portal or a standalone search application to search the indexed content.

The Seed list crawler can crawl Lotus Quickr content (document) libraries on a WebSphere Portal version 6 server.

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 Search portlet for enterprise search, you can search Web sites plus all of the other data source types that are supported by an enterprise search system.

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, the enterprise search workload can be distributed over multiple servers, with two servers providing support for search and retrieval processing.

Setup scripts for integrating enterprise search with WebSphere Portal

To integrate enterprise search with IBM WebSphere Portal, you can run setup scripts that are provided with the OmniFind Enterprise Edition installation program.

You must copy the JAR file that contains the setup scripts for your version of WebSphere Portal from the enterprise search server to the server where WebSphere Portal is installed. The setup scripts:

- Deploy EAR files that enable you to use enterprise search within WebSphere Portal and create crawlers for adding WebSphere Portal and IBM Workplace Web Content Management content to enterprise search collections.
- Deploy WAR files that are required by the enterprise search portlet.
- Create pages in WebSphere Portal and assign the enterprise search portlet files to those pages.
- Copy all required JAR files to the WebSphere Portal installation directories (JAR files already in the installation directories are backed up before the JAR files used for enterprise search are copied).
- Provide an integration point for WebSphere Information Integrator Content Edition to search Portal Document Manager documents.

After you run the scripts, you must use the WebSphere Portal administration interface to update search portlet properties and specify information about the search server for enterprise search.

Usage guidelines

- The scripts set up all integration points between enterprise search and WebSphere Portal. For example, you cannot selectively install the portlet and not install EAR files that support the WebSphere Portal and Web Content Management crawlers.
- If you do not set up WebSphere Information Integrator Content Edition, and later decide that you want to use a portlet to search Portal Document Manager documents, you must run a script to remove enterprise search from WebSphere Portal. You can then run the setup script again and specify the WebSphere Information Integrator Content Edition installation path.
- The scripts stop and restart WebSphere Portal. You might want to run the scripts after normal working hours to ensure that your user community is not affected by unavailability of portal services.
- If an errors occur while the setup scripts are running, run the setup script again. Tasks that completed successfully during the first attempt might report errors, but the setup process continues and completes the remaining tasks.
- The first time that you access the Enterprise Search portlet page after you run the setup script, the page might be slow to appear because the system must compile Java Server Pages (JSP files) for the portlet.

Setting up enterprise search in WebSphere Portal version 5.1

To integrate an enterprise search system with WebSphere Portal version 5.1.0 or later, you use the `wp5_install` script.

About this task

The files required to integrate enterprise search with WebSphere Portal are provided in the `es.wp5.install.jar` file. When you unpack this file, the following files are extracted:

- `ESSearchPortlet.war`
- `ESSearchAdapterPortlet.war`
- `ESSearchAdapter.ear`
- `ESPACServer.ear`
- `esapi.jar`
- `siapi.jar`
- `es.security.jar`
- Script, batch, XML, and JACL files that are needed by the installation

Procedure

To integrate enterprise search with a WebSphere Portal version 5.1 system:

1. Copy the `es.wp5.install.jar` file from the enterprise search server to the WebSphere Portal server, and then use the Java `JAR` command (or the `TAR` command) to unpack the file.
2. Optional: If you want to support integration with WebSphere Portal Document Manager (PDM), do one of the following steps:
 - Run the WebSphere Information Integrator Content Edition installation program, select the option to perform a connector only installation, and install the PDM connector on the WebSphere Portal server.
 - Create the WebSphere Information Integrator Content Edition directory structure on the WebSphere Portal server, and copy the following files from

an existing WebSphere Information Integrator Content Edition installation to the WebSphere Portal server, where CE_ROOT specifies the root WebSphere Information Integrator Content Edition installation directory:

```
CE_ROOT/lib/vbr.jar  
CE_ROOT/ejb/vbr_pdm.jar  
CE_ROOT/war/services.war  
CE_ROOT/vbr_services.properties
```

3. At a command prompt, run the **wp5_install.bat** command (on Windows) or the **wp5_install.sh** command (on AIX, Linux, or Solaris). The following example shows parameters on separate lines for readability; you must specify the parameters with the command:

```
wp5_install.bat  
-WASDir "C:\\Program Files\\WebSphere\\AppServer"  
-WASUser wpsbind -WASPassword wpsbind  
-WPSDir "C:\\Program Files\\WebSphere\\PortalServer"  
-WPSUser wpsadmin -WSPassword wpsadmin  
-WPSHost "portalserver.ibm.com:9081"  
-IICEDir "C:\\IICE"
```

WASDir

The fully qualified path for the WebSphere Application Server installation directory.

WASUser

The user name for the WebSphere Application Server administrative user; required only if global security is enabled in WebSphere Application Server.

WASPassword

The password for the WebSphere Application Server administrative user, if specified.

WPSDir

The fully qualified path for the WebSphere Portal installation directory.

WPSUser

The user name for the WebSphere Portal administrative user.

WSPassword

The password for the specified WebSphere Portal administrative user.

WPSHost

The WebSphere Portal server host name and port number.

IICEDir

The fully qualified path for the WebSphere Information Integrator Content Edition installation directory; required only if you previously set up the Portal Document Manager connector on the WebSphere Portal server.

4. After you run the script (WebSphere Portal is stopped and restarted), update the Enterprise Search portlet to identify the search server:
 - a. Log in to WebSphere Portal with the Portal administrator ID and password.
 - b. Click **Administration** in the upper right corner.
 - c. Click **Portlet Management** in the navigation area to the left, and then click **Portlets**.
 - d. Change the **Search by** option to **Title contains**.
 - e. In the **Search** field, type enterprise search and then click the **Search** button.
 - f. After new icons are displayed to the right, click the wrench icon to configure the search portlet for enterprise search.

- g. In the list of portlet parameters, modify the following parameters:

hostname

Specify the fully qualified host name of a search server for enterprise search.

- port** Specify the port number used by WebSphere Application Server on the search server for enterprise search. The default value is 80 (the default value for SSL communication is 443).

username

If global security is enabled in WebSphere Application Server on the search server, specify a user name that is valid in a WebSphere Application Server user registry.

password

If you specified a WebSphere Application Server user name, specify the corresponding password.

protocol

Specify the protocol used for communication between WebSphere Portal and the search server. The default is HTTP. If you use SSL, specify HTTPS.

trustStore

If you use SSL, specify the fully qualified path (with the file name) for the SSL certificate store.

trustPassword

If you use SSL, specify the password for the specified trustStore file.

ssoCookieName

Specify the name of the cookie that contains the single sign-on (SSO) token string. The default value is LtpaToken.

proxyHost

If a proxy server is required to access the search server for enterprise search, specify the fully qualified host name of a proxy server.

proxyPort

If you specified a proxy server, specify the port number for the proxy server.

proxyUser

If the proxy server requires basic authentication, specify a user name to use to log in to the proxy server.

proxyPassword

If you specified a user name for the proxy server, specify the corresponding password.

- h. Click **OK** to save your changes.

Configuring the WebSphere Portal version 5.1 Search bar to use enterprise search

You can configure WebSphere Portal version 5.1.0 or later to use enterprise search when users submit queries in the Search bar instead of the default WebSphere Portal search engine.

Before you begin

5. Open the AdminLinkBarInclude.jsp file and save the file. This step, which updates the modified date of the file to ensure that the file is recompiled, is optional if you use your own theme instead of the default WebSphere Portal theme.
6. Stop and restart the WebSphere Portal application server instance.

Removing enterprise search from WebSphere Portal version 5.1

To remove enterprise search from a WebSphere Portal version 5.1.0 or later system, you use the `wp5_uninstall` script.

About this task

When you remove enterprise search from WebSphere Portal, the portlet parameters that you specified for the Enterprise Search portlet as part of the setup process are not saved.

When you start the script, the script stops the WebSphere Portal server. After the enterprise search software is removed, the script restarts the WebSphere Portal server.

Procedure

To remove enterprise search from a WebSphere Portal version 5.1 system:

At a command prompt, run the **wp5_uninstall.bat** command (on Windows) or the **wp5_uninstall.sh** command (on AIX, Linux, or Solaris). The following example shows parameters on separate lines for readability; you must specify the parameters with the command:

```
wp5_uninstall.bat
-WASDir "C:\\Program Files\\WebSphere\\AppServer"
-WASUser wpsbind -WASPassword wpsbind
-WPSDir "C:\\Program Files\\WebSphere\\PortalServer"
-WPSUser wpsadmin -WSPassword wpsadmin
-WPSHost "portalserver.ibm.com:9081"
```

WASDir

The fully qualified path for the WebSphere Application Server installation directory.

WASUser

The user name for the WebSphere Application Server administrative user; required only if global security is enabled in WebSphere Application Server.

WASPassword

The password for the WebSphere Application Server administrative user, if specified.

WPSDir

The fully qualified path for the WebSphere Portal installation directory.

WPSUser

The user name for the WebSphere Portal administrative user.

WSPassword

The password for the specified WebSphere Portal administrative user.

WPSHost

The WebSphere Portal server host name and port number.

Setting up enterprise search in WebSphere Portal version 6

To integrate an enterprise search system with WebSphere Portal version 6, you use the `wp6_install` script.

About this task

The files required to integrate enterprise search with WebSphere Portal are provided in the `es.wp6.install.jar` file. When you unpack this file, the following files are extracted:

- `ESSearchPortlet.war`
- `ESPACServer.ear`
- `esapi.jar`
- `es.search.provider.jar`
- `es.security.jar`
- Search application source type icons that are used in the search provider results page
- Script, batch, XML, and JACL files that are needed by the installation

Procedure

To integrate enterprise search with a WebSphere Portal version 6 system:

1. Copy the `es.wp6.install.jar` file from the enterprise search server to the WebSphere Portal server, and then use the Java **JAR** command (or the **TAR** command) to unpack the file.
2. Optional: If you want to support integration with WebSphere Portal Document Manager (PDM), do one of the following steps:
 - Run the WebSphere Information Integrator Content Edition installation program, select the option to perform a connector only installation, and install the PDM connector on the WebSphere Portal server.
 - Create the WebSphere Information Integrator Content Edition directory structure on the WebSphere Portal server, and copy the following files from an existing WebSphere Information Integrator Content Edition installation to the WebSphere Portal server, where `CE_ROOT` specifies the root WebSphere Information Integrator Content Edition installation directory:

```
CE_ROOT/lib/vbr.jar
CE_ROOT/ejb/vbr_pdm.jar
CE_ROOT/war/services.war
CE_ROOT/vbr_services.properties
```

3. At a command prompt, run the **wp6_install.bat** command (on Windows) or the **wp6_install.sh** command (on AIX, Linux, or Solaris). The following example shows options on separate lines for readability; you must specify the options with the command:

```
wp6_install.bat
-WSPProfileDir "C:\\Program Files\\IBM\\WebSphere\\profiles\\wp_profile"
-WASDir "C:\\Program Files\\IBM\\WebSphere\\AppServer"
-WASUser wpsbind -WASPassword wpsbind
-WPSDir "C:\\Program Files\\IBM\\WebSphere\\PortalServer"
-WPSUser wpsadmin -WSPassword wpsadmin
-WPSHost "portalserver.ibm.com:9081"
-IICEDir "C:\\IICE"
```

WSPProfileDir

The fully qualified path for the WebSphere Portal profile directory. The

default path is /usr/IBM/WebSphere/AppServer/profiles/wp_profile on AIX systems, /opt/IBM/WebSphere/AppServer/profiles/wp_profile on Linux or Solaris systems, and C:\Program Files\IBM\WebSphere\profiles\wp_profile on Windows systems.

WASDir

The fully qualified path for the WebSphere Application Server root directory; required on AIX, Linux, and Solaris systems only. The default root directory path is /usr/IBM/WebSphere/AppServer on AIX systems, /opt/IBM/WebSphere/AppServer on Linux or Solaris systems, and C:\Program Files\IBM\WebSphere\AppServer on Windows systems.

WASUser

The user name for the WebSphere Application Server administrative user; required only if global security is enabled in WebSphere Application Server.

WASPassword

The password for the WebSphere Application Server administrative user, if specified.

WPSDir

The fully qualified path for the WebSphere Portal installation directory.

WPSUser

The user name for the WebSphere Portal administrative user.

WSPassword

The password for the specified WebSphere Portal administrative user.

WPSHost

The WebSphere Portal server host name and port number.

IICEDir

The fully qualified path for the WebSphere Information Integrator Content Edition installation directory; required only if you previously set up the Portal Document Manager connector on the WebSphere Portal server.

4. After you run the script, and stop and restart WebSphere Portal, update the Enterprise Search portlet to identify the search server:
 - a. Log in to WebSphere Portal with the Portal administrator ID and password.
 - b. Click **Administration** in the lower left corner.
 - c. Click **Portlet Management** in the navigation area to the left, and then click **Portlets**.
 - d. Change the **Search by** option to **Title contains**.
 - e. In the **Search** field, type enterprise search and then click the **Search** button.
 - f. After new icons are displayed to the right, click the wrench icon to configure the search portlet for enterprise search.
 - g. In the list of portlet parameters, modify the following parameters:

hostname

Specify the fully qualified host name of a search server for enterprise search.

port

Specify the port number used by WebSphere Application Server on the search server for enterprise search. The default value is 80 (the default value for SSL communication is 443).

username

If global security is enabled in WebSphere Application Server on the search server, specify a user name that is valid in a WebSphere Application Server user registry.

password

If you specified a WebSphere Application Server user name, specify the corresponding password.

protocol

Specify the protocol used for communication between WebSphere Portal and the search server. The default is HTTP. If you use SSL, specify HTTPS.

trustStore

If you use SSL, specify the fully qualified path (with the file name) for the SSL certificate store.

trustPassword

If you use SSL, specify the password for the specified trustStore file.

ssoCookieName

Specify the name of the cookie that contains the single sign-on (SSO) token string. The default value is LtpaToken.

proxyHost

If a proxy server is required to access the search server for enterprise search, specify the fully qualified host name of a proxy server.

proxyPort

If you specified a proxy server, specify the port number for the proxy server.

proxyUser

If the proxy server requires basic authentication, specify a user name to use to log in to the proxy server.

proxyPassword

If you specified a user name for the proxy server, specify the corresponding password.

- h. Click **OK** to save your changes.

Configuring the WebSphere Portal version 6 Search Center for enterprise search

You can configure WebSphere Portal version 6 to search enterprise search collections when users submit queries in the WebSphere Portal Search Center.

Restrictions

If the enterprise search collections to be searched are secure, users must run the Search portlet for enterprise search and configure a user profile. The profile is encrypted and stored in a secure enterprise search store. The profile must exist before users can submit queries to search secure collections from the WebSphere Portal Search Center.

About this task

The Search Center in WebSphere Portal version 6 supports federated search capabilities across multiple collections. The collections can contain various types of content, such as Portal Document Libraries and Portal Content (pages and portlets). After you run the setup scripts to integrate enterprise search with WebSphere Portal, you can configure the Search Center to also search enterprise search collections.

Procedure

To configure the Search Center to search enterprise search collections:

1. Log in to WebSphere Portal with the Portal administrator ID and password.
2. Click **Administration** in the lower left corner.
3. Click **Search Administration** in the navigation area to the left, and then click **Manage Search**.
4. Click **Search Services**, and then click **New Search Service**.
5. In the **Search service implementation** field, select the Enterprise Search search service, and then type the name that you want to use for the service in the **Service name** text box.
6. In the list of parameters, modify the following parameters:

hostname

Specify the fully qualified host name of a search server for enterprise search.

port

Specify the port number used by WebSphere Application Server on the search server for enterprise search. The default value is 80 (the default value for SSL communication is 443).

username

If global security is enabled in WebSphere Application Server on the search server, specify a user name that is valid in a WebSphere Application Server user registry.

password

If you specified a WebSphere Application Server user name, specify the corresponding password.

protocol

Specify the protocol used for communication between WebSphere Portal and the search server. The default is HTTP. If you use SSL, specify HTTPS.

trustStore

If you use SSL, specify the fully qualified path (with the file name) for the SSL certificate store.

trustPassword

If you use SSL, specify the password for the specified trustStore file.

ssoCookieName

Specify the name of the cookie that contains the single sign-on (SSO) token string. The default value is LtpaToken.

proxyHost

If a proxy server is required to access the search server for enterprise search, specify the fully qualified host name of a proxy server.

proxyPort

If you specified a proxy server, specify the port number for the proxy server.

proxyUser

If the proxy server requires basic authentication, specify a user name to use to log in to the proxy server.

proxyPassword

If you specified a user name for the proxy server, specify the corresponding password.

7. Click **OK** to save your changes.

Configuring the WebSphere Portal version 6 Search bar to use enterprise search

You can configure WebSphere Portal version 6 to use enterprise search when users submit queries in the Search bar instead of the default WebSphere Portal search engine.

Before you begin

Before you can redirect search requests to enterprise search, you must run the **wp6_install** setup script to integrate enterprise search with WebSphere Portal. You must also update the Enterprise Search portlet parameters to identify the host name, port, and other information about the search server for enterprise search.

About this task

The top-right corner of all WebSphere Portal interface themes includes a Search bar. The default behavior of this bar is to direct all search requests to the Search Center portlet. To use the more powerful enterprise search functions for query processing, you can change this default behavior so that all search requests are redirected to the Search portlet for enterprise search instead.

When you redirect the Search bar, the change affects pages that use the same WebSphere Portal theme as the Search portlet for enterprise search, and these pages must call the `banner_searchControl.jspf` file. Pages that use a different theme or that do not call the `banner_searchControl.jspf` file continue to use the default Search Center portlet.

After you complete this task, you cannot use the Search Center unless you undo the changes (for example, you can restore the original `banner_searchControl.jspf` file).

Procedure

To use the enterprise search portlet when users submit queries in the WebSphere Portal Search bar:

1. Stop the WebSphere Portal application server instance.
2. On the WebSphere Portal server, change to the `WPS_PROFILE_ROOT/installedApps/node_name/wps.ear/wps.war/themes/html/current_theme_name` directory, where `node_name` is node name for the WebSphere Portal server and `current_theme_name` is the currently applied theme for your WebSphere Portal server. The default theme name for a WebSphere Portal server is IBM.

3. Create a backup of the banner_searchControl.jspf file by copying this file and renaming it (for example, banner_searchControl.jspf.BACKUP).
4. Edit the banner_searchControl.jspf file and replace the contents with the following text. In the action= attribute of the form element, replace localhost:10038 with the host name and port number of your WebSphere Portal server.

```
<%@ taglib uri="/WEB-INF/tld/SearchMenuControl.tld" prefix="searchmenu" %>
<%String ic = (bidiImageRTL == null) ? "icons/scope_search_submit.gif" :
"icons/scope_search_submit"+bidiImageRTL+".gif";%>

<searchmenu:adminlinkinfo name="SEARCH_CENTER">
<div class="searchControl">
<form name="SearchForm" style="margin: 0px;" method="GET"
action="http://localhost:10038/wps/omnifind/portalSearchBar.jsp">
<table border="0" cellpadding="0" cellspacing="0">
<tr>
<td><span class="wpsInstructionText">
<portal-fmt:text key="search.theme.control.label" bundle="nls.engine"/></span></td>
<td valign="middle" style="padding: 0px 4px 0px 4px;">
<input type="text" name="q"></input></td>
<td valign="middle"><input tabIndex="4" valign="middle"
title="<portal-fmt:text key='search.theme.searchresultsicon.alttext' bundle='nls.engine' />"
alt="<portal-fmt:text key='search.theme.searchresultsicon.alttext' bundle='nls.engine' />"
src="<portal-logic:urlFindInTheme file=">"/>" type="image"></input></td>
</tr>
</table>
</form>
</div>
</searchmenu:adminlinkinfo>
```

5. Open the banner.jspf file and save the file. This step, which updates the modified date of the file to ensure that the file is recompiled, is optional if you use your own theme instead of the default WebSphere Portal theme.
6. Open the Default.jsp file and save the file.
7. Restart the WebSphere Portal application server instance.

Setting up the enterprise search portlet for Lotus Quickr

You can set up the enterprise search portlet in WebSphere Portal version 6 to search Lotus Quickr sources.

Before you begin

Run the **wp6_install.bat** command (on Windows) or the **wp6_install.sh** command (on AIX, Linux, or Solaris) and follow the procedures to set up enterprise search in WebSphere Portal version 6.

Procedure

To set up the enterprise search portlet in WebSphere Portal version 6 to search Lotus Quickr sources:

1. Update the portlet parameters in the WebSphere Portal configuration:
 - a. Log in to WebSphere Portal with the Lotus Quickr administrator ID and password.
 - b. Click **Site Administration** and then click **Advanced Administration**.
 - c. Click **Portlet Management** in the navigation area to the left, and then click **Portlets**.
 - d. Change the **Search by** option to **Title contains**.
 - e. In the **Search** field, type enterprise search and then click the **Search** button.

- f. After new icons are displayed to the right, click the wrench icon to configure the search portlet for enterprise search.
- g. In the list of portlet parameters, modify the following parameters:

hostname

Specify the fully qualified host name of a search server for enterprise search.

port

Specify the port number used by WebSphere Application Server on the search server for enterprise search. The default value is 80 (the default value for SSL communication is 443).

username

If global security is enabled in WebSphere Application Server on the search server, specify a user name that is valid in a WebSphere Application Server user registry.

password

If you specified a WebSphere Application Server user name, specify the corresponding password.

protocol

Specify the protocol used for communication between WebSphere Portal and the search server. The default is HTTP. If you use SSL, specify HTTPS.

trustStore

If you use SSL, specify the fully qualified path (with the file name) for the SSL certificate store.

trustPassword

If you use SSL, specify the password for the specified trustStore file.

ssoCookieName

Specify the name of the cookie that contains the single sign-on (SSO) token string. The default value is LtpaToken.

proxyHost

If a proxy server is required to access the search server for enterprise search, specify the fully qualified host name of a proxy server.

proxyPort

If you specified a proxy server, specify the port number for the proxy server.

proxyUser

If the proxy server requires basic authentication, specify a user name to use to log in to the proxy server.

proxyPassword

If you specified a user name for the proxy server, specify the corresponding password.

- h. Click **OK** to save your changes.
2. To access portlet after you set it up:
 - a. Log in to the Lotus Quickr server.
 - b. In the browser window, change the URL to the following:
`http://host_name:port/lotus/myquickr/ESSearchPortlet`

Removing enterprise search from WebSphere Portal version 6

To remove enterprise search from a WebSphere Portal version 6 system, you use the `wp6_uninstall` script.

About this task

When you remove enterprise search from WebSphere Portal, the portlet parameters that you specified for the Enterprise Search portlet as part of the setup process are not saved.

When you start the script, the script stops the WebSphere Portal server. After the enterprise search software is removed, the script restarts the WebSphere Portal server.

Procedure

To remove enterprise search from a WebSphere Portal version 6 system:

At a command prompt, run the **`wp6_uninstall.bat`** command (on Windows) or the **`wp6_uninstall.sh`** command (on AIX, Linux, or Solaris). The following example shows parameters on separate lines for readability; you must specify the parameters with the command:

```
wp6_uninstall.bat
-WPSProfileDir "C:\\Program Files\\IBM\\WebSphere\\AppServer\\profiles\\wp_profile"
-WASDir "C:\\Program Files\\IBM\\WebSphere\\AppServer"
-WASUser wpsbind -WASPassword wpsbind
-WPSDir "C:\\Program Files\\IBM\\WebSphere\\PortalServer"
-WPSUser wpsadmin -WSPassword wpsadmin
-WPSHost "portalserver.ibm.com:9081"
```

WPSProfileDir

The fully qualified path for the WebSphere Portal profile directory.

WASDir

The fully qualified path for the WebSphere Application Server root directory; required on AIX, Linux, and Solaris systems only.

WASUser

The user name for the WebSphere Application Server administrative user; required only if global security is enabled in WebSphere Application Server.

WASPassword

The password for the WebSphere Application Server administrative user, if specified.

WPSDir

The fully qualified path for the WebSphere Portal installation directory.

WPSUser

The user name for the WebSphere Portal administrative user.

WSPassword

The password for the specified WebSphere Portal administrative user.

WPSHost

The WebSphere Portal server host name and port number.

Enterprise search integration with WebSphere Portal clustered systems

You can set up the enterprise search portlet to run in a WebSphere Portal version 6 clustered system.

Usage guidelines

- Before you run the setup scripts for enterprise search, ensure that the WebSphere Application Server Network Deployment Manager is running and that each of the nodes in the cluster are running.
- The scripts set up all integration points between enterprise search and WebSphere Portal. For example, you cannot selectively install the portlet and not install EAR files that support the WebSphere Portal and Web Content Management crawlers.
- The scripts stop and restart all instances of the WebSphere Portal server in the cluster. You might want to run the scripts after normal working hours to ensure that your user community is not affected by the unavailability of portal services.
- If errors occur while the setup scripts are running, run the setup script again. Tasks that completed successfully during the first attempt might report errors, but the setup process continues and completes the remaining tasks.
- The first time that you access the Enterprise Search portlet page after you run the setup script, the page might be slow to appear because the system must compile Java Server Pages (JSP files) for the portlet.

Setting up enterprise search in a WebSphere Portal clustered system

To integrate an enterprise search system with a WebSphere Portal version 6 clustered system, you use the `wp6_cluster_install` script.

About this task

The files required to integrate enterprise search with WebSphere Portal are provided in the `es.wp6.install.jar` file. When you unpack this file, the following files are extracted:

- `ESSearchPortlet.war`
- `ESPACServer.ear`
- `esapi.jar`
- `es.search.provider.jar`
- `es.security.jar`
- Script, batch, XML, and JACL files that are needed by the installation

Procedure

To integrate enterprise search with a WebSphere Portal version 6 clustered system:

1. Copy the `es.wp6.install.jar` file from the enterprise search server to each node in the cluster where WebSphere Portal is installed, and then use the Java **JAR** command (or the **TAR** command) to unpack the file.
2. Optional: If you want to support integration with WebSphere Portal Document Manager (PDM), do one of the following steps on each node in the cluster:

- Run the WebSphere Information Integrator Content Edition installation program, select the option to perform a connector only installation, and install the PDM connector on the WebSphere Portal server.
- Create the WebSphere Information Integrator Content Edition directory structure on the WebSphere Portal server, and copy the following files from an existing WebSphere Information Integrator Content Edition installation to the WebSphere Portal server, where CE_ROOT specifies the root WebSphere Information Integrator Content Edition installation directory:

```
CE_ROOT/lib/vbr.jar
CE_ROOT/ejb/vbr_pdm.jar
CE_ROOT/war/services.war
CE_ROOT/vbr_services.properties
```

3. At a command prompt, run the **wp6_cluster_copyFiles.bat** command (on Windows) or the **wp6_cluster_copyFiles.sh** command (on AIX, Linux, or Solaris) on each node in the cluster where WebSphere Portal is installed. The following examples show the options on separate lines for readability; you must specify the options with the command:

```
wp6_cluster_copyFiles.bat
-WPSDir "C:\Program Files\IBM\WebSphere\PortalServer"
-WSPProfileDir "C:\Program Files\IBM\WebSphere\AppServer\profiles\wp_profile"

wp6_cluster_copyFiles.sh
-WASDir /opt/IBM/WebSphere/AppServer
-WPSDir /opt/IBM/WebSphere/PortalServer
-WSPProfileDir /opt/IBM/WebSphere/AppServer/profiles/wp_profile
```

4. At a command prompt, run the **wp6_cluster_install.bat** command (on Windows) or the **wp6_cluster_install.sh** command (on AIX, Linux, or Solaris). The following example shows options on separate lines for readability; you must specify the options with the command:

```
wp6_cluster_install.bat
-WPSClusterName MyCluster
-WSPProfileDir "C:\Program Files\IBM\WebSphere\profiles\wp_profile"
-WASDir "C:\Program Files\IBM\WebSphere\AppServer"
-WASUser wpsbind
-WASPassword wpsbind
-WPSDir "C:\Program Files\IBM\WebSphere\PortalServer"
-WPSUser wpsadmin
-WSPassword wpsadmin
-WPSHost "portalserver.ibm.com"
-webServerName webserver1
-webServerNodeName node1
-IICEDir "C:\Program Files\IBM\Content Edition"
```

WPSClusterName

The name of the cluster in which WebSphere Portal is installed.

WSPProfileDir

The fully qualified path for the WebSphere Portal profile directory. The default path is /usr/IBM/WebSphere/AppServer/profiles/wp_profile on AIX systems, /opt/IBM/WebSphere/AppServer/profiles/wp_profile on Linux or Solaris systems, and C:\Program Files\IBM\WebSphere\profiles\wp_profile on Windows systems.

WASDir

The fully qualified path for the WebSphere Application Server root directory; required on AIX, Linux, and Solaris systems only. The default root directory path is /usr/IBM/WebSphere/AppServer on AIX systems, /opt/IBM/WebSphere/AppServer on Linux or Solaris systems.

WASUser

The user name for the WebSphere Application Server administrative user; required only if global security is enabled in WebSphere Application Server.

WASPassword

The password for the WebSphere Application Server administrative user, if specified.

WPSDir

The fully qualified path for the WebSphere Portal installation directory.

WPSUser

The user name for the WebSphere Portal administrative user.

WPSPassword

The password for the specified WebSphere Portal administrative user.

WPSHost

The WebSphere Portal server host name and port number.

webServerName

The name of the Web server definition to which WebSphere Portal belongs.

webServerNodeName

The name of the WebSphere Application server node to which the Web server definition belongs.

IICEDir

The fully qualified path for the WebSphere Information Integrator Content Edition installation directory; required only if you previously set up the Portal Document Manager connector on the WebSphere Portal server.

5. After the script completes, open a Web browser and log in to the WebSphere Administration console on your Network Deployment server. The address is typically `http://hostname:9060/ibm/console`.
6. Expand the **Servers** section and select **Web servers**.
7. Select the **Select** box next to your Web server and then click the **Generate Plug-in** button.
8. Select the **Select** box next to your Web server and then click the **Propagate Plug-in** button.
9. Log out of the administration console.
10. Update the Enterprise Search portlet to identify the search server:
 - a. Log in to WebSphere Portal with the Portal administrator ID and password.
 - b. Click **Administration** in the lower left corner.
 - c. Click **Portlet Management** in the navigation area to the left, and then click **Portlets**.
 - d. Change the **Search by** option to **Title contains**.
 - e. In the **Search** field, type enterprise search and then click the **Search** button.
 - f. After new icons are displayed to the right, click the wrench icon to configure the search portlet for enterprise search.
 - g. In the list of portlet parameters, modify the following parameters:

hostname

Specify the fully qualified host name of a search server for enterprise search.

port Specify the port number used by WebSphere Application Server on the search server for enterprise search. The default value is 80 (the default value for SSL communication is 443).

username

If global security is enabled in WebSphere Application Server on the search server, specify a user name that is valid in a WebSphere Application Server user registry.

password

If you specified a WebSphere Application Server user name, specify the corresponding password.

protocol

Specify the protocol used for communication between WebSphere Portal and the search server. The default is HTTP. If you use SSL, specify HTTPS.

trustStore

If you use SSL, specify the fully qualified path (with the file name) for the SSL certificate store.

trustPassword

If you use SSL, specify the password for the specified trustStore file.

ssoCookieName

Specify the name of the cookie that contains the single sign-on (SSO) token string. The default value is LtpaToken.

proxyHost

If a proxy server is required to access the search server for enterprise search, specify the fully qualified host name of a proxy server.

proxyPort

If you specified a proxy server, specify the port number for the proxy server.

proxyUser

If the proxy server requires basic authentication, specify a user name to use to log in to the proxy server.

proxyPassword

If you specified a user name for the proxy server, specify the corresponding password.

- h. Click **OK** to save your changes.

Removing enterprise search from a WebSphere Portal clustered system

To remove enterprise search from a WebSphere Portal version 6 clustered system, you use the `wp6_cluster_uninstall` script.

About this task

When you remove enterprise search from WebSphere Portal, the portlet parameters that you specified for the Enterprise Search portlet as part of the setup process are not saved.

When you start the script, the script stops the WebSphere Portal server. After the enterprise search software is removed, the script restarts the WebSphere Portal server.

Procedure

To remove enterprise search from a WebSphere Portal version 6 clustered system:

1. At a command prompt, run the **wp6_cluster_uninstall.bat** command (on Windows) or the **wp6_cluster_uninstall.sh** command (on AIX, Linux, or Solaris) on one of the nodes in the cluster. The following example shows parameters on separate lines for readability; you must specify the parameters with the command:

```
wp6_cluster_uninstall.bat
-WPSClusterName MyCluster
-WSPProfileDir "C:\\Program Files\\IBM\\WebSphere\\profiles\\wp_profile"
-WASDir "C:\\Program Files\\IBM\\WebSphere\\AppServer"
-WASUser wpsbind
-WASPassword wpsbind
-WPSDir "C:\\Program Files\\IBM\\WebSphere\\PortalServer"
-WPSUser wpsadmin
-WSPassword wpsadmin
-WPSHost "portalserver.ibm.com"
-webServerName webserver1
-webServerNodeName node1
```

WPSClusterName

The name of the cluster in which WebSphere Portal is installed.

WSPProfileDir

The fully qualified path for the WebSphere Portal profile directory. The default path is /usr/IBM/WebSphere/AppServer/profiles/wp_profile on AIX systems, /opt/IBM/WebSphere/AppServer/profiles/wp_profile on Linux or Solaris systems, and C:\\Program Files\\IBM\\WebSphere\\profiles\\wp_profile on Windows systems.

WASDir

The fully qualified path for the WebSphere Application Server root directory; required on AIX, Linux, and Solaris systems only. The default root directory path is /usr/IBM/WebSphere/AppServer on AIX systems, /opt/IBM/WebSphere/AppServer on Linux or Solaris systems.

WASUser

The user name for the WebSphere Application Server administrative user; required only if global security is enabled in WebSphere Application Server.

WASPassword

The password for the WebSphere Application Server administrative user, if specified.

WPSDir

The fully qualified path for the WebSphere Portal installation directory.

WPSUser

The user name for the WebSphere Portal administrative user.

WSPassword

The password for the specified WebSphere Portal administrative user.

WPSHost

The WebSphere Portal server host name and port number.

webServerName

The name of the Web server definition to which WebSphere Portal belongs.

webServerNodeName

The name of the WebSphere Application server node to which the Web server definition belongs.

2. After the script completes, open a Web browser and log in to the WebSphere Administration console on your Network Deployment server. The address is typically `http://hostname:9060/ibm/console`.
3. Expand the **Servers** section and select **Web servers**.
4. Select the **Select** box next to your Web server and then click the **Generate Plug-in** button.
5. Select the **Select** box next to your Web server and then click the **Propagate Plug-in** button.
6. Log out of the administration console.

Migration from WebSphere Portal to enterprise search

Enterprise search provides a migration wizard that you can use to migrate collections and rule-based taxonomies from IBM WebSphere Portal to enterprise search.

In enterprise search, a taxonomy is called a *category tree*. After you migrate a taxonomy, you use the enterprise search administration console to edit the category tree and category rules. After you migrate a collection, you use the administration console to administer the collection.

To migrate taxonomies and collections, you run the migration wizard on the enterprise search index server.

Migrating a collection from WebSphere Portal

To migrate collections and rule-based taxonomies 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 taxonomies and collections, migrate the taxonomy files before you use this procedure to migrate collections. This approach ensures that your migrated categorization rules work with your migrated collections.

Procedure

To migrate a collection (and optionally migrate the taxonomy) 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 OmniFind Enterprise Edition was installed.
5. Change to the enterprise search installation directory:

```
UNIX: cd $ES_INSTALL_ROOT/bin  
Windows: cd %ES_INSTALL_ROOT%\bin
```

6. To migrate collections with security enabled, enter the following command to start the migration wizard, then click **Next**.

```
UNIX: ./eswpsmigrate.sh  
Windows: eswpsmigrate.bat
```

7. To disable collection-level security for the collections that you migrate, enter the following command to start the migration wizard, then click **Next**.

UNIX: `./eswpsmigrate.sh disable.security`

Windows: `eswpsmigrate.bat disable.security`

8. Select **Migrate the search settings from the Portal Search Engine in WebSphere Portal**, then click **Next**.
9. 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.
10. 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 the number of links to Web documents from other Web documents.
 - The type of categorization that you want to use for this collection. If you specify none, no taxonomy information is migrated to enterprise search. If you select rule-based categories, the taxonomy is migrated to enterprise search along with the collection.

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).

11. Start the crawling, parsing, and indexing processes for the migrated collection from the enterprise search administration console.
12. After you determine that the migrated collection is searchable in enterprise search, delete the original collection in the Portal Search Engine.
13. 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.

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 URLs
- The number of parallel crawling processes
- The crawling depth
- The timeout (in seconds) for retrieving a document
- The default character set
- Rules for crawling Web sites (include or exclude)

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 (rule-based or none)
- Whether to use the search cache and how many query responses 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 build delta indexes
- A schedule to build the main 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.

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 9. Enterprise search **esadmin** commands

Command	Description
<code>esadmin system startall</code>	Starts the enterprise search components on all enterprise search servers, including the Web server, ESSearchServer application, and information center on the search servers; crawler sessions on the crawler server; and index sessions on the index server. Starts the common communication layer (CCL) on the local server only. To recycle the CCL, you must manually stop and restart the CCL on each remote enterprise search server. Sample command: <code>esadmin system startall</code>
<code>esadmin system stopall</code>	Stops the enterprise search components on all enterprise search servers, including the information center, ESSearchServer application, and Web server on the search servers; crawler sessions on the crawler server; and index sessions on the index server. Stops the CCL on the local server only. To recycle the CCL, you must manually stop and restart the CCL on each remote enterprise search server. Sample command: <code>esadmin system stopall</code>
<code>esadmin system checkall</code>	Checks the status of all enterprise search components on all enterprise search servers. Sample command: <code>esadmin system checkall</code>

Table 9. Enterprise search **esadmin** commands (continued)

Command	Description
<code>esadmin crawler_session_id start</code>	<p>Starts a crawler session. This command does not start any crawling activity.</p> <p>Sample command: <code>esadmin col1.WEB1.esadmin start</code></p> <p>Sample messages and return codes: FFQC5310I WEBCrawler1 (sid: col1.WEB1.esadmin) is not running. FFQC5314I Result: 0</p>
<code>esadmin crawler_session_id startCrawl</code>	<p>Starts crawling.</p> <p>Sample command: <code>esadmin col3.DB21.esadmin startCrawl</code></p> <p>Sample messages and return codes: FFQC5303I DB2Crawler1 (sid: col3.DB21.esadmin) is already running. PID: 23650 FFQC5314I Result: 0</p>
<code>esadmin crawler_session_id pause</code>	<p>Pauses crawling.</p> <p>Sample command: <code>esadmin col3.DB21.esadmin pause</code></p> <p>Sample messages and return codes: FFQC5303I DB2Crawler1 (sid: col3.DB21.esadmin) is already running. PID: 23650 FFQC5314I Result: 0</p>
<code>esadmin crawler_session_id resume</code>	<p>Resumes crawling.</p> <p>Sample command: <code>esadmin col3.DB21.esadmin resume</code></p> <p>Sample messages and return codes: FFQC5303I DB2Crawler1 (sid: col3.DB21.esadmin) is already running. PID: 23650 FFQC5314I Result: 0</p>
<code>esadmin crawler_session_id stopCrawl</code>	<p>Stops crawling.</p> <p>Sample command: <code>esadmin col3.DB21.esadmin stopCrawl</code></p> <p>Sample messages and return codes: FFQC5303I DB2Crawler1 (sid: col3.DB21.esadmin) is already running. PID: 23650 FFQC5314I Result: 0</p>
<code>esadmin crawler_session_id stop</code>	<p>Stops a crawler session.</p> <p>Sample command: <code>esadmin col3.DB21.esadmin stop</code></p> <p>Sample messages and return codes: FFQC5303I DB2Crawler1 (sid: col3.DB21.esadmin) is already running. PID: 23650 FFQC5314I Result: 0</p>

Table 9. Enterprise search **esadmin** commands (continued)

Command	Description
esadmin <i>crawler_session_id</i> getCrawlerStatus	<p>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.</p> <p>Example for a Web crawler:</p> <pre>esadmin col1.WEB1.esadmin getCrawlerStatus</pre> <p>Possible return codes and messages for a Web crawler:</p> <pre>FFQC5303I WebCrawler1 (sid: col1.WEB1.esadmin) is already running. PID: 23650</pre> <p>Example for a non-Web crawler:</p> <pre>esadmin col3.DB21.esadmin getCrawlerStatus</pre> <p>Possible return codes and messages for a non-Web crawler:</p> <pre>FFQC5303I db2crawler (sid: db2col.DB2_96945) is already running. PID: 5936</pre> <p>For more information about returned status messages, see “Detailed information for status commands” on page 356.</p>
<pre>esadmin <i>dscrawler_session_id</i> getCrawlSpaceStatus esadmin <i>web_crawler_session_id</i> getCrawlStatus -selections <i>value</i></pre>	<p>Gets general crawl space status for any crawler other than the Web crawler.</p> <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 356.</p>

Table 9. Enterprise search **esadmin** commands (continued)

Command	Description
<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 356.</p>
<pre>esadmin monitor getCollectionParserMonitorStatus -cid collection_ID</pre>	<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 356.</p>
<pre>esadmin startMain -cid collection_id</pre>	<p>Starts the main index build.</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>
<pre>esadmin startDelta -cid collection_id</pre>	<p>Starts a delta index build.</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 9. Enterprise search **esadmin** commands (continued)

Command	Description
<pre>esadmin controller startIndexBuild -cid <i>collection_id</i> -buildType <i>type</i> -detectChanges</pre>	<p>Start a main or delta index build and specify that the build should proceed only if no changes that need to be applied to the index are detected.</p> <p>Sample command:</p> <pre>esadmin controller startIndexBuild -cid col_1 -buildType main -detectChanges</pre>
<pre>esadmin monitor getCollectionIndexMonitorStatus -cid <i>collection_id</i> -buildType [main delta] -numrecords <i>lastNrecords</i></pre>	<p>Gets the status of a main or delta index build. The option <code>numrecords</code> shows the last <i>N</i> index build status records. If <code>numrecords</code> 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 356.</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 9. Enterprise search **esadmin** commands (continued)

Command	Description
<pre>esadmin monitor getCollectionSearchMonitorStatus -cid collection_id esadmin searchmanager_session_id getStatus -cid collection_id</pre>	<p>Gets the status of the search server.</p> <p>Sample command:</p> <pre>esadmin monitor getCollectionSearchMonitorStatus -cid coll</pre> <p>Sample messages and return codes:</p> <pre>FFQC5303I Monitor (node1) (sid: monitor) is already running. PID: 12649</pre> <p>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.</p> <p>Sample command:</p> <pre>esadmin searchmanager.node1 getStatus -cid coll</pre> <p>Sample messages and return codes:</p> <pre>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</pre> <p>For more information about returned status messages, see “Detailed information for status commands.”</p>

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 351 provides possible returned information from each esadmin command. 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 10. 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 10. 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 the **getCrawlerStatus** command for non-Web crawlers:

```
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 11. Crawler status information for the NNTP, DB2, JDBC database, and Notes crawlers

Element and attribute name	NNTP crawler	DB2 and JDBC database crawlers	Notes 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 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 12. Crawler status information for the Exchange Server, DB2 Content Manager, and Content Edition crawlers

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 13. Crawler status information for the QuickPlace, Domino Document Manager, UNIX file system, and Windows file system crawlers

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 13. Crawler status information for the QuickPlace, Domino Document Manager, UNIX file system, and Windows file system crawlers (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.

Table 14. Crawler status information for the WebSphere Portal and Web Content Management crawlers

Element and attribute name	WebSphere Portal crawler	Web Content Management crawler
Status	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
NumberOfServers	The number of servers in the crawl space.	The number of sites in the crawl space.
NumberOfCompletedServers	The number of crawled servers.	The number of crawled sites.
NumberOfTargets	The number of servers in the crawl space.	The number of sites in the crawl space.
NumberOfCompletedTargets	The number of crawled servers.	The number of crawled sites.
NumberOfCompletedRecords	The number of crawled documents.	The number of crawled documents.
RunningThreads	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 15. 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.

Table 15. Selection mask values for the Web crawler crawl space status command (continued)

Mask bit	Selects
128	Number of URLs that are uncrawled.
256	Number of URLs that are overdue.
512	HTTP status 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"/>
    <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 16. Crawl space status information for the Web crawler

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.

Table 16. Crawl space status information for the Web crawler (continued)

Element	Attribute	Description
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.
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 the **getCrawlSpaceStatus** command for non-Web crawlers:

```
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 17. Crawl space status information for the NNTP, DB2, JDBC database, and Notes crawlers

Element and attribute name	NNTP crawler	DB2 and JDBC database crawlers	Notes crawler
Server@Name	News server name	Database name	Database name or directory name

Table 17. Crawl space status information for the NNTP, DB2, JDBC database, and Notes crawlers (continued)

Element and attribute name	NNTP crawler	DB2 and JDBC database crawlers	Notes crawler
Server/Status	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 	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 news groups in the crawl space.	The number of databases in the crawl space.	The number of views and folders or directories in the crawl space.
Server/NumberOfCompletedTargets	The number of crawled news groups.	The number of crawled tables.	The number of crawled views and folders or directories.
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 17. Crawl space status information for the NNTP, DB2, JDBC database, and Notes crawlers (continued)

Element and attribute name	NNTP crawler	DB2 and JDBC database crawlers	Notes crawler
Server/AggregationLevel	0: The crawler crawls documents with normal mode.	0: The crawler crawls documents with normal 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.) 1: The Notes crawler crawls documents with directory mode.

Table 18. Crawl space status information for the Exchange Server, DB2 Content Manager, and Content Edition crawlers

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, 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 	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.

Table 18. Crawl space status information for the Exchange Server, DB2 Content Manager, and Content Edition crawlers (continued)

Element and attribute name	Exchange Server crawler	DB2 Content Manager crawler	Content Edition crawler
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: The crawler crawls documents with normal mode.	0: The crawler crawls documents with normal mode.	0: The crawler crawls documents with normal mode.

Table 19. Crawl space status information for the QuickPlace, Domino Document Manager, UNIX file system, and Windows file system crawlers

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, 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 	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

Table 19. Crawl space status information for the QuickPlace, Domino Document Manager, UNIX file system, and Windows file system crawlers (continued)

Element and attribute name	QuickPlace crawler	Domino Document Manager crawler	UNIX and Windows file system crawlers
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.
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: The crawler crawls documents with normal mode.	0: The crawler crawls documents with normal mode.	0: The crawler crawls documents with normal mode.

Table 20. Crawl space status information for the WebSphere Portal and Web Content Management crawlers

Element and attribute name	WebSphere Portal crawler	Web Content Management crawler
Server@Name	WebSphere Portal server	Web Content Management search seed URL
Server/Status	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 20. Crawl space status information for the WebSphere Portal and Web Content Management crawlers (continued)

Element and attribute name	WebSphere Portal crawler	Web Content Management 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
Server/NumberOfTargets	The number of servers in the crawl space.	The number of sites in the crawl space.
Server/NumberOfCompletedTargets	The number of crawled servers.	The number of crawled sites.
Server/NumberOfErrors	The number of errors.	The number of errors.
Server/StartTime	The start time if applicable.	The start time if applicable.
Server/EndTime	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
Server/ScheduleTime	Schedule time if applicable.	Schedule time if applicable.
Server/TotalTime	The total time if applicable.	The total time if applicable.
Server/AggregationLevel	0: The crawler crawls documents with normal mode.	0: The crawler crawls documents with normal 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 21. 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.

Table 21. Selection mask values for the Web crawler detailed crawl space status command (continued)

Mask bit	Selects
512	HTTP status 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 22. Detailed crawl space status information for the Web crawler

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.

Table 22. Detailed crawl space status information for the Web crawler (continued)

Element	Attributes	Description
NumURLsOverdueBy	Threshold, Value: Integer (positive or negative) The value represents the number of URLs that are eligible to be recrawled. The threshold specifies the amount of time that the URLs have been waiting to be recrawled. The threshold is measured as the number of seconds offset from the current time. If the threshold is negative, it means that a recrawl of the URLs is overdue. If the threshold is positive, it means that a recrawl of the URLs is due to occur.	The number of URLs that became eligible to be recrawled at least some number of seconds ago or that are becoming eligible to be recrawled in the next so many seconds.
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 status code or another internal code.	How many times a particular HTTP status 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 **getCrawlSpaceStatusDetail** 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 23. Detailed crawl space status information for the NNTP, DB2, JDBC database, and Notes crawlers

Element and attribute name	NNTP crawler	DB2 and JDBC database crawlers	Notes crawler
Target@Name	News group name	Table name	View or folder name
Target@CrawlType	Not applicable.	0,1 (DB2); 0 (JDBC database) <ul style="list-style-type: none"> 0: Active crawl (Normal) 1: Passive crawl (DB2 Event Publishing) 	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	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.

Table 23. Detailed crawl space status information for the NNTP, DB2, JDBC database, and Notes crawlers (continued)

Element and attribute name	NNTP crawler	DB2 and JDBC database crawlers	Notes crawler
Target/AggregationLevel	0: The crawler crawls documents with normal mode.	0: The crawler crawls documents with normal mode.	0, 1: <ul style="list-style-type: none"> 0: The crawler crawls documents with normal mode. The 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.
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 24. Detailed crawl space status information for the Exchange Server, DB2 Content Manager, and Content Edition crawlers

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.

Table 24. Detailed crawl space status information for the Exchange Server, DB2 Content Manager, and Content Edition crawlers (continued)

Element and attribute name	Exchange Server crawler	DB2 Content Manager crawler	Content Edition crawler
Target/NumberOf DeletedRecords	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 crawler crawls documents with normal mode.	0: The crawler crawls documents with normal mode.	0: The crawler crawls documents with normal mode.
Target/LastUpdatedTime	Not applicable.	Not applicable.	Not applicable.
Target/LastResetTime	Not applicable.	Not applicable.	Not applicable.

Table 25. Detailed crawl space status information for the QuickPlace, Domino Document Manager, UNIX file system, and Windows file system crawlers

Element and attribute name	QuickPlace crawler	Domino Document Manager crawler	UNIX and Windows file system crawlers
Target@Name	PPlace 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/NumberOf Records	Not applicable.	Not applicable.	Not applicable.
Target/NumberOf CompletedRecords	The number of crawled documents.	The number of crawled documents.	The number of crawled files.
Target/NumberOf InsertedRecords	The number of inserted records.	The number of inserted records.	The number of inserted records.
Target/NumberOf UpdatedRecords	The number of updated records.	The number of updated records.	The number of updated records.

Table 25. Detailed crawl space status information for the QuickPlace, Domino Document Manager, UNIX file system, and Windows file system crawlers (continued)

Element and attribute name	QuickPlace crawler	Domino Document Manager crawler	UNIX and Windows file system crawlers
Target/NumberOf DeletedRecords	The number of deleted records.	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 crawler crawls documents with normal mode.	0: The crawler crawls documents with normal mode.	0: The crawler crawls documents with normal mode.
Target/LastUpdatedTime	Not applicable.	Not applicable.	Not applicable.
Target/LastResetTime	Not applicable.	Not applicable.	Not applicable.

Table 26. Detailed crawl space status information for the WebSphere Portal and Web Content Management crawlers

Element and attribute name	WebSphere Portal crawler	Web Content Management crawler
Target@Name	WebSphere Portal server name	The search seed URL that represents the site
Target@CrawlType	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
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
Target/NumberOf Records	Not applicable.	Not applicable.
Target/NumberOf CompletedRecords	The total number of crawled records.	The total number of crawled records.
Target/NumberOf InsertedRecords	The number of inserted records.	The number of inserted records.
Target/NumberOf UpdatedRecords	The number of updated records.	The number of updated records.
Target/NumberOf DeletedRecords	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.
Target/EndTime	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.

Table 26. Detailed crawl space status information for the WebSphere Portal and Web Content Management crawlers (continued)

Element and attribute name	WebSphere Portal crawler	Web Content Management crawler
Target/AggregationLevel	0: The crawler crawls documents with normal mode.	0: The crawler crawls documents with normal mode.
Target/LastUpdatedTime	Not applicable.	Not applicable.
Target/LastResetTime	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>
    <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>
    <NumberOfDocsAlreadyParsed>29</NumberOfDocsAlreadyParsed>
  </CrawlerStatus>
</Monitor>
```

The following table describes the XML elements for information that is returned by the parser status command:

Table 27. Elements for the parser status command

Element	Description
Status	<ul style="list-style-type: none"> • 0: The parser session for this collection is stopped. • 1: The parser session for this collection is running.

Table 27. Elements for the parser status command (continued)

Element	Description
State	<p>The possible states are: Initializing, Idle, Restart, Parsing, Stopped, Paused, Resuming, NoParserServiceIsAvailable.</p> <p>A status of Initializing means that the parser is starting and initializing its state.</p> <p>A status of Idle indicates that the parser is sleeping for <i>N</i> minutes waiting for more documents to arrive from the crawlers in this collection. The default sleep time is 300 seconds.</p> <p>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.</p> <p>A status of Parsing indicates that the parser is processing documents.</p> <p>A status of Paused indicates that the parser was paused by the index build session for this collection.</p> <p>A status of Resuming indicates that the parser was changed from a Paused state to a Parsing state by the index build session for this collection.</p> <p>A status of NoParserServiceIsAvailable indicates that there are no parsing/tokenizing JVMs available to process the documents for this collection. This status means that all parsing/tokenizing JVMs are being used by other collections.</p>
NumberOfDocsToBeIndexed	The number of documents in the store for this collection. This number also includes documents that are marked to be deleted 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.
Crawlerid	The ID created for this crawler by the system.
Type	The type of crawler (Web, NNTP, DB2, and so on.)
ParserStatus	<ul style="list-style-type: none"> • 0: Documents from this crawler are not being parsed (the parser session is stopped). • 1: Documents from this crawler are being parsed (the parser session is running).
NubmerOfDocsAlreadyParsed	The number of documents from this crawler that were parsed.

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:

```
<?xml version="1.0"?>
<Monitor Type="MainIndexHistory" Count="1">
  <IndexStatus Id="1">
    <StartTime>1131987633901<StartTime>
    <Progress>0</Progress>
```

```

<CurrentPhase>0</CurrentPhase>
<TotalPhase>3</TotalPhase>
<IndexCopyTime>49822</IndexCopyTime>
<CurrentServer>0</CurrentServer>
<TotalServer>0</TotalServer>
<IndexBuildTime>46158</IndexBuildTime>
<Status>0</Status>
<JobID>1131987633899</JobID>
<MessagesAvailable>false</MessagesAvailable>
<StopTime>1131987734199</StopTime>
<TotalTime>100298</TotalTime>
<NumberOfDocuments>43</NumberOfDocuments>
</IndexStatus>
<CurrentIndexWildcardSupport/>
<NextIndexWildcardSupport Type="None" Limit="0"/>
<ScheduleStatus>
  <Status>1</Status>
  <ScheduledTime Enabled="false"></ScheduledTime>
</ScheduleStatus>
</Monitor>

```

The following table describes each XML element for information that is returned by the index build status command:

Table 28. Elements for the index build status command

Element	Description
IndexStatusId	The index status ID.
StartTime	The time in seconds since 1970 when this index build was started. To compute the present time that this time represents, use the formula January 1, 1970 %2B StartTime. To learn more about epoch time, see http://en.wikipedia.org/wiki/Unix_epoch .
Progress	The percentage of 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	The 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	Total time to copy index to all the search servers.
IndexBuildTime	The total time for all phases of index build
Status	<ul style="list-style-type: none"> • 0: index build and copy • -1: index build request failure • 1: index build, copy, or both are in progress
JobID	A unique ID that is associated with each request for an index build.
MessagesAvailable	A boolean value that indicates whether error messages are available (in case of failure).
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.

Table 28. Elements for the index build status command (continued)

Element	Description
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 time in seconds since 1970 when the next index build for this collection and index type will be run. To compute the present time that this time represents, use the formula January 1, 1970 %2B ScheduledTimeEnabled. To learn more about epoch time, see http://en.wikipedia.org/wiki/Unix_epoch .

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 29. 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 30. Items for the detailed search server status command

Item	Description
CacheHits	The number of results retrieved from the search cache.

Table 30. Items for the detailed search server status command (continued)

Item	Description
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 31. 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 32. 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
discovery	index server	10649	Started
monitor	index server	10682	Started


Table 32. Examples of sessions names, origin servers, session IDs, and session states (continued)


Session	Server where the session is running	Session ID	Session state
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

“Monitoring enterprise search activity” on page 285

“Backing up and restoring an enterprise search system” on page 315

 Messages for enterprise search

 Messages for enterprise search

Related tasks

“Monitoring crawlers” on page 288

“Starting an enterprise search system” on page 279

“Stopping an enterprise search system” on page 281

“Administering the search servers in stand-alone mode” on page 283

Case sensitivity in enterprise search

The components of enterprise search, such as query syntax, quick links, field names, and so on, treat case differently.

Query syntax

Search is case-insensitive except in the following cases:

XML element names and attribute names

Case-sensitive. Terms and attribute values are case-insensitive even in XML queries. For example, in this document:

```
<book>
  <Author>
    <Name>Ferdinand</Name>
    <Contact Type="eMail">ferdi@nand.org</Contact Type>
    <Contact Type="Phone">+1 408 876 4242</Contact Type>
  </Author>
</book>
```

The following queries will not return the document:

- @xmlns::'author[Name ftcontains ("Ferdinand")]'
- @xmlns::'//contact[@type="eMail"]'
- @xmlf2::'<author><name>Ferdinand</name></author>'
- @xmlf2::'<CONTACT TYPE="email">ferdi</contact>'

But the following queries will return the document:

- @xmlns::'Author[Name ftcontains ("ferdinand")]'
- @xmlns::'//Contact[@Type="email"]'
- @xmlf2::'<Author><Name>ferdinand</Author><Name>'
- @xmlf2::'<Contact Type="email">ferdi</Contact>'

Access control lists (ACLs)

Case-sensitive.

URLs in docid: and samegroupas: terms

Case-sensitive. However, URL parts in site: or url: terms are case-insensitive. For example, in a document with URL `http://www.here.com/HR/`:

- The query `docid:http://www.here.com/hr` will not return the document.
- The queries `url:hr` and `url:HERE` will return the document.
- The query `site:HERE.com` will return the document.

Category IDs and taxonomy IDs

Case-insensitive. For example, in the query `taxonomy_id::category_id`, the case of both `taxonomy_id` and `category_id` does not matter. The query `RuleBased::c42` matches the category `c42` in the rule-based taxonomy, and also matches `rulebased::C42`. You cannot search by category name, but you can search by category ID.

Scopes

Case-insensitive. For example, both `Scope:RESEARCH` and `scope::research` will return documents from the scope named `Research`.

Wildcard terms

Case-insensitive. For example, the term Fer*n*d is equivalent to fer*n*d.

Field names

Case-insensitive. For example, the queries Title:Expenses, TITLE:expenses, and title:expenses are all equivalent. All field names are case-insensitive, even if they originate from an XML mapping file. However, external sources might treat field names as case-sensitive according to their own query semantics.

Quick links

Case-insensitive.

Rule-based categorizer

URL rules are case-sensitive, but document content rules are case-insensitive.

Collapsed URIs and URI pattern-based boost definitions

URIs are case-sensitive, but collapsed URI group names are case-insensitive. You cannot define two collapsed URI group names that differ only in case.

Field names

Case-insensitive. This rule applies to field names in queries, XML mappings, boost class definitions, and all other administrative interfaces where field names are specified. If you configure an uppercase or mixed case field name in the interface, it might be made lowercase by the system so that it will appear in lowercase the next time you view the configuration. Also, a field name can be interpreted case sensitively by an external source.

Dictionaries

Case-insensitive. This rule applies to synonym dictionaries, stop word dictionaries, spelling suggestion dictionaries, and boost word dictionaries.

Collection names and IDs

Case-sensitive. If you specify a collection name or a collection ID in the administration or search API, it must match exactly the case of the collection with that name. However, even though collection IDs are case-sensitive, you cannot specify two collection IDs that differ only in case. This same restriction applies to crawler and data source names and IDs.

Search applications

Case-sensitive. You cannot specify two application names or IDs that differ only in case.

Enterprise search documentation

You can read the OmniFind Enterprise Edition documentation in PDF or HTML format.

The OmniFind Enterprise Edition installation program automatically installs the information center, which includes HTML versions of the documentation for enterprise search. For a multiple server installation, the information center is installed on both search servers. If you do not install the information center, when you click help, the information center opens on an IBM Web site.

To see installed versions of the PDF documents, go to `ES_INSTALL_ROOT/docs/locale/pdf`. For example, to find documents in English, go to `ES_INSTALL_ROOT/docs/en_US/pdf`.

To access the PDF versions of the documentation in all available languages, see the OmniFind Enterprise Edition, Version 8.5 documentation site.

You can also access product downloads, fix packs, technotes, and the information center from the OmniFind Enterprise Edition Support site.

The following table shows the available documentation, file names, and locations.

Table 33. Documentation for enterprise search

Title	File name	Location
Information center		http://publib.boulder.ibm.com/infocenter/discover/v8r5/
<i>Installation Guide for Enterprise Search</i>	iiysi.pdf	ES_INSTALL_ROOT/docs/locale/pdf/
<i>Quick Start Guide</i> (This document is also available in hardcopy for English, French, and Japanese.)	OmniFindEE850_qsg_ <i>two-letter</i> locale.pdf	ES_INSTALL_ROOT/docs/locale/pdf/
<i>Administering Enterprise Search</i>	iiysa.pdf	ES_INSTALL_ROOT/docs/locale/pdf/
<i>Programming Guide and API Reference for Enterprise Search</i>	iiysp.pdf	ES_INSTALL_ROOT/docs/en_US/pdf/
<i>Troubleshooting Guide and Messages Reference</i>	iiysm.pdf	ES_INSTALL_ROOT/docs/locale/pdf/
<i>Text Analysis Integration</i>	iiyst.pdf	ES_INSTALL_ROOT/docs/locale/pdf/

Accessibility features

Accessibility features help users who have a disability, such as restricted mobility or limited vision, to use information technology products successfully.

IBM strives to provide products with usable access for everyone, regardless of age or ability.

Accessibility features

The following list includes the major accessibility features in OmniFind Enterprise Edition:

- Keyboard-only operation
- Interfaces that are commonly used by screen readers

The OmniFind Enterprise Edition Information Center, and its related publications, are accessibility-enabled. The accessibility features of the information center are described at http://publib.boulder.ibm.com/infocenter/discover/v8r5m0/topic/com.ibm.classify.nav.doc/dochome/accessibility_info.htm.

Keyboard navigation

This product uses standard Microsoft Windows navigation keys.

You can also use the following keyboard shortcuts to navigate and advance through the OmniFind Enterprise Edition installation program.

Table 34. 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

Interface information

The user interfaces for the administration console, sample search application, and search application customizer 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.

Related accessibility information

You can view the publications for OmniFind Enterprise Edition in Adobe Portable Document Format (PDF) using the Adobe Acrobat Reader. The PDFs are provided on a CD that is packaged with the product, or you can access them at

<http://www.ibm.com/support/docview.wss?rs=63&uid=swg27010938>.

IBM and accessibility

See the IBM Human Ability and Accessibility Center for more information about the commitment that IBM has to accessibility.

Glossary of terms for enterprise search

This glossary defines terms that are used in the enterprise search interfaces and documentation.

access control list (ACL)

In computer security, a list associated with an object that identifies all the subjects that can access the object and their access rights.

administrative role

A classification of a user that prescribes access to a user.

analysis engine

See text analysis engine.

analysis results

The information that is produced by annotators. Analysis results are written to a data structure called a common analysis structure. Analysis results produced by the custom text analysis engines (annotators) can be made available for search by inclusion in the enterprise search index.

annotation

Information about a span of text. For example, an annotation could indicate that a span of text represents a company name. In the Unstructured Information Management Architecture (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

An object that contains specifications that can influence the relative rank of a document in the search results.

boost word

A word that can influence the relative 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 tree

A hierarchy of categories.

certificate

In computer security, 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 and is digitally signed by that authority.

certificate authority

A trusted third-party organization or company that issues the digital

certificates used to create digital signatures and public-private key pairs. The certificate authority guarantees the identity of the individuals who are granted the unique certificate.

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 (CAS)

A structure that stores the content and metadata of a document, and all analysis results that are produced by a text analysis engine. All data exchange during document analysis is handled by using the common analysis structure.

common analysis structure consumer (CAS consumer)

A consumer that does the final processing on the analysis results that are stored in the common analysis structure. For example, a consumer indexes the contents of the common analysis structure in a search engine or it populates a relational database with specific analysis results.

common communication layer (CCL)

The communication infrastructure that unites the various components (controller, parser, crawler, index server) of OmniFind Enterprise Edition.

concept extraction

A text analysis 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 Uniform Resource Locators (URLs), database names, file system paths, domain names, and IP addresses) 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, 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. For example, the sign-on information (user ID and password) for a user are credentials that allow the user to access an account.

custom text analysis engine

A text analysis engine that is created by using the Unstructured Information Management Architecture (UIMA) software development kit (SDK) and can be added to the set of standard enterprise search text analysis engines (also known as enterprise search base annotators). See also text analysis engine.

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.

data store

A data structure where documents are kept in their parsed form.

delta index build

In an enterprise search system, the process of adding new information to an existing index. Contrast with main index build.

dequeue

To remove items from a queue.

diacritic

A mark indicating a change in the phonetic value of a character or a combination of characters.

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 put a message or item in a queue.

enterprise search administrator

An administrative role that enables a user to administer the entire enterprise search system.

enterprise search base annotators

A set of standard text analysis engines used in enterprise search for default document analysis processing.

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 OmniFind Enterprise 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 Unstructured Information Management Architecture (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 An area into which a particular category of data or control information is entered.

fielded search

A query that is restricted to a particular field.

free-form text

Unstructured text consisting of words or sentences.

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 a search to 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

A set of enterprise search APIs that control access to secure data and

enable users to search a collection without being required to specify a user ID and password for each repository in the collection.

index See full-text index.

index queue

A list of requests for main and delta index builds to be processed.

information extraction

A type of concept extraction that automatically recognizes significant vocabulary items, such as names, terms, and expressions, in text documents.

IP address

A unique address for a device or logical unit on a network that uses the IP standard.

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.

key database file

See key ring. key ring.

key ring

In computer security, a file that contains public keys, private keys, trusted roots, and certificates. See also keystore file.

keystore file

A key ring that contains both public keys that are stored as signer certificates and private keys that are stored in personal certificates.

language identification

In enterprise search, a search function that determines the language of a document.

lemma

The base form of a word. Lemmas are significant in highly inflected languages such as Czech.

lemmatization

A process that identifies the root form and different grammatical forms of a word. For example, a search for mouse also finds documents that contain the word mice, and a search for go also finds documents that contain going, gone, or went.

lexical affinity

The relationship of search words in a document that are close to each other in meaning. 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 they appear as one character. For example, *ff* and *ffi* are characters that can be presented as ligatures.

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 (DAP). For example, LDAP can be used to locate people, organizations, and other resources in an Internet or intranet directory.

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

In an enterprise search application, a client object created by the search and index APIs that enables users to search a set of heterogeneous collections and obtain a unified set of search results.

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.

main index build

In enterprise search, the process of building the entire index. Contrast with delta index build.

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.

monitor

An enterprise search user who has the authority to observe collection-level processes.

newline character

A control character that causes the print or display position to move down one line.

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 that page.

no-index directive

A directive in a Web page that instruct robots (such as the Web crawler) to not include the contents of that page in the index.

Notes remote procedure call (NRPC)

A communication mechanism 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.

parser driver

In enterprise search, a service that feeds the parser service with documents. There is one parser driver for each collection. A collection's parser driver service corresponds to the collection's parser in the enterprise search administration console.

parser service

The enterprise search service that handles all document parsing and text analysis processing across document collections. At least one parser service is running at all times.

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 type of ranking that raises a document's existing ranking based on the document's popularity.

Portal Document Manager (PDM)

Allows users to have one central document repository for team collaboration. Administrators have the ability to effectively manage their documents and can control the way users interact with information.

processing engine archive

A .pear zip archive file that includes a Unstructured Information

Management Architecture (UIMA) analysis engine and all of the resources required to use it for custom analysis in enterprise search.

proximity search

A text search that returns a result when two or more matching terms occur within a certain distance from each other, such as in the same sentence or paragraph.

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 Uniform Resource Identifier (URI) and keywords or phrases.

ranking

The assignment of an interger 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.

raw data store

A data structure where crawled documents are stored before they are sent to the parser. Crawlers write to the raw data store, and the parser reads from the raw data store. When documents have been parsed, they are removed from the raw data store. Not to be confused with data store.

regular expression annotator

A software component that detects entities or units of information in a text document, such as product numbers, based on regular expressions that describe the exact patterns that are searched in the document text. If one of the regular expressions matches parts of the document text, the regular expression annotator creates the corresponding annotations that cover the match or part of it. These annotated expressions are then stored, either in the enterprise search index using an index mapping file, or a JDBC-capable database using a database mapping file.

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 Uniform Resource Identifier (URI) pattern, with specific categories.

search application

In enterprise search, a program that processes queries, searches the index, returns the search results, and retrieves the source documents.

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. With SSL, client/server applications can communicate in a way that is designed to prevent eavesdropping, tampering, and message forgery.

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 list page

In WebSphere Portal, an XML page that contains links to the pages that are available on a portal. Crawlers use the seed list to identify the documents to crawl. The seed list page also contains metadata that is stored with the crawled documents in the enterprise search index.

start Uniform Resource Locator (URL)

The starting point for a crawl.

segmentation

The division of text into distinct lexical units. Nondictionary-based processing includes white space and n-gram segmentation, while dictionary-based support includes word, sentence, and paragraph segmentation, and lemmatization.

semantic search

A type of keyword search that incorporates linguistic and contextual analysis. See also text analysis.

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.

shingle

A string of consecutive tokens (words) that are taken from a sentence. For example, from "This is a very short sentence.", the 3-word shingles (or trigrams) are:

This is a
is a very
a very short
very short sentence

Shingles can be used in statistical linguistics. For example, if two different texts have a lot of common shingles, the texts are probably related somehow.

soft error page

A type of Web page that provides information about why the requested Web page cannot be returned. For example, instead of returning a simple status code, the HTTP server can return a page that explains the status code in detail.

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 non-redundant 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. See also semantic search.

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.

text segmentation

See segmentation.

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.

tokenization

The process of parsing input into tokens.

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.

type system

The type system defines the types of objects (feature structures) that may be discovered by a text analysis engine in a document. The type system defines all possible feature structures in terms of types and features. You can define any number of different types in a type system. A type system is domain and application specific.

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)

The unique address of an information resource that is accessible in a network such as the Internet. The URL includes the abbreviated name of the protocol used to access the information resource and the information used by the protocol to locate the information 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 type of crawler 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 is designed to uniquely identify or address parts of source XML data, for use with XML-related technologies, such as XSLT, XQuery, and XML parsers. XPath is a World Wide Web Consortium standard.

Notices and trademarks

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant 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 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 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 which 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.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

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.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

Portions of this product are:

- Oracle® Outside In Content Access, Copyright © 1992, 2008, Oracle. All rights reserved.
- IBM XSLT Processor Licensed Materials - Property of IBM © Copyright IBM Corp., 1999-2008. All Rights Reserved.

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:

Adobe, Acrobat, Portable Document Format (PDF), PostScript, and all Adobe-based trademarks are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

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

Linux is a registered trademark of Linus Torvalds 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.

UNIX is a registered trademark of The Open Group in the United States and other countries.

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

Index

A

- access controls
 - current user validation 253
 - description 247
 - disabling for a collection 277
 - document-level security 252
 - identity management 254, 257
 - requirements for Lotus Domino 269, 270
 - requirements for Windows file systems 273
 - single sign-on security 258
- accessibility features for this product 385
- active Web sites, monitoring 289, 290
- address rules for Web crawlers 88
- administration console
 - description 8
 - interface 15
 - logging in 18
 - task summary 15
- administrative roles
 - collection administrator 248, 249
 - configuring 249
 - description 248
 - enterprise search administrator 248, 249
 - monitor 248, 249
 - operator 248, 249
- administrator password
 - changing on a single server 19
 - changing on multiple servers 20
- AdminLinkBarInclude.jsp file 329
- AIX operating system
 - Content Edition crawler
 - configuration 44
 - DB2 Content Manager crawler
 - configuration 56
 - DB2 crawler configuration 49
 - Domino Document Manager crawler
 - configuration 73
 - event publishing configuration 49
 - Notes crawler configuration 73
 - QuickPlace crawler configuration 73
- alerts
 - collection-level 285, 306
 - description 305
 - documents crawled 306
 - documents indexed 306
 - e-mail options 306, 307
 - free space on servers 307
 - index limits 285
 - receiving e-mail for 310
 - search response times 306
 - SMTP server configuration 309
 - system-level 307
- anchor text analysis
 - collection security 260
 - description 245
 - global analysis 260
 - indexing documents 261

- annotators 134
- APIs
 - description 10
 - Search and Index 10, 211
- application IDs 250
- archive files
 - crawling 112
 - supported formats 112
 - URI formats 113
- ASCII parser 149
- authentication
 - description 247
 - disabling for enterprise applications 275
- authorization, description 247
- automatic detection
 - code pages 161
 - languages 160

B

- backing up enterprise search 315, 316
- backup scripts
 - description 315
 - running 316
- banner_searchControl.jspf file 336
- banner.jspf file 336
- bar charts
 - Java classes for top results 234
 - top results 234
- boost classes
 - configuration 206, 207
 - default values 207
 - description 204
 - duplicate document detection 204
 - high recall queries 204, 207
 - low recall queries 204, 207
 - mapping fields to 206
- boost factors
 - boost class configuration 204, 207
 - for boost classes 206, 207
 - for boost word dictionaries 200
 - for URI patterns 203
- boost word dictionaries
 - adding to the system 202
 - associating with a collection 202
 - description 200
 - redeploying 191
- bos.iocp.rte module 77
- building indexes 165

C

- categories
 - categorization type 127
 - category trees 126
 - creating 127
 - description 124
 - migrating from WebSphere Portal 347

- categories (*continued*)
 - nesting subcategories 126
 - rule-based 124, 127
 - searching 124
 - URI formats 113
- categorization type
 - rule-based 124
 - selecting 32, 127
- category rules
 - configuring 127
 - document content 124, 127
 - URI patterns 124, 127
- category trees
 - description 126
 - migrating from WebSphere Portal 347
- ccl.properties file 25
- CCLServer_date.log file 23
- Chinese
 - n-gram segmentation 162
 - removing new line characters 162
- cloning
 - crawlers 37
 - search applications 233
- clusters
 - WebSphere Portal 340
- code pages
 - automatic detection 161
 - supported 161
- collapsed search results
 - configuring 182
 - description 181
 - security restrictions 278
- collapsed URIs
 - configuring 182
 - description 181
 - security restrictions 278
- collection administrator
 - description 248
 - role configuration 249
- collection ID, syntax rules 32
- collection IDs 36
- Collection wizard 31
- collection-level security
 - anchor text analysis 260
 - application IDs 250
 - description 245, 249
 - duplicate document detection 249
 - enabling 32
- collections
 - anchor text security 260
 - application ID security 250
 - associating with search applications 212
 - bypassing document-level access controls 277
 - creating with Collection wizard 31
 - creating with Collections view 32
 - default migration settings 349
 - deleting 35
 - description 3

- collections (*continued*)
 - determining the ID 36
 - draft 31
 - duplicate document detection 175
 - duplicate document security 249
 - editing 34
 - estimating resources 285
 - estimating the size 32
 - federation 31
 - migrating from WebSphere Portal 347
 - MigrationWizard.log file 347
 - monitoring 286
 - parsing 123
 - search servers 185
 - searching 171
 - security 249
 - system status 286
 - ways to create 31
 - Collections view
 - creating collections 32
 - description 15
 - commands, enterprise search 351
 - common analysis structures
 - description 134
 - mapping to relational databases 140
 - mapping to the index 139
 - mapping XML elements to 138
 - complete match search fields, description 171
 - complex text languages 159
 - compound terms, parsing 141
 - concurrent index builds 168
 - config.properties file 264, 265
 - cloning 233
 - customizing 231
 - editing 230
 - property descriptions 214
 - Content Edition crawlers
 - configuration 41
 - direct mode 43
 - server mode 44
 - setting up in Solaris operating environment 44
 - setting up on AIX operating system 44
 - setting up on Linux operating system 44
 - setting up on Windows 45
 - URI formats 113
 - cookies for Web crawling
 - configuring 99
 - description 98
 - format 98
 - cookies.ini file
 - configuring 99
 - description 98
 - format 98
 - Crawl page, description 15
 - crawl rate, monitoring 291
 - crawl space
 - alerts about 306
 - description 4
 - editing 40
 - Web crawler configuration 88
 - crawl.rules file 99
 - crawled document dates
 - configuring for Web crawlers 102
 - crawler history reports
 - creating 292
 - description 289
 - HTTP status code report 292
 - Site report 292
 - crawler plug-ins 110
 - crawler properties
 - description 4
 - editing 39
 - crawler servers
 - starting 279, 288
 - stopping 281, 288
 - crawler types
 - base values for 37
 - combining in a collection 37
 - crawler_rdb_plugin.xml file 65
 - crawlers
 - archive files 112
 - base values for 37
 - combining crawler types 37
 - configuration overview 37
 - Content Edition 41, 43, 44
 - creating 38
 - Data Listener applications 109
 - DB2 46
 - DB2 Content Manager 55
 - default migration settings 349
 - deleting 40
 - description 4
 - document-level security 251
 - Domino Document Manager 59
 - editing crawl spaces 40
 - editing crawler properties 39
 - enabling document-level security 37
 - Exchange Server 61, 269
 - initial values for 38
 - JDBC database 62, 63, 65
 - monitoring 288
 - NNTP 69
 - Notes 70, 72
 - plug-ins 110
 - QuickPlace 78
 - scheduling 37, 41
 - Seed list 81
 - support for external 109
 - system status 288
 - UNIX file system 83
 - URI formats 113
 - Web 84
 - Web Content Management 103, 106
 - WebSphere Portal 105, 106
 - Windows file system 107
 - creating
 - collections 31, 32
 - crawlers 38
 - HTML search fields 133
 - quick links 195
 - rule-based categories 127
 - scopes 180
 - Web crawler reports 292
 - XML search fields 130
 - custom text analysis
 - description 134
 - mapping analysis results to a relational database 140
 - custom text analysis (*continued*)
 - mapping analysis results to the index 139
 - mapping the common analysis structure to a relational database 140
 - mapping the common analysis structure to the index 139
 - mapping XML elements 138
 - text analysis engines 136, 137
 - customizing search applications 230, 231
- ## D
- data flow, enterprise search system 11
 - Data Listener
 - configuring 109
 - monitoring 301
 - restarting 109, 301
 - data source types
 - CA-Datcom databases 46
 - Content Edition repositories 41, 43, 44
 - DB2 Content Manager item types 55
 - DB2 databases 46, 62, 241
 - DB2 for iSeries databases 46
 - DB2 for z/OS 46
 - Domino Document Manager databases 59
 - Exchange Server public folders 61
 - IMS databases 46
 - Informix databases 46
 - JDBC databases 62, 63, 65, 241
 - Lotus Quickr for Domino 78
 - Lotus Quickr for WebSphere Portal 81
 - NNTP news groups 69
 - Notes databases 70, 72
 - Oracle databases 46, 62, 241
 - QuickPlace databases 78
 - relational databases 46
 - Software AG Adabas databases 46
 - SQL Server databases 46, 62
 - support for external 2, 10
 - supported by enterprise search 2
 - Sybase databases 46
 - UNIX file systems 83
 - VSAM databases 46
 - Web Content Management sites 103
 - Web sites 84
 - WebSphere Portal sites 105
 - Windows file systems 107
 - DB2 Content Manager crawlers
 - configuration 55
 - setting up in Solaris operating environment 56
 - setting up on AIX operating system 56
 - setting up on Linux operating system 56
 - setting up on Windows 57
 - URI formats 113
 - DB2 crawlers
 - configuration 46
 - event publishing 46
 - event publishing configuration 49
 - setting up in Solaris 49

- DB2 crawlers (*continued*)
 - setting up on AIX 49
 - setting up on Linux 49
 - setting up on Windows 49
 - URI formats 113
 - WebSphere II Classic Federation 54
 - WebSphere II Event Publisher Edition configuration 50
 - WebSphere MQ configuration 52
 - WebSphere MQ installation on AIX 49
 - WebSphere MQ installation on Linux 49
 - WebSphere MQ installation on Solaris 49
 - WebSphere MQ installation on Windows 49
 - DB2 databases
 - access as external source 241
 - access with DB2 crawlers 46
 - access with JDBC database crawlers 62
 - Default search application 230
 - Default.jsp file 329, 336
 - deleting
 - collections 35
 - crawlers 40
 - indexes from the queue 298
 - delta indexes
 - concurrent builds 168
 - description 6, 165
 - detecting changes 169
 - scheduling 166
 - DIIOp protocol, crawler configuration 76
 - direct mode, Content Edition repositories 43
 - Directory Assistance configuration 272
 - disabling index schedules 167
 - discovery 4
 - distinctRecentQueryCheck parameter 300
 - document content, description 171
 - document importance
 - boost classes 204, 207
 - boost word dictionaries 202
 - enabling for a collection 32
 - in migrated collections 347
 - restore default values 199
 - static 198
 - URI patterns 203
 - document ranking
 - restore default values 199
 - document summaries
 - customizing 192
 - editing properties for 193
 - document tracking
 - description 302
 - disabling 302
 - enabling 302
 - log file configuration 302
 - log files 304
 - reports 303
 - document types
 - detecting 145
 - for parser services 147, 149
 - for Stellent parsers 152
 - document types (*continued*)
 - parsing 147
 - supported by Stellent parsers 154
 - document-level security
 - crawler configuration 37
 - crawler plug-ins 110
 - current credential validation 253
 - description 245, 251
 - for Lotus Domino documents 269
 - for Windows file systems 273
 - identity management 254, 259
 - indexed access controls 252
 - Lotus Domino documents 270
 - post-filtering results 251
 - pre-filtering results 251
 - real time validation 253
 - security tokens 252
 - single sign-on support 258
 - user profiles 257
 - documentation
 - finding 383
 - HTML 383
 - PDF 383
 - domain rules for Web crawlers 88
 - Domino Document Manager crawlers
 - configuration 59
 - DIIOp protocol configuration 76
 - IOCP configuration 77
 - NRPC protocol 73, 75
 - setting up in Solaris operating environment 73
 - setting up on AIX operating system 73
 - setting up on Linux operating system 73
 - setting up on Windows 75
 - URI formats 113
 - Domino user configuration, QuickPlace crawlers 271
 - dropped documents
 - description 302
 - log file configuration 302
 - log files for 304
 - reports about 303
 - Dublin Core elements 133
 - duplicate document detection
 - boost class configuration 204
 - description 175, 245
 - enabling security 249
 - global analysis 175, 249
 - dynamic ranking 197
 - dynamic summarization 192
- E**
- e-mail notifications
 - for alerts 310
 - for messages 310
 - SMTP server configuration 309
 - EAR files
 - ESAdmin application 275
 - ESSearchApplication application 275
 - ESSearchServer application 275
 - editing
 - collections 34
 - crawl spaces 40
 - crawler properties 39
 - editing (*continued*)
 - Data Listener applications 109
 - search application properties 214, 230
 - enabling index schedules 167
 - enterprise applications
 - ESAdmin application 275
 - ESSearchApplication application 275
 - ESSearchServer application 275
 - enterprise search
 - administration console 8
 - administrative roles 248
 - APIs 10
 - backing up 316
 - backup scripts 315
 - collection-level security 249
 - commands 351
 - components 3
 - crawler servers 4, 37
 - data flow diagram 11
 - document-level security 251
 - index servers 6, 165
 - integration with Lotus Notes 323
 - integration with WebSphere Portal 325
 - log files 305
 - monitoring 285
 - overview 1
 - parsers 5, 123
 - port number configuration 23
 - restore scripts 315
 - restoring from a backup 317
 - return codes 351
 - search applications 11
 - search servers 8, 185
 - security 245
 - session IDs 351
 - starting search servers 283
 - starting the servers 279
 - stopping search servers 283
 - stopping the servers 279, 281
 - URI formats 113
 - enterprise search administrator
 - changing the password on a single server 19
 - changing the password on multiple servers 20
 - description 248
 - role configuration 249
 - enterprise search servers
 - changing IP addresses 25
 - dual IP support 26
 - IPv6 protocol support 27
 - error messages
 - receiving e-mail for 308, 310
 - SMTP server configuration 309
 - viewing dropped document log files 304
 - viewing log files 312
 - ES_INSTALL_ROOT, description 19, 20
 - ES_NODE_ROOT, description 19, 20
 - es_special_field.default_field reserved field 207
 - es_special_field.default_metadata_field reserved field 207
 - es_special_field.regular_text reserved field 207

- es.cfg file 19, 20, 25, 26, 238, 264, 265
- es.search.provider.jar file 332, 340
- es.security.jar file 327, 332, 340
- es.wp5.install.jar file 327
- es.wp6.install.jar file 332, 340
- ESAdmin application
 - disabling security 275
 - logging in to 18
- esadmin command 351
- esadmin startSearch command 283
- esadmin stopIndex command 170
- esadmin stopSearch command 283
- esadmin system startall command 351
- esadmin system stopall command 351
- esapi.jar file 327, 332, 340
- esbackup.bat script 316
- esbackup.sh script 316
- exchangeproxypw command 239
- exchangepw script 19, 20
- eschangetrustpw command 238
- eschangewaspw command
 - multiple server configuration 265
 - single server configuration 264
- escrm.sh script 56
- escrm.vbs script 57
- escrdb2.sh script 49
- escrdb2.vbs script 49
- escrnote.sh script 73
- escrnote.vbs script 75
- escvbr.sh script 44
- escvbr.vbs script 45
- ESPACServer.ear file 327, 332, 340
- esrestore.bat script 317
- esrestore.sh script 317
- ESSearchAdapter.ear file 327
- ESSearchApplication application
 - config.properties file 214, 230
 - disabling security 275
 - starting 237
- ESSearchPortlet.war file 327, 332, 340
- ESSearchRegistrationPortlet.war file 327
- ESSearchServer application
 - disabling security 275
 - restarting 230, 231
- estimating system resources 285
- event publishing
 - DB2 crawler configuration 50, 52
 - description 46
 - setting up in Solaris operating environment 49
 - setting up on AIX operating system 49
 - setting up on Linux operating system 49
 - setting up on Windows 49
- Exchange Server crawlers
 - configuration 61
 - secure documents 269
 - URI formats 113
- external crawlers
 - configuring 109
 - Data Listener applications 109
- external sources
 - application ID security 250
 - associating with search applications 243
 - configuration 241

- external sources (*continued*)
 - description 241
- F**
- federated collections 31
- fielded search
 - description 171
 - string sort 171
- fields, mapping to boost classes 206
- file extensions
 - excluding from Web crawl spaces 88
 - supported by collection parsers 147, 149
 - supported by Stellent parsers 152
 - supported Stellent parsers 154
- firewalls, crawling Exchange Server documents 269
- followindex.rules file
 - configuring 101
 - description 101
- form-based authentication 95, 96
- free space alerts 307
- free text search, description 171

- G**
- global analysis
 - anchor text analysis 245, 260
 - description 6
 - duplicate document detection 175, 245, 249
- global Web crawl space 99
- global.rules file 99

- H**
- high recall queries
 - default boost factors 207
 - description 204
- HTML documentation for enterprise search 383
- HTML documents
 - parsing 150, 151
 - replacement rules 150, 151
 - searching 132, 133
- HTML replacement rules 150, 151
- HTML search fields
 - creating 133
 - description 132
 - Dublin Core elements 133
 - mapping elements to 132, 133
- HTTP basic authentication 95
- HTTP proxy servers 97
- HTTP status codes
 - received by Web crawlers 292
 - Web crawler report 292
- HTTPS, search server configuration 238

- I**
- I/O completion port module, crawler configuration 77
- identity management
 - configuration 259

- identity management (*continued*)
 - description 254
 - disabling 254
 - group extraction 254
 - single sign-on support 258
 - user profiles 257
 - user security context 254
 - XML query string 254
- ideographic languages 159
- index builds
 - concurrent 168
 - description 165
 - detecting changes 169
 - parallel 168
 - scheduling 167
 - startIndexBuild command 169
 - starting 297
 - stopping 170, 297, 298
 - system status 298
- index expansion
 - description 176
 - effect on index build time 178
 - effect on index size 178
- Index page, description 15
- index queue 298
- index servers
 - starting 279
 - stopping 281
- indexes
 - alerts about 306
 - anchor text 261
 - changing the schedule 167
 - collapsed URIs 171, 181, 182
 - concurrent builds 168
 - deleting from the queue 298
 - description 6, 165
 - detecting changes 169
 - disabling the schedule 167, 297
 - effect of wildcard characters 178
 - enabling the schedule 167, 297
 - monitoring 297, 298
 - parallel builds 168
 - removing URIs 171, 183
 - scheduling 166
 - scopes 171, 179
 - startIndexBuild command 169
 - URI formats 113
 - wildcard characters 171, 176, 179
- integration with WebSphere Portal
 - clustered system 340
 - description 325
 - es.wp5.install.jar file 327
 - es.wp6.install.jar file 332, 340
 - Lotus Quickr 325
 - setup scripts 326
 - Web Content Management 325
 - wp5_install script 327
 - wp6_cluster_install script 340
 - wp6_install script 332
- IOCP, crawler configuration 77
- IP address rules for Web crawlers 88
- IP addresses
 - IPv6 support 27
 - loopback adapter 26
 - support for dual 26
- IP addresses, changing 25
- IPv6 protocol 27

J

- Japanese
 - n-gram segmentation 162
 - removing new line characters 162
- Java connector for DB2 Content Manager 56, 57
- JavaScript support in Web crawlers 88
- JDBC database crawlers
 - configuration 62
 - crawling multiple tables 63, 65
 - plug-in to crawl multiple tables 63, 65
 - supported drivers 62
 - URI formats 113
- JDBC drivers
 - for JDBC database crawlers 62
 - for JDBC external sources 241
- JDBC external sources
 - configuration 241
 - deleting 241
 - editing 241
 - JDBC drivers 241

K

- keystore files 238
- keywords in quick links 194, 195
- Korean
 - compound term analysis 141
 - n-gram segmentation 162

L

- languages
 - automatic detection 160
 - searching 159
 - supported 159, 160
 - two-character codes 159
- LDAP external sources
 - configuring 241
 - deleting 241
 - editing 241
- LDAP user registry 262
- limiting the Web crawl space 88
- linguistic support
 - boost word dictionaries 200
 - code page detection 161
 - custom text analysis 134
 - language codes 159
 - language detection 160
 - locales 159
 - n-gram segmentation 162
 - native XML search 142
 - semantic search 134, 142
 - stop word dictionaries 189
 - synonym dictionaries 186
 - white space removal 162
- Linux operating system
 - Content Edition crawler
 - configuration 44
 - DB2 Content Manager crawler
 - configuration 56
 - DB2 crawler configuration 49
 - Domino Document Manager crawler
 - configuration 73
 - event publishing configuration 49

- Linux operating system (*continued*)
 - Notes crawler configuration 73
 - QuickPlace crawler configuration 73
 - Solaris operating environment
 - event publishing configuration 49
- Local User security, QuickPlace crawlers 271
- locales
 - parsing 159
 - searching 159
- log files
 - default location 305
 - description 305
 - e-mail options 310
 - filtering 312
 - for document tracking 302
 - maximum size 308
 - migration wizard 350
 - monitoring 304, 312
 - query logs 312
 - rotating 308
 - severity levels 308
 - size configuration 312
 - SMTP server configuration 309
 - viewing 312
 - viewing dropped documents 304
- Log page, description 15
- logging in to the administration
 - console 18
- loopback adapter configuration 26
- Lotus Domino domains 269, 270
- Lotus Domino Trusted Servers 270
- Lotus Notes
 - integration with enterprise search 323
 - plug-in installation 323
 - plug-in update site 323
 - search bar 323
- Lotus Quickr
 - integration with WebSphere Portal 325, 337
 - QuickPlace crawler configuration 78
 - Seed list crawler configuration 81
- low recall queries
 - default boost factors 207
 - description 204

M

- main indexes
 - concurrent builds 168
 - description 6, 165
 - detecting changes 169
 - scheduling 166
- mapping
 - analysis results to relational databases 140
 - common analysis structures to the index 139
 - fields to boost classes 206
 - HTML search fields 133
 - the common analysis structure to relational databases 140
 - XML elements to the common analysis structure 138
 - XML search fields 130
- maximum recrawl interval 93

- metadata fields, top result bar charts 234
- migrating
 - collections 347
 - rule-based taxonomy 347
- migration wizard
 - collections 347
 - default collection settings 349
 - default crawler settings 349
 - description 347
 - log file 350
 - rule-based taxonomies 347
 - starting 347
- MIME types, including in Web crawl spaces 88
- minimum recrawl interval 93
- monitor
 - description 248
 - role configuration 249
- Monitor view, description 15
- monitoring
 - collections 286
 - crawlers 288
 - Data Listener 301
 - dropped documents 303, 304
 - enterprise search 285
 - log files 312
 - parsers 296
 - popular queries 299, 300
 - recent queries 299, 300
 - response time history 299
 - search servers 299, 300
 - URI details 286
 - Web crawler active sites 290
 - Web crawler crawl rate 291
 - Web crawler thread details 290
 - Web crawlers 289
- multiple structured table plug-in 63, 65
- multiple-byte encoding 161

N

- n-gram segmentation 162
- native XML search 142
- new line character removal 162
- newHtmlTagReplacement parameter 150
- NNTP crawlers, configuring 69
- no-follow directives
 - configuring 101
 - description 101
- no-index directives
 - configuring 101
 - description 101
- nodes.ini file 25, 238
- Notes crawlers
 - configuration 70
 - DIOP protocol configuration 76
 - document-level security
 - configuration 269
 - field mapping rules 72
 - IOCP configuration 77
 - Lotus Domino Trusted Server 270
 - NRPC protocol 73, 75
 - setting up in Solaris operating environment 73
 - setting up on AIX operating system 73

- Notes crawlers (*continued*)
 - setting up on Linux operating system 73
 - setting up on Windows 75
 - tips for using 72
 - URI formats 113
 - validation of current credentials 270
- NRPC protocol, crawler configuration 73, 75

O

- OmniFind Enterprise Edition
 - administration console 8
 - APIs 10
 - changing IP addresses 25
 - changing the password on a single server 19
 - changing the password on multiple servers 20
 - commands 351
 - components 3
 - crawler servers 4
 - data flow diagram 11
 - dual IP support 26
 - index servers 6
 - integration with Lotus Notes 323
 - integration with WebSphere Portal 325
 - IPv6 protocol support 27
 - overview 1
 - parsers 5
 - port number configuration 23
 - return codes 351
 - search applications 11
 - search servers 8
 - session IDs 351
- operator
 - description 248
 - role configuration 249
- Oracle databases
 - access as external source 241
 - access with DB2 crawlers 46
 - access with JDBC database crawlers 62

P

- parallel index builds 168
- parametric fields
 - description 171
 - numeric sort 171
- Parse page, description 15
- parser servers
 - thread configuration 141
- parserdriver.collection.properties file 150
- parsers
 - ASCII parser 149
 - code page detection 161
 - compound term analysis 141
 - data analysis tasks 5
 - description 5, 123
 - document format detection 145
 - document types for parser services 147, 149

- parsers (*continued*)
 - document types for Stellent parsers 152
 - files with no extensions 149
 - HTML replacement rules 150, 151
 - language detection 160
 - linguistic processing 159
 - monitoring 296
 - n-gram segmentation 162
 - native XML search 142
 - new line character removal 162
 - parser type selection 145
 - parsing document types 147
 - starting 296
 - stopping 296
 - supported languages 159
 - supported Stellent document types 154
 - system status 296
 - threads 141
 - unknown document types 149
 - white space removal 162
- parserTypes.cfg file 145, 147, 149
- password-protected Web sites 95
 - form-based authentication 96
 - HTTP basic authentication 95
- password, enterprise search administrator 19, 20
- PDF documentation for enterprise search 383
- plug-in for Lotus Notes
 - installation 323
 - update site 323
- plug-ins
 - crawling multiple structured tables 63, 65
 - JDBC database crawlers 63, 65
- plug-ins, for crawlers 110
- popular queries
 - calculating 300
- popular queries, monitoring 299
- port number, enterprise search 23
- portlets
 - description 325
 - enterprise search 325
 - integration with WebSphere Portal 5.1 327
 - integration with WebSphere Portal 6 332, 337
 - integration with WebSphere Portal clusters 340
 - removing from WebSphere Portal 5.1 331
 - removing from WebSphere Portal clusters 343
 - removing from WebSphere Portal version 6 339
 - setting up for Lotus Quickr sources 337
- prefix rules for Web crawlers 88
- proxy servers 97
 - search server configuration 239

Q

- query expansion
 - description 176

- query expansion (*continued*)
 - effect on index build time 178
 - effect on index size 178
- query log configuration 312
- query validation 253
- quick links
 - creating 195
 - description 194
 - searching 194
 - URI formats 113
- QuickPlace crawlers
 - configuration 78
 - DIIOOP protocol configuration 76
 - Directory Assistance configuration 272
 - Domino user configuration 271
 - IOCP configuration 77
 - Local User security 271
 - NRPC protocol 73, 75
 - setting up in Solaris operating environment 73
 - setting up on AIX operating system 73
 - setting up on Linux operating system 73
 - setting up on Windows 75
 - URI formats 113

R

- ranking search results
 - boost classes 204, 206, 207
 - boost word dictionaries 202
 - description 197
 - dynamic 197
 - restore default values 199
 - runtime.properties file 199
 - static 198, 199
 - text-based scoring 197
 - URI patterns 203
- recent queries
 - calculating 300
- recent queries, monitoring 299
- recently crawled URLs, monitoring 289
- recrawl intervals for Web crawlers 93
- removeCjNewlineChars option 162
- removeCjNewlineCharsMode option 162
- removing URIs from an index 183
- response time history, monitoring 299
- restore scripts
 - description 315
 - running 317
- restoring enterprise search 315
- return codes, enterprise search 351
- revisiting URLs as soon as possible 93
- Robots Exclusion protocol
 - user agent identification 85
 - Web crawler compliance 86
- robots.txt files
 - user agent identification 85
 - Web crawler compliance 86
- rule-based categories
 - creating 127
 - description 124
 - selecting the categorization type 127
- rule-based taxonomy, migrating from WebSphere Portal 347

runtime-generic.properties file 193, 300

S

sample search application

cloning 233

config.properties file 214, 230

default deployment 230

description 11, 212

disabling security 275

HTTPS enforcement 238

search functions 211, 212

SSL enforcement 238

scheduling

crawlers 37, 41

index builds 166, 167

scopes

creating 180

description 179

searching 179

URI formats 113

URI patterns 179, 180

scripts

esbackup.bat 316

esbackup.sh 316

escrcm.sh 56

escrcm.vbs 57

escrdb2.sh 49

escrdb2.vbs 49

escrnote.sh 73

escrnote.vbs 75

escrvbr.sh 44

escrvbr.vbs 45

esrestore.bat 317

esrestore.sh 317

startcl 317

Search and Index API 10, 211

Search Application Customizer

config.properties file 231

starting 231

search applications

accessing 237

application IDs 250

associating with collections 212

associating with external sources 243

collection-level security 250

custom 211

customizing 231

description 11

sample 211, 212

starting 237

Search bar, WebSphere Portal

version 5.1, redirection to enterprise

search 329

version 6, redirection to enterprise

search 336

search cache

configuring 186

description 186

Search Center, WebSphere Portal

description 325

version 6, integration with enterprise

search 334

search options

complete matching 171

document content 171

fielded search 171

search options (*continued*)

for search results 171

free text search 171

parametric search 171

sortable fields 171

Search page, description 15

Search portlet deployment

es.wp5.install.jar file 327

es.wp6.install.jar file 332, 340

wp5_install script 327

wp5_uninstall script 331

wp6_cluster_install script 340

wp6_cluster_uninstall 343

wp6_install script 332

wp6_uninstall script 339

search response time

alerts about 306

monitoring 299

search result fields, description 171

search results

boost class configuration 204, 206, 207

collapsing 181, 182, 278

customizing summaries 192, 193

description 197

dynamic ranking 197

dynamic summarization 192

grouping 181, 182

post-filtering 251

pre-filtering 251

ranking 203

static ranking 198

summaries 192, 193

text-based scoring 197

URI pattern configuration 203

wildcard character expansion 179

wildcard characters 176

search servers

associating boost word

dictionaries 202

associating stop word

dictionaries 190

associating synonym dictionaries 188

boost word dictionaries 200

calculating query counts 300

description 8, 185

HTTPS configuration 238

monitoring 299, 300

popular queries 299, 300

proxy server configuration 239

recent queries 299, 300

redeploying dictionaries 191

response time history 299

search cache 186

SSL configuration 238

starting 279, 283, 299

stop word dictionaries 189

stopping 281, 283, 299

synonym dictionaries 186, 188

system status 299

SearchBarInclude.jsp file 329

searching

categories 124

collections 171

HTML documents 132, 133

quick links 194

XML documents 129, 130, 138

security

access controls 247

administrative roles 249

anchor text analysis 260

authentication 247, 275

bypassing document-level access

controls 277

collapsed search results 278

collection-level 249, 277

crawler plug-ins 110

description 245

disabling for enterprise

application 275

document-level 251, 252, 253, 259,

277

duplicate document detection 249

enabling for a collection 32, 245

enabling for enterprise search 261

global, WebSphere Application

Server 261, 262

HTTPS configuration for search 238

identity management 254, 259

LDAP user registry 262

Lotus Domino documents 269, 270

multiple server setup 265

search application IDs 250

single server setup 264

single sign-on support 258

SSL configuration for search 238

user profiles 257

WebSphere global security 275

Windows domains 273

security tokens

crawler configuration 252

disabling for a collection 277

document-level security 252

Security view, description 15

Seed list crawlers

configuration 81

integration with WebSphere

Portal 325, 337

URI formats 113

semantic search 134, 138, 142

server mode, Content Edition

repositories 44

session IDs, enterprise search 351

setup scripts

WebSphere Portal 326

SI-API (Search and Index API) 10, 211

siapi.jar file 327

simple text languages 159

single sign-on security

configuration 259

identity management 258

single-byte encoding 161

site detail reports

creating 292

description 289

SMTP server configuration 309

soft error pages, Web crawlers 94

Solaris operating environment

Content Edition crawler

configuration 44

DB2 Content Manager crawler

configuration 56

Domino Document Manager crawler

configuration 73

- Solaris operating environment (*continued*)
 - Notes crawler configuration 73
 - QuickPlace crawler configuration 73
- Solaris operating system
 - DB2 crawler configuration 49
- sortable fields
 - numeric sort 171
 - string sort 171
- SQL Server databases
 - access with DB2 crawlers 46
 - access with JDBC database crawlers 62
- SSL, search server configuration 238
- start URLs for Web crawlers 88, 93
- startcl script 317
- startIndexBuild command 169
- starting
 - crawler servers 288
 - Data Listener 301
 - enterprise search servers 279
 - index builds 297
 - migration wizard 347
 - parsers 296
 - Search Application Customizer 231
 - search applications 237
 - search servers 283, 299
- static ranking
 - description 198
 - enabling for a collection 32
 - in migrated collections 347
- Stellent parser
 - associating document types 152
 - default document types 154
 - description 145
 - parsing document types 147
- stellent.properties file 152
- stellentypes.cfg file 152
- stellentTypes.cfg file 145
- stop word dictionaries
 - adding to the system 190
 - associating with a collection 190
 - description 189
 - redeploying 191
- stopping
 - crawler servers 288
 - enterprise search servers 279, 281
 - index builds 297, 298
 - parsers 296
 - search servers 283, 299
- summaries
 - customizing 192, 193
 - dynamic 192
- synonym dictionaries
 - adding to the system 188
 - associating with a collection 188
 - description 186
 - redeploying 191
- system backup 315, 316
- system resources
 - estimating 285
- system restore 315, 317
- system status
 - collections 286
 - crawlers 288
 - index builds 298
 - parsers 296
 - search servers 299

- system status (*continued*)
 - Web crawlers 289
- System view, description 15

T

- tar files
 - crawling 112
 - URI formats 113
- task summary, administration
 - console 15
- taxonomies, migrating from WebSphere
 - Portal 347
- text analysis
 - common analysis structures 139, 140
 - mapping XML elements 138
 - text analysis engines 136, 137
- text analysis engines
 - adding to the system 136
 - associating with collections 137
 - description 134
 - mapping analysis results to relational databases 140
 - mapping analysis results to the index 139
 - mapping the common analysis structure to relational databases 140
 - mapping XML elements 138
- text processing
 - annotators 134
 - common analysis structures 134
 - text analysis engines 134
- text-based scoring 197
- thread details, monitoring 289
- threads
 - parser 141
 - Web crawler 290
- top results
 - bar charts 234
- Trusted Server configuration 270

U

- UIMA
 - adding text analysis engines to the system 136
 - associating with collections 137
 - common analysis structures 139, 140
 - description 134
 - mapping analysis results to relational databases 140
 - mapping analysis results to the index 139
 - mapping the common analysis structure to relational databases 140
 - mapping the common analysis structure to the index 139
 - mapping XML elements 138
- unicode encoding 161
- UNIX file system crawlers
 - configuration 83
 - URI formats 113
- unknown document types 149

- URI details
 - dropped documents 303
 - monitoring 286
- URIs
 - category rules 124, 127
 - collapsed in search results 181, 182
 - formats in enterprise search 113
 - influencing static scores 203
 - quick links 194, 195
 - removing from an index 183
 - scopes 179, 180
 - viewing details about 286
- URL path depth 88
- USC string 254
- user agents 85
- user profiles
 - configuration 259
 - description 257
- user security context string 254

V

- validation of current credentials 253, 269, 270, 273
- vbr_access_services.jar file 44, 45
- viewing
 - dropped document log files 304
 - log files 312
 - URI details 286
- visiting URLs as soon as possible 93

W

- Web Content Management
 - integration with WebSphere Portal 325
- Web Content Management crawlers
 - configuration 103
 - copying site URLs 106
 - URI formats 113
- Web crawlers
 - active sites 289, 290
 - configuration 84
 - cookie configuration 99
 - cookie format 98
 - cookies 98
 - crawl rate 291
 - crawled document dates 102
 - crawler history 289
 - crawling rules 88
 - creating reports about 292
 - followindex.rules file 101
 - global crawl space 99
 - JavaScript support 88
 - limiting the crawl space 88
 - monitoring 289
 - no-follow directives 101
 - no-index directives 101
 - password-protected Web sites 95, 96
 - proxy servers 97
 - recently crawled URLs 289
 - recrawl intervals 93
 - robots.txt files 85, 86
 - site details 289
 - soft error pages 94
 - start URLs 88, 93

- Web crawlers (*continued*)
 - system status 289
 - thread details 289, 290
 - URL status 289
 - user agents 85
 - visiting URLs as soon as possible 93
- WebSphere Application Server user
 - password for multiple server configuration 265
 - password for single server configuration 264
- WebSphere global security
 - disabling 275
 - search application customizer 231
 - search application properties 230
- WebSphere II Classic Federation 54
- WebSphere II Event Publisher Edition,
 - DB2 crawler configuration 50
- WebSphere MQ, crawler server
 - configuration 49
- WebSphere MQ, DB2 crawler
 - configuration 52
- WebSphere Portal
 - category tree migration 347
 - clustered system 340
 - collection migration 347
 - default migration settings 349
 - integration with enterprise search 325
 - Search bar, description 325
 - Search Center, description 325
 - setup scripts for enterprise search 326
 - taxonomy migration 347
 - version 5.1, integration scripts 327
 - version 5.1, removing enterprise search 331
 - version 5.1, Search bar configuration 329
 - version 6, integration scripts 332
 - version 6, removing enterprise search 339
 - version 6, Search bar configuration 336
 - version 6, Search Center configuration 334
- WebSphere Portal clusters
 - integration guidelines 340
 - integration scripts 340
 - removing enterprise search 343
- WebSphere Portal crawlers
 - configuration 105
 - URI formats 113
- white space removal 162
- wildcard characters
 - in queries 176
 - index expansion 176, 178, 179
 - query expansion 176, 179
- Windows
 - IPv6 protocol support 27
- Windows domains 273
- Windows file system crawlers
 - configuration 107
 - document-level security configuration 273
 - URI formats 113
- Windows operating system
 - Content Edition crawler configuration 45
 - crawler configuration 75
 - DB2 Content Manager crawler configuration 57
 - DB2 crawler configuration 49
 - event publishing configuration 49
 - wp5_install script 327
 - wp5_uninstall script 331
 - wp6_cluster_install script 340
 - wp6_cluster_uninstall script 343
 - wp6_install script 332
 - wp6_uninstall script 339
 - WpsMigratorLog.log file 350

X

- XML documents
 - native XML search 142
 - searching 130
- XML elements
 - mapping to search fields 130
 - mapping to the common analysis structure 138
 - searching 129, 138
- XML fragments, native XML search 142
- XML query syntax, native 142
- XML search fields
 - creating 130
 - description 129, 138
 - mapping elements to 129, 130, 138
- XPath, native XML search 142

Z

- zip files
 - crawling 112
 - URI formats 113

IBM



Java[™]
COMPATIBLE

SC18-9283-04



Spine information:

OmniFind Enterprise Edition

Version 8.5

Administering Enterprise Search

