

IBM Enterprise Information  
Portal for Multiplatforms



# Managing Enterprise Information Portal

*Version 8 Release 1*



IBM Enterprise Information  
Portal for Multiplatforms



# Managing Enterprise Information Portal

*Version 8 Release 1*

**Note**

Before using this information and the product it supports, read the information in "Notices" on page 119.

**First Edition (May 2002)**

This edition applies to Version 8 Release 1 of IBM Enterprise Information Portal for Multplatforms (product number 5724-B43) and to all subsequent releases and modifications until otherwise indicated in new editions. This edition replaces SC27-0875-00.

Portions of this product are: Copyright © 1990-2000 ActionPoint, Inc. and/or its licensors, 1299 Parkmoor Drive, San Jose, CA 95126 U.S.A. All rights reserved.

Outside In<sup>®</sup> Viewer Technology © 1992--2000 Inso Corporation. All rights reserved.

© **Copyright International Business Machines Corporation 1999, 2002. All rights reserved.**

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

---

# Contents

<b>About this guide</b> . . . . .	<b>v</b>
Who should use this guide . . . . .	v
Skills required for administrators . . . . .	v
Skills that are required for business analysts or process modeler . . . . .	vi
Where to find more information . . . . .	vi
Information included in your product package. . . . .	vi
Support available on the Web . . . . .	vii
How to send your comments . . . . .	viii
What's new in EIP Version 8.1 . . . . .	viii

## Chapter 1. Introducing Enterprise

<b>Information Portal</b> . . . . .	<b>1</b>
Searching for customer information. . . . .	1
The need . . . . .	1
The solution . . . . .	2
Overview . . . . .	2
Introducing the Enterprise Information Portal components. . . . .	2

## Chapter 2. Introducing the administration client . . . . . 7

Using <i>First Steps</i> to understand the system administration client. . . . .	7
Administering EIP . . . . .	7
Managing users and groups . . . . .	7
Using the administration client tools . . . . .	7
Introducing privileges . . . . .	9
Privileges . . . . .	9
Privilege groups . . . . .	9
Privilege sets . . . . .	10
Switching product views and databases . . . . .	10
Improvements and enhancements to the administration client . . . . .	10
Switching product views . . . . .	11
Defining document types. . . . .	11
Changing the server MIME type file (cmbcc2mime.ini) . . . . .	12
Changing the client MIME type file (cmbmime2app) . . . . .	12

## Chapter 3. Creating search templates 15

Defining servers . . . . .	15
Guidelines for defining servers . . . . .	16
Working with the OnDemand connector: TCP/IP tuning and sockets . . . . .	20
Working with the Extended Search connector . . . . .	21
Creating federated entities . . . . .	21
Understanding federated entities . . . . .	21
Using the Create Federated Entity wizard . . . . .	22
Creating federated text indexes . . . . .	23
Creating search templates . . . . .	23
Define search template . . . . .	23
Define search criteria . . . . .	24

Determine search settings. . . . .	24
Assign privileges . . . . .	24

## Chapter 4. Managing user access . . . 27

Creating user IDs and passwords . . . . .	27
Understanding DB2 administration authority . . . . .	27
Importing users from LDAP. . . . .	28
Creating privilege sets. . . . .	29
Creating privilege groups. . . . .	29
Assigning a privilege set to a user. . . . .	29
Assigning a user ID a grant privilege set . . . . .	29
Assigning users to resource managers . . . . .	30
Assigning users to collections . . . . .	30
Creating user groups . . . . .	30
Creating access control lists . . . . .	30
Assigning a privilege set to an access control list . . . . .	30
Creating domains . . . . .	31
Administering domains . . . . .	32
Accessing domains . . . . .	32
Assigning a user to a domain . . . . .	32
Assigning a user group to a domain . . . . .	32
Assigning a privilege set to a domain . . . . .	32
Assigning a resource manager to a domain. . . . .	33
Assigning a collection to a domain . . . . .	33
Moving a user from one domain to another . . . . .	33
Moving a user group from one domain to another. . . . .	34
Moving a resource manager from one domain to another. . . . .	34
Moving a collection from one domain to another . . . . .	34
Moving a privilege set from one domain to another. . . . .	34
Moving an access control list from one domain to another. . . . .	34

## Chapter 5. Managing information mining . . . . . 35

What is information mining? . . . . .	35
The EIP information mining services . . . . .	35
Components of the information mining services . . . . .	36
Using information mining in a business environment . . . . .	38
An example of using information mining . . . . .	39
Supported languages and formats . . . . .	43
Concepts . . . . .	43
System architecture. . . . .	43
The information mining concepts . . . . .	45
The information mining tools . . . . .	46
Programming interfaces . . . . .	53
<i>First Steps</i> . . . . .	54
Building a taxonomy . . . . .	54
Installing the Information Structuring Tool . . . . .	55
Getting started . . . . .	55
Access rights . . . . .	55
Defining a taxonomy . . . . .	56

Selecting training documents . . . . .	57
Uploading training documents . . . . .	58
Evaluating a categorization model. . . . .	60
Training a catalog . . . . .	64
Performance tuning. . . . .	65
Using the IBM Web Crawler. . . . .	65
IBM Web Crawler capabilities . . . . .	66
Configuring and running IBM Web Crawler for the Web . . . . .	66
The IBM Web Crawler configuration file. . . . .	68
Logging in IBM Web Crawler . . . . .	76
Troubleshooting . . . . .	77
Choosing summarizers . . . . .	78
IBM Web Crawler for Notes . . . . .	79
Excluding IBM Web Crawler from a server . . . . .	83

**Chapter 6. Introducing workflow. . . . . 85**

Understanding workflow . . . . .	85
How to use workflow . . . . .	85
Planning a workflow . . . . .	86
Information to be processed . . . . .	86
How information is handled. . . . .	86
Actions to take . . . . .	86
How information flows through the process . . . . .	87
How everything fits together . . . . .	87
Using Enterprise Information Portal workflow components . . . . .	87
Using workflow builder . . . . .	87
Using workflow services . . . . .	88
Defining worklists . . . . .	88
Defining action lists . . . . .	89
Creating a workflow . . . . .	89
Enabling workflow builder . . . . .	89
Starting the MQSeries Workflow server . . . . .	89

**Chapter 7. IBM Web Crawler sample files . . . . . 91**

config-sample2.xml sample . . . . .	91
IBM Web Crawler log analysis file example. . . . .	93

**Chapter 8. Using text search and QBIC® . . . . . 97**

Searching documents using the text search engine . . . . .	97
Enabling a text search server . . . . .	97
Searching images using Query by Image Content (QBIC) . . . . .	97

Introducing image search. . . . .	97
Setting up image search . . . . .	98
Loading and indexing sample data . . . . .	100
Before you load the sample data . . . . .	100
Creating a text search index . . . . .	101
Creating the image search database, catalog, and features . . . . .	102
Running the loader program . . . . .	103
Indexing the sample text data . . . . .	104

**Chapter 9. Document formats . . . . . 105**

Information mining document formats . . . . .	105
Word processing: Generic . . . . .	105
Word processing: DOS . . . . .	105
Word processing: International . . . . .	106
Word processing: Windows. . . . .	106
Word processing: Macintosh . . . . .	107
Spreadsheets formats . . . . .	107
Database formats . . . . .	107
Standard graphic formats . . . . .	108
High-end graphics formats . . . . .	110
Presentation formats . . . . .	110
Compressed and encoded formats . . . . .	110
Other . . . . .	111

**Chapter 10. Rights management . . . . . 113**

Protecting your intellectual property. . . . .	113
Using marking techniques . . . . .	114
Visible marking. . . . .	115
Invisible marking . . . . .	115

**Chapter 11. Accessibility features. . . . . 117**

Keyboard input and navigation . . . . .	117
Features for accessible display . . . . .	117
Alternative alert cues . . . . .	118
Compatibility with assistive technologies . . . . .	118
Accessible documentation . . . . .	118

**Notices . . . . . 119**

Trademarks . . . . .	121
----------------------	-----

**Glossary . . . . . 123**

**Index . . . . . 129**

---

## About this guide

This guide provides an introduction to all of the basic concepts you need to understand to manage your Enterprise Information Portal (EIP) system. Because EIP offers several components which you can manage from the administration client, and because you can access other product functionality by using EIP, this guide is not a typical system administration guide. This document focuses on the following topics and explains how to:

- Use EIP to meet business needs
- Access and use the administration client
- Manage user access
- Use EIP to search for content on multiple content servers, including structured data stored in relational databases, unstructured or multimedia content, or text documents
- Design, implement, and manage workflows

---

## Who should use this guide

This guide helps EIP administrators perform the following tasks:

### **System Administration**

Including database, server and network administration

### **User management**

Defining and granting access to individuals and groups, maintaining access control lists

### **Federated searches**

Defining and using federated search templates to retrieve content from your content management system

### **Information mining**

Extracting information from documents, categorizing documents and search results

### **Web Crawling**

Using IBM® Web Crawler to search and import content from the web

### **Text searching**

Using IBM DB2® TIE or IBM Text Search Engine (Content Manager Version 7.1 and earlier only) to search and index documents

### **Image searching**

Using Content Manager Version 7.1 (and earlier) to perform image searches

### **Workflow management**

Using EIP workflow tools to manage an enterprise's information workflows

---

## Skills required for administrators

Depending on which tasks you perform, you must understand:

- Security protocols for user access
- Windows NT®, Windows® XP, Windows 2000, AIX®, or Solaris operating systems

- Network administration
- Data models of the content servers in your content management system
- Database administration
- How to apply a working knowledge of content and search criteria when creating search templates.
- Information mining techniques and tools
- Principles of workflow design
- The business processes you wish to support with EIP workflows

---

## Skills that are required for business analysts or process modeler

Business analysts and process modelers will also find conceptual information in this guide concerning how to define and model EIP workflows for their enterprise.

To use Enterprise Information Portal workflow builder, you must:

- Understand staff requirements, programs, and data structures used in the business processes of your enterprise.
- Make decisions regarding the business or workflow processes of your enterprise.

---

## Where to find more information

Your product package includes a complete set of information to help you plan for, install, administer, and use your system. Product documentation and support are also available on the Web.

## Information included in your product package

The product package contains an information center and each publication in portable document format (.PDF).

### The information center

The product package contains an information center that you can install when you install the product. For information about installing the information center see *Planning and Installing Your Content Management System*.

The information center includes the documentation for Content Manager, Enterprise Information Portal, and IBM Content Manager VideoCharger for Multiplatforms. Topic-based information is organized by product and by task (for example, Administration). In addition to the provided navigation mechanism and indexes, a search facility also aids retrievability.

### 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>.

Table 1 shows the Content Manager publications included with IBM Content Manager for Multiplatforms.

*Table 1. Content Manager publications*

File name	Title	Publication number
install	<i>Planning and Installing Your Content Management System</i> <sup>1</sup>	GC27-1332-00
migrate	<i>Migrating to Content Manager Version 8</i>	SC27-1343-00



Table 1. Content Manager publications (continued)

File name	Title	Publication number
sysadmin	<i>System Administration Guide</i>	SC27-1335-00

**Notes:**

1. You receive a printed copy of *Planning and Installing Your Content Management System* with IBM Content Manager for Multiplatforms.

When you order IBM Content Manager for Multiplatforms, you also receive IBM Enterprise Information Portal for Multiplatforms. Or, you can separately order IBM Enterprise Information Portal for Multiplatforms. Table 2 shows the Enterprise Information Portal publications that are included with the product.

Table 2. Enterprise Information Portal publications

File name	Title	Publication number
apgwork	<i>Application Programming Guide for Windows</i> <sup>1</sup>	SC27-1347-00
ecliinst	<i>Installing, Configuring, and Managing the eClient</i>	SC27-1350-00
eipinst	<i>Planning and Installing Enterprise Information Portal</i> <sup>2</sup>	GC27-1345-00
eipmanag	<i>Managing Enterprise Information Portal</i>	SC27-1346-00
messcode	<i>Messages and Codes</i> <sup>3</sup>	SC27-1349-00

**Notes:**

1. The *Application Programming Guide for Windows* contains information about programming applications for both Content Manager and Enterprise Information Portal.
2. When you separately order IBM Enterprise Information Portal for Multiplatforms, you receive a printed copy of *Planning and Installing Enterprise Information Portal* with the product.
3. *Messages and Codes* contains the messages and codes for Content Manager and Enterprise Information Portal.

## Support available on the Web

Product support is available on the Web. Click **Support** from the product Web sites at:

<http://www.ibm.com/software/data/cm/>

<http://www.ibm.com/software/data/eip/>

The documentation is included in softcopy with the product. To access product documentation on the Web, click **Library** on the product Web site.

An HTML-based documentation interface, called Enterprise Documentation Online (EDO), is also available from the Web. It currently contains the API reference information. Go to the Enterprise Information Portal Library Web page for information about accessing EDO.

## How to send your comments

Your feedback helps IBM to provide quality information. Please send any comments that you have about this publication or other Content Manager or Enterprise Information Portal documentation. You can use either of the following methods to provide comments:

- Send your comments from the Web. Visit the IBM Data Management Online Reader's Comment Form (RCF) page at:  
<http://www.ibm.com/software/data/rcf>  
You can use the page to enter and send comments.
- Send your comments by e-mail to [comments@vnet.ibm.com](mailto:comments@vnet.ibm.com). Be sure to include the name of the product, the version number of the product, and the name and part number of the book (if applicable). If you are commenting on specific text, include the location of the text (for example, a chapter and section title, a table number, a page number, or a help topic title).

---

## What's new in EIP Version 8.1

The following changes have been made to the product:

### Support for Sun Solaris

You can install connectors, features, and databases on Solaris systems.

### Common system administration

A single client application provides separate access to Content Manager and Enterprise Information Portal administration.

### New connectors

- The ICM connector for Content Manager Version 8 Release 1 allows you to take advantage of Content Manager Version 8's powerful document storage features.
- The new C++ Extended Search Version 3.7 connector runs on AIX.

### Improved connectors

- Parametric text searches are supported from the federated layer and through a direct Extended Search connection.
- Functional enhancements and performance improvements to the OnDemand connector, including:
  - Modifications to the structure of an OnDemand DDO.
  - Asynchronous search is now supported

### New information mining services

- Feature extraction
- Clustering
- Language identification

### IBM Web Crawler

IBM Web Crawler is a feature that allows users to search for and summarize information on the Web and in Lotus Notes® databases.

### Workflow enhancements

Workflow is now fully supported on AIX and Solaris. The workflow builder, APIs, and JavaBeans™ provide improved workflow function and usability.

### Information center

The browser-based information center includes the documentation for

Content Manager, Enterprise Information Portal, and IBM Content Manager VideoCharger for Multiplatforms™. Topic-based information is organized by product and by task (for example, Administration). In addition to the provided navigation mechanism and indexes, a search facility also aids retrievability.

### **Accessibility**

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully. The major accessibility features for this product include:

- The ability to operate all features using the keyboard instead of the mouse.
- Support for enhanced display properties.
- Options for video and audio alert cues.
- Compatibility with assistive technologies
- Compatibility with operating system accessibility features
- Accessible documentation formats



---

## Chapter 1. Introducing Enterprise Information Portal

Many paper-intensive enterprises, such as insurance companies and financial institutions, administer large volumes of business-related content. The need for an enterprise solution for managing and accessing business information spans many industries.

A *content server* is a software system that stores multimedia, business forms, documents, and related data, along with metadata that allows employees to process and work with the content. When there is no way to effectively connect disparate content servers, a business can waste time and money by duplicating information or training employees to perform multiple searches.

Enterprise Information Portal (EIP) provides leading-edge technology to bring all of your enterprise resources to your workstation desktop. EIP can help you maximize the value of your information and multimedia assets by connecting disparate content servers through a single client. With an EIP client, users can quickly and concurrently access all connected content servers. Users can also do information mining or perform advanced searches across content servers, including the Web or an intranet. They can perform workflow tasks within your business processes that you define.

With EIP, you can customize applications for your enterprise. Using the EIP samples, application programmers can write both desktop and Web-based applications.

This section provides an overview of EIP. A scenario about a fictitious insurance company, XYZ Insurance, demonstrates the features and functionality of EIP.

---

### Searching for customer information

XYZ Insurance (XYZ), a large property and casualty insurance company, has an extensive collection of photographs, claims, policies, adjuster's notes, reports from experts, and other business documents.

XYZ keeps all memos that are sent to policy holders, along with medical and appraisal electronic forms in Lotus® Domino™.Doc file cabinets. XYZ archives all policy declarations, notices, and invoices in a Content Manager OnDemand server for long-term storage and quick access. XYZ stores all claim forms, photographs, and letters received from policy holders in a Content Manager for iSeries system folder. XYZ keeps reports from experts in a DB2 Universal Database™ (DB2 UDB) Data Warehouse Center Information Catalog Manager. XYZ also stores corporate media assets such as high-resolution graphics in a Content Manager system for the advertising, public relations, and new business departments to share. In addition, XYZ keeps information, such as company procedures, on its company intranet.

### The need

Claims, customer calls, and general policy holder servicing cannot be handled with the content from one server because employees need to access all customer information. To provide customer service, employees require simultaneous access to a variety of content servers. XYZ Insurance needs a solution that connects their

content servers and their company intranet for searching and retrieving information. They also want to expand their use of workflow processing.

Many different employees need to access documents, from clerks to claim adjusters to agents. XYZ must restrict access to certain items, while providing unlimited access to others. XYZ also wants an easy-to-use interface to reduce the need for training.

## The solution

XYZ Insurance deploys EIP because the comprehensive search technologies allow them to connect and search all of their content servers for the retrieval of data. Now, when an XYZ Call Center representative receives a call, a single federated search retrieves all of the necessary policy holder information.

XYZ Insurance also uses the EIP information mining feature to search for and retrieve information from the company's intranet. They also want to expand their use of workflow processes.

---

## Overview

EIP is a comprehensive product; its components work together to provide a solution uniquely suited to your enterprise. Centered on a multiple-tier architecture, EIP provides an administration client for managing searches, clients for running searches, and connectors for connecting to disparate content servers such as IBM Content Manager, Content Manager ImagePlus<sup>®</sup> for OS/390<sup>®</sup>, Content Manager OnDemand, Lotus Domino.Doc, DB2 Universal Database, DB2 DataJoiner<sup>®</sup> and DB2 Data Warehouse Center Information Catalog Manager. You can write additional connectors for additional content servers by using the EIP connector toolkit and samples.

EIP's architecture allows your client applications to run single searches on one or more content servers. To perform searches, a client uses search templates defined by the EIP administrator.

Using search templates, the client runs a *federated search*, a search that runs simultaneously across content servers whose native attributes have been mapped with the federated attributes used in the search template. EIP search templates contain search criteria, which reference federated attributes that are mapped to native attributes on each of the content servers. The EIP administrator creates the search templates. EIP provides connectors to access and search for data stored on multiple content servers. The content servers then return data objects to the client.

EIP's architecture provides the following advantages:

- Access using a single query to multiple and varying content servers that support e-business<sup>™</sup> transactions and customer service applications.
- Information mining capability across multiple content servers, including the Web.
- Workflow process access to data across multiple, heterogeneous content servers.
- Support for the development of client applications that are independent of data's location on any content server, because of the separation of client applications, indexes, and data.

## Introducing the Enterprise Information Portal components

This section explains each EIP component.

Table 3 lists the components and the compatible operating systems.

*Table 3. EIP component operating system compatibility*

Component	Windows	AIX	Solaris	Notes
Administration database	yes	yes	yes	Database includes workflow builder functionality
Administration client	yes	no	no	Client can connect to databases installed on Windows, AIX or Solaris operating systems.
Connectors	yes	yes	yes	
Information mining	yes	yes	yes	
IBM Web Crawler	yes	yes	yes	
Text search client	yes	yes	yes	
Image search client	yes	yes	yes	
Connector toolkit and samples	yes	yes	yes	<ul style="list-style-type: none"> <li>• Windows version includes source code to compile sample client. No sample client code installed on AIX.</li> <li>• Workflow samples and APIs are installed with the federated connector sample.</li> </ul>
Viewer	yes	no	no	Installs OnDemand client and viewer.
Information center	yes	yes	yes	

## Administration

The administration component provides the administration database and administration client subcomponents. When you install the administration database, you also install the workflow feature.

**Administration database:** The administration database is a DB2 database that manages information about EIP users and groups, privilege levels, passwords, user IDs, and other information. The database also provides the workflow and, optionally, the information mining functionality. You can install multiple databases. Each database provides the EIP workflow functionality. If you have a Content Manager Version 8 system, you can add EIP tables to a Content Manager Version 8 Library Server database.

**Administration client:** The administration client can be installed only on Windows workstations. You can install multiple clients. If you have a Content Manager Version 8 system, you can administer EIP and Content Manager Version 8 from the same client.

The client provides the interface that allows the administrator to:

- Define each content server for federated searching.
- Identify native entities and attributes on content servers and map them to federated entities.
- Maintain an inventory of the search criteria for all content servers.
- Create search templates.
- Identify and manage users and groups.
- Assign privileges to users and groups.
- Define access to search templates and set conditions on the actions users can take with the information retrieved from a search.
- Design and administer business workflow processes.

## Connectors

The connectors provide the communications interface between EIP clients, the content servers, and the administration database. The content server connectors, such as Content Manager Version 7.1 connector, provide the functionality that allows EIP to log in to the server, search for information, and return the information to the administration or end-user clients. The federated connector connects the administration client to the administration database.

EIP provides the following connectors:

- Federated connector connects EIP client to the administration database.
- Relational database connector for DB2 Universal Database 7.1, JDBC driver 1.3 (Java™ only), ODBC 3.0 (C++ only), DataJoiner 2.1.1.
- Content Manager connector for Content Manager Version 7.1 servers
- Content Manager connector for Content Manager Version 8.1 servers
- Content Manager OnDemand connector for Content Manager OnDemand Version 7.1
- Content Manager for VisualInfo™ for 400® Version 4.3, and Version 5.1
- Content Manager ImagePlus for OS/390 connector for ImagePlus/390 Folder Application Facility Version 3.1, Image Plus/390 ODM Version 3.1
- Lotus Domino.Doc connector for Domino.Doc Version 3.0a, Desktop Enabler Version 3.0a
- Extended Search connector for Version 3.7
- Information Catalog Manager connector for DB2 Universal Database Visual Warehouse™ Version 5.2, DB2 Universal Database Version 7.2

## Features

EIP has four optional features.

### Information mining

Information Mining provides linguistic services to find hidden information in text documents on content servers. During text document processing, metadata is created that can be summarized, categorized, and searched. WebSphere® Application Server 4.0 (standard or advanced edition) is an information mining prerequisite. Further, you can cluster similar



documents, extract features from documents, for example person or company names, and determine the language of a document.

#### **Image Search client**

Provides the interface required to access and administer Image Search functionality on a Content Manager Version 7 content server.

#### **Text Search client**

Provides the interface required to access and administer Text Search functionality on a Text Search server.

#### **IBM Web Crawler**

Web Crawler is a Java-based content crawler and miner. Web Crawler can crawl content in an intranet, extranet, or Internet web, Lotus Notes databases, or through Domino, local file systems, FTP collections, and NNTP newsgroups.

Web Crawler can mine metadata and text from many types of content. For example, HTML content can be mined for URL, title, body, time of last modification, and metatags such as author, keywords, description, and so forth. Users select from a set of predefined miners for a given type of content. The content and/or mined metadata are saved to local disk.

#### **Content viewer**

Installing the OnDemand viewer also installs the OnDemand client and other files required to view documents retrieved from an OnDemand server.

#### **Connector toolkits and samples**

Install the Enterprise Information Portal connector toolkit and samples to build your own Web or desktop client applications that access data and content on individual content servers. You must install the toolkits to create custom client applications.

You can use the toolkits to create customized clients and custom connectors for content servers. The toolkits provide:

- Java, C++, and ActiveX classes
- Content server-specific samples

#### **Information center**

The information center component contains the Enterprise Information Portal information center. The information center is a Web-based, searchable version of the Enterprise Information Portal library.



---

## Chapter 2. Introducing the administration client

The administration client provides the interface between the EIP administration database and the EIP administrator. This section describes the many features and functions the client offers to help manage your EIP system.

You access some features and functions, such as server definitions and user management, from icons located on the left pane of the client. You access other functions through the Tools menu bar.

---

### Using *First Steps* to understand the system administration client

*First Steps* is a module that comes with every installation of EIP. *First Steps* provides you with sample data and populates objects so you do not have to use real data. Use *First Steps* if you want to explore server definitions, users and groups, and other features to help understand the basic structure, look, and feel of the administration client.

---

### Administering EIP

As the system administrator, you can complete one or more of the following tasks through the administration client:

- Defining content servers
- Managing users and groups
- Managing privileges and access levels
- Creating federated search templates
- Creating federated entities
- Create subdomains, if administrative domains are enabled.
- Work with workflow, if workflow is enabled.
- Create a federated text entity in Content Manager Version 7.

---

### Managing users and groups

You allow users access to search for and work with documents on multiple content servers by creating user IDs and privileges. You restrict access to the data stored in the system by defining and assigning appropriate privileges to the users.

---

### Using the administration client tools

This section describes the 12 tools provided by the administration client.

#### LDAP configuration

When you click this option, EIP launches a window that contains four tabs:

- LDAP tab - you can enable importing of datasources from an LDAP server, enable LDAP user import and authentication, or select both.
- Server tab - contains fields to define LDAP server specifications, including hostname, user name, referral type and so forth.
- Authentication tab - contains fields to define Secure Sockets Layer information.

- Advanced tab - defines settings about maximum records and server timeout.

#### **User mapping option**

This option allows you to disable the default setting of User mapping enabled.

#### **Fed user mapping editor**

The federated user editor displays a list of users and gives you the option to map users to specific content servers.

#### **Search template viewer**

The search template viewer provides detailed information about all search templates. The viewer provides three options to view search template details:

- Associated Mappings (default) - provides details about the federated entities and other details about the search template
- Search Template - provides details about default operator, default values and so forth
- Display Results - provides details about the display name, display width, criteria order and so forth.

#### **Server inventory viewer**

Displays the inventories of the selected server or servers.

#### **Log viewer**

Use the log viewer to view the log generated after you refresh the server inventory. the log shows a list of messages when differences are found between new and previous inventories.

#### **Services**

Select Services to enable workflow and/or information mining.

#### **Administrative domains**

Select administrative domains to enable administrative domains. Administrative domains cannot be disabled after they have been enabled.

#### **MIME type editor**

The MIME type editor lists the following information for each content server:

- Content Class
- File Extension
- Relational Database (RDB) Column
- MIME type

The content server names listed in the MIME type editor are abbreviated and correspond to the list of content server names that appears when you define a new content server. **Tip:** DL is the abbreviation for the Content Manager Version 7.1 content server. V4 is the abbreviation for the Content Manager for AS/400<sup>®</sup> content server.

You can add to, remove and edit the default information in the MIME type editor.

#### **MIME to application editor**

Use the MIME to application editor to add to, delete or edit the five default MIME to application associations. The values and settings defined in the MIME to application editor impact the viewer used by the end-user clients.

### Server Type Definition

Use this Tool to define any custom servers developed by your system programmers.

### Change DB2 ID/password

Select this option to modify the connect-only DB2 user ID and password. This is a completely separate user ID from the administrator user ID.

---

## Introducing privileges

This section describes the Enterprise Information Portal privileges. Expand the Authorization icon to access the four features. **Tip:** because you can administer Content Manager Version 8 and EIP Version 8 from the same client, the client displays all privileges for both privileges to each side of the client.

## Privileges

A privilege defines the actions user can take when using EIP and EIP features. EIP creates a list of privileges during database installation. Click Privileges to view the default privilege list. You can create a privilege. When you view the privileges and descriptions, EIP organizes the information by related tasks:

### Client privileges

These privileges define the actions a client can take when they work with EIP. For example, if you associate the default client privilege named **ClientAddtoNoteLog** to a user, that user can add a note to a note log.

### System privileges

The system privileges define actions users and administrators can take on an EIP system. For example, if you associate the default system privilege named **DomainDefineGroup** to a user, the user can create, update, or delete group or group members in a domain.

### EIP privileges

EIP privileges define the actions users and administrators can take on EIP systems. For example, if you associate the default system privilege named **EIPAdminServer** with a user, the user can administer a server.

### IKF privileges

This group of privileges defines actions users can take to manage information mining, an EIP feature. For example, if you associate the default system privilege named **IKFAllPermissions** with a user, the user can perform all Information Mining operations.

### Workflow privileges

These privileges define what actions users can take when using the EIP workflow feature. For example, if you associate the default system privilege named **WFWorklist** with a user, the user can add, update, delete, or retrieve the workflow work list.

When you administer users and groups, you associate a privilege set with a user and/or a user group. When you assign privilege sets to a user group, all users in the group have those associated privilege sets.

## Privilege groups

Enterprise Information Portal installs nine default privilege groups.

- ItemsPrivGroup (Content Manager Version 8 only)
- LogonPrivGroup (EIP and Content Manager)

- DocRoutingPrivGroup (Content Manager Version 8 only)
- DataModelingPrivGroup (Content Manager Version 8 only)
- WorkflowPrivGroup (EIP only)
- ClientPrivGroup (EIP and Content Manager)
- EIPPrivGroup (EIP only)
- IKFPrivGroup (EIP only)
- SysAdminPrivGroup (EIP and Content Manager)

## Privilege sets

EIP installs nine default privilege sets.

- AllPrivSet (EIP and Content Manager)
- DomainAdminPrivSet (EIP and Content Manager)
- EIPAdminPrivSet (EIP only)
- EIPUserPrivSet (EIP only)
- ICMConnectPrivSet (Content Manager only)
- ICMTrustedLogonPrivSet (Content Manager only)
- ItemAdminPrivSet (Content Manager only)
- ItemLoadPrivSet (Content Manager only)
- ItemReadPrivSet (Content Manager only)

---

## Switching product views and databases

If you have Content Manager and Enterprise Information Portal as part of your enterprise solution, you can access both system administration clients from one user interface. In the past, if you had both products installed, you would have to open two separate clients. Switching from one client view to the other provides a convenient way to modify information that applies to both clients and fast access to either product.

To switch from administering EIP to administering Content Manager without logging off, go to the main system administration window and use the pull-down menu above the left pane and select Content Manager.

To switch between federated databases, go to the right pane in the client window and double-click a federated database icon.

You can also administer different databases without exiting the client and logging into the new database. The administration client displays an icon for all the administration databases listed in the `cmbds.ini` file. To switch to a different database, click the icon. If the new database has a different user ID than the one you entered when you logged in to the client, you will be prompted to enter a different user ID.

---

## Improvements and enhancements to the administration client

EIP Version 8.1 features significant enhancements to the EIP administration client, including:

### Improved wizards and dialogs

New dialogs make managing users easier. New wizards make defining and modifying federated entities and search templates easier. Users can still choose to use the dialogs supported by EIP Version 7.1.

### Shared administrative client

When you install EIP Version 8.1 and Content Manager Version 8.1 on the same system, the two products share one administrative client. If you are an administrator for both products, you log on to the client once and toggle between the two applications within the client. You can also switch between administration databases without having to log out and back in.

### Domain administrators

You can create domain administrators who have only administration privileges for a defined domain.

### Single sign-on and LDAP support

EIP now uses Windows Active Directory and LDAP to enable users to have single sign-on access to multiple content servers.

---

## Switching product views

If you have Content Manager and Enterprise Information Portal as part of your enterprise solution, you can access both system administration clients from one user interface. In the past, if you had both products installed, you would have to open two separate clients. Switching from one client view to the other provides a convenient way to modify information that applies to both clients and fast access to either product.

To switch from one product to the other without logging off, go to the main system administration window and use the pull-down menu above the left pane. If the pull-down menu lists any other product other than the one that you are currently using, you can switch to that product.

---

## Defining document types

EIP provides viewer support for some document types. If you define a document type to the server, you can launch documents within their native applications. For example, if you are storing Lotus Word Pro® documents in your Content Manager OnDemand server, you can set EIP to launch documents with a .lwp extension in Lotus Word Pro rather than the client document viewer.

You can define a document type by changing the following files:

#### **cmbcc2mime.ini**

Translates content classes to a MIME type stream, so content from content servers can be read by a client. The `cmbcc2mime.ini` file is in `installation_dir\cmb` where `installation_dir` is the Enterprise Information Portal installation directory. The default installation directory is defined by CMBROOT.

#### **cmbmime2app.ini**

Translate the MIME type stream to the client application that you use to view the documents. The `cmbmime2app.ini` file is in `installation_dir\cmb` where `installation_dir` is the Enterprise Information Portal installation directory. The default installation directory is CMBROOT.

**Important:** When launching an application based on MIME type, only the base object is displayed. Any markup that was made to the document is not displayed. If the document has multiple parts, only the first part is displayed. The MIME type in both files must match.

## Changing the server MIME type file (cmbcc2mime.ini)

When adding server MIME types, verify that the document type you are adding is a MIME type created for that file. For more information, see the Web site: <ftp://ftp.isi.edu/in-notes/iana/assignments/media-types>.

To add values to the cmbcc2mime.ini file, complete the following steps:

1. Open cmbcc2mime.ini in a text editor.
2. Use the following format for user-defined values:
  - Content class starts at 4096
  - The equal sign (=) follows the content class value
  - MIME type should follow the equal sign. If this is not a standard MIME type for that content class, follow these steps:
    - a. A MIME type is composed of a type and a subtype. Valid types are application, text, image, model, message, audio, and video.
    - b. A slash (/) follows the type
    - c. To create the subtype, the token (x-) must precede the token used for that document; for example  
`x-mydocumentclass (4096=application/x-mydocumentclass)`
  - Repeat 2b and 2c as necessary for every new MIME type.

**Tip:** OnDemand content servers map file extensions rather than content class numeric values to MIME type streams.

## Changing the client MIME type file (cmbmime2app)

The cmbmime2app file is installed with every client and is used to add a client MIME type or change the application values of existing MIME types.

To add a client MIME type, or change the application values of existing MIME types, complete the following steps:

1. Open cmbmime2app.ini in a text editor.
2. Add or change the MIME type to the application association:
  - a. Verify that the MIME type is defined in the server file cmbcc2mime.ini.
  - b. Initially, when the MIME type file is opened, all existing MIME type definitions are commented out with the number sign (#). For MIME type definitions that the administrator chooses, the number sign (#) must be removed.
  - c. Specify the MIME type followed by the equal sign (=).

**Important:** Specify only the name of the executable program. Its fully qualified path name must be defined in the PATH environment variable. If the fully qualified path name is specified, you can use double backslashes (\\) or forward slashes (/) as directory separators. You can specify only valid DOS directory and file names (8.3).
  - d. Specify a space or blank, then x=.
  - e. Specify the most common file extension used for the document type; include a period (.) before the file extension. For example:

```
text/plain=a=notepad x=.txt
text/richtext=a=c:\\progra~1\\window~1\\access~1\\wordpad x=.rtf
image/tiff\\c:/progra~1/window~1/access~1/imagevue/wangimg x=.tiff
```
  - f. Save the cmbmime2app.ini file.



**If you are using Content Manager OnDemand servers:** The cmbmime2app file is included on the Enterprise Information Portal CD with the value application/pdf=a=acrord32 x=.pdf defined to allow OnDemand PDF documents to be displayed by the Adobe Acrobat Reader. The Adobe Acrobat Reader must be installed on the same machine as the Enterprise Information Portal client, and the directory of acrord32.exe must be included in your PATH.



---

## Chapter 3. Creating search templates

A federated search is a query issued from a client application that simultaneously searches one or more content servers. EIP provides you with the tools that create search templates for federated searches. Because each content server stores and organizes information differently, the search template must account for these differences for each server. The search template maps federated entities and their federated attributes to native attributes in order to search the content servers.

Creating federated searches involves:

- Defining connections to content servers using EIP connectors
- Creating federated entities
  - Defining federated entities
  - Creating federated attributes
  - Mapping federated attributes to native attributes
  - Assigning parameters
- Creating search templates
  - Defining the search template
  - Defining search criteria
  - Defining template settings
  - Assigning access to client users

Two wizards, provided in EIP Version 8.1, make creating federated entities and search templates easier. The federated entity wizard includes a server inventory that can be filtered to make finding native attributes easy. It also generates valid default parameters for federated attributes, reducing the chance for misconfiguring them. The search template wizard helps you create search criteria. It also helps you design how the search criteria and results displays will look and act. It even provides you with a preview of what the search template might look like in your client application. Additionally, the dialogs for creating federated entities and search templates for EIP Version 7.1 are also available for those who prefer them.

All wizards, dialogs, and fields are documented in the EIP online help.

---

### Defining servers

You must define the server before you can connect to a server and perform a server inventory. When you right-click the Server icon and click New, the client displays all the connectors supported by EIP. Before you define a server, you must know some basic information about the connectors:

- Which connectors did the installer select? The installed connectors are listed in the `cmbs.ini` configuration file. On a Windows server, the default path is `x:\Program Files\IBM\CMgmt`. Ask your AIX or Solaris administrator for the location of `cmbs.ini` files.
- Did the installer select a local or remote connector option? The `cmbs.ini` file contains the Local or Remote connector types.
- If your system is configured for RMI, is the RMI server started? To start RMI on the local RMI server, use **Start→Programs→IBM Enterprise Information Portal for Multiplatforms 8.1→Start RMI servers**. If your system uses remote

RMI, look in `cmbsvclient.ini` to find the remote server where the RMI connectors are installed. Ask the RMI server administrator for more information.

- If the person who installed EIP included the CM for AS/400 connector, what information was included in the network table named `frnolint.tbl`? The AS/400 `frnolint.tbl` is in `%CMBROOT%`.
- If you are defining content servers that contain relational databases, such as Content Manager Version 8 and DB2, DataJoiner and Information Catalog, you must catalog or add the database from the workstation where you are using the client.

The following list provides general steps you take when you define a server:

1. Right-click Servers and select **New**.
2. Select a server from the list. The New Server window appears.
3. Type in the server name and description in the Server Name field on the General tab. For some servers, you type only the database name. For other servers, you type in the fully qualified name of the server where the database was installed.
4. Specify initialization parameters, if required. Some servers require initialization parameters, such as connect string and configuration strings. Other servers only require the database name.
5. Click Test Server connection. EIP logs in to some servers using the user ID and password you entered to start the administration client. If server requires a different user ID and password, EIP prompts you to enter a valid user ID and password specific to the content server you are defining.

**Tip:** You can also define a content server type that is not one of the predefined server types, but you must provide the Java or C++ connector classes and the server definition class for the new server type. You also need the Java connector to run the server inventory. For instructions about adding content servers, see the *Workstation Application Programming Guide* and the online API reference.

If your configuration to the content server was unsuccessful, see *Messages and Codes* for more information about how to troubleshoot the situation or about the error message you received.

You can also consult with the administrator of the server you want to connect to for more assistance.

## Guidelines for defining servers

This section provides guidelines to help perform the initial server definition.

### Connecting to DB2 (relational) databases

This section applies to DB2, DataJoiner, JDBC, ODBC, Information Catalog, and Content Manager Version 7 and Version 8 servers.

- **Important:** You must catalog each DB2 database before you define the server. You can use DB2 CCA to catalog the databases, or you can use a DB2 Command Prompt. Contact the DB2 administrator for more information.
- In the Server name field on the General tab, you must type the name of the database you want to connect to. Use capital letters when you type the server name.
- When you define DB2, DataJoiner, JDBC, ODBC and Information Catalog, click the Initialization Parameters tab and type the Schema Name that is associated with the database tables you are connecting to, for example `SCHEMA=ICMADMIN`.

- When you define a Content Manager Version 7.1 or Version 8.1 server, you are only required to type the database name. Do not change the default settings in the Initialization Parameters tab.
- When you define a Content Manager Version 7.1 server, you must have a network table named `frnlint.tbl` on your local drive in `x:\CMBROOT`. The network table contains the hostname, port number, and server type information EIP requires to locate and log on to the Content Manager Version 7.1 Library Server. If you define multiple Content Manager Version 7.1 servers, each server must have a separate entry in the `frnlint.tbl` file before you define the server.
- To connect to DB2 DataJoiner, ensure that the authentication method for Enterprise Information Portal is defined as server for the database instance defined in DB2 Universal Database.
- To connect to DataJoiner 2.1, you must download a bind program from the DataJoiner website and bind the DataJoiner database before you can define a DataJoiner server.

### Connecting to a Text Search server

To define a Text Search Server, you must first define the Content Manager Version 7.1 server that is associated with the Text Search Server.

You type the Text Search server name into the "Select the name of the associated Content Manager Version 7.1 server" from a drop-down box. This box is on the Associated Server tab.

The Content Manager Version 7.1 server and the Text Search server must be up and running before EIP can connect to them.

### Connecting to multiple Content Manager for AS/400 servers

If you use more than one AS/400 server, you must define the additional servers in the network table. The network table (`frnlint.tbl`) is located in `x:\<cmbroot>`. For the new server, type the server name, connection type (for example, TCP/IP), host name, port and server type. For the first server, the installer types in Server, Hostname and Port values during installation to create `frnlint.tbl`.

The following is a typical example of information stored in `frnlint.tbl`:

```
/* VI/400 Network Table */
SERVER: VI400 REMOTE TCP/IP
      HOSTNAME = vi400
      PORT     = 29000
      SERVER_TYPE = FRNLS400
```

### Configuring the Extended Search connector

The information you enter to define an Extended Search server depends on two factors:

- The Web server type on which Extended Server was installed - Domino Web Server, WebSphere, IIS.
- The port number defined for the Web Server where Extended Search was installed.

When you define an Extended Search connector, perform the following steps:

1. In the Server Name field on the General tab, enter the fully-qualified hostname of the Web server where Extended Search was installed.
2. On the Initialization Parameters tab, enter 80 in the Port Number field if the installer selected the default settings for the Web Server port number when installing Extended Search.

3. In the Application ID field, type Demo. Type the name as shown.
4. In the Password field, type Demo.
5. In the Additional Parameters field:
  - a. Do not change the two semicolons if you know that Extended Search was installed on a Domino Web Server and that the installer used the default port number settings for the Web Server and Extended Search port numbers during installation.
  - b. See the section below for information on how to modify the Additional Parameters field for Extended Search servers that were installed using custom settings.

If you use the Extended Search connector to communicate with an HTTP server, you might need to configure the connector to find the correct relative paths of CGI program servlets and ES port number. If the relative paths for CGI program servlets and ES port on the HTTP server are different from \CGI-BIN or \SERVLET or 6001, you can create a configuration file, for example, `desclient.cfg`, by completing the following steps: Set the directory to the directory in which your applications or samples are located. Create a configuration file; for example, `desclient.cfg`. This file is not provided with Enterprise Information Portal. Add the following lines to the `desclient.cfg` file:

```
DESCGIPATH=/cgi-bin/desReflector
DESREQURI=/servlet/ESAdmin
DESPORT=6001
```

where *cgi-bin* and *servlet* are the directory paths on the HTTP server that support the Extended Search connector. If the application server is WebSphere, the DESREQURI should be `/lotuskms/ESAdmin` instead of `/servlet/ESAdmin`

If you intend to search ES sources from Thin/Fat clients, define an additional parameter called

```
"DESCFGPATH=<absolute path of desclient.cfg>"
```

in the DES server definition dialogs of the administration client.

If you want to run ES samples, pass the absolute path of `desclient.cfg` in the command line arguments.

Example 1:

```
TConnectDES es.stl.ibm.com user password
PORT=80;DESAPPID=Demo;DESAPPPW=password;DESCFGPATH
=<absolute path of desclient.cfg>;
```

Example 2:

```
java TConnectDES es.stl.ibm.com user password
PORT=80;DESAPPID=Demo;DESAPPPW=password;DESCFGPATH=<absolute path of
desclient.cfg>;
```

### Defining the Information Catalog server

You must catalog the Information Catalog server before you define the server. Type the server name, for example, `SAMPLE1`, in the Server Name field. In the Initialization parameters tab, type `SCHEMA=<Schema name associated with SAMPLE1>`.

## Defining the OnDemand server

The OnDemand server and Library Server daemon must be running before you can define an OnDemand server. You can ping the OnDemand server before defining the server in EIP to verify that the server and daemon are running.

On the General tab, enter the fully-qualified host name of the server where OnDemand was installed.

On the Initialization Parameters tab, enter the port number that was assigned when the OnDemand server was installed. If the person who installed OnDemand selected the default port value of 0 during OnDemand installation, type 0 in the port number field. If the installer selected a different port number, enter that port number preceded by a # sign. For example, # 5000 might be an alternate port number chosen for OnDemand on a Windows server.

If you are defining an OnDemand server that was installed on an AS/400 server running Version 4 software, you must enter the following information in the Additional parameters field: STATECONNECT=#1.

If you are defining an OnDemand server that was installed on an OS/390 server running Version 2.1 software, enter the custom port number designated when OnDemand was installed on the OS/390 Version 2.1 server.

OnDemand requires that a socket is kept alive during connection.

## Defining the text search server

To define a Text Search Server, you must first define the Content Manager Version 7 server associated with the Text Search Server.

Type the name of the Text Search server in the Server Name field on the General tab. Select the associated Content Manager Version 7.1 server from a drop-down box on the Associated Server tab.

The Content Manager Version 7.1 server and the Text Search server must be up and running before EIP can connect to them.

## Defining the Domino.Doc server

In the server name field, type the path to the server name and library name of the Domino.Doc server. For example: oakley/DominoDoc1/Lib.nsf.

If you are using local connectors, you must install the Domino Doc Desktop Enabler on the workstation that has the EIP client. If you are using RMI, you must install the Domino Doc Desktop Enabler on the RMI server. The Domino Doc Desktop Enabler must be at the same version as the Domino Doc server.

Do not modify the two semicolons on the Initialization Parameters tab.

## Defining the ImagePlus for OS/390 server

When you define an ImagePlus for OS/390 server, you must obtain the following parameters to connect to the server. The list below contains sample values:

- FAF Port Number: 3061
- FAF Application ID: 01
- FAF Protocol: 4000
- FAF IP Address: 9.67.43.83
- Object Distribution Manager CICS: 4000

- Object Distribution Manager IP Address: 9.67.43.83
- Object Distribution Manager Port Number: 3082
- Object Distribution Manager Terminal ID: *leave this field blank*
- Additional parameters: *FAFSITE=CS61;*

### Using tracing with Content Manager ImagePlus for OS/390

Tracing can help you solve problems if you cannot connect to the Content Manager ImagePlus for OS/390 server. If you installed the connector for Content Manager ImagePlus for OS/390, you can turn on tracing for ImagePlus for OS/390 by modifying the `eypapi.ini` file that is located in `cmbroot`.

The `eypapi.ini` file contains the following lines:

```
; Path where the IPFAF files are stored
;   (MUST NOT have a trailing '\')
; -- default is the <ROOT Directory>\
;
IPFAFPath=d:\cmbroot
; Flag for Logging (EYPLmdd.LOG files)
; -- default is Logging OFF (0)
; -- 0 All Logging OFF
; -- 1 Log files created only error conditions logged
; -- 2 Log files created all conditions logged
;
Logging = 0

;-----
;
; Flag for Logging the FAF Parameters Types created by APIs
;   -- default is Logging OFF (0)
;   -- 0 Parameter types Not logged
;   -- 1 Log Faf Parameter Types
;
FafTypeLogs = 0
```

#### IPFAFPath

Specifies the directory where the logs are written. The log files are named: `EYPmdd.LOG`

where `mdd` is the month and day the log was created.

#### Logging

Specifies when a log file is created.

- 0 Do not log. The default setting is 0.
- 1 Created log files contain only error conditions.
- 2 Created log files contain all conditions.

#### FafTypeLogs

Specifies logging for the FAF parameter types created by APIs.

- 0 Do not log parameter types; the default setting is 0/.
- 1 Log FAF parameter types.

## Working with the OnDemand connector: TCP/IP tuning and sockets

A known Windows problem might affect performance when connecting to an OnDemand server. During repeated searches and retrievals on an OnDemand server, many Windows sockets are opened and closed. Two default Windows settings might impact heavy traffic between EIP and an OnDemand server:



- When an application closes a Windows socket, Windows places the sockets port into TIME\_WAIT status for 240 seconds; during this time the port cannot be reused.
- Windows limits the number of ports that an application can use to 5000.

To avoid the problems that might result, change the values for the timeout wait time and number of ports using the Windows registry editor.

- Change the value of the timeout wait time from 240 seconds to a lower number (the valid range is 30-300 seconds). The key's name is  
HKEY\_Local\_Machine\System\CurrentControlSet\services\Tcpip\Parameters\TcpTimedWaitDelay.
- Increase the maximum port number from its default of 5000 to a higher number (the valid range is 5000-65534). The key's name is  
HKEY\_Local\_Machine\System\CurrentControlSet\services\Tcpip\Parameters\MaxUserPort

For more information on TcpTimedWaitDelay and MaxUserPort, consult your Windows documentation.

## Working with the Extended Search connector

This section describes a change to the Extended Search connector in EIP Version 8.1.

User-specified locale is supported by passing the locale value in the DESLOCALE key. You can pass this pair value in the command line if you invoke the ES connector directly. You can set this value in the **Additional** arguments for ES properties.

**Tip:** The Extended Search server software is included in the EIP Version 8.1 product box.

---

## Creating federated entities

After you define your connections to content servers, your next step in creating federated searches is to create federated entities, which become the building blocks for search templates. This section explains federated entities and how to use the Create Federated Entity wizard.

## Understanding federated entities

Most of the time, client application users do not want to search for information on a server-by-server basis. Instead, they want to conduct a single federated search. Search templates allow client application users to bundle their searches into a single query. As an EIP administrator, you can create these search templates for use in client applications. Before you create the search template, you must first create federated entities, which map their federated attributes to native attributes on content servers.

For example, DB2 stores information in tables whose columns represent attributes of the information stored in a table. A table named Customer\_Demographics might contain columns such as Name, Pol\_Number, Address, Phone, and Occupation.

Content Manager, on the other hand, uses items, itemtypes, and attributes instead of tables and columns. The same information stored in DB2 could be stored in an

entity name `CustInfo`. Its attributes could be `CustName`, `Acct`, `HomeAddress`, `HomePhone`, and `Job`. In both of these cases, identical information is stored and identified differently.

EIP solves the problem of having to account for all the different ways content servers store identical information. Federated entities keep track of this information for you. A federated entity does not actually store data; it stores metadata about how each content server stores data. When you create a federated entity, you map all of its attributes to corresponding native attributes on the content servers you want to query.

For the example given above, you could create a federated entity called `Policy_Info`, with federated attributes `Policy_Name`, `Policy_Number`, `Home_Address`, and `Job_Title`. You can then map the federated attributes to each of their corresponding native attributes.

EIP can generate a server inventory that contains this information. A server inventory contains this information, and the Create Federated Entities wizard allows you to obtain a server inventory that can be filtered on content servers. Filtering on content servers is not available when you perform the optional server inventory that does not use the wizard. Once you generate the server inventory, you can then begin to map federated attributes to native attributes.

It is not enough to map federated attributes to native attributes. Each native attribute can have different properties as well. Attributes can be (1) nullable, (2) queryable, (3) updateable, and (4) text-searchable. Depending on which data type you select, you may also have options regarding the data length, precision, scale, and minimum and maximum values.

When you define these properties, you cannot make them more restrictive than the properties already defined by the native attributes mapped to the federated attribute. The wizard provides default properties that meet this criterion. If, after you have customized the default properties for the default federated attribute, you want to revert to the default properties suggested by the wizard, you can still choose the default settings.

To summarize, federated attributes map to corresponding native attributes on multiple content servers. Each federated attribute's properties encompass all of the properties for the native attributes. Once you create a federated entity, you now have the path to information stored on different content servers. You can then use the federated entities to create search templates for particular queries.

## Using the Create Federated Entity wizard

The Create Federated Entity wizard is new in EIP Version 8.1. While you can continue to use the same dialogs to create federated entities as in EIP Version 7.1 and earlier, the wizard makes it easier for you to create and modify a federated entity.

To create a federated entity using the wizard, follow the steps listed below:

1. **Define Federated Entity** Name and describe a federated entity. You can also determine if you want the federated entity to be text-searchable.
2. **Define Federated Attributes** Name and modify federated attributes.
3. **Map Federated Attributes** Map federated attributes to native attributes. Tools are provided to obtain a server inventory, to select the native attributes you wish to map, and to modify your mappings at a later date.

4. **Define Properties** Define the properties for each federated attribute. You can either customize the properties or accept default settings.
5. **Confirm Federated Entity** Review the settings you have selected for your federated entity. You can return to previous panels to modify your settings. Click **Finish** when you are done.

These steps correspond to using the wizard. See the EIP online help for more information on how to use the wizard.

---

## Creating federated text indexes

The Text Search Engine can be integrated with your Content Manager Version 7.1 and earlier content servers, so that you can automatically index, search for, and retrieve text information stored in Content Manager. Users can locate documents by searching for words or phrases. The text search server supports both single-byte and double-byte character sets.

If you are using Content Manager Version 7.1 and earlier servers with the Text Search Engine, you can create a federated text index. You then map the federated text index to the Content Manager text search index on the Content Manager text search servers.

When you create the federated text search index, you can enable it for combined searching, that is, for searching both native text indexes and native attributes. When enabling a federated text index for combined searches, you also map that index to a federated entity. You then map the native attributes mapped by the federated entity, and its federated attributes, to the native text search indexes on the text search servers.

---

## Creating search templates

After you create a federated entity, you create a search template. Remember that a search template uses a federated entity as a map to where content is stored. When you create the search template, you must still define what you want to search for, what you want to do with the search results, and who has permission to use the template. While you can only use one federated entity for each template, you can use a federated entity for multiple templates. You can also search on any combination of the federated entity's attributes as search criteria. To create a search template, complete these steps using the Search Template wizard:

1. Define the search template
2. Define the search criteria
3. Determine the search settings
4. Assign access privileges

These steps correspond to the steps for the Search Template wizard. See the EIP online help for details to complete the search template creation process.

### Define search template

After you start the wizard, it prompts you to define the search template. Be prepared to:

- Provide a name and description for the search template
- Select a federated entity for the search template. **Restriction:** You can only use one federated entity per search template.

- Select a federated text index, if applicable

**Tip:** The check box for federated text index applies only if you are using Text Search Engine for Content Manager Version 7.1 and earlier. If you use DB2 TIE for text search, it is a parametric search and can be configured as such in the search template.

## Define search criteria

After you define the search template, the wizard prompts you to

1. Choose a search type, either attribute or document. Document is only available if a federated text index was selected in the previous step.
2. Name the search criterion
3. Select a federated attribute
4. Select available operators
5. Provide default search string (document search only)

The wizard provides a drop-down menu that lists all of the federated attributes associated with the selected federated entity. These attributes become the search criteria for the search template. The wizard also supplies a list of available operators.

**Tip:** You can create more than one search criterion per template or delete existing criteria from the template.

## Determine search settings

This panel allows you to define your default search settings, criteria settings, and your display value settings. Each of these settings has a default value that you can modify. To modify these settings, click the corresponding button for each setting.

The Default Settings window allows you to:

- Control what happens if a server is unavailable when a client application user wants to use a search template
- Define a wildcard character for parametric searches
- Specify the name of the folder for saving search results
- Select whether the search has to use all (AND) or any of the criteria (OR)

The Criteria Settings window allows you to control the order of the search criteria, the order of the results display columns, the column headers, and column widths.

The Display Value Settings window provides you with a way to define the display value of search results. For instance, if the value for Weekday is Monday on one server, but Mon. on another, you can specify that Monday be used as a search result display value for both servers.

## Assign privileges

Besides defining where to look (with federated entities), what to look for (search criteria), and how to display results (settings), you must also define who has access to search templates.

The Search Template wizard's Assign Privileges window provides tools to assign access to the template to existing users or user groups.

Assigning a user access privileges for a search template does not grant that user access to the content servers mapped to the template. Users must meet the security requirements for each individual content server. You must use access control lists and user management to make sure users have the proper privileges before assigning them access to a search template.

When you use the wizard to search for users or user groups, EIP returns only users who have appropriate access to the requested content servers.



---

## Chapter 4. Managing user access

A user cannot access the EIP system without a user ID, password, or a privilege set. Before creating users and assigning them privileges, however, you must decide who will have access to the system and what their jobs require. You do not want users having the right to delete an object when they do not understand the ramifications of deleting that object. On the other hand, you do not want to prevent users from doing their jobs by not giving them the correct privileges. So, before assigning users privileges, you need to determine the types of tasks each job requires.

When users create an object in the EIP system, they must define the access that other users will have to that object. Users who create an object must define who can access the object and what operations can be done to the object. This definition is what is known to the EIP system as an access control list, or an ACL.

---

### Creating user IDs and passwords

Whether you are a system administrator with exclusive access to the system or a domain system administrator, you must define user access to the system and to objects on the system. You have the ability to create, modify, and change a user ID and password. You can change your own password at the time you log on. You cannot modify or change your own user ID.

Each user ID must be unique. You must consult the database administrator to determine whether user IDs and passwords are case-sensitive, because user authentication to the EIP system is always passed through the database on which the administration database runs.

If you want to create users who administrate domains, then you must grant privileges that allow them to manage user access.

For more information about privileges, see “Assigning a privilege set to a user” on page 29. For more information about granting privileges to a user, see “Assigning a user ID a grant privilege set” on page 29. For more information about domains, see “Creating domains” on page 31.

---

### Understanding DB2 administration authority

There are two kinds of administrative users. The first is a user who is going to work with the data model and item types (for instance, defining item types). This user must be granted both the DB2 privilege and EIP privilege.

The second kind of administrator is a user who is going to perform other activities, such as defining users. This administrator will not need database administration privileges, but will need EIP privileges.

The administration database creation script encrypts two passwords. One is for the system administrator and the other is for the default resource manager user ID. These passwords reside in the INI file after installation, and should be removed for security reasons.

When defining a user without providing a password, EIP will assume the user is a system (DB2) user and an administrative user in EIP. This user must be defined in the system and will not have a password for EIP.

For a user to log on to the administrative client:

- The operating system ID must have DB2 authority (dbadmin).
- Every system administrator who logs in to the system administration client must have an operating system ID and password.
- Every system administrator who logs into the system administration client must have a valid system administrator ID and password that has been defined as a administration database administrator.
- The library server ID and password are only used when connecting with OO APIs.
- The Change Database Password window (available in the **Tools** menu) is used to update the INI files, and is also only used by OO APIs.

---

## Importing users from LDAP

LDAP supports managing a user's ID and password at an enterprise level, rather than on a system-by-system basis. EIP makes use of two LDAP technologies: IBM SecureWay<sup>®</sup> Directory and Windows 2000 Active Directory. Both the user ID and password reside in a central directory. When a user logs onto Content Manager or Enterprise Information Portal, the user ID and password are authenticated and the user ID's specific privileges are checked by the user profile in the EIP database.

LDAP might have been enabled during your EIP installation. If LDAP was not enabled during installation, you can activate it by using the LDAP User Registry Import utility. To enable LDAP, click **Start** → **Programs** → **IBM Content Manager for Multiplatforms 8.1** → **LDAP User ID Import Scheduler** → **Start** → **Programs** → **EIP for Multitplatforms 8.1** → **LDAP User ID Import Scheduler**, and then going into the system administration client and activating the LDAP feature (**Tools** → **LDAP Configuration**).

After you enable LDAP, you can import users when you create or modify a user by clicking **LDAP**. The EIP administration database uses the same user ID and password that a user has in LDAP for authentication in the system. The administration database tries to connect automatically to the LDAP server to authenticate the user. If the LDAP server is not running, authentication fails.

If you find that a user that you specified in your current configuration is not currently listed in the LDAP server, then you can change your LDAP server by going to the main system administration client window and clicking **Tools** → **LDAP Configuration**.

You can also change your current LDAP server by going to the LDAP User Registry Import Utility from the **Start** → **Programs** → **IBM Content Manager for Multiplatforms 8.1** → **LDAP User ID Import Scheduler** → **Start** → **Programs** → **EIP for Multitplatforms 8.1** → **LDAP User ID Import Scheduler**.

For information about planning for LDAP, see *Planning and Installing Your Content Management System*. For information about how to implement LDAP, see the system administration client online help.



---

## Creating privilege sets

When you plan your EIP system configuration, you must also decide who will have access to your system and how much access these users will have to the objects on your system. The EIP system defines access through privileges.

A privilege grants the right to access a specific object in a specific way. Privileges include rights such as creating, deleting, and selecting objects stored in a system. A group of privileges assigned to a user is a privilege set.

Your first task in managing access is to create privilege sets for users. A *privilege set* identifies the tasks or actions that a user can perform. Privilege sets combine privileges and are tailored for certain types of users. For example, you might want one set of administrators to manage your document routing server and another set of administrators to manage a domain. When an administrator logs on, EIP checks the administrator's privilege set.

The system administration client has a number of predefined privileges that you can group together into a privilege set. You then assign the privilege sets that you create to individual users. You cannot assign a privilege set to a user group.

## Creating privilege groups

Privilege groups are like user groups for users. You create a privilege group to put similar privileges together to easily find the privileges you want to include in a privilege set. For example, if you have two privileges that you assign to almost every user in your system, instead of searching through the many privileges you have each time that you create a privilege set, you group these two basic privileges into a privilege group called BasicPrivs.

## Assigning a privilege set to a user

The system administration client has a number of predefined privileges that you can group together into a privilege set. You then assign the privilege sets you create to individual users. You cannot assign a privilege set to a user group.

You can create privilege names, but you cannot create the privilege itself. You need to work with the system programmer to create any privileges that are not yet defined to the system administration client.

You can use the privilege sets that come with EIP, or you can create your own.

## Assigning a user ID a grant privilege set

To prevent users from creating a user ID with more privileges than they have, EIP has implemented the use of a grant privilege set. When you assign a user ID with a grant privilege set, you give them authority to create user IDs within the limits of their granted privileges. For example, you can give a user ID a set of system administration privileges to manage a domain. You might, however, want to ensure that the user ID does not have the privilege to create users. So, when you create this user ID, in the grant privilege set field, you would select NoPrivSet. In effect, the user ID can manage the domain but cannot create users for that domain.

## Assigning users to resource managers

To allow users to access a specific resource manager, you assign a resource manager to a domain that users have access to. For more information about assigning resource managers to domains, see “Assigning a resource manager to a domain” on page 33.

## Assigning users to collections

To allow users to access to collections, you assign a collection on a resource manager to a domain that users have access to. For more information about assigning collections to domains, see “Assigning a collection to a domain” on page 33.

---

## Creating user groups

Often, users with the same job description have the same or similar tasks, and therefore, the same access to objects on your system. You can group users with common access needs together in a user group. You cannot nest user groups.

A user group is solely a convenience grouping of individual users with similar tasks. You do not assign a user group a privilege set. Each user in a user group has his or her own privilege set. A user group makes it easier to create access control lists for objects in your system.

If you have domains enabled, before you assign a user ID to a group, check to see if that user group is in a specific domain or the PUBLIC domain (see “Administering domains” on page 32 for more information about domains). Make sure that the user group is in the domain that you want your user ID to be in. If you want to create a user ID specifically for a domain, you can click **New User** within the User Group window. You can then add the user that you create to the user group, and ensure that the user is in the same domain.

---

## Creating access control lists

You provide users with the privileges that they need to accomplish their tasks. Objects, on an individual basis, have certain access control issues.

An access control list (ACL) is a list consisting of one or more individual user IDs or user groups and their associated privileges. You use ACLs to control user access to objects in the EIP system. The objects that can be associated with access control lists are: the data objects stored by users, item types and item type subsets, worklists, and processes.

Privilege sets define the individual user’s maximum ability to use the system, an ACL restricts an individual user’s access to an object. An ACL that has a privilege not defined by a user’s privilege set does not grant the user with that privilege. Only users that have that privilege can use that privilege on an object. An ACL limits user access, it does not grant more access. Access control lists provide another level of security when managing a system.

## Assigning a privilege set to an access control list

Each user ID that you add to an access control list (ACL) needs a privilege set associated with it. The user ID and privilege set define which users have access to an object and what kind of access they have to that object.

Users cannot access any object unless they are on the ACL. To add a user or user group to an ACL, you need to select a user ID and a privilege set for the ACL and click **Add**. For each defined ACL, you will find the user IDs and groups listed in the Access Control List window. You can modify this table by adding and removing user IDs and groups. For more information about creating or modifying an ACL, see the system administration client online help.

---

## Creating domains

A domain is a section of an administration database that one or more administrators manage. Domains consist of user IDs, user groups, access control lists, privilege sets, access control lists, resource managers, and SMS collections. Domains are not visible to users, so what you name your domains will only have meaning to you and the system administrators who manage them. Users do not know that you have limited them to a part of the administration database, meaning that they only know about items within that domain.

Domains limit administrative and user access to a subsection of the administration database. An administrator with full privileges to the administration database can delegate limited administrative privileges to another administrator. The administrator with full privileges, a super administrator, has access to all sections of a administration database while an administrator with limited privileges, a subadministrator, has access to only a section of the administration database.

Domains restrict the access a subadministrator has to access control lists (ACLs). Only super administrators can create ACLs that subadministrators can use to either add or delete user IDs and user groups. Subadministrators cannot create, update, or delete ACLs.

A subadministrator might share different combinations of the super administrator responsibilities but only for their domain. By creating domains and assigning administrators to manage those domains, the super administrators can delegate subtasks while concentrating on the overall system and managing it efficiently as the subadministrators manage users and tasks specific to their domain.

Before you enable domains, consider the following conditions:

- You cannot disable domains
- Resource managers, collections, user IDs, and user groups can exist in only one domain at a time
- Privilege sets and access control lists can exist in more than one domain at a time
- Except for the PUBLIC (shared) domain, domains do not overlap
- Any object created in the super administrative domain cannot be moved, whether if it is system generated or user created.

To enable domains, go to the file menu, select **Tools** → **Administrative Domains** and then select **Enable Administrative Domains**. For specific instructions about how to configure your administration database for domains, see the system administration client online help.

## Administering domains

Depending on your privilege set, you administer either the whole administration database or a specific domain. An administrator who has full access to the administration database is a super administrator. A subadministrator has full access to the objects in a specific domain.

Each type of administrator has the ability to create, retrieve, update, and delete the objects in their domains, including users and collections. Subadministrators can see and retrieve objects only in their domain and list or retrieve in the PUBLIC, or shared, domain.

## Accessing domains

Subadministrators cannot change the domain of an object. They can, however, access the contents of their own domain and list or retrieve any object in the PUBLIC, or shared, domain.

Super administrators have access to all domains on the administration database. They can create an object and assign it to a domain. Some objects, such as privilege sets and ACLs, only they can create for subadministrators to use.

Subadministrators can only do create, retrieve, update, and delete (CRUD) for any objects in their domain.

## Assigning a user to a domain

When you create a user ID, you have the choice to assign it to a domain, or leave it in the default domain. You can change the domain of the user ID at a later time through user properties.

A user ID can have access to only one domain at a time. You cannot add a user to the PUBLIC, or shared, domain.

Only super administrators have the authority to create domains and assign users to those domains. A domain can have more than one subadministrator, but only the super administrator can define who those administrators are by giving them system administration privileges within a privilege set. The **Grant privilege set** field in the New User or User Properties window will indicate which administrative privileges a subadministrator has within a domain.

## Assigning a user group to a domain

Assigning a user group to a domain changes the domain designated for each user ID in that user group. A user ID can have access to only one domain at a time. So, any user ID included in a group that you assign is also moved to the new domain.

A user group name cannot be in only one domain at a time. You can assign the user group into the PUBLIC, or shared, domain.

## Assigning a privilege set to a domain

Any user ID that you add to a domain must also have an associated privilege set. If you do not include the associated privilege sets, then the users cannot perform their tasks. The best place to store privilege sets to make them available to any user is the PUBLIC, or shared, domain.

## Assigning a resource manager to a domain

You can restrict user access to certain resource managers by assigning them to a specific domain. When you define a new resource manager for a administration database to access, you have the option to select a domain.

The default for all resource managers is PUBLIC. If you do not want everyone to have access to the resource manager, you need to assign it to a domain. If you do not see a domain that you can assign the resource manager to, you can still define the resource manager and then create the domain you need. After you have the appropriate domain defined, open the resource manager properties and select the domain.

## Assigning a collection to a domain

You can restrict user access to a certain collection on a resource manager by assigning it to a specific domain. If the resource manager is in the PUBLIC domain, you can assign a collection to any other defined domain. If the resource manager, however, is defined to a specific domain already, then you cannot assign the collection to another domain, even if you want to assign the collection to the PUBLIC domain.

A user needs access to the resource manager to access the collections on it, so you cannot restrict access to the resource manager without imposing the same restrictions to the collections on it.

## Moving a user from one domain to another

You might find reason to remove certain users from one domain and add them to another. Consider using the **Description** field in the User definition window as a way to remember which user groups a user is grouped in. It might make this task a little easier.

**Important:** This task is very time consuming and can result in problems with accessing the system if you do not do it right. You must be a super administrator to change the domain of a user.

Follow these steps carefully:

1. Find all of the groups that the user belongs to.
2. For all the groups the user belongs to, either move these groups to the PUBLIC domain, or remove the user from all the groups.
3. Move any resource manager associated with this user to the PUBLIC domain, followed by all of the collections for each resource manager that you move to the target domain.
4. Create, *do not move*, all of the privilege sets associated with the user in the target domain, if they are not already in the target domain.
5. Create, *do not move*, all of the access control lists associated with this user, if they are not in the target domain.
6. Move the user to the target domain by opening the user's Properties and changing the user's domain.
7. **Optional:** You can move the groups and resource manager that you moved in steps 1, 2, and 3 from the PUBLIC domain to the target domain, but you can only do so if there are no more users remaining in the source domain who are associated with the groups and resource managers that you move. Otherwise, the groups and resource managers need to stay in the PUBLIC domain to allow sharing for users in different domains.

**Reminder:** At no time can a user be in the PUBLIC domain. Users cannot be shared.

## Moving a user group from one domain to another

**Important:** This task can result in problems with accessing the system if you do not do it right. You must be a super administrator to change the domain of a user group.

Follow these steps to move a user group to a different domain:

- If the user group is empty, delete the group from its current domain then recreate the group and assign it to the target domain.
- If the user group is not empty, follow these steps:
  1. Find all of the users that belong to this group.
  2. Delete the group from its current domain, which will delete all of the users.
  3. Recreate the group and assign it to the target domain.
  4. Add all of the users to this newly created group.

## Moving a resource manager from one domain to another

You must be a super administrator to change the domain of a resource manager.

To move a resource manager to another domain, follow these steps:

- If the resource manager contains no collections, move the resource manager to the target domain by opening its properties and changing the domain to the target domain.
- If the resource manager contains collections, follow these steps:
  1. Move the resource manager to the PUBLIC domain.
  2. Move the collections to the target domain by opening Properties and selecting the target domain.
  3. Move the resource manager to the target domain by opening Properties and selecting the target domain.

## Moving a collection from one domain to another

You must be a super administrator to change the domain of a collection.

Follow these steps to move a collection from one domain to another:

1. Find out the resource manager the collection belongs to.
2. Move the associated resource manager to the PUBLIC domain.
3. Move the collection to the target domain by opening Properties and selecting the target domain.
4. Move the resource manager to the target domain by opening Properties and selecting the target domain.

## Moving a privilege set from one domain to another

Because privilege sets can reside in multiple domains, you can add them to the target domain without moving them.

## Moving an access control list from one domain to another

Because access control lists can reside in multiple domains, you can add them to the target domain without moving them.

---

## Chapter 5. Managing information mining

This section begins with what information mining is and how you can use information mining in a business environment. This is followed by sections on the information mining First Steps, concepts in information mining, the Information Structuring Tool, and finishes with remarks on performance tuning.

---

### What is information mining?

Information mining is a key technology that helps companies provide users with easy access to relevant information at a low cost by automating many aspects of information extraction and analysis.

The first challenge in information mining is to readily access information in unstructured text for use by computers. The complete interpretation of only factual knowledge stated in unrestricted natural language is still out of reach using current technology. However, tools that apply pattern recognition techniques and heuristics are capable of extracting valuable information from arbitrary free-text. Extracted information ranges from identifying so-called important words, like names, institutions, or places mentioned in a document to whole summaries of a document.

Yet there is more to information mining than just extracting bits of information from single documents. When you are dealing with huge collections of documents, information mining comes into play. The "mining" metaphor is used for both the knowledge discovery process, that is, identifying and extracting codified information from single documents and storing this information as metadata, as well as the analysis process of the distribution of these features across document collections detecting interesting phenomena, patterns, or trends.

### The EIP information mining services

The EIP information mining services provide an infrastructure for the creation and maintenance of information related to individual documents or collections of documents. This information is referred to as metadata. Examples of information characterizing the document content and stored as metadata include:

- Title
- Abstract or summary
- Names, terms, or expressions
- Categories a document belongs to

What makes information mining different from a traditional metadata store that associates documents with metadata is that information mining provides capabilities to automatically create metadata, even if it is not explicitly available. The mining and retrieval algorithms are able to access relevant information in huge document collections, either by using the metadata to guide the retrieval process or by running statistical models against the metadata to find interesting relations between documents that may not be obvious when looking at the individual documents of the collection.

As the mining and retrieval operations work on well-defined sets of metadata instead of considering the original document content, the speed of these processes

can be increased significantly by keeping the metadata in a dedicated store, the so-called *information mining data store*. This speeds up access to the metadata significantly since applications can fetch metadata from a single repository and need not go back to an arbitrarily remote content server. Since metadata is crucial for retrieval and navigation and is often used to narrow down a search result, many steps can be performed without ever going back to a content server.

Another advantage of using the metadata store is that it keeps data associated with a document separate from the actual document. Obviously, for documents that are available for read access only, such as foreign documents on the Web, keeping the content and metadata within the same repository is not an option at all.

The EIP information mining services provide the following mechanisms for the automatic creation of metadata, namely:

- **Categorization** which assigns one or more categories to a document based on a user-defined taxonomy. The categorization component contains an application that provides a graphical user interface for the creation and maintenance of taxonomies called the *Information Structuring Tool*.
- **Summarization** which extracts the most important sentences of a document, and so can help the user decide whether to read the entire document or not. The user can specify the length of the intended summary and in this way directly influence the balance between the complexity of the extracted metadata and the amount of information in the document.
- **Language identification** which determines the language a document is written in. This is a useful preprocessing step before applying other services.
- **Information extraction** which recognizes significant vocabulary items, like names, terms, and expressions, in text documents automatically.
- **Clustering** which divides up a set of documents into similar groups or clusters. Clusters are derived from the document collection automatically.

## Components of the information mining services

An information mining application typically encompasses the following tasks:

1. Organizing data in such a way that it can be browsed and navigated
2. Accessing disparate data sources
3. Using the advanced search operation to filter out the required data for a forecast or trend

Figure 1 on page 37 illustrates these tasks.





Figure 1. The information mining tasks

To exploit the information mining functionality, documents must be organized so that they can be browsed and navigated. This task is typically performed by a librarian or knowledge engineer. The librarian uses the Information Structuring Tool (IST) to define a taxonomy, a hierarchical thematic characterization of the data in the documents to be further explored. The IST is an application with a graphical user interface that allows you to create and maintain taxonomies. The categories are trained and, once the taxonomy is stable, different data sources can be accessed.

Documents can be imported from any EIP content server or from the Web using the Web Crawler service (only available as JavaBeans) and assigned to categories.

The text analysis and metadata creation functions that are subsequently applied to the document content are available as a programming interface on the level of non-visual JavaBeans and a Java Service API.

The information mining JavaBeans are software components for rapid application development and are JavaBeans conform. The Java Service API contains the full information mining functionality as individual building blocks for creating applications. JavaBeans based samples and sample JSPs are provided to support the creation of applications.

Identifying the content of a document is a prerequisite for all information mining operations that process documents. Substeps of this task include:

1. Identifying the document code page
2. Identifying the text sections that need to be processed, in other words, ignoring markup information or binary data such as images.

As documents in content servers may be arbitrarily structured, the information mining services provide a means to write specific modules that identify and extract relevant text portions from document formats. A default module that covers a wide range of frequently used document formats is also available in the information mining services. See Chapter 9, "Document formats" on page 105 for a list of the supported formats, and the *Application Programming Guide for Windows* for details on how to use the sample default module.

Creating metadata for each of the selected documents involves processing the document's content and applying statistical methods or heuristics based on knowledge resources, for example, dictionaries or frequency profiles

The information mining APIs support the following operations:

- Summarization
- Categorization
- Language identification
- Information extraction
- Clustering

The created metadata for all the documents is stored in the information mining data store.

Once the data store has been filled, one more option for performing document selection becomes available, namely the selection of documents based on information in this data store. The advanced search operation combines a textual query with a category, thus restricting search to documents that belong to a certain category.

## **Using information mining in a business environment**

The infrastructure of an organization that supports the implementation of any information mining technology usually comprises at least the following roles:

- A system administrator for IT in general, not necessarily restricted to information mining
- An application programmer
- A librarian or knowledge engineer
- People working with an information mining application (end users)

Depending on the nature of the application, you may also find the following more specific roles in addition to those listed above:

- A Web designer
- An architect or consultant

Figure 2 on page 39 illustrates these roles and actions.

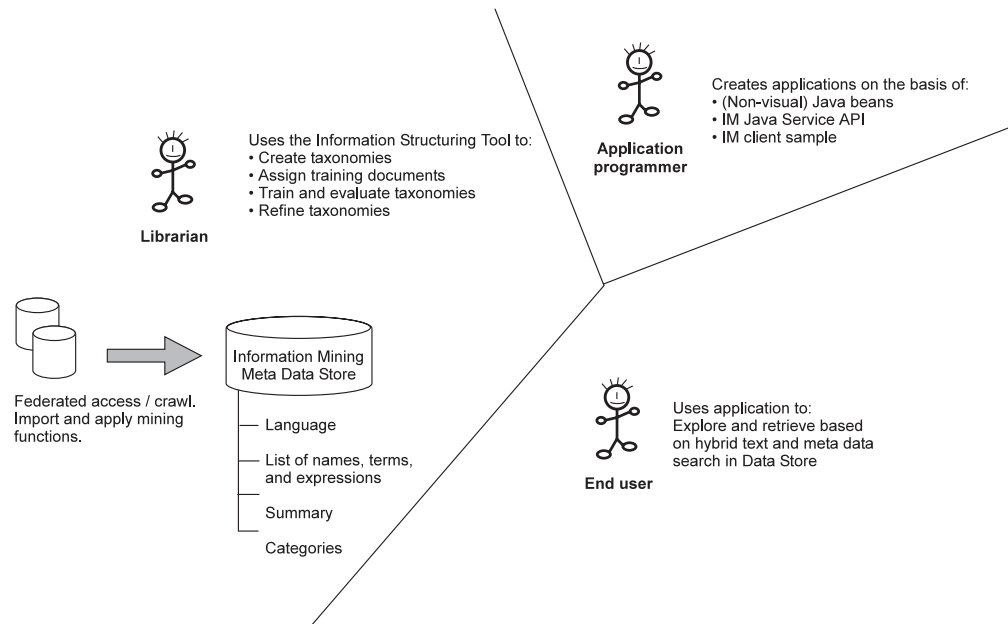


Figure 2. The information mining roles and actions

The *system administrator* sets up the hardware and software environment and maintains the required resources, for example, the file system space and access rights. The system administrator installs the required EIP components, and configures content sources and the EIP administration application so that access to the various repositories is available. The system administrator also administers the metadata store on a database level.

The *application programmer* creates an application using the JavaBeans or the Service API. See the *Application Programming Guide for Windows* for JavaBeans samples or for a description of the Service API. The application programmer may do this with the help of a *Web designer*.

The *librarian* or *knowledge engineer* is responsible for setting up and maintaining the document collections and resources to be used for mining and retrieval. The librarian uses the EIP administration application to create metadata mappings and search templates, and the Information Structuring Tool (see “Building a taxonomy” on page 54) to define catalogs and taxonomies. Populating the information mining metadata store with documents from data sources or the Web, using applications written by the application programmer, is usually also the librarian’s responsibility.

The *end users* work with the application created by the application programmer to perform information mining and retrieval tasks based on the resources which the librarian or knowledge engineer has created and maintains. Depending on the distribution of work between the end users and the librarian, the end users may also be involved in selecting documents from content servers and populating the information mining metadata store.

## An example of using information mining

Electro Corp. is a company that produces electronic devices for the mass market. Its portfolio covers at least five different products with a wide range of individual configurations.

The sales department has information on customer preferences for certain device application areas. These usage profiles explain how the customers use the products and the various configuration options. Each profile relates to a specific marketing, rollout, and relationship management strategy.

The services department has information on which parts make up the devices, how they are assembled, who the suppliers of these parts are, and on the maintainability and reliability of the individual parts.

The contract administration department keeps information on resellers and subcontractors. They also have access to legal documents related to the terms and conditions that are valid for specific types of contracts.

Lately, a marked drop in the sales to certain customers has been noticed. Competition between methods of applying the varying electronic devices has shifted and customer expectations have changed to meet these new technological advances.

To keep up with these changes and exploit them to its own benefit, Electro Corp. sets up a task force to develop a strategy that will move the company back into business.

The first step is to design an IT infrastructure to access this relevant information so that planners can make quick and well-informed decisions.

Examples of this kind of information are:

- Data on competitor products, features, prices, and customer acceptability
- Knowledge of the strengths and weaknesses of Electro Corps. product line from a customer's point of view
- Trends and perspectives in the areas in which these products are typically used

This information resides on disparate data sources, with different hardware and software platforms, levels of organization, for example, hierarchical, indexed, or flat file, and document types, for example, database records or HTML.

Figure 3 on page 41 illustrates using information mining.

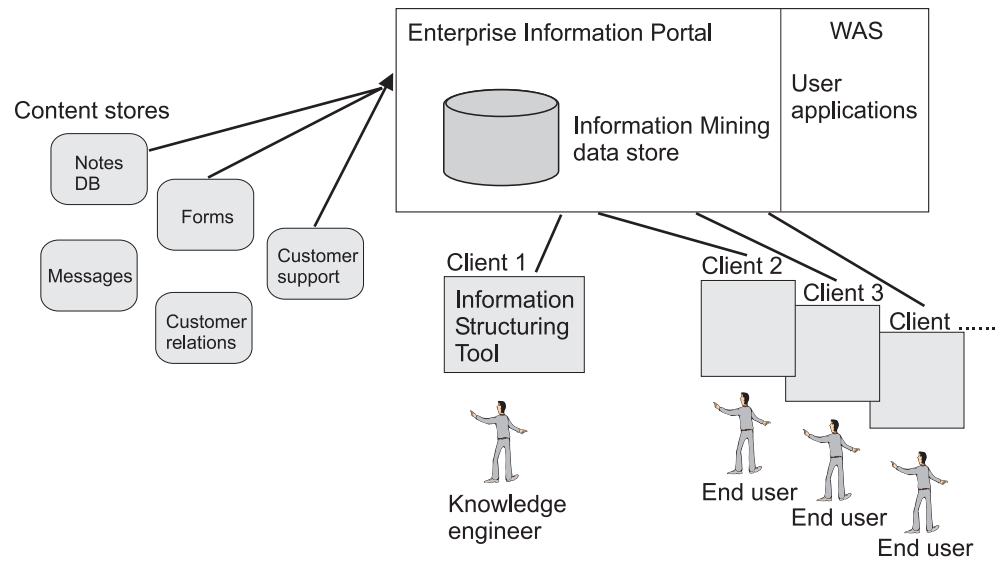


Figure 3. An example of information mining

The task force decides to create a portal to access all these disparate sources from a thin client (browser) that resides on the planner's client machine. They choose EIP since it provides all the building blocks to create such a portal and even allows for the creation of custom connectors to legacy data sources that were created when standards in the area of business applications were rare.

The following steps are involved:

1. Setting up the hardware and software infrastructure
2. Defining access methods to data sources, setting up the necessary connections, and creating mappings to the relevant data
3. Organizing the data in a way that planners can browse and navigate this data
4. Creating an end user application

Step 3 is where information mining comes into play. Once the infrastructure is available and running, the data sources have been identified, the corresponding connections established, and the relevant mappings created, the required data can be accessed from a single point, and subsets of this data can be defined using a federated search. The next question is how to filter out the required data for a certain forecast or trend, and how to organize this data in such a way that it fits the strategic planning process.

The task force defines a *knowledge engineer* who is responsible for maintaining, organizing, and updating the strategic planning information. To extract the relevant information from the huge set of documents residing on the various data sources, the knowledge engineer interviews the staff involved in past strategic planning to learn how the processes interact and what proved to be good practice, and searches the customer relations and support databases.

Using the search capabilities of EIP, documents from these databases can easily be accessed by customer name, address, or by device properties. However, the information needed to determine usage profiles is hidden within the text and the only way to get at this information is to analyze the document content in an intelligent way using the EIP information mining services.

A useful type of information provided by these services is the thematic characterization of a document's content called a category, for example, *this document is about PDAs*. The information mining categorization service assigns documents to categories by analyzing their content. Categories are structured in a topical hierarchy called a taxonomy. Both, the explicitly available and the automatically created metadata resides in repositories maintained by the information mining service called catalogs that help speed up access and retrieval.

Using the Information Structuring Tool, the knowledge engineer defines a catalog that illustrates the way in which the devices are used by customers.

Figure 4 shows a catalog.

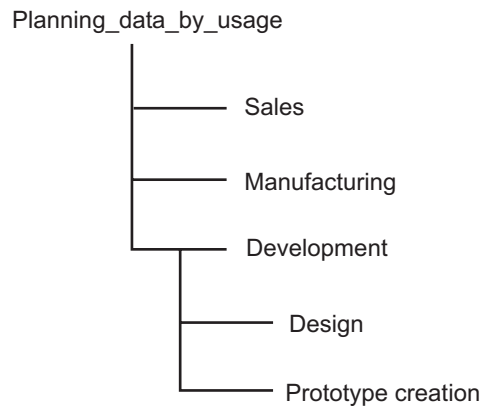


Figure 4. A catalog example

By performing a federated search on all customers and the way they use the devices in the sales and support databases, the knowledge engineer is able to identify sets of training documents that are representative of each of the categories in the taxonomy. This can lead to a re-organization of the taxonomy as new categories may show up when relevant data is looked at more closely.

Once the taxonomy is stable and each category has a sufficient number of training documents, the knowledge engineer trains the taxonomy using the Information Structuring Tool. Training creates a categorization model that can be used to assign documents to categories using the Categorization service.

In the meantime, programmers from the IT department create an end user application for the strategic planners using the EIP thin client and, either the JavaBeans or the Java Service API.

This application consists of a number of search templates customized for use in strategic planning. Using these templates, the planners can populate the catalog with documents retrieved from a mix of different content servers. When populating the catalog, the information mining service automatically assigns categories to the documents. If a new usage profile is identified, the knowledge engineer reorganizes the taxonomy accordingly using the new documents identified by the planners as training material. The catalog is retrained and the new results are passed on to the planners.

The above example illustrates that, by combining the information mining functionality to reflect a customer's expectations and needs, a company like Electro Corp. can keep abreast with shifts in the marketplace and so remain competitive.

## Supported languages and formats

Table 4 shows the languages supported by information mining services.

Table 4. The supported languages

	Language Identification	Information Extraction	Summarization	Categorization	Clustering
English	x	x	x	x	x
German	x		x	x	
French	x		x	x	
Danish	x				
Finnish	x				
Italian	x		x	x	
Norwegian	x				
Portuguese	x				
Spanish	x		x	x	
Swedish	x				
Korean	x		x	x	
Japanese	x	x	x	x	
Chinese (Traditional and Simplified)	x		x	x	

For a list of the supported document formats, see Chapter 9, “Document formats” on page 105.

---

## Concepts

The amount of information consumed is growing constantly. Most organizations have larger and increasing numbers of online documents which contain information of great potential value, for example, customer feedback data, strategic information vital in an increasingly competitive market, or information providing insights into new and changing business opportunities. The information mining services are designed to be used as applications that deal with large amounts of online documents.

## System architecture

Figure 5 on page 44 illustrates the information mining system architecture.

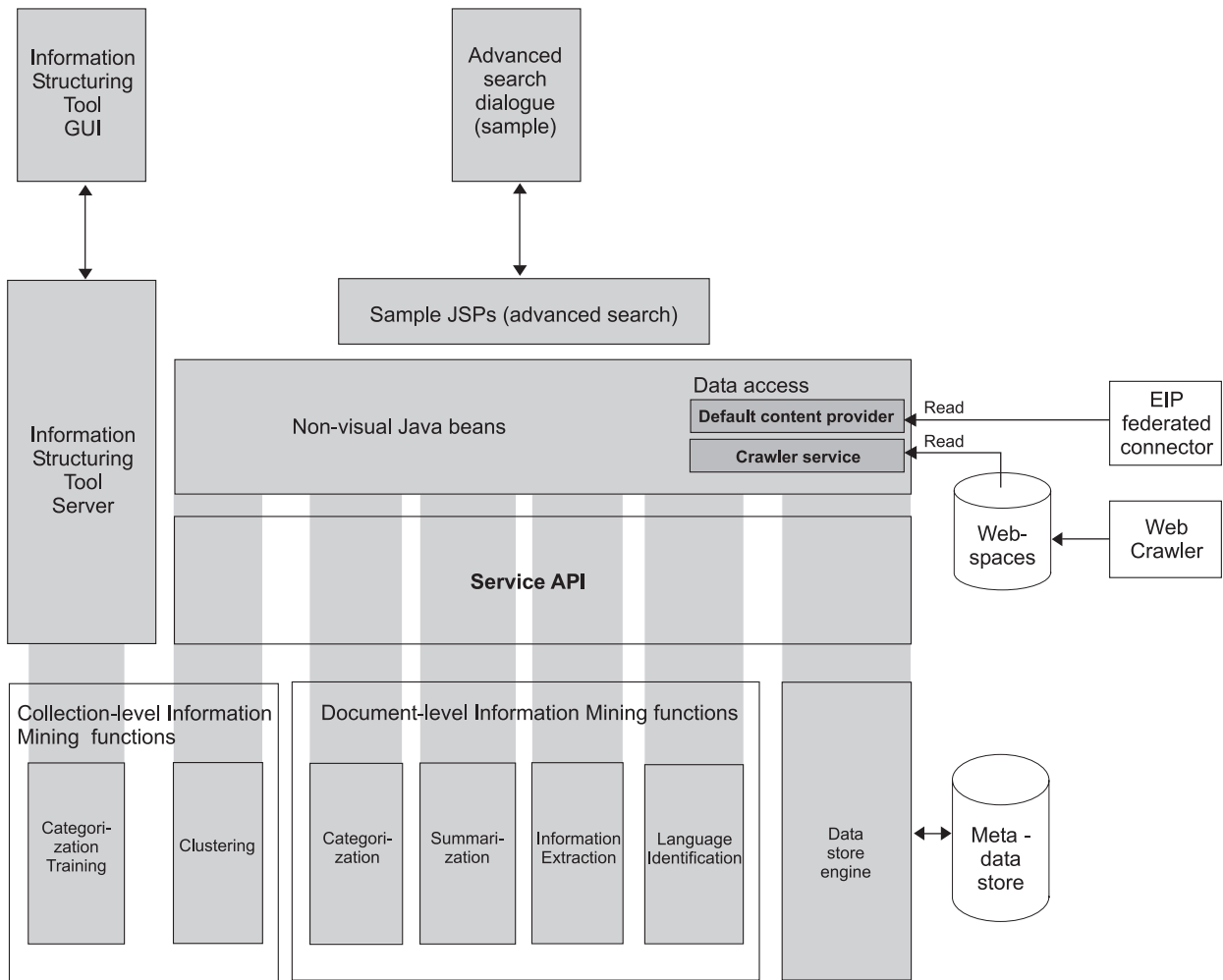


Figure 5. Information mining system architecture

The boxes on the right-hand side of the diagram refer to components that are used by the information mining services but are not considered to be part of the services, namely:

- EIP federated connector (part of the EIP OO API)
- Web Crawler

There are different layers of information mining functionality, namely:

1. Java Service API.  
This layer exposes the information mining functionality and metadata persistency as a consistent Java API.
2. Non-visual JavaBeans.  
This layer uses ready-to-use components based on the JavaBeans specifications applying event types and conventions from the standard EIP beans.
3. Sample Java Server Pages.



This level consists of sample code using the non-visual JavaBeans that illustrates an application for advanced search, in other words, textual search with category restriction.

#### 4. Information Structuring Tool

An application with a graphical user interface for creating and maintaining taxonomies.

## The information mining concepts

To fully understand and to be able to use the information mining functionality effectively, this section deals with the main concepts.

The information mining services provide an infrastructure for the creation and maintenance of information related to individual documents or collections of documents. This information about a document is referred to as its **metadata**.

The **library** is a conceptual view of the contents of the information mining database. The library contains a set of catalogs.

A **catalog** is the metadata store for text documents and contains:

- A **catalog schema** defining which attributes get stored for each document.
- A **taxonomy** which is a hierarchical tree structure of **categories**.
- A **categorization model** based on the document training results, that can be used to automatically assign categories to documents. This model is generated using the **Information Structuring Tool** where a taxonomy can be created and trained. The model serves as input to the categorization service.

The schema specifies the names and types of attributes that can be generated or stored for a document in the catalog. The schema is predefined and contains the following attributes:

- IKF\_CONTENT of type string
- IKF\_TITLE of type string
- IKF\_AUTHOR of type string
- IKF\_CATEGORIES of type string
- IKF\_SUMMARY of type string
- IKF\_LANGUAGE of type string
- IKF\_FEATURES of type string
- IKF\_COMMENTS of type string
- IKF\_DATE of type timestamp
- IKF\_IDNUMBER of type integer

The catalog creates a **record** according to the catalog schema to store information extracted or created from an imported document. A record has a unique identifier and a set of name value pairs. The unique identifier, known as a Persistent Object ID, or PID, links the created records back to the original document source.

Figure 6 on page 46 shows a sample record.

Record	
IKF_TITLE	"Birds"
IKF_AUTHOR	"J. Smith"
IKF_SUMMARY	"This is a summary of the book called Birds"
IKF_CATEGORIES	Birds/Insect eaters
IKF_DATE	07/01/2001

Figure 6. A sample record

Once a record has been created, it is stored in the catalog by assigning it to a category. Normally, the categorization results can be used to select the appropriate category although the category can also be chosen depending on another value stored in the record. Records have to be assigned to a category as this also includes indexing the document content to enable text search. Each catalog has a single text index which means that all search results are always automatically within the catalog search scope.

The **data store engine** is the component that maintains access to the persistent data store.

## The information mining tools

The information mining services provide functions to deal with online documents. These include the following:

- The Information Structuring Tool creates and maintains catalogs.
- The Language Identification service automatically detects the language in which a document is written.
- The Categorization service automatically assigns documents to categories that you have previously defined using the Information Structuring Tool.
- The Summarization service analyses words and sentences in a document to produce a summary of the document.
- The Information Extraction service recognizes significant items in text automatically, without requiring you to define domain-dependent vocabulary.
- The Clustering service divides up a set of documents into groups or clusters. The documents in each cluster share common features. The clusters are not predefined; they are derived automatically.
- The Advanced Search searches for text in documents stored in the catalog restricted to particular categories.

### The Information Structuring Tool

The Information Structuring Tool is a Web-based application that provides a means to create and maintain a set of catalogs called a library. A catalog is used to store metadata extracted from a document and is associated with a taxonomy used to organize documents with respect to a predefined organization. A taxonomy is a hierarchical structure of categories that classifies documents according to their thematic content.

For example, using the Information Structuring Tool, a librarian can define a catalog that illustrates the feeding habits of birds.

Figure 7 shows a sample catalog.

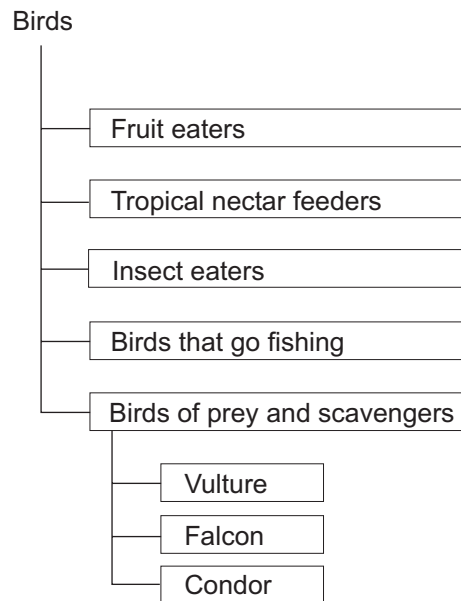


Figure 7. A sample catalog

A well-structured system of categories significantly helps to find relevant information in a mass of data. The categories are chosen to match the intended use of a collection of documents and have to be trained using sample documents beforehand. The categorization model created using the Information Structuring Tool can subsequently be used by the categorization service to automatically assign categories to documents.

The Information Structuring Tool functionality includes:

- Create, rename, and delete catalogs.
- Edit the description of a catalog.
- Create, rename, and delete categories.
- Add and remove training documents to and from a category.
- View the content of a training document.
- Start and stop the catalog training process.
- Obtain feedback on the quality of the training data in a catalog.

For detailed information on how to install and use the Information Structuring Tool, refer to “Building a taxonomy” on page 54.

## Language identification

The Language Identification service selects, for a given set of languages, the most likely language in which a text document is written.

The Language Identification service returns a ranked list of languages and confidence values for each document. The language key is specified in a two-letter code according to the ISO Standard 639. The confidence value is a measure indicating how well the document fits the language and is represented as a float number ranging between 0 (bad) and 1 (good). The language identification algorithm is designed to determine the language in monolingual documents. Hence, for multilingual documents, the ranking by confidence value cannot be guaranteed to reflect the correct language of the document.

The following languages can be detected:

- English EN
- German DE
- French FR
- Danish DA
- Finnish FI
- Italian IT
- Norwegian/Bokmal NB
- Norwegian/Nynorsk NO
- Portuguese PT
- Spanish ES
- Swedish SV
- Korean KO
- Japanese JA
- Simplified and traditional Chinese ZH

The following property can be set:

- **maxResults** (only when using the Java Service API):  
The maximum number of languages to be determined and returned for each document. It is an integer value greater or equal to 0. The default value is 1, meaning the best ranked result is returned. If the value is set to 0, all of the recognized languages are returned in a ranked order, except if the confidence value for the language is below 0,01.

You can use language identification as a preprocessing step for the other information mining services. For example, you might use language identification to find all the English or Japanese documents before doing information extraction.

## Categorization

Categorization is a means of assigning categories to documents and organizing documents with respect to a predefined organizational model created using the Information Structuring Tool.

Therefore, before you can use the categorization service, you must define and train a taxonomy to create such a model using the Information Structuring Tool.

The categorization result contains the category and a confidence value indicating how well the document fits the category. A set of such results is returned for each document. The list is ranked according to the confidence values returned.

The following properties can be set:

- **maxResults:**  
The maximum number of categories to return for each document. The default value is -1, meaning all of the categories are returned. The result list is ranked.
- **minConfidence:**  
The confidence value assigned to a document indicates how well the document fits the category. The parameter minConfidence specifies the minimum value ranging between 0 (less well suited) and 1 (better suited). The default value is set to 0, meaning that all of the categories assigned to a document are returned. The result list is ranked from better suited to less well suited.
- **catalogName:**  
This specifies the catalog to be used for categorization. A catalog can be created and trained for categorization using the Information Structuring Tool.

Refer to Table 4 on page 43 for a list of the supported languages.

### Summarization

A document summary consists of a collection of sentences extracted from a document that is characteristic of the document content. The Summarization tool can, for example, help you decide whether a document is relevant and should be read completely or, when returned as part of the result of a query, to help you decide whether to follow the document link.

The summarization result contains the summary as a single string and as a data structure (a matrix) that can be used by an application to select individual sentences and determine if they occur adjacent to one another.

The Summarization service can be used in different modes. The modes determine how the values `maxLength` and `ratio` are used to determine the length of the summary.

- **maxLength:**

The maximum number of sentences in the summary. The created summary will not be any longer than `maxLength`. The default value is 3.

- **ratio:**

The number of sentences with respect to the total length of the document. The length of the created summary is determined by the overall length of the document. The default value is 0.1.

- **mode:**

Determines the relation between `maxLength` and `ratio` needed for setting the length of the summary. The different modes are:

- **MODE\_LESS\_THAN\_MAXLENGTH:**

The summary has at most `maxLength` number of sentences. This is the default mode.

- **MODE\_EQUALS\_RATIO:**

The summary has exactly `ratio` number of sentences. This is determined by multiplying `ratio` by the total number of sentences in the document.

- **MODE\_EQUALS\_RATIO\_BUT\_AT\_MOST\_MAXLENGTH:**

The summary has at least `ratio` number of sentences (`ratio` times the total number of sentences in the document), but not more than `maxLength` number of sentences.

Refer to Table 4 on page 43 for a list of the supported languages.

### Information extraction

An important task when analyzing documents is the extraction of items that provide information about the document content. These key elements can be used:

- To indicate important information to help evaluate whether a document is of interest
- To find and store key concepts to use for query refinement
- As criteria for collecting documents that are related

Examples of key elements are vocabulary items, such as words, names, or multiword terms.

For English, the Information Extraction service normalizes key elements that it finds and groups together occurrences of such key elements in a text if they refer to the same entity or express the same concept. For example, if James J. Smith, Mr.

Smith, James and Smith occur in a document, they are all marked as referring to the same person by mapping them to the same normalized form. Inflected forms of words are also mapped to their normalized form, for example, children to child.

For Japanese, however, all key elements are extracted as they occur in the document. No normalization is done except for date, time, and currency expressions, where normalization conforms to ISO8601 and ISO4217.

The Information Extraction service enables you to analyse documents for:

- Single and multiword vocabulary items, for example, announcement, value, product cycle
- Names of places, people, and organizations, for example, Washington, Bush, Data Management Academy
- Abbreviations, for example, MB (megabytes)
- Terms for dates, money, and numbers, for example, 11 Jan. 1958, 01/11/58, \$30, thirty pence, 4.5, 5000

You can specify three types of information to extract:

- Name
- Term
- Expression

Using the Java Service API, you can also identify the subtypes listed below and a confidence value stating how well the subtype fits the extracted vocabulary item. The confidence value ranges between 0 (bad) and 1 (good). The subtypes for type name, term, and expression include:

- **Name**
  - Place, for example, Montreal or London
  - Person, for example, Tim Brown
  - Organization, for example, Smith and Son
  - Unknown, for example, Smashing Pumpkins, Silicon Valley, CCTV (abbreviation with no full form)
  - Other, for example, AIS Plan, ISO Conference, Internet, Privacy Act Officer, JCAHO Performance Report
- **Term**
  - Unspecified term, for example, entertainment conglomerate, art world, class variable, source code, data definition, process improvement initiative
- **Expression**
  - Cardinal, for example, four, fifty, 70
  - Ordinal, for example, fourth, fiftieth
  - Percent, for example, 12%, sixty percent
  - Date, for example, 07/28/98
  - Time, for example, 18 hrs, 4 o'clock
  - Money, for example, DM90, thirty pounds
  - Abbreviation, for example, NY

Information extraction only works on English-language and Japanese-language documents. You can use the Language Identification service as a preprocessing step to identify documents in your document collection that are not in English or Japanese. You can combine the Information Extraction service with other mining

functionality, for example, use it as a preprocessing step for the Summarization service to summarize only those documents about Bush as president and not as governor of Texas.

## Clustering

The Clustering tool arranges a collection of documents so that similar documents are grouped together, and documents in different groups (clusters) are distinct from one another with respect to their content. In this way, clustering can be used as a means of providing an overview of a large document collection and identifying related documents. It can likewise be used to support building a taxonomy using the Information Structuring Tool by clustering training documents within an application area. Clustering is also useful to find both similar documents within a collection that might point you to new trends or new technologies, and duplicates, or very similar documents, that might be interesting for competitive analyses.

Clustering is an iterative process that organizes documents into clusters so that the documents in each cluster are as similar as possible to one another with respect to their content, and the clusters are as different as possible from one another. Clustering works on a collection of documents as a whole, in contrast to the above information mining services, like categorization or summarization, which work at document level. Clustering works by comparing the representative features of each document with one another and grouping the documents according to their feature similarity.

During a clustering phase, no new documents can be added to the document set.

The following properties can be set:

- **maxClusterCount**  
The maximum number of clusters to be returned.
- **minClusterCount**  
The minimum number of clusters to be returned.
- **clusterFeatureCount**  
The number of labels (keywords) returned per cluster.

These values, however, are not binding for the clusterer, but merely serve as guideline boundaries. The output of the Clustering service is a result list.

Clustering only works on English documents. You can use the Language Identification service as a preprocessing step to identify the documents in your document collection that are not in English.

## Advanced search

In contrast to a standard EIP search which is carried out on the entire EIP Content Server, the so-called advanced search only searches on documents whose IDs are stored in a catalog created by the Information Structuring Tool. To narrow the search even further, the advanced search query not only searches for text but can also restrict this search to documents in particular categories.

The following parameters can be set:

- **catalogName:**  
This specifies the catalog to be used for search. A catalog can be created and trained using the Information Structuring Tool.
- **maxResults:**

The maximum number of search results returned for each query. The default value is 0 and means that all of the results are returned.

The following types of queries can be submitted:

1. Pure text query. This search returns all of the documents that match the text query. The result list is ordered by relevance.
2. Pure category search. This search returns all of the documents that are assigned to the category. The results are in an arbitrary order.
3. Combined text and category search. This search returns all of the documents that match the text query and are assigned to the category. The result list is ordered by relevance.

Advanced search queries submitted against the system are always bound to a specific catalog. This is referred to as the catalog search scope. No cross-catalog search is possible because catalogs represent a view on the imported documents that must be respected.

The BNF (query syntax) for a query string is as follows:

```
query_string ::= term
term ::= '(' term ')';
term ::= single_term | compound_term
compound_term ::= single_term binary_bool_op single_term
compound_term ::= unary_bool_op single_term
single_term ::= category_term | CLOB_term | string_term | number_term
CLOB_term ::= '(' '''attribute_name''' CLOB_operator string_value ')'
string_term ::= '(' '''attribute_name''' string_operator string_value ')'
number_term ::= '(' '''attribute_name''' basic_operator number_value ')'
category_term ::= '(' '''DKIKFCategory''' category_operator category_path ')'
category_path ::= '''category''' | '''category '/' category_path'''
category_operator ::= '=' | '>='
binary_bool_op ::= AND | OR
unary_bool_op ::= NOT
string_value ::= "string"
CLOB_operator ::= CONTAINS
string_operator ::= LIKE | CONTAINS | basic_operator
basic_operator ::= '=' | '>' | '<' | '<=' | '>=' | '!='
```

- The terminals, string and number, represent common terms.
- The category\_operator '=' restricts the search scope to only one category.
- The category\_operator '>=' expands the search to this category and all its subcategories in the category tree.
- The string search in a CONTAINS clause may include the wildcards ('\_') for a single character and ('%') for an arbitrary amount of characters. For example, \_LOB may match BLOB and CLOB, whereas %name could match filename. Only schema attributes that have been flagged as searchable, for example IKF\_CONTENT, can be queried using the string operator CONTAINS.
- The search string in a LIKE clause can also include wildcards as they are used in SQL.
- For a complete list of the currently supported attribute names, refer to “The information mining concepts” on page 45.

Query examples:

- Pure text queries:

```
("IKF_CONTENT" CONTAINS "southern Africa") AND NOT
("IKF_CONTENT" CONTAINS "Cape")
```
- Pure category queries:

```
("DKIKFCATEGORY" >= "birds/Fruit eaters")
```



- Combined text and category query:  
 ("IKF\_CONTENT" CONTAINS "'South Africa'") AND  
 ("DKIKFCATEGORY" >= "birds/Birds of prey and scavengers/Falcon")
  - Attribute query:  
 ("IKF\_SUMMARY" LIKE "humming birds in the tropics")
- or
- ("IKF\_FEATURES" LIKE "Goethe") AND ("IKF\_TITLE" = "Faust")

## Programming interfaces

The information mining functionality is available to build applications as:

- Java Service API
- Information mining JavaBeans

The **Java Service API** integrates the full information mining functionality, except catalog maintenance, which is part of the Information Structuring Tool, as an EIP service. It provides client/server communication based on Java RMI.

An application using the Java Service API can:

- Determine the language a document is written in
- Create summaries of text documents
- Assign categories to documents
- Extract information, for example names, terms or expressions, from a text document
- Group together similar documents
- Store and look up metadata for documents in a catalog
- Perform a text search on documents restricted to certain categories and on attributes, for example a summary

The Java Service API is able to run in local mode, using direct method calls, or in remote mode, using Java Remote Method Invocation (RMI). Running remote enables you to configure one server as the application server running your Web applications and another server as the information mining server that performs text analysis, indexing, and search. The server task mechanism is the means by which complete tasks can be sent to the information mining server (remote machine) and all processing is carried out on that machine.

When using the Web Crawler, the corresponding access mechanisms have to be implemented using the JavaBeans. Web Crawler access is not available on the Java Service API level.

For a detailed description of the information mining Java Service API, refer to the *Application Programming Guide for Windows*.

Figure 8 on page 54 illustrates the information mining remote configuration.

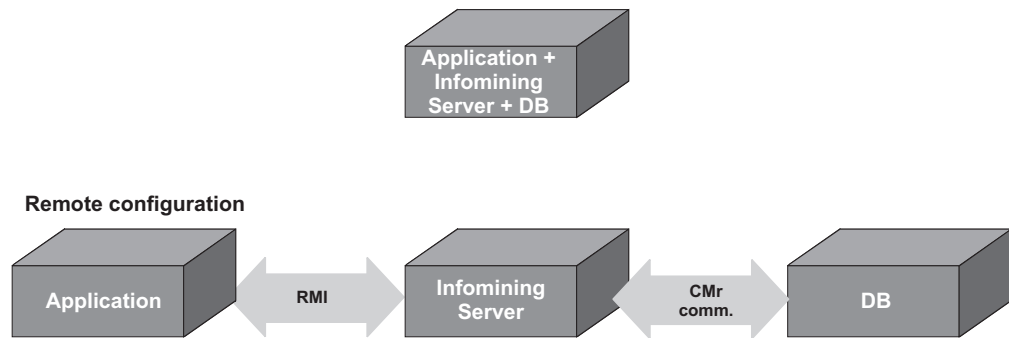


Figure 8. Information mining remote configuration

The **information mining beans** are a high-level Java API for rapid application development and are built according to the JavaBeans specification. The beans do not support server task processing, so, for performance reasons, all application development using the beans should be done on the same machine.

Each of the beans can be used event-driven and some provide methods that can be called directly. Integration with existing EIP beans is done through support of the result event used by the EIP beans. This means that an EIP federated search and Web Crawler result events are compatible with the information mining services and information mining results can be processed by EIP. For a detailed description of the information mining beans, refer to the *Application Programming Guide for Windows*.

---

## First Steps

The EIP information mining *First Steps* is a storyboard tutorial that shows knowledge engineers, administrators, or application programmers how the IBM information mining technology can be applied in a business environment based on a realistic scenario. The tutorial is structured as follows:

- Short introduction
- What needs to be done before the First Steps can be started
- Organizing the sample data
- Accessing this data
- Using the sample client
- Removing the sample data
- Further reading references

To access the Information Mining First Steps, execute  
 <CMBROOT>\ikf\firststeps\first\_steps.html.

The EIP Information Mining First Steps also serve as an installation verification.

---

## Building a taxonomy

The Information Structuring Tool is a Web-based application that provides a means to create and maintain a set of catalogs, called a library. A catalog is used to store metadata and is associated with a taxonomy that organizes the information in the catalog. The taxonomy is a hierarchical structure of categories that classify documents according to their thematic content. For example, the top level of a taxonomy could contain categories like *business*, *culture*, *sports*, while *sports*

subdivides into *team sport* and *athletics* at the next lower level. One level further down *team sports* subdivides into *soccer, baseball, tennis*.

By assigning training documents to categories and training a catalog, the Information Structuring Tool creates a categorization model that can subsequently be used by the categorization service bean to assign categories to a document.

## Installing the Information Structuring Tool

The Information Structuring Tool must be deployed as a Web application in a servlet container, for example, the IBM WebSphere Application Server (WAS) servlet engine.

Note, however, that no two Information Structuring Tool Web applications (for example, one named IST1, and the other IST2) are permitted to work against the same information mining instance.

Before deploying the Information Structuring Tool, ensure that WAS Version 4.01 (and not lower) is installed and running.

The user access rights required for the deployment of the Information Structuring Tool are:

- For Windows: Administrator authority
- For AIX: Root user privileges

For information on how to configure the WebSphere Application Server for the Information Structuring Tool, refer to *Planning and Installing Enterprise Information Portal*.

## Getting started

The application provides a Web-based interface where taxonomies can be defined, maintained, and trained. There are two frames. The left-hand frame is called the *catalog* view and is used to create and maintain the taxonomies. The right-hand frame is called the *notebook* and provides information on using the application. The notebook displays a series of tabs used to train and evaluate documents for the catalog. To use the application, catalogs and categories must first be created in the left-hand catalog view frame.

## Access rights

In the Information Structuring Tool, the user name and password information is maintained by EIP. You can only enter a user name and password known to EIP.

When you start the Information Structuring Tool, a security warning message appears because the Java Applet used for uploading training documents requires read access to your file system. If you decline, you will no be able to upload training documents, and hence not be able to train taxonomies.

The Information Structuring Tool can run in a multi-user environment. To allow multiple users to view the same taxonomy, the Information Structuring Tool provides a locking mechanism to control access to a catalog and its categories.

A user can choose to explicitly lock a catalog by selecting and locking it before starting to work with the catalog, or can start working with the catalog, that is, adding or renaming categories, and adding training documents, in which case the

catalog is automatically locked to prevent access conflicts. Other users can view this catalog but cannot change anything in the catalog until it is unlocked again by the user that locked the catalog.

Note, that, if the application server the Information Structuring Tool is deployed in, shuts down, all locks are deleted.

## Defining a taxonomy

A catalog is the anchor point of a taxonomy which is a tree-like structure consisting of categories.

The steps to define a new catalog and select appropriate categories are:

1. Decide which categories to define and create a new taxonomy.

In the catalog view, select **Library** and click the right mouse button. A menu is displayed. Select **new catalog**, and a catalog icon is created. Rename this icon by entering the catalog name and click **Enter**. A folder is created with the same name. This is the root category. The content of the notebook changes.

Another alternative method to add a new catalog to the Library is to import an existing taxonomy created outside the Information Structuring Tool, for example, in the file system. Refer to “Uploading training documents” on page 58 for more information.

To actively work on a taxonomy, the catalog must be locked for you. When you create a catalog, this is done automatically. To lock an existing catalog, select the catalog, and click the right mouse button. A menu is displayed. Select **lock catalog** and the catalog status icon changes. Icons are used to display the different types of catalog status:



The taxonomy tree is collapsed and is not locked by any user.



The taxonomy tree is expanded and is not locked by any user.



The taxonomy tree is collapsed and is locked by the current user.



The taxonomy tree is expanded and is locked by the current user.



The taxonomy tree is collapsed and is locked by another user.



The taxonomy tree is expanded and is locked by another user.

Figure 9 on page 57 is an example of two catalogs.

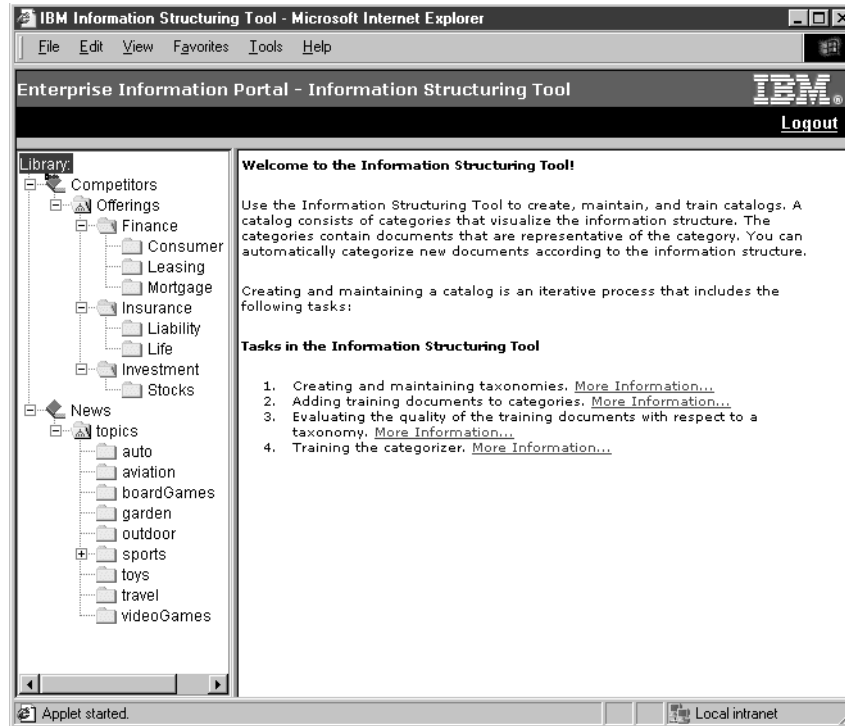


Figure 9. An example of a catalog

2. Once the catalog has been created and locked, categories can be added, renamed, or deleted. When you first create a new catalog, a root category with the same name as the catalog is also created. You can rename this category. To do this, select the category and click the right mouse button. Select **rename**, type in the new name, and press **Enter**.

To add new categories, highlight the category you want to add a new subcategory to and click the right mouse button. Select **new category**, type in the new name, and press **Enter**. Category names at the same level in the tree structure must be unique.

The root category cannot be deleted. It is only deleted when the catalog is deleted. Deleting a category also removes all subcategories (lower categories in the tree structure) that may contain training documents and all records.

3. Further descriptive information related to the selected catalog is displayed in the **Properties** tab. The first time, the description field is empty. To add or edit a catalog description, click **Edit Description**. A window is displayed, in which a description can be entered.

## Selecting training documents

The quality of a categorization model is strongly dependent on the quality of the training documents assigned to each category.

The training documents must be in one of the document formats supported by EIP. See Chapter 9, "Document formats" on page 105 for a list of the supported formats.

Selecting an appropriate set of training documents is essential. Documents should:

- Be representative of the category
- Contain a significant amount of descriptive text, without too much markup or lists of words

- All be written in the same writing style, for example, avoid prose writing if the other documents are in report style
- All be about the same length, and preferably not too long; the training documents should also be about the same length as the documents you want to categorize using the Categorization service

A collection of about 40 training documents per category is recommended; however, if the chosen category is more general, more documents are required. The categories must be chosen with care and be meaningful; categories and training documents that appear vague and indistinct to a human indexer will certainly pose problems during automatic processing.

## Uploading training documents

To add training documents, select the respective category and the **Training Document List** window is displayed. To add documents to this list, click **Add Document** .

The **Add Training Documents** window appears. You do not have to close this window after uploading files; it can be used for subsequent document uploads to another category or another catalog.

To add training documents, click **Browse** and select the relevant files or a directory on the **Open** window.

You can either select one or more files from the same directory or a complete directory to upload. If the directory you select is empty, you are notified.

This enables you to import and work with existing taxonomies which have been created outside of the Information Structuring Tool, for example, in the file system.

If the selected directory in your file system, for example, Development, contains a subdirectory Design, and the category Development in your taxonomy tree also contains a subcategory Design, the files in the subdirectory are added to the subcategory. If the subcategory does not exist, it is created and the files are added to this newly created subcategory.

Select the document language (English (US, UK), German, French, Japanese, Italian or Spanish) and the format for all the selected files. With the exception of plain text files, always use the format "automatic detection".

To add the files to the list of training documents, click **Submit**.

Figure 10 on page 59 shows adding training documents.

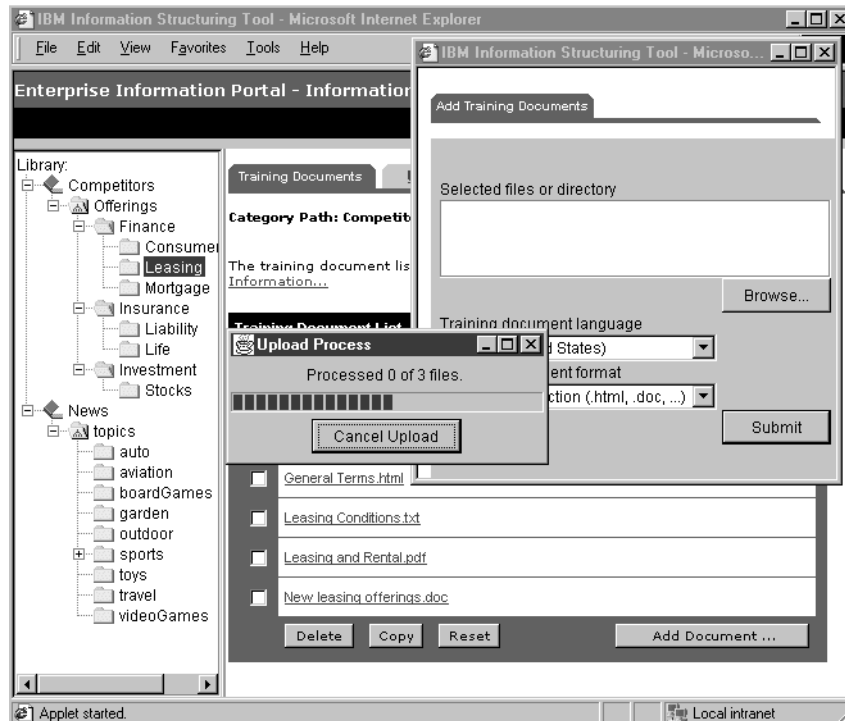


Figure 10. Adding training documents

An **Upload Process** window appears. You can cancel the document upload process from this window. If the upload process is cancelled, no more files are added. The files that have already been added as training documents are removed. However, if any subcategories were added during the upload, they are not deleted from the catalog.

If files cannot be uploaded successfully, an **Upload Status** window is automatically displayed with information on why the files could not be uploaded. For example:

- A file with the same name already exists.
- The file is empty.
- The file could not be uploaded to the server.

From the Upload Status window, press the **Training Documents** tab to return to the Training Document list. All the training documents that were successfully uploaded into the category are displayed here. Use the **Previous** and **Next** buttons to see all documents in the list.

If you want to add the same document as a training document to more than one category, upload the file into the first category and then copy it to the other categories. Do not upload this file again. To copy, select one or more documents in the Training Document List window and press **Copy**. On the window that appears, press **Browse** to select one or more categories you want to copy documents to and press **Submit**.

The following actions are not permitted during file upload:

- Unlock the catalog
- Logout from the Information Structuring Tool
- Start catalog training or evaluation
- Rename the catalog

- Delete the upload status information for the catalog

However, the following actions are permitted:

- Start another file upload process, either into the same category or a different one
- Work on another catalog

Figure 11 shows the list of training documents.

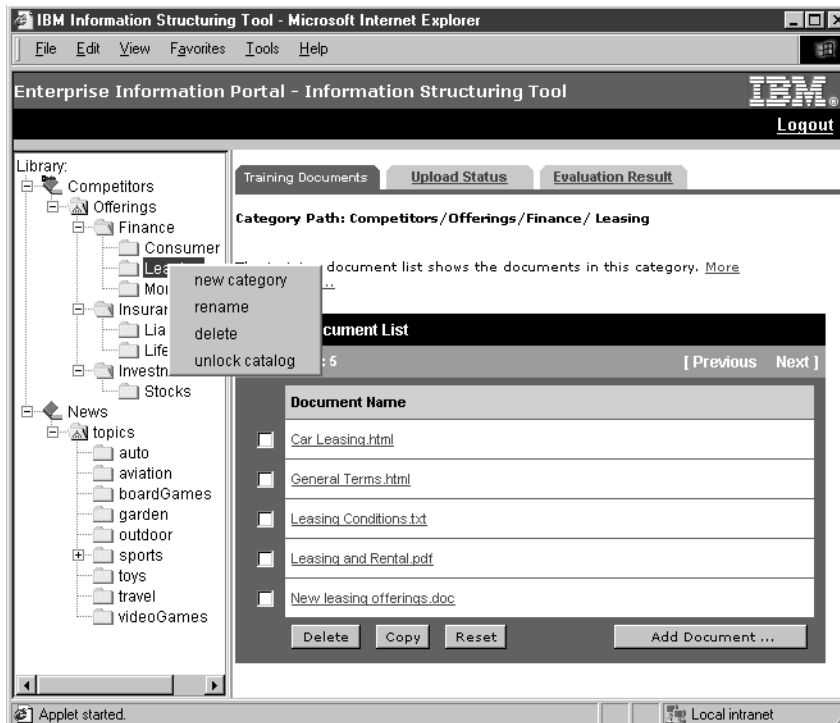


Figure 11. List of training documents

If the browser is closed during file upload, the files will be added as training documents if they have already been transferred to the server. If not, no files are added. After you have uploaded all of your training documents, select the catalog to begin evaluation.

## Evaluating a categorization model

Once the taxonomy has been defined and training documents have been assigned to each category, the taxonomy should be evaluated. Evaluating your taxonomy helps you assess how good your training documents are with respect to your predefined taxonomy. This is an iterative process consisting of the following steps:

1. Start the evaluation
2. Assess the evaluation results
3. Make changes to the taxonomy or the training documents
4. Rerun evaluation

During each evaluation iteration, the evaluation process:

- Splits the training documents up into a training set (approximately 80% of the documents) and a test set (approximately 20% of the documents). It trains the catalog using the training set and uses the Categorization service on the test set.



- Checks if the documents are assigned to the correct categories with a sufficiently high confidence value. The range is between 0 and 1, whereby 1 means the document is ideal. You can set the confidence value. The default is 0.5.

You can select between three to five iterations. Three iterations is the default and gives a meaningful perspective of the strengths and weaknesses of the taxonomy. When you select five, all of the documents will have been in both the training and test set.

Press **Start Evaluation** to begin the evaluation process.

For each category, the following is calculated during each evaluation iteration:

- *Correct* documents. The number of training documents in category *c* assigned to *c* during evaluation.
- *Outbound* documents. The number of training documents in category *c* assigned to a different category during evaluation.
- *Inbound* documents. The number of training documents assigned to category *c* during evaluation but originating from a different category.
- *Unassigned* documents. The number of training documents in category *c* that were not assigned to any category during evaluation. These may include documents that were assigned to a category but fell below the given confidence value.

An overview of how the evaluation process is progressing at catalog level is displayed, showing:

- Evaluation status indicating if evaluation is still running or has stopped
- Last evaluation listing the date of the last evaluation
- Number of completed evaluation iterations
- Total number of iterations
- Average overall recall showing the percentage of correct documents, that is those originally allocated to the category as well as the inbound documents
- Average overall precision showing the percentage of documents in the category that are correct
- Correct documents, that is, the number of documents correctly assigned
- Misplaced documents, that is, the number of inbound or outbound documents
- Unassigned documents

Precision and recall are closely related to the assigned confidence value. If the confidence value is low, precision drops and recall increases, or vice versa. A high precision value means that many training documents have been correctly assigned; a high recall value, on the other hand, means that most of the training documents have been assigned to a category, in other words, there are no or very few unassigned documents.

### **Evaluation results**

Detailed evaluation results are obtained by pressing the **Evaluation Result** tab. If you select the catalog and press the tab, the results for the whole catalog are displayed; if you select a category, the results for the category are shown.

Figure 12 on page 62 shows the evaluation results at catalog level:

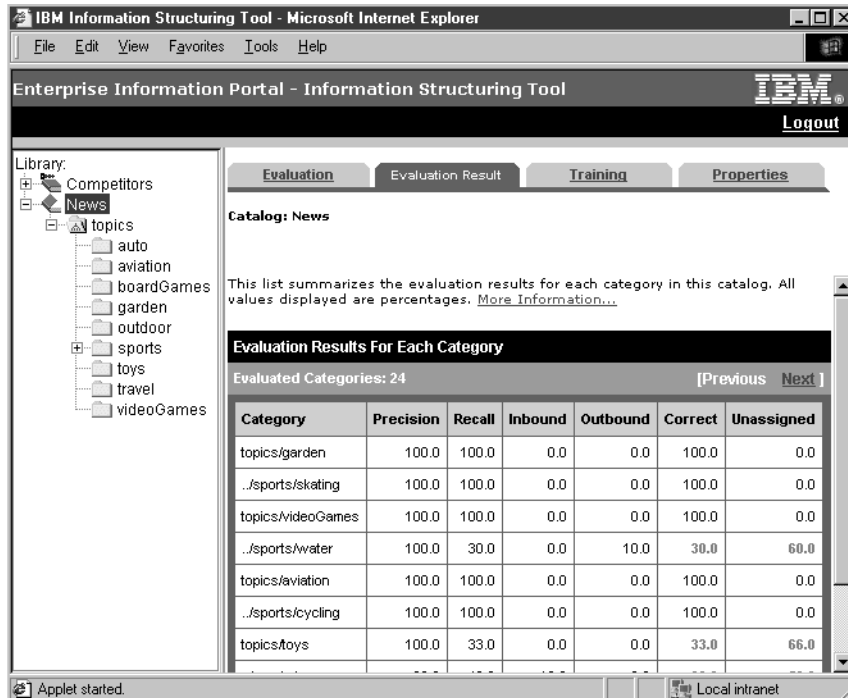


Figure 12. Evaluation results at catalog level

Begin with the overall catalog results. The values in red (critical) and blue (less critical) indicate that something is amiss, either with the category or the training documents. The quality of a category is defined by its training documents, hence it is advisable to study the movement of the documents to (inbound) and from (outbound) the category.

Changes made to a category or to training documents are strongly dependent on whether emphasis is placed on both precision and recall equally, or more so on precision only. The higher the precision value, the more distinctive the category is with respect to the others in the taxonomy. The higher the recall value, on the other hand, the fewer unassigned training documents there are.

The evaluation results are displayed at two levels:

1. **Catalog level:**

For each category in the catalog:

- The precision and recall percentages
- The percentage of inbound documents
- The percentage of outbound documents
- The percentage of correct documents
- The percentage of unassigned documents

2. **Category level:**

For document type inbound and outbound:

- The training documents and the originating or destination categories

For document type correct and unassigned:

- The training documents

## Interpreting the evaluation results

The following section suggests how you can interpret the evaluation results, but always bear in mind that the taxonomy operates as a whole, which means that changes made to one section of the taxonomy may have an adverse effect on the results produced elsewhere in the taxonomy.

Figure 13 shows the evaluation results at category level:

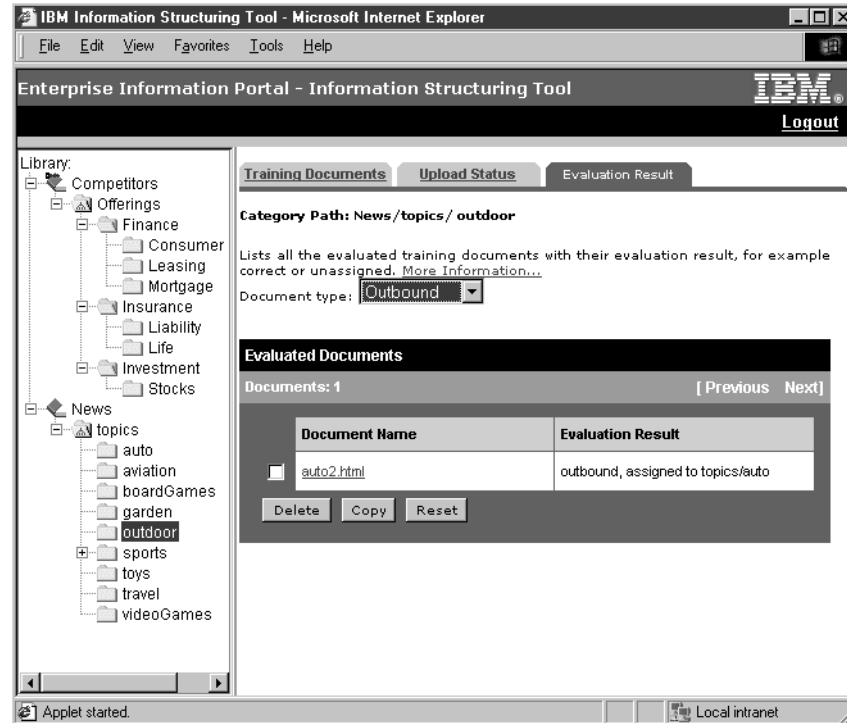


Figure 13. Evaluation results at category level

Beginning with the evaluation results at catalog level, select the categories with potentially low precision and recall values, and with training document values that have been marked in either red (more critical) or blue (less critical). For each category in turn (selectable on the left-hand frame), check the following:

- Many **inbound** documents:
  - Category gains documents from many other categories:
    - The category is not distinctive enough. Copy these documents to this category.
    - Make this category more distinctive by selecting better suited training documents or split the category up into subcategories.
  - Category gains documents from one or two categories:
    - Copy these documents to this category.
    - Check the originating category. Does it make sense to keep this category or should the categories be merged? To merge, copy all the training documents to the category you want to keep and delete the other(s).
- Many **outbound** documents:
  - Category lost documents to many other categories:
    - The category is not distinctive enough. Select better suited training documents or consider deleting the category.

- Category lost documents to one or two categories:
  - Copy these documents to the other categories.
  - Check this category. Does it make sense to keep the category or should categories be merged?
- Many **unassigned** documents:
  - Compare the unassigned documents with the correct ones for:
    - Size
 

If the unassigned documents are shorter, for example, consider joining two to make a new longer document to match the size of the correct documents.
    - Style
 

If the styles differ, delete the document.
    - Topic
 

If the topic is slightly different, yet still related to the topic in the correct documents, consider finding additional training documents that cover this topic and uploading these into the same category, or a newly created category.

If the recall and precision values are good, and the same documents remain unassigned during repeated evaluation iterations, delete them.

You only need to start the evaluation process again if you have made changes to the taxonomy, for example, have deleted a category or merged two, or you have moved many documents around. Smaller changes at category level, like adding a new training document or copying documents, can be made for several categories before you need to restart the evaluation process. Note, that there is no undo function if you want to reverse your changes.

Stop evaluating when there are no more red or blue values of interest to you, or when you have attained a precision and recall level of above 90%.

## Training a catalog

After you have evaluated your taxonomy and are satisfied with the results, it needs to be trained using all the training documents to produce a special type of metadata, namely a categorization model, which can subsequently be used to categorize new documents using the information mining Categorization service.

To begin the training phase, select the catalog which requires training, press the **Training** tab and then the **Start Training** button.

Displayed at the top of the right-hand frame is the name of the selected catalog. You can also see the training status of the catalog. The types of training status are:

- Documents cannot be categorized. Train the catalog to update the categorizer. (This is also the default for a newly created catalog. In this case, the **Last training** date is empty.) Either the catalog is new and has never been trained or the latest training results are invalid, because, for example, a category has been renamed or deleted since the catalog was last trained. An attempt to use the categorization service to categorize new documents will result in an error.
- The set of training documents in the catalog has changed. Documents are still categorized according to the last training results. Train the catalog to update the categorization.
- Training is running on the catalog.

- The catalog is up to date, no training is required.

To stop the training process, click on the **Stop Training** button. Note that, if training is stopped, the catalog training process has to be rerun.

You cannot upload new training documents to the catalog while it is being trained.

---

## Performance tuning

When you create records in a catalog, the text indexes for the text-searchable attributes, for example IKF\_CONTENT, grow very large and search performance decreases considerably. To optimize storage and increase performance, you need to reorganize the text indexes periodically, and especially after large updates have been made to the indexes.

Reorganization of a text index is best started at computer off-peak times, for example, at night. To run an index organization, switch to:

- For Windows: ...\\ikf\\IkfReorg.cmd
- For AIX: .../ikf/bin/IkfReorg.sh
- For Solaris: .../ikf/bin/IkfReorg

The parameters include: IkfReorg <UserID><Password><DBName>

---

## Using the IBM Web Crawler

This section describes and explains how to configure the IBM Web Crawler feature. The feature is installed by the EIP installation program if you select the Features checkbox.

EIP Version 8.1 contains a Web crawler, a Lotus Notes crawler, summarizers for extracting data from crawled files, HTML based documentation, configuration examples, and supporting utilities. IBM Web Crawler (also referred to as GCS) requires Java Version 1.3 or higher.

IBM Web Crawler is a Java-based content crawler and miner. When pointed at content, it acquires and mines that content.

IBM Web Crawler can crawl content in an intranet, extranet, or Internet web, in Lotus Notes databases natively or through Domino, in local file systems. IBM Web Crawler is built for easy addition of new protocols. The content can be of any type, for example, HTML, Notes attachments, and multimedia.

IBM Web Crawler can mine metadata and text from many types of content. For example, HTML content can be mined for:

- URL
- Title
- Body
- Time of last modification
- Meta tags such as author, keywords, description, and so forth

You select from a set of predefined miners for a given type of content. The content and/or mined metadata are saved to local disk. IBM Web Crawler can use Network Solutions Outside In technology to extract text from over 200 types of

content, an ideal partnership for use in search applications. IBM Web Crawler is also built for easy addition of new miners.

IBM Web Crawler is available for the Windows NT 4.0 and Windows 2000 operating systems. You can install, configure and use IBM Web Crawler in about a half-hour. It acquires and mines content at about ten files per second on a 500 MHz PC. It is tested to scale to 1 million objects (200000 Notes). It supports multiple users and multiple crawl/mine configurations per user, and is enabled to support the National Language preferred by the user.

## IBM Web Crawler capabilities

The installation program installs two files:

`x:<install directory>/run`

IBM Web Crawler for the Web batch files and sample configurations.

`x:<install directory>/notes-run`

Web Crawler for Notes batch files and sample configurations.

`x:<install directory>/lib`

IBM Web Crawler .jar and .zip and filtering files.

## Configuring and running IBM Web Crawler for the Web

This section describes how to configure and run IBM Web Crawler for the Web. IBM Web Crawler for the Web accesses HTTP, FTP, news or file servers and creates summaries of HTML documents and other objects. Summaries are files, one per document or object, containing metadata and full text.

### A basic configuration

This section contains instructions that explain how you can edit an IBM Web Crawler configuration file in XML format. Two sample configurations are provided to help you get started:

- A `config-db2.xml` file to use IBM Web Crawler with DB2 UDB.
  - A `config-sample.xml` file to use IBM Web Crawler without DB2 UDB.
1. Open a command prompt.
  2. Change directory to the run subdirectory where you installed IBM Web Crawler. For example, if you installed IBM Web Crawler on a Windows server, type `cd x:<cmbroot>\gcs\run`. If you installed IBM Web Crawler on AIX, type `cd /usr/lpp/cmb/gcs`.

**Tip:** It is very important to keep a copy of the original file. An error in the file can break IBM Web Crawler. Be careful while editing.

3. To run IBM Web Crawler with a DB2 UDB database (more scalable, slower), edit the `config-db2.xml` file. For example, type `edit config-db2.xml` in the command prompt.
4. To run IBM Web Crawler without a DB2 UDB database (less scalable, faster), edit the `config-sample.xml` file. For example, type `config-sample.xml` in the command prompt.

To run crawls of  $n$  URLs without a database, you need approximately  $n/1000$  MB of RAM on the machine to contain crawled URL metadata. For example, to crawl 500,000 URLs you need 512 MB of RAM. To take advantage of this memory, edit `crawlweb.bat` file and increase the value of `JVMXmx`.

### Configuring the IBM Web Crawler DB2 option

To configure the DB2 option you must create a database. This requires DB2 administrator authority. You may need to switch to the DB2 administrator account.

You can name the database as DB2 allows, but if your database name is not `gcs`, you must update the `dbname` in the Web Crawler configuration file.

If you have database administrator authority, you can run this command in a DB2 command prompt to create the database:

```
db -createdb <user><password>[database_name]
```

If you do not specify a database name, `gcs` is used. Once the database is created, add IBM Web Crawler tables by issuing the command:

```
db -createtables<user><password>[database_name]
```

IBM Web Crawler database and table creation must be complete to use DB2 with IBM Web Crawler.

The following configuration file settings (in the `urlpool-config` section) are required to use your new database:`dbname`:

- The name of the database (as created above): Example `gcs`.
- User name: The user name, for example: `db2admin`
- Password: The user password, for example: `db2admin`.

Set the database, username, and password properties to the appropriate values. Do not change the cache size or the driver. Continue editing the file to set the crawl scope for your system.

### Setting the crawl scope

These configuration file settings are required to establish the crawl scope regardless of whether you use DB2 or not.

Check the following settings in the `crawler-config` section and set these entries appropriately for your needs.

#### **seed list**

One or more starting absolute URLs. The URL must be available. Verify using your browser, for example: `http://www.<mysite>.com/`

#### **content-type-pattern-list**

Crawl URL found on pages only if any file extension matches these patterns, for example: `htm*`

#### **include-pattern-list**

Crawl URL found on pages only if they match these patterns, for example: `<mysite>.com`

You can also set these entries:

#### **recursion-depth**

The maximum distance in links to crawl from any starting point. Use `-1` for unlimited depth.

#### **exclude-pattern-list**

Crawl URL found on pages only if they do not match these patterns, for example: `*cgi-bin*`

#### **system properties**

To crawl through a firewall from an un-socksified machine you will also need to set the `socksProxy` values in this file.

## Starting IBM Web Crawler

If you edited the .xml configuration file, save it.

To start the IBM Web Crawler, use the crawlweb batch file and a configuration file. Open a command prompt and type:

- For Windows: crawlweb.bat<CONFIGFILE>
- For AIX: crawlweb.sh<CONFIGFILE>

To run with DB2 UDB, type: crawlweb config-db2.xml and press Enter. To run without DB2 UDB, type: crawlweb config-sample.xml and press Enter.

**Tip:** Plan to report crawl/summarize progress on a regular basis. As your targets are crawled, summaries are written to the location configured in summaries-dir. The default summarizers write the original object plus a metadata prologue as .html files in a tree. During or after crawling, you can examine the log-file for additional information.

## Advanced configuration

At this point you can study the configuration options. See the config-sample2.xml file in Chapter 7, “IBM Web Crawler sample files” on page 91 for a configuration example. The sample illustrates the configuration of:

- Crawler and summarizer threads
- The graphing monitor
- Logging options
- SOCKS
- Lotus Domino crawling
- Multiple content types
- More exclusions
- The use of InsoSummarizable to obtain summaries of objects such as .pdf files

See the config.dtd file for a formal definition of the available parameters in the configuration file. **Recommendation:** *Do not* edit this file. Make a copy of the file and rename the copy.

## The IBM Web Crawler configuration file

The configuration file is an XML file that tells IBM Web Crawler what web-based resources to gather and how to summarize them. This section describes each element and attribute that you can set in config.xml. For information on how to use IBM Web Crawler for Notes, see “IBM Web Crawler for Notes” on page 79.

IBM Web Crawler checks that the contents of the configuration file complies with gcs-config.dtd. If there are significant errors, for example, no URLs to crawl, IBM Web Crawler exits and prints an error message. For minor problems (an unknown attribute or value), the program will log a warning to the log file and continue.

**Recommendation:** Back up your configuration file before editing. An error in the file can break IBM Web Crawler.

Sample configuration files are shipped with IBM Web Crawler.

### <gcs-config>

The gcs-config file contains two sections: **globals** and **group-list**. See Chapter 7, “IBM Web Crawler sample files” on page 91 for an example of a gcs-config file.



**globals**

The `globals` element captures settings for IBM Web Crawler such as file system, performance, and network information.

**group-list**

The `group-list` element configures the crawl and summarization of groups, where a group is a set of resources such as a business or network domain.

**<globals>**

The `globals` element represents the global settings for IBM Web Crawler. The settings are encoded as global attributes and child elements.

The following list defines the global attributes. For definitions of global child elements, see “<logger-config>” on page 71.

**max-urls**

The maximum number of URLs to crawl. It should be a positive integer, and defaults to 100000.

**summaries-dir**

The directory in which to write the resource summaries. By default the `summaries/` directory is used.

**summaries-filepool-class**

The type of file pool used for resource summaries. This determines how the summary files are named and what subdirectory structure (if any) is used. By default, the `FullPathFilePool` is used, which creates a directory for the host, and then uses the same subdirectory structure and file name as the URL.

**num-crawlers**

The number of crawler threads to use. It should be a positive integer, and defaults to 20.

**num-summarizers**

The number of summarizer threads to use. It should be a positive integer, and the default is 5. Use these steps to configure `num-crawlers` and `num-summarizers`:

1. Set `crawlers` to the speed of your machine in MHz/20. For example, on a 600 MHz system, use 30.
2. Set the number of summarizers to 1/4th the number in step 1, for example, 8.
3. Do a trial run while observing the Windows Task Manager Performance panel. If the CPU *ever* goes for more than a second at 100%, return to step 1 and set a smaller number, for example, 3/4ths of the last time, until the CPU is basically never at 100%.

If, during your trial, you see that the text-monitor consistently reports Summarizer: ToDo numbers well below the number of configured summarizers, you may decrease the number of summarizers (the fewer the better) and increase the number of crawlers (the more the better) while not in violation of step 3. For best performance, use the fastest network you can and spread summaries, database, temp space and logs across separate disks, when possible.

**text-monitor**

When set to on, `text-monitor` prints the status of IBM Web Crawler every five seconds to standard out. When given a decimal value, `text-monitor` sets the time between refreshes (in seconds) of the text output. The default setting is off.

**graph-monitor**

When set to on, graph-monitor tells IBM Web Crawler to display its status using a graph GUI. When given a decimal value, graph-monitor sets the time between refreshes (in seconds) of the monitor GUI. The default setting is off.

**log-file**

Specifies the main log file to use. The default is log/log.txt.

**Tip:** You can specify additional logger info in the logger-config element.

**log-priority**

Sets the default log priority. Enter a value of info, warn or error. The default value is warn.

**Tip:** you can specify additional logger info in the logger-config element.

**temp-dir**

The directory in which to write temporary files. **Tip:** All files in this directory can be deleted by IBM Web Crawler. You should not need to change this from the default setting of x:/temp/gcs.

**temp-filepool-class**

The type of file pool to use for temporary files. **Recommendation:** Do not change this from the default setting of TempFilePool.

**content-dir**

The directory where IBM Web Crawler writes the content files. Normally, content-dir is the same as temp-dir.

**content-filepool-class**

The type of file pool to use for content files. Normally this is the same as the temp-filepool-class.

**how-often-to-gc**

The number of URLs to crawl between requesting garbage collection.

**Recommendation:** Define an integer  $\geq 50$ . The default setting is 100.

**max-resource-pool-size**

The maximum queue size of resources waiting to be summarized.

**Recommendation:** Define an integer  $\geq 10$ . The default setting allows 10 waiting resources per summarizer.

**connect-timeout**

Defines the milliseconds to wait before timing out a connect on the network. The default value is 4000. The valid range is 1000-60000.

**read-timeout**

Defines the milliseconds to wait before timing out a read on the network. The default value is 6000. The valid range is 1000-60000.

**cookies**

Defines whether to check for cookies in the HTTP header and store them in a database. The default setting is off. You can enable cookies by setting the value to on.

**locale** Defines the language to use for summaries and logging. The default value is en\_US.

Global child elements include logger-configs, a urlpool-config, and system-properties.

### <logger-config>

The logger-config file provides advanced control over what is logged, how it is formatted, and where the log file is written. The default log-file and log-priority are specified as global attributes. For more information on logging, see “Logging in IBM Web Crawler” on page 76.

#### category

The category of the logger being configured for example, `gcs.crawler`. If not specified, then the default logger is configured. Remember that settings to a particular category will affect all child categories.

#### priority

The minimum priority a message must have to be logged. If not specified, then this logger gets the priority from its parent category (and ultimately from the global default log-priority).

#### log-file

Defines where to write the log file. If it starts with a '+', then this log file will be used in addition to any other (parent) log files. If not specified, then the parent log file will be used (and ultimately the global default log-file).

**Tip:** Be careful not to specify the same log-file for multiple loggers, because they will overwrite each other.

#### log-layout

Defines the layout used for each message printed to the log file.

### <urlpool-config>

The urlpool-config file configures the component of IBM Web Crawler where URLs are stored. There are several options for the URL pool. You can store the pool in memory, you can use DB2, or you can use a special small memory version that does not store as much information about each URL. If you do not specify a urlpool-config element, the URL pool is stored in memory. The urlpool-config may have child urlpool-param elements, for specifying things such as database information.

#### urlcontainer-class

The type of URL container to use. Specify:

- DB2URLContainer to crawl with DB2 UDB
- MemoryURLContainer to crawl without DB2 UDB (default).
- BigMemoryURLContainer to crawl without DB2 UDB and use extra memory (stores some referring URLs and other information).

#### urlcollection-class

The type of URL collection to use. Specify:

- DB2URLCollection to crawl with DB2 UDB
- MemoryURLCollection to crawl without DB2 UDB (default).
- BigMemoryURLCollection to crawl without DB2 UDB and use extra memory (stores some referring URLs and other information.)

### <urlpool-param>

Used to pass parameters to the urlcollection-class. For an example, see the database connection information in the sample configuration using DB2 UDB in Chapter 7, “IBM Web Crawler sample files” on page 91.

**name** Defines the parameter name.

**value** Defines the parameter value.

**Tip:** Be careful using these parameters, because there is no error checking for them.

### **<system-properties>**

System properties represents a list of system property settings.

### **<property>**

For example, see the configuration for using a SOCKS gateway in the advanced configuration sample.

**name** The name of the parameter.

**value** The value of the parameter.

Alternatively, you can configure IBM Web Crawler access to external servers through a PROXY gateway using:

```
<system-properties>
  <property name="proxySet" value="true"/>
  <property name="proxyHost" value="proxy.hostname"/>
  <proxy port name="proxyPort" value="80"/>
</system-properties>
```

**Tip:** There is no error-checking on these parameters, so be careful using them.

### **<group-list>**

The group-list is a list of one or more group elements.

### **<group>**

The group element represents a single group of resources that will be crawled and summarized in a similar way. Each group must have a unique name attribute and at least one crawler-config child element that tell it what to crawl. Group can have a child summarizer-config element if you do not want to use the default summarizers. **Tip:** The overlapping groups (same URL is in two or more groups) may lead to unexpected results. Multigrouped URLs are associated only with the first group within which they are found.

**name** a unique name for this group (required).

### **<crawler-config>**

Use these rules to set the scope of crawling. The crawler retrieves each URL in the seed-list, parses URL from their content, and adds to the to-be-crawled list those URLs that match:

- at least one rule in the content-type-pattern-list
- and at least one rule in the include-pattern-list
- and none of the rules in the exclude-pattern-list.

The crawler-config also requires a single attribute: **recursion-depth**. The recursion-depth defines how many links radius from each seed the crawler can travel. The default is -1, which represents infinite depth.

### **<seed-list>**

This is a list of URL seeds, possibly with authentication information.

### **<seed>**

Seed represents a URL seed for starting the crawler, with a URL attribute and possibly authentication information. Each seed must be an absolute URL, for example `http://<your.server>.com/`. Avoid seeds that are redirected, unavailable,

or point to pages that are not text. It is useful to point to a page that you have edited to contain your seeds. Such a page is easy to update, to review, and test with a browser.

**URL** A seed URL to start the crawling at.

### **<authentication>**

Optional authentication sent for a seed URL protected by *Basic Authentication* as defined in rfc2617.

#### **username**

The user name used for authentication.

#### **password**

The password used for authentication.

For example:

```
<seed url="http://your.server.com/"><authentication username="me"
password="mine"/></seed>
```

### **<content-type-pattern list>**

This is a list of patterns for including content types to be crawled as identified by file extension. Any URL file extension (.html, .gif, .doc, and so forth) that matches any url-name-pattern in this list passes this test. URLs that do not have an extension pass the test by default. If the content-type-pattern-list is not specified or empty, then only URLs without a file extension are accepted.

### **<include-pattern list>**

This is a list of patterns for including URLs to be crawled, for example, by server or domain name. Any URL that matches any url-obj-pattern, url-regex-pattern, url-name-pattern, or url-predicate-pattern in this list passes the test. If the include-pattern-list is not specified or empty, then all URLs will be accepted.

### **<exclude-pattern list>**

This is a list of patterns for excluding URLs from being crawled. Any URL that matches any url-obj-pattern, url-regex-pattern, url-name-pattern, or url-predicate-pattern in this list will not be crawled. If the exclude-pattern-list is not specified or empty, then no URLs will be rejected.

### **<url-obj-pattern>**

This is a pattern for matching the different parts of a URL (protocol, host, and so forth.) with wildcards. It may be used in both the exclude-pattern-list and the include-pattern-list. The pattern for each part can have a '\*' wildcard at the beginning and/or end, which will match anything. However, it cannot have wildcards in the middle of any pattern. The matching is case insensitive. Any omitted URL part pattern is automatically matched.

The following list contains an example of how Java and IBM Web Crawler break up the URL `http://www.ibm.com/products/index.html?query#ref:`

- Protocol is: http
- Host is: www.ibm.com
- Port is: -1 (not specified)
- File is: /products/index.html?query
- Path is: /products/index.html
- Dir is: /products/
- Filename is: index.html

- Extension is: .html
- Query is: query
- Ref is: ref

The following list provides more details about each element of `url-obj-pattern`:

**protocol**

The wildcard pattern that a URL protocol must match, for example, `http`.

**host** The wildcard pattern that a URL host must match, for example, `*.ibm.com`

**port** The wildcard pattern that a URL port must match, for example, `80`.

**file** The wildcard pattern that a URL file must match, for example, `*.htm*`. The file part of a URL starts with the first slash after the host and may include a query but not a ref. The *file* part of `http://www.ibm.com/products/index.html?query#ref` is `/products/index.html?query`.

**path** The wildcard pattern that a URL path must match, for example, `*.html`. The path part of a URL starts with the first slash after the host and does not include a query or a ref. In the example, the *path* part of `http://www.ibm.com/products/index.html?query#ref` is `/products/index.html`

**dir** The wildcard pattern that the directory in a URL must match, for example, `/products/`. The directory is the part of the path starting with the first slash and ending with the last slash. In the example, the *dir* part of `http://www.ibm.com/products/index.html?query#ref` is `/products/`. It does not include a query or ref. Note that bad URLs that leave off the last slash, for example, `http://www.ibm.com/products`, will not match the *dir* correctly. In the example of the bad URL, the URL *dir* is `/`.

**filename**

The wildcard pattern that the filename in a URL must match, in the example, `index.html`. The filename is the part of the path following the last slash. In the example, the *filename* of `http://www.ibm.com/products/index.html?query#ref` is `index.html`. It does not include a query or ref.

**extension**

The wildcard pattern that a URL file extension must match, for example, `htm*`. It is preferred to use the `content-type-pattern-list` where possible.

**query** The wildcard pattern that a URL query must match.

**ref** The wildcard pattern that a URL ref must match (not used for HTTP). For example, `<url-obj-pattern host="*.ibm.com"/>` will match HTML pages on any IBM site.

**<url-regex-pattern>**

The `url-regex-pattern` is a pattern that matches a URL using a regular expression. It may be used in both the `exclude-pattern-list` or the `include-pattern-list`. It uses the `com.ibm.regex` package (`regex4j`), and has most of the functionality of Perl 5 regular expressions. It can have two regular expressions, one that the URL *must* match and the other that the URL *must not* match. Other options can be specified, such as `i` for case insensitivity. See `Regex4j Regular Expressions` for details.

**match** A Perl 5 style regular expression that a URL must match.

**no-match**

A Perl 5 style regular expression that a URL must not match.

**options**

Optional modifiers, such as `i` for case insensitivity.

For example, `<url-regex-pattern match="^http://www\.ibm\.com/.*\.html?$"/>` will match HTML pages on the main IBM web site.

**<url-name-pattern>**

This is a simple pattern with wildcards for matching a full URL or a URL file extension. It may be used in the `content-type-pattern-list` and in the `include-pattern-list` and in the `exclude-pattern-list`. It can have a `'*'` at the beginning and/or end of the pattern string, which match anything. However, it cannot have wildcards in the middle of the name. The matching is case insensitive.

For example, `<url-name-pattern name="*.ibm.com/*"/>` would match all files on the IBM site, but `<url-name-pattern name="*.ibm.com/*.html"/>` is invalid because there is a wildcard in the middle.

**name** Wildcard pattern that a URL string must match, with optional `'*'` wildcards at the beginning and/or end.

**<url-predicate-pattern>**

This pattern loads a Java UnaryPredicate class for matching a URL. It may be used in both the `exclude-pattern-list` or the `include-pattern-list`. The class must have a `public boolean execute(URL url)` method that returns true if the URL matches the predicate.

**class** The fully qualified UnaryPredicate class name.

**<summarizer-config>**

This is the configuration for a summarizer, with a list of child resource-handlers. There can currently be only one `summarizer-config` per group.

**<resource-handler>**

Determines what type of summaries are made for a resource (such as a web page or a newsgroup article), based on its content type, for example, `(text/html)` or filename extension `(htm)`. When a resource is ready to be summarized, IBM Web Crawler checks the resource-handlers, in order, and uses the first one that matches the content type or file extension. If none match, the `Copy2RdfSummarizable` and `Copy2RdfSummaryMaker` will be used by default. You can override this by adding a resource-handler with no content-type or file-extension at the bottom of the list.

A resource-handler can also have `summarizer-param` children that pass special parameters to its `SummaryMaker` class.

**content-type**

A wildcard pattern that the content-type of a resource must match, for example: `*htm*`

**file-extension**

A wildcard pattern that the file extension of a resource must match, for example: `htm*`

**summarizable**

The resource Summarizable class name, for example: `HtmlRawSummarizable`

### **summary-maker**

The resource SummaryMaker class name, for example:  
HtmlRawSummaryMaker

The content-type and file-extension patterns allow wildcards. A pattern can have a \* at the beginning and/or end of the pattern string, which match anything. However, it cannot have wildcards in the middle. The matching is case insensitive.

A resource-handler will match if both the content-type and file-extension patterns match, and an unspecified pattern will always match, so

```
<resource-handler content-type="*htm*"
summarizable="*HtmlRawSummarizable" summary-maker=
"HtmlRawSummaryMaker"/>
```

will match all files with content type text/html, regardless of the file extension.

For the summarizable and summary-maker, you do not have to specify the full path of the classes if the classes are in the com.ibm.IBM Web Crawler.summarizer.resource package.

### **<summarizer-param>**

These are special parameters that are passed to a SummaryMaker class. The usage is specific to that class.

**name** The name of the parameter.

**value** The value of the parameter.

**Tip:** There can be no error checking on these parameters, so be careful using them.

## **Logging in IBM Web Crawler**

This is an introduction to the logging features of IBM Web Crawler.

IBM Web Crawler provides powerful control over what gets logged, where it is logged to, and how it is formatted. For example, you can choose to write the response code of each page crawled to one file, IBM Web Crawler status (how many URLs crawled, how many threads working, and so forth) to another file, the URLs summarized to a third file, all IBM Web Crawler warnings to a fourth file, and all log messages in the net utilities package to another file for debugging.

See "IBM Web Crawler log analysis file example" on page 93 for a sample of a log analysis file.

### **Uses for logs**

Logging is useful for network/web/crawl/summarization accounting, communicating with other application components, and for IBM Web Crawler debugging.

Crawl and mining accounting can reveal a broad spectrum of interesting features, for example misconfigured servers, missing pages, and number of objects per content-type. The loganalysis.pl Perl script provides a sample of log summary accounting. Applications may need information from IBM Web Crawler, such as when content has been removed.



## Configuring the loggers

You can specify the configuration of one or more loggers in the IBM Web Crawler config file. The `log-priority` and `log-file` attributes of the `globals` element establish the default logging policy.

To extend the logging policy, create `logger-config` statements as children of the `globals` element. Each statement selects a subset of IBM Web Crawler log messages, routes them to a particular file, and writes them using a particular format. The subset of messages logged is selected using `priority` and `category` attributes. Legal `priority` values are: `trace`, `debug`, `info`, and `warn` (case insensitive).

- Setting the `priority` value determines how verbose a logger is, with `trace` being the most verbose.
- `trace` and `debug` are maintenance levels - messages are hard-coded in English.
- `info` and `warn` are user levels, with support for national languages.
- `info` produces many messages. Reduce message output by specifying a `priority` of `warn`.

## Logging configuration examples

### Log from/to hyperlinks, without date/time/thread info, to the file `log/fromto.txt`

```
<logger-config category="gcs.url.fromto" priority="info"
log-layout="%m\n" log-file="log/fromto.txt"/>
```

### Log summarized objects to the file `log/resources.txt`

```
logger-config category="gcs.summaries.list.resource"
priority="info" log-file="log/resources.txt"/
```

### Log URL skipped and the reason for skipping

```
<logger-config category="gcs.url.skipped"
priority="info" log-file="log/urls_skipped.txt"/>
```

### Log specially processed HTTP response codes

```
<logger-config category="gcs.http.302"
priority="info" log-file="log/urls_redirected.txt"/>
<logger-config category="gcs.http.404"
priority="info" log-file="log/urls_not_found.txt"/>
```

### Log all messages in the summarizer category, including their priority

```
<logger-config category="gcs.summarizer"
priority="TRACE" log-file="summarizer_trace.txt"
log-layout="%d: %t: %c: %p: %m\n"/>
```

## Troubleshooting

If you encounter problems, the first things to check are:

### Are the pages on your seed list reachable?

The pages must exist (avoid redirected seeds) and must be reachable from your system, and through SOCKS, if SOCKS is being used.

### Are the pages on your seed list regular HTML?

Frames, Flash, javascripts, and other such items are not good choices as seeds. Choose regular HTML pages.

### If you are using DB2 UDB, have you already crawled?

DB2 UDB keeps track of what you have crawled. If all the pages have been processed, DB2 UDB is silent. Use the `db -emptytables` command to start a new crawl.

**If you are using DB2 UDB, are your configuration file database access entries correct?**

if the database connection fails the crawl will fail.

**Have you carefully checked your configuration file edits?**

Errors can break IBM Web Crawler. Have you overlooked a restrictive max-urls or recursion-depth value?

**Still have a problem?**

Edit your configuration file and change the log priority to "debug". Now start Web Crawler again, and, after it stops, examine the log file.

## Choosing summarizers

The purpose of a summarizer is to take a resource (such as a web page) or a host (such as a web server) and produce a file that contains the information in which you are interested in a format that is easy to use.

IBM Web Crawler includes a variety of summarizers that handle different content types, extract different kinds of data from the resource, and output in different file formats. This section describes the features and requirements of the available summarizers. If none of these has the functions that you want, then you can also write your own summarizer.

There are two kinds of summarizers in IBM Web Crawler. A resource summarizer produces a summary of a single resource, such as a web page, and a host summarizer produces a summary of a host, such as a web server. For now, only the resource summarizers are configurable.

### Choosing a resource summarizer

The things to consider when choosing or writing a resource summarizer are:

- What is the input format? HTML web page, PDF, WordPro document, XML file,
- What metadata do you want to extract? HTTP header, title, annotated links, body text
- What output format do you want? XML, HTML, RDF

You specify which summarizer to use for a particular type of resource using the resource-handler element in the IBM Web Crawler config file. First you specify the content type and/or filename extension that the summarizer is used for. Then you specify the Java Summarizable and SummaryMaker classes that do the work. The summarizable class represents the resource to be summarized, and the summary-maker class represents the type of summary that will be made.

### Default summarizer (Copy + RDF Summarizer)

The copy + RDF summarizer is the summarizer that is applied to any object with a content-type not explicitly configured for processing by another summarizer. This summarizer can be used with any type of resource, and it writes two files. The first file is an exact copy of the original resource, and the second file is an RDF summary containing the original URL, the file name of the stored file, and HTTP header information. It can also be explicitly configured using the DefaultSummarizable and Copy2RdfSummaryMaker.

### Summarizers for HTML pages (raw HTML summarizer)

For HTML resources, the raw HTML Summarizer simply produces a copy of the file, with the URL and HTTP header information enclosed in a comment at the top. It is configured using the HtmlSummarizable and HtmlRawSummaryMaker.

```
<resource-handler content-type="*htm*"
    summarizable="HtmlSummarizable"
    summary-maker="HtmlRawSummaryMaker" />
```

### Summarizers for HTML pages (EIP HTML summarizer)

For HTML resources, the raw HTML Summarizer simply produces a copy of the file, with the URL and HTTP header information enclosed in a comment at the top. It is configured using the `EIPHtmlSummarizable` and `EIPHtmlRawSummaryMaker`.

```
<resource-handler content-type="*htm*"
    summarizable="EIPHtmlSummarizable"
    summary-maker="EIPHtmlRawSummaryMaker" />
```

### Summarizers for HTML pages (no write HTML summarizer)

This summarizer will crawl HTML and follow links, but not write any summary of the file. This might be useful if, for example, you want to crawl all of the PDF files on a site (using the INSO to XML Summarizer) but not store the HTML files. It is configured using the `HtmlSummarizable` and `NoWriteSummaryMaker`.

```
<resource-handler content-type="*htm*"
    summarizable="InsoSummarizable"
    summary-maker="InsoSummaryMaker" />
```

### Summarizers for other content types (INSO to XML summarizer)

This summarizer will create an XML summary for more than 200 types of resources, such as Microsoft Word documents, PDF files, PowerPoint presentations, and others. It has some meta information, and the body text is extracted by Network Solutions INSO filters (an INSO license is required). It is configured using the `InsoSummarizable` and `InsoSummaryMaker`.

```
<resource-handler content-type="pdf"
    summarizable="InsoSummarizable"
    summary-maker="InsoSummaryMaker" />
```

### Other summarizers

If you need to summarize other resource types, mine other data, or output in other formats, contact IBM or create a custom summarizer.

## IBM Web Crawler for Notes

This section describes how to configure and run IBM Web Crawler for Notes. IBM Web Crawler for Notes accesses Notes databases and creates summaries of Notes documents and attachments. Summaries are XML format files, one per document or attachment, containing object and full text.

### Prerequisites

The following prerequisites are required before running IBM Web Crawler for Notes:

- Lotus Notes Version 5.0.5 or later.
- PKZIP Version 2.50, if you want to handle attachment files that are self-extracting zip files.

### Performing a test crawl

Select **Start** → **Programs** → **Command Prompt**. In your new window, change directory to where you installed IBM Web Crawler for Notes and then change to the `notes-run` subdirectory. For example:

```
cd c:\<install directory>\gcs\notes-run
```

A Notes crawl is controlled by two files:

- A sources list that you edit to identify Notes databases that you can crawl. It includes Notes server names, IP addresses, `.nsf` filenames, and so on. For

example, a sources list might name 34 Notes databases; which ones are crawled is established in a configuration described below. The sources list can be either an .xml file or a Notes database (an .nsf file).

- A configuration file that specifies a sources list, which of the sources are to be crawled, what attachment types to process, output formats, and so forth. The configuration file is always an .xml file.

To verify that IBM Web Crawler is correctly installed, crawl its test database. Using an editor, ensure that the testSources.xml source list has the correct path and file name to the test.nsf database, which is in the notes-run subdirectory where you installed IBM Web Crawler. Make a backup copy of the original files.

**Recommendation:** *Edit carefully:* an error in the file will cause IBM Web Crawler to fail. Save any changes.

Test your installation by crawling the included test.nsf database. Type:  
crawlNotes crawlTestXml

The crawlNotes.bat file starts IBM Web Crawler with crawlTestXml as its configuration file; .xml is automatically appended to the configuration filename. IBM Web Crawler should report the crawl and summarize two documents, each with an attachment.

When IBM Web Crawler is finished, you can view the summaries in the summary directory, and the crawl log files in the log summary directory specified in the configuration file.

## Configuring a custom Notes crawl

After a successful test crawl you will want to crawl other databases.

1. Create your database source list. Add Notes databases to be crawled to a sources file.

To identify Notes databases to be crawled in an XML file, start by editing the testSources.xml file. To identify Notes databases to be crawled in a Notes database, use Notes to open and update the testSources.nsf Notes database. Parameters that you can set in sources files are explained in Editing Source Lists.

2. Set the crawler configuration. You will need to edit a configuration file in XML format.
  - If your sources are listed in an XML file, start by editing crawlTestXml.xml and set sourcesInXmlFile to point to your sources file.
  - If your sources are in a Notes database, start by editing crawlTestNsf.xml and set sourcesInNotesDB to point to your sources database. Parameters that you can set in configuration files are explained in “The IBM Web Crawler configuration file” on page 68.

Once your source list and configuration are complete, invoke IBM Web Crawler:  
crawlNotes your\_config

Or, if your source list is in a Notes database, start IBM Web Crawler as in the following working example: crawlNotes crawlTestNsf

When the IBM Web Crawler Notes crawler is done, you can view the summaries in the summary directory and the crawl log files in the log summary directory specified in the configuration file.

## Source list parameters

A source list contains descriptions of Notes databases that can be crawled. Source lists in file.xml format contain a notesDataSources element with one or more oneDBInfo elements. Each oneDBInfo element contains:

**id** A numeric id for this database. It is referred to by the range parameter in the configuration file.

**serverName**

The name of the server that serves the database. Use the null string "" for the local database.

**pathAndFileName**

The full path and file name of the database on the server. End the path and file name with .nsf.

**viewName**

The name of the Notes view of the database to be crawled.

**ipAddress**

Optional. The IP address of the server; if given, DNS is not used. If DNS cannot resolve the target server name, you can specify its IP address here. On Windows, the IP address can be determined using the nslookup server\_name command.

**dateLastCrawled**

Optional. The date on which the database was last crawled. It will be modified automatically unless you set update Date Last Crawled to no in your configuration file.

**tries** Optional. The number of times you want to try recrawling the database if the crawl does not succeed (time out).

**fieldSubstitutions**

Mappings specifying how Notes database field names are replaced in the output XML document. It contains one or more substitute elements, each with two attributes:

- Original: the field name which will be replaced in the output XML document if it exists
- Replace: With the new field name which will replace the original field name in the output XML document.

Source lists in a Notes database can be examined and updated using the Notes client. Start Notes and select **File** → **Database** → **Open**. Click **Browse** to locate and open the testSources.nsf database in the x:\<install directory>\gcs\notes-run directory. The shipped test database can also be examined and updated using the Notes client. Start Notes and select **File** → **Database** → **Open**. Click **Browse** to locate and open the test.nsf database in the x:\<install directory>\gcs\notes-run directory.

## Configuration file parameters

Parameters that can be set in a configuration file are described below. You can omit parameters with defaults listed.

A sourcesInXml file or a sourcesInNotesDB element identifies the sources list. Sources is an XML format file or a Notes database, respectively, that contains information identifying the database(s) to be crawled.

A runInfo element containing parameters controlling a single run. That is, these parameters apply to all Notes databases crawled in given use of the crawler:

**rangeSpecify**

The ids of databases to crawl. The ids are the numbers given in the id field of your sources list. It can be specified as a comma-separated list of individual ids and/or hyphen-indicated ranges, as in 1-4, 15, 25-31.

**SummaryDirectory**

Specifies the root directory for output summaries. The summaries are written into subdirectories of this directory.

**MaxThreads**

Specifies the number of parallel crawl threads. Each Notes database is crawled by a single thread. Multiple databases are crawled in parallel.

**doIncrementalCrawl**

Default is no. If yes, only process Notes documents new/modified since 'summarizeThisDateAndLater'. If 'summarizeThisDateAndLater' is not specified, then the crawler will use each database's own DateLastCrawled field specified in the database sources list. If doIncrementalCrawl=no, then all documents will be processed regardless of date.

**summarizeThisDateAndLater**

The format for this field is: MM/dd/yyyy hh:mm a tz, for example 01/01/2000 01:11 PM PDT. If no date and time is given, summarize all documents since the last crawl recorded in the sources list (if doIncrementalCrawl is set to yes), or since forever (if doIncrementalCrawl is set to no.)

**detachAttachments**

Default is yes. If yes, detach and summarize attachments. The types of attachment files to process are listed in the configuration file. If no, attachments are ignored.

**attachmentFilenameFormat**

Default is l (long). It can also be s (short). The long file name encodes the type, the server, the database name, and the Notes id. The short file name encodes the type and the Notes id.

**processAttachmentsAfterwards**

Default is no. If yes, attachment files are not summarized during Notes database summarization. Instead, a record is written to notesCrawl-attachments.bat for each attachment, specifying commands for summarizing the files. You write and run a batch file afterwards that summarizes and then deletes the attachments. Processing attachments afterwards typically requires substantial disk storage.

**saveAttachmentFiles**

Default is no. If yes, attachment file originals are not erased after processing. This option is only valid if processAttachmentsAfterwards is no. If processAttachmentsAfterwards is set to yes, disk space is required to store the saved attachments.

**MaximumNumberOfDetachingErrors**

Default is 10. The maximum number of errors in handling attachment, for example, running out of disk space when saving attachment files, that the crawler will tolerate before aborting the crawl.

**saveURLsToFile**

Default is no. If specified, URLs found in the Notes document items are written to a file of the name in the form:  
databasename(without path and .nsf) + ".html".

**updateDateLastCrawled**

Default is yes. If no, do not update the dateLastCrawled in the sources file.

**tempDirectory**

Default is c:\temp. This directory is used for writing any temporary files.

**logSummaryDirectory**

Default is log. The directory in which log files are saved.

**loggerPriority**

Default is info. These settings define the priority of the logger. The settings can be, from highest to lowest, error, warn, or info. For example, if the priority of the logger is set as warn, only log messages with a priority of warn and error will be logged.

An attachments element containing include elements identifying attachment file extensions to be processed, for example .prz.

## Excluding IBM Web Crawler from a server

For security and performance reasons, an EIP administrator might want to exclude certain servers or pages from being crawled. You might need to be able to limit the crawlers activities on your servers and pages.

You can instruct the IBM Web Crawler to avoid servers or pages using an *access policy* file. The file is built according to guidelines published in *A Standard for Robot Exclusion* (see <http://info.webcrawler.com/mak/projects/robots/norobots.html>).

- The IBM Web Crawler requests the access policy file `http://yourserver/robots.txt` before crawling a server and periodically thereafter.
- The file consists of lines in the form  
*field*:<optionalspace>*value*<optionalspace>

If *field* is User-Agent, and *value* is IBM-WebCrawler or \*, the following Disallow lines (through the next User-Agent line) specify partial addresses to avoid. This can be a full path or a partial path; any address that starts with this value is not retrieved.

For example:

```
Disallow: /help
```

disallows both /help.html and /help/index.html.

```
Disallow: /help/
```

disallows /help/index.html but allows /help.html.

An empty value allows all addresses to be retrieved.

- These lines can be separated by blank lines.
- You can include comments by typing a # character. The rest of the line is considered a comment.

Here are some examples:

- This /robots.txt specifies that all robots should avoid this server.  
# disallow everybody

```
User-agent: *
Disallow: /
```

- This /robots.txt specifies that only IBM Web Crawler can crawl this server, and that the crawler has no limits.

```
# allow only IBM

User-agent: *
Disallow: /
User-agent: IBM-WebCrawler
Disallow: # disallow nothing
```

- This /robots.txt specifies that all robots should avoid addresses within the temp, development, and testing htmldocs trees, that a user agent called IBM-WebCrawler has an exception for the development and testing trees (it is allowed there) and that the xyz and wxyz robots should stay away entirely.

```
# a more realistic example

User-agent: *
Disallow: /htmldocs/temp
Disallow: /htmldocs/development
Disallow: /htmldocs/testing

User-agent: xyz
User-agent: wxyz
Disallow: /

User-agent: IBM-WebCrawler
Disallow: /htmldocs/temp
```



---

## Chapter 6. Introducing workflow

You can use EIP workflow to control the flow and performance of work in your business. When users work with the results of federated searches, they often must make decisions on what actions to perform. You can use EIP workflow to determine in advance how you want users to perform the work.

You can automate the workflows by setting up profiles and rules that control the way workflow components work together. You also choose how restrictive to make your system by controlling user access and authority through privilege sets and access control lists.

---

### Understanding workflow

Most business operations can be characterized as a set of interrelated processes. Work flows from one employee to another, and from one department to another. Some simple processes might require only a few steps, while more complex processes involve a number of employees in different departments.

Workflow allows you to move work through a process and make decisions about work throughout the process. For example, XYZ Insurance receives large volumes of claims forms in the mail. During the verification process, insurance claims adjusters need to gather documents such as photographs, appraisals, and expert reports. Employees spend several hours each day opening, sorting, filing, and monitoring information, as well as collecting pertinent documents for final approval.

This information moves from one employee to another as the information is received and checked. As the claim is completed, it might be handled by employees in more than one department.

---

### How to use workflow

As in our XYZ Insurance example, most enterprises that handle documents perform some or all of the following tasks:

- File documents for later retrieval.
- Collect documents, forms, reports, and information from different sources, then deliver these documents to somewhere to be processed.
- Match incoming mail with documents currently being processed.

A *workflow* represents the flow of work. It describes the actions that can be performed on a group of one or more documents or content and the path this group of documents takes throughout the workflow. A workflow reflects work the way that it is performed, with a clearly defined scope and boundaries. It defines the sequence of activities and tasks, and the connections and relationships among those activities and tasks. A workflow determines the criteria that are used to make decisions about the flow of work. For information about the workflow creation process, see the *Workstation Application Programming Guide*. For information about using a client with workflow, see *Installing, Configuring, and Managing the eClient*.

---

## Planning a workflow

Before you begin to define a workflow, you must analyze the work that your business performs, where and how it is performed, and by whom. An administrator or business analyst does this planning step.

What is the final product? The final product might be the result of all the work accomplished by your business, by one department in your business, or by certain employees from different departments. For example, the final product of the claims compensation process of XYZ Insurance is the letter sent to the policy holder approving or rejecting the claim.

Analyze the information that must be processed to produce the final product, determine the actions that must be performed and where they are performed, and decide how you want the information to flow through the workflow.

### Information to be processed

Consider the information that must be handled by users in your enterprise. What types of input support the final product? What are the specific documents that must be processed?

A *work item* can be any content (documents or objects) from a content server. For example, XYZ Insurance initially receives claims forms and later receives follow-up documents, such as photographs, appraisals, and expert reports.

### How information is handled

Who can best handle each step of the process? For example, an administrative assistant might verify that a claims form is complete, then file the claims form until a certain document is received from the policyholder. When the document arrives, the claims adjuster might be responsible for matching the document with the claims form, and for approving that document.

The claims forms can be grouped into a *worklist* that is accessible to a number of claims adjusters. A *worklist* can be thought of as a queue of work that you create for one or more employees to use. The worklist is a filtered view of the work items. Employees only see items in a worklist that they are allowed to see.

Worklists can be defined to filter work items in a manner that handles each part of the claims process, such as gathering photographs, appraisals, and reports. A worklist can also consist of work from different workflows. For example, the worklist for one claims adjuster can contain appraisals for one claim, photographs for a second claim, and an expert report for a third claim. The actions the adjuster performs for each item in this worklist can be different. The adjuster might review the appraisal and approve the first claim. She might need to wait for more information about the second claim before taking action on the photographs. For the third claim, she could send the expert report to another employee for action.

### Actions to take

Consider what actions you want to take on the contents of a work item during the workflow. For example, a claims adjuster can accept a claims form or reject it as incomplete. An *action list* defines the actions that a user can perform on the work. An action list can describe the following:

- Selectable options that are available to the user
- Customer-defined options

For example, depending on whether a claim meets the initial requirements, an adjuster can select one option to continue the claims form through the workflow, or another option to reject the form.

## How information flows through the process

Consider how you want information and activities to flow. For example, when is the initial claims form reviewed? What supporting documents are needed to move on to the next step in the process? What criteria determine whether a claim is accepted or rejected? This flow of information is the basis of your workflow.

A workflow consists of the paths that guide the work throughout processing. Where does the input originate? Your workflow must begin at some point. For XYZ Insurance, the claims form submitted by the policy holder is the document that starts the workflow.

When all of the documents are received, the work item can continue along the path to a final action—for example, approval of the claim.

## How everything fits together

After you analyze the information that you want to process, determine the actions that you want to perform, and decide how you want the information to flow, you are ready to create a workflow diagram, which is the graphical representation of your workflow. You use EIP's workflow builder feature to create the diagram.

A workflow diagram shows how work moves through the various activities in the process, noting what tasks the activity involves. It describes the flow, the main elements, and the key points of a workflow.

Each symbol in the workflow diagram represents a point at which work is done. An insurance claim must be reviewed, supporting documentation must be collected, and the claim must be approved or rejected, depending on certain criteria. See "Creating a workflow" on page 89 for more information about the process symbols used in the workflow builder.

---

## Using Enterprise Information Portal workflow components

This section describes the workflow components. You access all components through the administration client. **Tip:** The EIP Version 8 workflow includes several changes, including changes to the Version 7.1 container, to accommodate the new Content Manager Version 8 architecture.

### Using workflow builder




You use the workflow builder to graphically define and build the workflow of a workgroup, department, or enterprise. **Restriction:** The EIP migration process migrates users from Version 7.1 databases. EIP Version 8.1 does not provide any automated migration of workflow data. You must redraw your Version 7.1 workflow diagrams using the EIP Version 8.1 workflow builder and redeploy the EIP Version 7.1 workflow processes.

Before you use the workflow builder to create a model of your workflows, you must define your privilege sets, access control lists, users, user groups, actions, action lists, and worklists. When you define a workflow in the administration client, you can set a default action list for the entire workflow. You can also assign

a different action list at each node in the workflow. For more information about these tasks, see “Defining action lists” on page 89, “Defining worklists”, and the online help.

Although you use the workflow builder to build workflows, you cannot use workflow builder to run the workflow. Using a client, your users view and perform actions on worklists and work items. For more information about how to program a client to work with Enterprise Information Portal workflow, see the *Workstation Application Programming Guide* and online API reference. Table 5 describes the three workflow icons that are common to every workflow. The EIP online help describes the toolbar icons in detail.

Table 5. Workflow builder process icons

	<p>A start node begins the workflow process. The workflow process diagram must have one and only one start node.</p>
	<p>A stop node ends the workflow process. Every new workflow process diagram contains a stop node. When you create a process, a stop node will be generated for you. You can move the stop node to any position on the drawing surface. The workflow process diagram must have one and only one stop node.</p>
	<p>A user exit routine node calls a line-of-business application to use with work items on a workflow process. Values from the workflow process can then be passed to the line-of-business application, and control values from the line-of-business application can be passed back to the workflow process.</p>

## Using workflow services

Enterprise Information Portal provides workflow services that maintain workflow information. The workflows and action list definitions that you create using the workflow builder are maintained in the Enterprise Information Portal administration database and the IBM MQSeries® Workflow database.

When the system administrator creates a worklist, the information associated with the worklist is permanently stored in the administration database. The system administrator can update, delete, and add worklists using EIP. When a system administrator checks out a workflow, the workflow is locked in the Enterprise Information Portal database, and is marked within the database as being checked out to the user, preventing anyone else from updating it until the user is finished.

---

## Defining worklists

A worklist can be thought of as filter of work available. A worklist is a filtered list of items assigned to specific users or user groups. When users log in to Enterprise Information Portal they can see filtered lists of work items that are assigned to them. You use the Enterprise Information Portal administration client to define your worklists.

A worklist definition includes the rules that govern the presentation, status, and security of its work item. You specify the rules for each worklist at the same time as you create the worklist. To manage the access to a worklist, you create an access control list for the worklist. For a complete description of how to define worklists, see the online help. The worklist definition includes the following:

**Access control list**

An access control list consists of one or more individual user IDs or user groups and the privilege set associated with each. The privilege set is used to define a user's authorization to access or perform certain tasks on the work. You use access control lists to limit user access to items in a worklist.

**Filtering and sorting worklists**

Criteria by which a user can view a filtered and sorted worklist.

**Maximum number of items in a worklist**

Maximum number of items that you want a worklist to contain.

---

## Defining action lists

An action list is a comprehensive list of all actions that a user can perform on work in a workflow.

The administration client online help provides step-by-step instructions that explain how to define actions and action lists.

---

## Creating a workflow

After you define actions, action lists, and worklists, you use workflow builder to create the model of your workflow. The administration client online help provides step-by-step instructions that explain how to define actions and action lists. The workflow builder provides visual cues for creating a workflow.

---

## Enabling workflow builder

In this step, you are starting workflow on an administration database. **Restriction:** The database you select for workflow must be on the same server where you installed MQ Series, and the MQSeries services must be started.

To enable EIP workflow and create a workflow definition:

1. Log in to the administration client.
2. If you have multiple administration databases, click the icon for the database where you want to enable workflow.
3. Click Tools->Services. Click Enable Workflow.
4. Log out of the client and log in again. If you have multiple databases, select the icon for the database where you enabled EIP workflow. The Workflows folder icon will be displayed.
5. In the left pane of the Enterprise Information Portal administration main window, double-click the Workflows folder **Workflows**.
6. Right-click Workflow Definitions icon and select **New** to create a workflow definition.

**Requirement:** You must create at least one access control list, one action, and one action list before defining a workflow.

---

## Starting the MQSeries Workflow server

Start the MQSeries Workflow server by entering `cmbwfstart` at a command prompt. Two windows open for the MQSeries Workflow server. Leave these command windows open to continue running the server.

If you install the workflow after the initial installation of Enterprise Information Portal, you must configure the Enterprise Information Portal system for the workflow feature. You also need to change the configuration if you install the workflow feature on a different workstation than the workstation on which the administration client is installed.

1. From the administration window, click on the file member **Tools**.
2. Click **Services** from the menu.
3. Select the **Workflow** check box.
4. After the configuration is complete, log off of the Enterprise Information Portal administration client and log on again to initialize the workflow feature. After you log on to the Enterprise Information Portal administration client, the **Workflow Definitions** icon appears in the left pane.

**Tip:** Administrators do not see the **Workflow Definitions** icon unless they have authority to administer the workflow feature. If you want to restrict access to the workflow feature, see the appropriate system administration books for each content server. See the online help for more information about authorizing administrators to manage the workflow feature.

The client can be created from a custom application using the EIP Connector Toolkit and samples, or you can use the EIP sample client.

---

## Chapter 7. IBM Web Crawler sample files

This section provides two code samples. The config-sample2.xml sample file provides examples of the <gcs-config> configuration parameters. The log analysis sample provides an example of a report containing information from a completed crawl.

---

### config-sample2.xml sample

The code sample in this section is an example of the gcs-config file.

```
<!DOCTYPE gcs-config SYSTEM "config.dtd">
<gcs-config>
  <!-- Global settings: -->
  <globals max-urls="1000000"
    num-crawlers="30"
    num-summarizers="8"
    summaries-dir="summaries"
    log-file="log/LOG.txt"
    temp-dir="temp"
    log-priority="warn"
    text-monitor="60"
    graph-monitor="2"
    connect-timeout="120"
    read-timeout="100">

    <!-- sample logger settings -->
    <logger-config category="gcs.summaries.list.resource"
priority="info" log-file="log/resources.txt"/>
    <logger-config category="gcs.summaries.list.host" priority="info"
log-file="log/hosts.txt"/>
    <logger-config category="gcs.url.skipped" priority="info"
log-file="log/skipped_urls.txt"/>
    <logger-config category="gcs.url.fromto" priority="info"
log-layout="%m\n" log-file="log/fromto.txt"/>
    <logger-config category="gcs.http" priority="info"
log-file="log/http.txt"/>
    <logger-config category="gcs.http.connect" priority="info"
log-file="log/connecterrs.txt"/>

    <!--use this to specify a database
    <urlpool-config urlcontainer-class="DB2URLContainer"
urlcollection- class="DB2URLCollection">
      <urlpool-param name="dbname" value="gcs"/>
      <urlpool-param name="user" value="xxxxxx"/>
      <urlpool-param name="password" value="xxxxxx"/>
      <urlpool-param name="cachesize" value="1000"/>
      <urlpool-param name="driver"
value="COM.ibm.db2.jdbc.app.DB2Driver"/>
    </urlpool-config -->

    <!--use this to specify a SOCKS proxy
    <system-properties>
      <property name="socksProxySet" value="true"/>
      <property name="socksProxyHost" value="socks2.server.ibm.com"/>
      <property name="socksProxyPort" value="1080"/>
    </system-properties -->

  </globals>

  <group-list>
    <group name="ibm">
```

```

<crawler-config recursion-depth="-1">
  <seed-list>
    <!-- URLs to start crawling at: -->
    <seed url="http://gcs.stl.ibm.com/gcs/testurl.html"/>
    <seed url="http://gcs.stl.ibm.com/gcs/stl.html"/>
    <seed url="http://gcs.stl.ibm.com/gcs/ibm.html"/>
  </seed-list>

  <content-type-pattern-list>
    <!-- URL file extensions that don't match these patterns
    won't be crawled: -->
    <url-name-pattern name="htm*"/>
    <url-name-pattern name="pdf"/>
    <url-name-pattern name="gif"/>
    <url-name-pattern name="zip"/>
    <url-name-pattern name="txt"/>
  </content-type-pattern-list>

  <include-pattern-list>
    <!-- URLs that don't match these patterns won't be crawled: -->
    <url-obj-pattern host="*.ibm.com"/>
    <!-- sbo - url-obj-pattern query="*OpenDocument*" / -->
    <!-- sbo - url-obj-pattern query="*OpenView*" / -->
    <!-- url-obj-pattern query="*OpenDocument =>
OpenDocument&amp;ExpandAll*" / -->
    <!-- url-obj-pattern query="*OpenView =>
OpenView&amp;ExpandAll&amp;Count=999999*" / -->
  </include-pattern-list>

  <exclude-pattern-list>
    <!-- URLs that match these patterns won't be crawled: -->
    <!-- skip these common patterns in our intranet -->
    <url-obj-pattern file="*news*"/>
    <url-obj-pattern file="*search*"/>
    <url-obj-pattern file="*/afs*/>
    <url-obj-pattern file="*/...*/>
    <url-obj-pattern file="*bluepages*"/>
    <!-- skip personal home pages -->
    <url-obj-pattern file="*/~*"/>
    <!-- skip SOCKS: no URL should specify this directly -->
    <url-regex-pattern match=".*:1080/.*"/>
    <!-- skip gateways: recommended for mere mortals -->
    <url-regex-pattern match=".*[?|\=|\+|\;|\%&quot;&amp;];.*"/>
    <!-- else crawl gateways as configured here... -->
    <!-- skip Domino -->
    <url-obj-pattern file="*.nsf*"/>
    <!-- else crawl Domino: allow only OpenDocument -->
    <url-obj-pattern query="*OpenServer*"/>
    <url-obj-pattern query="*OpenDatabase*"/>
    <url-obj-pattern query="*OpenElement*"/>
    <url-obj-pattern query="*OpenView*"/>
    <url-obj-pattern query="*OpenAbout*"/>
    <url-obj-pattern query="*OpenHelp*"/>
    <url-obj-pattern query="*OpenIcon*"/>
    <url-obj-pattern query="*OpenForm*"/>
    <url-obj-pattern query="*OpenNavigator*"/>
    <url-obj-pattern query="*OpenAgent*"/>
    <url-obj-pattern query="*CreateDocument*"/>
    <url-obj-pattern query="*DeleteDocument*"/>
    <url-obj-pattern query="*EditDocument*"/>
    <url-obj-pattern query="*SaveDocument*"/>
    <url-obj-pattern query="*SearchSite*"/>
    <url-obj-pattern query="*SearchView*"/>
    <url-obj-pattern query="*&login*"/>
    <url-obj-pattern query="*Command*"/>
    <!-- crawl Domino: avoid OpenDocument permutations -->

```



```

        <url-obj-pattern    query="*ExpandSection*"/>
        <url-obj-pattern    query="*Navigate*"/>
        <url-obj-pattern    query="*Start*"/>
    <!-- -->

    </exclude-pattern-list>
</crawler-config>

<summarizer-config>
<!-- Copy2Rdf is the default summarizer.  For these types, use: -->

    <resource-handler content-type="*htm*"
        summarizable="EipHtmlSummarizable"
        summary-maker="EipHtmlRawSummaryMaker" />
    <resource-handler content-type="*pdf"
        summarizable="InsoSummarizable"
        summary-maker="InsoSummaryMaker" />
</summarizer-config>
v    </group>
    </group-list>
</gcs-config>

```

---

## IBM Web Crawler log analysis file example

```
D:\gcs\run\log>perl loganalysis.pl log.txt
```

Elapsed time in Log.txt for 7710 lines was 1.84 minutes.

```

GCS was configured for 20 crawlers
  999 total crawls attempted
  137 - total crawl failures:
      21  GCSHttpConnection.ABANDONING
      12  GCSHttpConnection.CONNECT_ERROR
      16  GCSHttpConnection.UNKNOWN_HOST
       4  HTTP 403
      29  HTTP 404
       2  HTTP 500
       8  HTTP 599
       1  Read timed out
      39  Robots not allowed
       4  over max redirects
       1  unknown protocol

-----
  862 = successfully crawled
   0  - unchanged since earlier crawl
-----
  862 = new or changed
  468 crawled per minute

```

```

GCS was configured for 5 summarizers
  855 total summaries attempted
   0  - total summary failures:
-----
  855 = successfully summarized
  144 gcs.summaries.list.host
  855 gcs.summaries.list.resource
  465 summarized per minute

```

```

GCS successfully crawled 134 servers to obtain 862 URL:
afqa0854.mop.ibm.com: 15
alslf1.yamato.ibm.com: 1
apache.btv.ibm.com: 1
apc.endicott.ibm.com: 2
as400service.ibm.com: 1

```

atlas.bocaraton.ibm.com: 1  
autoproxy.ibm.com: 1  
cer.si.ibm.com: 1  
commerce.www.ibm.com: 1  
crmweb.boulder.ibm.com: 3  
d02ntcl01.ibm.com: 1  
dacs.endicott.ibm.com: 1  
duke.toraix.can.ibm.com: 1  
ebcweb.austin.ibm.com: 1  
ecspubs.ibmus2.ibm.com: 5  
edaw3.fishkill.ibm.com: 1  
endwww.endicott.ibm.com: 1  
gcs.stl.ibm.com: 1  
gustwick.austin.ibm.com: 1  
ibmfnsys.somers.hqregion.ibm.com: 1  
ibmpny1.somers.hqregion.ibm.com: 2  
ifw-www.mul.ie.ibm.com: 1  
iplswww.nas.ibm.com: 2  
itirc.ibm.com: 1  
logosite.services.ibm.com: 1  
lt.lahulpe.ibm.com: 17  
messaging.ibm.com: 1  
mrsmrn04.leeds.uk.ibm.com: 1  
online.lahulpe.ibm.com: 1  
page.sg.ibm.com: 1  
procure.sby1.ibm.com: 1  
reso.somers.hqregion.ibm.com: 1  
risctal.leipzig.de.ibm.com: 1  
rrhhar.argentina.ibm.com: 1  
seashore.stl.ibm.com: 1  
secureway.raleigh.ibm.com: 15  
service.software.ibm.com: 1  
software.ibmus2.ibm.com: 1  
techcenter.austin.ibm.com: 1  
tr2.fishkill.ibm.com:8080: 1  
ucd.torolab.ibm.com: 1  
usmweb.boulder.ibm.com: 1  
w3-1.ibm.com: 32  
w3-2.ibm.com: 3  
w3-3.ibm.com: 108  
w3-5.ibm.com: 4  
w3.a-nz.au.ibm.com: 1  
w3.academy.ibm.com: 1  
w3.almaden.ibm.com: 2  
w3.alphaworks.ibm.com: 1  
w3.ap.ibm.com: 1  
w3.asca.ibm.com: 7  
w3.austin.ibm.com: 3  
w3.boulder.ibm.com: 1  
w3.br.ibm.com: 1  
w3.btv.ibm.com: 1  
w3.can.ibm.com: 40  
w3.chq.ibm.com: 4  
w3.coc.ibm.com: 1  
w3.corporatetechnology.ibm.com: 1  
w3.cupertino.ibm.com: 1  
w3.dds.dfw.ibm.com: 17  
w3.demopkg.ibm.com: 4  
w3.design.ibm.com: 1  
w3.developer.ibm.com: 3  
w3.education.ibm.com: 1  
w3.emea.ibm.com: 14  
w3.enterlib.ibm.com: 7  
w3.finsys.ibm.com: 1  
w3.gcg.ibm.com: 1  
w3.globalfinancing.de.ibm.com: 1  
w3.hakozaki.ibm.com: 1

w3.houston.ibm.com: 1  
w3.hursley.ibm.com: 5  
w3.iabc.ibm.com: 1  
w3.ibm.com: 180  
w3.ibmfax.ibm.com: 1  
w3.ibmlla.ibm.com: 14  
w3.isicc.de.ibm.com: 1  
w3.itso.ibm.com: 1  
w3.japan.ibm.com: 1  
w3.knowledge.raleigh.ibm.com: 1  
w3.linux.ibm.com: 1  
w3.marketing.ibm.com: 1  
w3.micro.ibm.com: 2  
w3.mtlisc.can.ibm.com: 1  
w3.munich.ibm.com: 1  
w3.ode.raleigh.ibm.com: 1  
w3.paylink.au.ibm.com: 1  
w3.pisc.uk.ibm.com: 1  
w3.pl.ibm.com: 1  
w3.printers.ibm.com: 1  
w3.pssc.mop.ibm.com: 1  
w3.pssed.au.ibm.com: 1  
w3.raleigh.ibm.com: 3  
w3.rchland.ibm.com: 1  
w3.research.ibm.com: 3  
w3.reserve.ibm.com: 1  
w3.rs6000.ibm.com: 1  
w3.security.ibm.com: 1  
w3.software.ibm.com: 6  
w3.ssd.ibm.com: 1  
w3.stl.ibm.com: 1  
w3.techline.ibm.com: 1  
w3.techsupp.yamato.ibm.com: 1  
w3.torolab.ibm.com: 2  
w3.usergroup.ibm.com: 1  
w3.vendor.pok.ibm.com: 1  
w3.viewblue.ibm.com: 1  
w3.watson.ibm.com: 2  
w3.wdg.uk.ibm.com: 1  
w3.ytal.yasu.ibm.com: 1  
w3.zurich.ibm.com: 1  
w3chq.disbursements.ibm.com: 1  
w3is.lagaude.ibm.com: 1  
w3md.btv.ibm.com: 1  
w3ssd.mainz.de.ibm.com: 1  
w3vm.demopkg.ibm.com: 1  
widweb.raleigh.ibm.com: 1  
wtscpok.itso.ibm.com: 1  
wwas.raleigh.ibm.com: 1  
www-1.ibm.com: 63  
www-3.ibm.com: 4  
www-4.ibm.com: 86  
www.almaden.ibm.com: 1  
www.as400.ibm.com: 1  
www.chips.ibm.com: 1  
www.ibm.com: 52  
www.ieg.ibm.com: 1  
www.patents.ibm.com: 1  
www.pc.ibm.com: 2  
www.rs6000.ibm.com: 23  
www.software.ibm.com: 9  
www.storage.ibm.com: 1  
www.watson.ibm.com: 1

GCS timed out 1 times:  
w3-3.ibm.com: 1

GCS ignored 42 URL prohibited by robots.txt:

- reso.somers.hqregion.ibm.com: 1
- w3.education.ibm.com: 1
- w3.rchland.ibm.com: 34
- w3.zurich.ibm.com: 1
- www.ibm.com: 5

GCS skipped 3846 URL (requires gcs.url logging)

59 specified an unsupported protocol:

- protocol not supported gopher: 6
- protocol not supported mailto: 53

1206 had content-types (lower or UPPER case, > 10) that were not included

- .2: 12
- .faq: 13
- .1: 14
- .asp: 16
- .cgi: 21
- .shtml: 90
- .pl: 92
- .gif: 157
- .nsf: 160
- .jpg: 214
- .css: 240

516 URL were on servers and/or paths that were not included

2065 were excluded for these reasons:

- URL longer than 254: 1
- excluded by rule 1: 1210
- excluded by rule 2: 854

---

## Chapter 8. Using text search and QBIC®

The first section of this appendix explains how to configure and use text search and Query by Image Content (QBIC), two features available only when you select the Content Manager Version 7.1 connector during EIP installation. The second section of this appendix provides information on loading the sample text and image data used with sample applications.

---

### Searching documents using the text search engine

Text search can be integrated with your Content Manager Version 7.1 server, so that you can automatically index, search, and retrieve documents stored in Content Manager. Users can locate documents by searching for words or phrases. The text search server supports both single-byte and double-byte character sets and runs on both AIX and Windows.

Text search includes structured document support for XML, HTML, and tagged ASCII documents, which allows you to search for terms within specified sections of a document. You can search for data in nested sections. You can search by full XML context, for example, you can search for IBM within titles, and search for IBM within a title that is in a specific section. When you specify a DTD path, text search can dynamically use the appropriate DTD for each document, assuming that a reference to the DTD is stored as metadata for that document.

See *Planning and Installing Enterprise Information Portal* for information on planning and installing an EIP system using text search.

### Enabling a text search server

To use a text search server, you must enable administration for the server before starting the IBM Content Manager for Multiplatforms administration client. To enable administration:

1. Start your IBM Content Manager for Multiplatforms library server.  
Allow the library server to finish building the index classes.
2. Start the text search server on the workstation where it is installed by entering:  

```
imlss -start dinst
```

where *dinst* is the name of your text search server instance chosen at installation time or when using the *imlcfgsv* command utility.

---

### Searching images using Query by Image Content (QBIC)

This section introduces Query by Image Content (QBIC) and explains how to configure and use QBIC. The QBIC feature is only available if you install the Content Manager Version 7.1 connector. QBIC is compatible with Windows and AIX operating systems.

### Introducing image search

The image search server uses IBM's QBIC (query by image content) technology to help you search for objects by certain visual properties, such as color and texture. The image search server analyzes images and stores the image information in a database. Then users can run image queries, which use the visual properties of

images, to match colors, textures, and their positions without describing them in words. You can combine content-based queries with text and keyword searches for more powerful retrieval of image and multimedia data.

Each image search server has a data directory containing one or more image search databases, which hold the image search catalogs. An image search catalog stores the data about the visual features of a collection of images. The actual image objects are stored on object servers in the IBM Content Manager for Multiplatforms system. The image search server runs on AIX and Windows.

See *Planning and Installing Enterprise Information Portal* for information on installing image search.

## Setting up image search

These instructions apply after you install Image Search, which is automatically installed if you select the Content Manager Version 7.1 connector. Setting up image search consists of:

1. Setting up the environment
2. Configuring the image search server
3. Configuring the image search client
4. Loading the sample image data

**If you use the installation wizard on AIX:** You do not have to run the configuration and setup scripts or issue the server configuration command. The wizard completes these tasks for you.

**If you are installing on Windows:** You must complete these tasks.

### Setting up the environment

Complete the environment setup tasks in this section on both the server and client machines. The image search server requires the following environment variables:

#### **QBICTOP**

Resolves file names during image search configuration

#### **QbicImagePath**

Resolves file names on a server image file

#### **QbicMaskPath**

Resolves file names on a server mask file

#### **QbicSketchPath**

Resolves file names on a server sketch file

#### **QbicTextPath**

Resolves file names on a server text file

The image search client requires only the QBICTOP environment variable.

**AIX example:** On AIX, run the configuration script, which generates the setup script, and then run the setup script to set up the environment.

1. Run the following configuration script:  

```
/usr/lpp/cmb/bin/frnconfig.iss QBICTOP
```

where QBICTOP is the directory path to the control files (\*.ini). Set QBICTOP to /user1/cmb/qbic, where /user1 is the home directory of the image search administration user ID. The image search user ID must have read/write access to this directory.

This script generates the setup script: frnsetup.iss.

2. From the home directory of the image search user ID, run:

```
. ./frnsetup.iss
```

This script populates the environment variables for image search servers and clients.

**Windows example:** To set the environment variables:

1. Select **Start** → **Settings** → **Control Panel**.
2. Double-click **System**.
3. Click the Environment tab.
4. Set the variables and the values shown in Table 6 by typing them in the appropriate fields and clicking **Set**.

**Requirement:** The image search client requires only the QBICTOP variable. For client environments, set only the QBICTOP variable.

Table 6. Image search environment variables

Variable	Value
QBICTOP	d:\cmbroot\iss
QbicImagePath	d:\cmbroot\iss
QbicMaskPath	d:\cmbroot\iss
QbicSketchPath	d:\cmbroot\iss
QbicTextPath	d:\cmbroot\iss

Where d: is the drive where image search is installed.

## Configuring the image search server

Before starting the image search server, you must configure it. Configuring the server consists of completing the initial configuration and verifying the connection.

To configure the server:

1. Start the command interpreter by entering: qbicadm
2. Enter the **config server** command. For example,  
config server LIBSRVRN FRNADMIN PASSWORD 9999

where LIBSRVRN is the library server name, FRNADMIN is the Content Manager user ID, PASSWORD is the Content Manager password, and 9999 is the port number of the image search server.

See “Verifying the connection” on page 100 for more information.

## Configuring the image search client

Before starting an image search client, including the image search system administration program, you must configure it. Content Manager system administration requires that you assign one alias. Test your configuration by verifying the connection.

**Assigning an alias:** Before you can use the image search Content Manager system administration, which acts as an image search client, you must assign at least one server alias.

To assign an alias:

1. Start the command interpreter by entering: `qbicadm`
2. Enter the **add alias** command. For example,  
`add alias QBICSRV HOSTNAME 9999`

where QBICSRV is the alias name, HOSTNAME is the image search server's host name, and 9999 is the port number of the image search server.

**Verifying the connection:**

**Important:**

1. The library server must be running to connect to the image search server.
2. Image search system administration requires an existing Content Manager user ID. To successfully connect to the library server, the image search user ID and library server user ID must be the same. The default value for this user ID is `frnadmin`. If you change this value, make sure that the IDs match.

To verify the connection:

1. After configuring the image search server and adding the alias, start the server by entering `commsrv` from the server command line.
2. To start the command interpreter, enter: `qbicadm`.
3. In the command interpreter, enter the `connect` command.  
`connect QBICSRV FRNADMIN PASSWORD`

where QBICSRV is the alias name, FRNADMIN is the Content Manager user ID, PASSWORD is the Content Manager password.

After successfully connecting, the message `Library Server is LIBSRVRN` displays.

4. To disconnect from the server, enter: `disconnect`.
5. To exit the command interpreter, enter: `quit`.

---

## Loading and indexing sample data

This section explains how to load and index sample text and image data, which you use with the sample applications. This section only applies if you installed the Content Manager Version 7.1 connector and selected the text search option.

There are several sample loaders provided on the Enterprise Information Portal CD. This section describes how to load both image and text data using the sample loader `LoadSampleTSQLCDL`. You can load text and image search data separately to ensure both features are working properly.

### Before you load the sample data

Before running the loader program, you must:

1. Log in to the EIP administration client. Click `Start` → `Programs` → **Enterprise Information Portal** for Multiplatforms 8.1 → `Administration`.



2. Choose a database and log in using the correct user ID and password. If you select the default database `icmn1sdb`, enter **icmadmin** as the user ID and enter password in the password field. If you use another database, enter the applicable user ID.
3. Create a library server configuration using the Content Manager system administration program. See the system administration program online help for assistance with this task.
4. Change the **Access** properties of the library server configuration by completing the following steps:
  - a. Right-click your new configuration and click **Properties** to open the Properties notebook.
  - b. Click the Access tab.
  - c. Click the **Unlimited sessions from any workstation** radio button.

## Creating a text search index

Before you load data you must create an empty text search index that you can use to index the text samples. **Tip:** You can only create a text search index on a Content Manager Version 6.1 or Version 7.1 server.

To create a text search index:

1. Start the text search server on the workstation where it is installed by using the following command:

```
imlss -start d1inst
```

where `d1inst` is the name of your text search server instance chosen at installation time or when using the `imlcfgsv` command utility.

2. Start and log in to the Content Manager system administration program.
3. Select **Text Search** from the list at the top left pane.
4. Double-click **Search Servers** in the left pane.
5. Double-click **TM**. TM is the search server alias for the text search server.
6. Double-click the **Indexes** folder from the left pane. If the message `RC_EMPTY_LIST` displays, then, from the menu bar, click **Selected** → **New** to create an index.
7. In the New Index window, define your index. Click **Help** for a detailed description of each field.

For example:

**For Windows:**

**Name** TMINDEX

**Type** Precise

**Index files**

`x:\cmbroot\ts\index\tmlindex` where `x` is your installation drive; if the path does not exist, it is created.

**Index work files**

`x:\cmbroot\ts\work\tmlindex` where `x` is your installation drive; if the path does not exist, it is created.

**Information entry**

Name of the Content Manager library server.

Do not change the client and server default DLL names.

**For AIX:**

**Name** TMINDEX

**Type** Precise

**Index files**

/home/c1tadmin/tsindex/index/tminindex; if the path does not exist, it is created.

**Index work files**

/home/c1tadmin/tsindex/work/tminindex; if the path does not exist, it is created. Ensure that the user has authorization to write to the directory.

**Information entry**

Name of the Content Manager library server.

8. Click **OK**.
9. Double-click **TMINDEX** to open the TMINDEX Administration Notebook.

## Creating the image search database, catalog, and features

After you create a text search index for the sample text search data, you must create an image search database and catalog for the sample image data.

To create the image search database, catalog, features:

1. Start the image search server on the workstation where it is installed by entering the following command:  
`commsrv`
2. Start and log in to the Content Manager system administration program.
3. Select **Image Search** from the list at the top left pane.
4. Click **Image Search Servers** in the left pane.
5. Click **QBICSRV**.  
where QBICSRV is the image search server name you specified during installation.
6. Right-click **Databases** in the left pane and select **New Database**.
7. In the New Database window, enter SAMPLEDB in the **Name** field, and click **OK**.
8. In the left pane, click **Databases** to display the **SAMPLEDB** icon in the left pane.
9. Click **SAMPLEDB**.
10. In the left pane, right-click **Catalogs** and click **New Catalog**.
11. In the New Catalog window, enter SAMPLECAT in the **Name** field, and click **OK**.
12. In the left pane, click **Catalogs** to display the **SAMPLECAT** icon.
13. Click **SAMPLECAT**.
14. In the left pane, right-click **Features** and click **New Features**.
15. In the New Features window, select each feature in the **Name** field and click **Apply**. When all four features are selected the following message displays:  
All possible features have been added to catalog.
16. Click **OK**.
17. Click **Cancel**.

## Running the loader program

You can load sample data to test your text and image search.

The sample image data is in the following files:

**On Windows:**

```
x:\cmbroot\samples\java\d1\samples.jar
```

**On AIX:**

```
/usr/lpp/cmb/samples/java/d1/samples.jar
```

The sample loader program loads the data into Content Manager and indexes it. Read the prolog of the source program for instructions about the syntax for running the program. The sample loader programs are:

**On Windows:**

```
x:\cmbroot\samples\java\d1\LoadSampleTSQBICDL.jar
```

**On AIX:**

```
/usr/lpp/cmb/samples/java/d1/LoadSampleTSQBICDL.jar
```

To run the sample data loader program:

1. Decompress the .jar files by entering:

```
jar -xvf samples.jar
```

The files decompress into the correct directories.

2. Set your workstation environment variables to compile the sample loader program by completing the following tasks:

**On Windows:**

- a. Open x:\cmbroot\cmbenv71.bat in a text editor and change the first three lines to set your workstation environment variables:

```
set CMBROOT = e:\cmbroot  
set DB2HOME = e:\sql1ib  
set JAVAHOME = d:\jdk117
```

- b. Save cmbenv71.bat and set the environment variables by entering:

```
cmbenv71
```

**On AIX:**

- a. Go to /usr/lpp/cmb/bin/ and run setup by entering

```
./cmbenv71.sh
```

- b. Ensure that the subdirectories under /usr/lpp/cmb/samples/java/d1/ and the sample files are writable by all users.

3. Compile loader program by entering the case-sensitive command:

```
javac LoadSampleTSQBICDL.java
```

4. **Requirement:** The following servers must be running before you start the loader program:

- Library server
- Object server
- Text search server
- Image search server

If you are running in a National Language Version of Content Manager, set the FRNDEFLANG variable to ENU before running the loader program. The AIX command to set the environment variable is: export FRNDEFLANG=ENU

5. Load the sample data with the loader program by entering:  

```
java LoadSampleTSQBICDL sampleQBIC.dat load.log frnadmin password LIBSRVRN
```

where your user ID is frnadmin, your password is password and your library server is LIBSRVRN.

6. Check the load.log to ensure that the sample data loaded successfully.

After you finish loading the sample data, use the Content Manager system administration program or text search command line tools to index the sample text data.

## Indexing the sample text data

To index the sample data:

1. Start and log on to the Content Manager system administration program.
2. Select **Text Search** from the list at the top left pane.
3. Double-click **Search Servers**.
4. Double-click **TM**. TM is the search server alias for the text search server.
5. Right-click **new text index** and click **Properties**.
6. On the Explicit page of the Properties notebook, click **Refresh**.
7. The **Index count** field should display the number of documents that you loaded with the loader program.
8. Click **Index** to index the files.
9. After a few moments, click **Refresh** to see the number of successfully indexed documents in the **Primary document index** field.

After you index the data, you can use the sample Java application to query the collection or run a simple query by using the `iml srch` command line tool.

---

## Chapter 9. Document formats

---

### Information mining document formats

This appendix describes the document formats supported by information mining.

#### Word processing: Generic

ANSI Text (7 & 8 bit)	All versions
ASCII Text (7 & 8 bit versions available)	All versions
HTML	Versions through 3.0 (some limitations)
IBM FFT	All versions
IBM Revisable Form Text	All versions
Microsoft Rich Text Format (RTF)	All versions
Unicode Text	All versions

#### Word processing: DOS

DEC WPS Plus (DX)	Versions through 4.0
DEC WPS Plus (WPL)	Versions through 4.1
DisplayWrite <sup>®</sup> 2 & 3 (TXT)	All versions
DisplayWrite 4 & 5	Versions through release 2.0
Enable	Versions 3.0, 4.0 and 4.5
First Choice	Versions through 3.0
Framework	Version 3.0
IBM Writing Assistant	Version 1.01
Lotus Manuscript	Versions through 2.0
MASS11	Versions through 8.0
Microsoft Word	Versions through 6.0
Microsoft Works	Versions through 2.0
MultiMate	Versions through 4.0
Navy DIF	All versions
Nota Bene	Version 3.0
Office Writer	Version 4.0 to 6.0
PC-File Letter	Versions through 5.0
PC-File+ Letter	Versions through 3.0
PFS:Write	Versions A, B, and C

<b>Professional Write</b>	Versions through 2.1
<b>Q&amp;A</b>	Version 2.0
<b>Samna Word</b>	Versions through 4.0
<b>SmartWare II</b>	Version 1.02
<b>Sprint</b>	Versions 1.0
<b>Total Word</b>	Version 1.2
<b>Volkswriter 3 &amp; 4</b>	Versions through 1.0
<b>Wang PC (IWP)</b>	Versions through 2.6
<b>WordMARC</b>	Versions through Composer Plus
<b>WordPerfect</b>	Versions through 6.1
<b>WordStar</b>	Version through 7.0
<b>WordStar 2000</b>	Version through 3.0
<b>XyWrite</b>	Version through III Plus

## **Word processing: International**

<b>JustSystems Ichitaro</b>	Versions 5.0, 6.0, 8.0, 9.0 and 10.0
-----------------------------	--------------------------------------

## **Word processing: Windows**

<b>AMI/AMI Professional</b>	Versions through 3.1
<b>Corel WordPerfect for Windows</b>	Versions through 9.0
<b>JustWrite</b>	Versions through 3.0
<b>Legacy</b>	Versions through 1.1
<b>Lotus WordPro (Win32 / Intel platforms)</b>	SmartSuite® 96, 97 and Millennium
<b>Lotus WordPro (Unix platforms - text only)</b>	SmartSuite 97 and Millennium
<b>Microsoft Windows Works</b>	Versions through 4.0
<b>Microsoft Windows Write</b>	Versions through 3.0
<b>Microsoft Word 97</b>	Word 97
<b>Microsoft Word 2000</b>	Word 2000
<b>Microsoft Word for Windows</b>	Versions through 7.0
<b>Microsoft WordPad</b>	All versions
<b>Novell Perfect Works</b>	Version 2.0
<b>Novell WordPerfect for Windows</b>	Versions through 7.0
<b>Professional Write Plus</b>	Version 1.0
<b>Q&amp;A Write for Windows</b>	Versions 3.0
<b>WordStar for Windows</b>	Version 1.0

## Word processing: Macintosh

Microsoft Word	Versions 4.0 through 6.0
Microsoft Word 98	Word 98
WordPerfect	Versions 1.02 through 3.0
Microsoft Works	Versions through 2.0
MacWrite II	Version 1.1

## Spreadsheets formats

VP Planner 3D	Version 1.0
Enable	Versions 3.0, 4.0 and 4.5
First Choice	Versions through 3.0
Framework	Version 3.0
Lotus 1-2-3 <sup>®</sup> (DOS & Windows)	Versions through 5.0
Lotus 1-2-3 for SmartSuite	SmartSuite 97 and Millennium
Lotus 1-2-3 Charts (DOS & Windows)	Versions through 5.0
Lotus 1-2-3 (OS/2 <sup>®</sup> )	Versions through 2.0
Lotus 1-2-3 Charts (OS/2)	Version through 2.0
Lotus Symphony	Versions 1.0, 1.1 and 2.0
Microsoft Excel 97	Excel 97
Microsoft Excel 2000	Excel 2000
Microsoft Excel Macintosh	Version 3.0 through 4.0, 98
Microsoft Excel Windows	Versions 2.2 through 7.0
Microsoft Excel Charts	Versions 2.x through 7.0
Microsoft Multiplan	Version 4.0
Microsoft Windows Works	Versions through 4.0
Microsoft Works (DOS)	Versions through 2.0
Microsoft Works (Mac)	Versions through 2.0
Mosaic Twin	Version 2.5
Novell Perfect Works	Version 2.0
QuattroPro for DOS	Versions through 5.0
QuattroPro for Windows	Versions through 9.0
PFS:Professional Plan	Version 1.0
SuperCalc 5	Version 4.0
SmartWare II	Version 1.02

## Database formats

SmartWare II	Version 1.02
--------------	--------------

<b>Access</b>	Versions through 2.0
<b>dBase</b>	Versions through 5.0
<b>DataEase</b>	Version 4.x
<b>dBXL</b>	Version 1.3
<b>Enable</b>	Versions 3.0, 4.0 and 4.5
<b>First Choice</b>	Versions through 3.0
<b>FoxBase</b>	Version 2.1
<b>Framework</b>	Version 3.0
<b>Microsoft Windows Works</b>	Versions through 4.0
<b>Microsoft Works (DOS)</b>	Versions through 2.0
<b>Microsoft Works (Mac)</b>	Versions through 2.0
<b>Paradox (DOS)</b>	Versions through 4.0
<b>Paradox (Windows)</b>	Versions through 1.0
<b>Personal R:BASE</b>	Version 1.0
<b>R:BASE 5000</b>	Versions through 3.1
<b>R:BASE System V</b>	Version 1.0
<b>Q &amp; A</b>	Versions through 2.0
<b>Reflex</b>	Version 2.0

## Standard graphic formats

<b>PNG - Portable Network Graphics Internet Format</b>	Version 1.0
<b>Binary Group 3 Fax</b>	All versions
<b>BMP(including RLE, ICO, CUR &amp; os/2 DIB)</b>	Windows
<b>CDR (if TIFF image is embedded in it)</b>	Coral Draw versions 2.0 – 9.0
<b>CGM - Computer Graphics Metafile</b>	ANSI, CALS, NIST, Version 3.0
<b>CMX - Corel Clip Art Format</b>	Versions 5 through 6
<b>DCX (multi-page PCX)</b>	Microsoft Fax
<b>DRW - Micrografx Designer</b>	Version 3.1
<b>DRW - Micrografx Draw</b>	Versions through 4.0
<b>DXF (Binary and ASCII) AutoCAD Drawing Interchange Format</b>	Versions through 14
<b>EMF</b>	Windows Enhanced Metafile
<b>EPS Encapsulated PostScript</b>	If TIFF image is embedded in it
<b>FMV - FrameMaker graphics</b>	Vector and raster format through Version 5.0
<b>FPX - Kodak Flash Pix</b>	No specific format



<b>GDF - IBM Graphics Data Format</b>	Version 1.0
<b>GEM - Graphics Environment Manager Metafile</b>	Bitmap and Vector
<b>GIF - Graphics Interchange Format</b>	Compuserve
<b>GP4 - Group 4 CALS Format</b>	Type I and Type II
<b>HPGL - Hewlett Packard Graphics Language</b>	Version 2.0
<b>IMG - GEM Paint</b>	No specific version
<b>JFIF (JPEG not in TIFF format)</b>	All versions
<b>JPEG - Joint Photographic Experts Group format</b>	All versions
<b>MET - OS/2 PM Metafile</b>	Version 3.0
<b>PBM - Portable Bitmap</b>	No specific version
<b>Kodak Photo CD</b>	Version 1.0
<b>PCD - PCX Bitmap</b>	PC Paintbrush
<b>Perfect Works (Draw)</b>	Novell version 2.0
<b>PGM - Portable Graymap</b>	No specific version
<b>PIC - Lotus 1-2-3 Picture File Format</b>	No specific version
<b>PICT1 &amp; PICT2 (Raster)</b>	Macintosh Standard
<b>PIF - IBM Picture Interchange Format</b>	Version 1.0
<b>PNTG</b>	MacPaint
<b>PPM - Portable Pixmap</b>	No specific version
<b>Progressive JPEG</b>	No specific version
<b>PSP - Paintshop Pro (Win32 only)</b>	Versions 5.0, 5.0.1
<b>RND - AutoShade Rendering File Format</b>	Version 2.0
<b>SDW Ami Draw Snapshot (Lotus)</b>	All versions
<b>SRS - Sun Raster File Format</b>	No specific version
<b>Targa</b>	Truevision
<b>TIFF</b>	Versions through 6
<b>TIFF CCITT Group 3 &amp; 4</b>	Fax Systems
<b>VISO (Page Preview mode only for Version 4) Visio 4, 5, 2000</b>	Visio 4, 5, 2000
<b>WMF</b>	Windows Metafile

<b>WordPerfect Graphics [WPG and WPG2]</b>	Versions through 2.0
<b>XBM - X-Windows Bitmap</b>	x10 compatible
<b>XPM - X-Windows Pixmap</b>	x10 compatible
<b>XWD - X-Windows Dump</b>	x10 compatible

## High-end graphics formats

<b>PSD - Adobe Photoshop File Format</b>	Version 4.0
<b>AI - Adobe Illustrator File Format</b>	Versions through 7.0
<b>CDR - Corel Draw</b>	Versions through 8.0
<b>DSF - Micrografx Designer</b>	Windows 95, version 6.0
<b>DWG - AutoCAD Native Drawing Format</b>	Versions 12 through 14
<b>IGES - Initial Graphics Exchange Specification</b>	Version 5.1
<b>PDF - Portable Document Format</b>	Acrobat version 2.1, 3.0, 4.0, including Japanese PDF
<b>PS - Postscript</b>	Level 2

## Presentation formats

<b>Microsoft PowerPoint for Macintosh</b>	Version 4.0, 98
<b>Corel Presentations</b>	Version 8.0 and 9.0
<b>Novell Presentations</b>	Versions 3.0 and 7.0
<b>Harvard Graphics for DOS</b>	Versions 2.x and 3.x
<b>Harvard Graphics</b>	Windows versions
<b>Freelance 96</b>	Freelance 96
<b>Freelance for Windows 95</b>	SmartSuite 97 and Millennium
<b>Freelance for Windows</b>	Version 1.0 and 2.0
<b>Freelance for OS/2</b>	Versions through 2.0
<b>Microsoft PowerPoint for Windows</b>	Versions through 7.0
<b>Microsoft PowerPoint 97</b>	PowerPoint 97
<b>Microsoft PowerPoint 2000</b>	PowerPoint 2000

## Compressed and encoded formats

<b>ZIP PKWARE</b>	Versions through 2.0g
<b>GZIP</b>	No specific version

<b>LZA Self Extracting Compress</b>	No specific version
<b>LZH Compress</b>	No specific version
<b>Microsoft Binder</b>	Version 7.0, Binder 97
<b>MIME (text mail)</b>	No specific version
<b>UUEncode</b>	No specific version
<b>UNIX<sup>®</sup> Compress</b>	No specific version
<b>UNIX TAR</b>	No specific version

## **Other**

<b>vCard Electronic Business Card</b>	Version 2.1
<b>Executable (EXE, DLL)</b>	No specific version
<b>Executable for Windows NT</b>	No specific version
<b>MSG (text only)</b>	Microsoft Outlook mail format
<b>Microsoft Project (text only)</b>	Project 98



---

## Chapter 10. Rights management

This chapter introduces the EIP rights management feature and describes rights managements concepts. It explains marking techniques that you can use to protect your property.

---

### Protecting your intellectual property

Multimedia objects stored in digital form are your intellectual property. Protection of these objects can be critical to your business objectives, especially when these objects are on the World Wide Web where copying is relatively easy. You can use the marking technology provided with Content Manager to deter unauthorized use of your intellectual property by marking your multimedia digital objects for protection or by fingerprinting your objects for identification.

You can apply a mark to your valuable objects to:

- Identify the source, to deter unauthorized copying or reusing. This is known as a watermark and is typically visible.
- Identify the recipient of the content, to deter unauthorized copying or reusing. This is known as a fingerprint and is typically invisible.
- Provide a contact for obtaining additional information.
- Give information, such as time and date, for use in a value-added chain of distribution.

You can mark your digital objects before the object is delivered to your customers. Both watermarks and fingerprints can be applied before delivery. However, applying a fingerprint before delivery implies that the recipient is known and might require the mark to be applied dynamically in the delivery process. Applying the mark from your own controlled environment before delivery increases the security, because there is less risk of tampering.

You can apply marks at several stages in the management and delivery process. Your situation will influence what is appropriate. Marks can be applied at the following points in your process:

- Before the object is stored

If a common mark is to be used for the object (for example, a visible watermark to identify the owner), you can apply the mark before or while storing the object. You can store both the original, unmarked object and the marked object in your Content Manager system. Or you can store only the marked object and keep the unmarked object in a separate repository.
- After the object is stored

If you want to mark objects that are stored in your Content Manager system, you can retrieve the object, mark it, and either replace the unmarked object with the marked version or store the marked object as a new item.
- When the object is retrieved

If the mark to be applied varies based on the recipient, you can apply the mark dynamically after the object is retrieved. The marked object can then be delivered instead of the original object.

If you have many legacy objects in your system that are not marked and you do not want to take the time or use the resources to go back and mark each object, you can mark objects when they are retrieved.

---

## Using marking techniques

You have a variety of content-marking techniques to choose from. Each technique addresses a specific problem and differs in its resistance to removal and modification.

Markings are characterized by:

- Information conveyed

### **Watermark**

Identifies the source of the content. It can contain information such as the owner and version of the object.

### **Fingerprint**

Identifies the recipient of the content. It can contain information such as where and to whom the object was delivered.

- Visibility

### **Visible**

The mark is visible and can be noticed.

### **Invisible**

The mark is hidden within an image.

- Integrity

### **Fragile**

The mark is broken by any modification.

### **Robust**

The mark resists modifications to the marked object such as re-sizing, compressing, rotating, and clipping.

- Time of application

- When the object is captured
- When the object is stored
- When the object is retrieved for distribution to a customer
- When the object is received at the recipient's workstation

- Location

- If a visible mark is intended to be a deterrent to illegal reuse, it can be applied to a large portion of an image. The mark can be moved to overlap a more textured area of the image to make it more difficult to remove.
- If an invisible mark is used, a textured area of the image allows the data to be embedded with the least effect on the image.
- If a visible mark is intended to denote ownership, it can be placed unobtrusively in a corner of the image.
- If both a visible and an invisible mark are to be used, the visible mark should be applied first.

- Format

### **Binary**

The mark can be a random sequence of bits repeated throughout the image. This random sequence is the key that can be used to mark or unmark the image.

The mark can also be an image.

#### **Structured data**

The mark can be embedded textual data.

### **Visible marking**

A visible mark is a transparent mask that is placed over an image in such a way that both the mask and image are visible. A visible mark that is difficult to remove is an effective deterrent to misappropriation of your objects.

Use a visible mark for the following situations:

- When you want to provide images for your customers to review, yet discourage your customers from reusing the review copies
- When you want to use an image in an advertisement on the World Wide Web

### **Invisible marking**

An invisible mark is data hidden within an image in such a way that the image appears to be unaltered. An application is needed to apply, detect, and decipher the mark.

Use an invisible mark for the following situations:

- When you want to embed information to identify ownership and discourage illegal copies of marked objects (watermarking)
- When you want to embed information to track a distribution path (fingerprinting)
- When you want to embed an annotation or a caption in an image





---

## Chapter 11. Accessibility features

This product includes a number of features that make it more accessible for people with disabilities. These features include:

- The ability to operate all features using the keyboard instead of the mouse.
- Support for enhanced display properties
- Options for video and audio alert cues
- Compatibility with assistive technologies
- Compatibility with operating system accessibility features
- Accessible documentation formats

---

### Keyboard input and navigation

The following features are available for keyboard input and navigation:

#### Keyboard input

You can use the keyboard instead of a mouse to operate the product.

Menu items and controls provide access keys that allow you to activate a control or select a menu item directly from the keyboard. These keys are self-documenting; the access keys are underlined on the control or menu where they appear.

#### Keyboard focus

In Windows-based systems, the position of the keyboard focus is highlighted, indicating which area of the window is active and where your keystrokes will have an effect.

#### Response time adjustments

In Windows-based systems, you can adjust response times through your control panel.

---

### Features for accessible display

The clients have a number of features that enhance the user interface and improve accessibility for users with low vision. These enhancements include support for high-contrast settings and customizable font properties.

#### High-contrast mode

The clients support the high-contrast mode option that is provided by the operating system. This feature supports a higher contrast between background and foreground colors.

#### Font settings

In Windows-based systems, you can specify display settings that determine the color, size, and font for the text in menus and dialog windows. The client allows you to select the font for the document list.

#### Non-dependence on color

You do not need to distinguish between colors in order to use any function of this product.

---

## **Alternative alert cues**

In Windows-based systems, the SoundSentry feature can be used to provide visual feedback for general application and system alerts such as warning beeps. You can also adjust the volume of sound alerts.

---

## **Compatibility with assistive technologies**

The clients are compatible with screen reader applications such as Narrator and Via Voice. The clients have properties required for these accessibility applications to make onscreen information available to visually impaired users.

---

## **Accessible documentation**

Documentation for this product is available in PDF format. You can convert the PDF files to HTML or text using free tools available from Adobe at [access.adobe.com](http://access.adobe.com). This allows users to view documentation according to the display preferences set in their browsers. It also allows the use of screen readers and other assistive technologies.

---

## 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 give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation  
Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokyo 106, 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  
J74/G4  
555 Bailey Avenue  
San Jose, CA 95141  
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, results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

#### COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

---

## Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

IBM	DisplayWrite	PowerPC
400	e-business	PTX
Advanced Peer-to-Peer Networking	HotMedia	QBIC
AIX	Hummingbird	RS/6000
AIXwindows	ImagePlus	SecureWay
APPN	IMS	SP
AS/400	Micro Channel	VideoCharger
C Set ++	MQSeries	Visual Warehouse
CICS	MVS/ESA	VisualAge
DATABASE 2	NetView	VisualInfo
DataJoiner	OS/2	WebSphere
DB2	OS/390	
DB2 Universal Database	PAL	

Approach, Domino, Lotus, Lotus 1-2-3, Lotus Notes and SmartSuite are trademarks or registered trademarks of the Lotus Development Corporation in the United States, other countries, or both.

Intel and Pentium are trademarks or registered trademarks of Intel Corporation in the United States, other countries, or both.

Microsoft, Windows, and Windows NT are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, and service names may be trademarks or service marks of others.



---

## Glossary

This glossary defines terms and abbreviations specific to this system. Terms shown in *italics* are defined elsewhere in this glossary.

### A

**abstract class.** An object-oriented programming *class* that represents a concept; classes derived from it represent implementations of the concept. You cannot construct an object of an abstract class; that is, it cannot be instantiated.

**access control.** The process of ensuring that certain functions and stored *objects* can be accessed only by authorized users in authorized ways.

**access control list.** A list consisting of one or more user IDs or user groups and their associated *privileges*. You use access control lists to control user access to *search templates* in the Enterprise Information Portal system.

**action list.** An approved list of the actions, defined by a system administrator or some other *workflow coordinator*, that a user can perform in a *workflow* or document routing process.

**ADSM.** See *Tivoli® Storage Manager*.

**API.** See *application programming interface*.

**application programming interface (API).** A software interface that enables applications to communicate with each other. An API is the set of programming language constructs or statements that can be coded in an application program to obtain the specific functions and services provided by the underlying licensed program.

**attribute.** A unit of data that describes a certain characteristic or property (for example, name, address, age, and so forth) of an item, and which can be used to locate that item. An attribute has a type, which indicates the range of information stored by that attribute, and a value, which is within that range. For example, information about a file in a multimedia file system, such as title, running time, or encoding type (MPEG1, H.263, and so forth). For Enterprise Information Portal, see also *federated attribute* and *native attribute*.

**Audio/Video Interleaved (AVI).** A RIFF (*Resource Interchange File Format*) file specification that permits audio and video data to be interleaved in a file. The

separate tracks can be accessed in alternate chunks for playback or recording while maintaining sequential access on the file device.

**AVI.** See *Audio/Video Interleaved*.

### B

**binary large object (BLOB).** A sequence of bytes with a size ranging from 0 bytes to 2 gigabytes. This string does not have an associated code page and character set. Image, audio, and video objects are stored in BLOBs.

**BLOB.** See *binary large object*.

### C

**cache.** A special-purpose buffer, smaller and faster than main storage, used to hold a copy of data that can be accessed frequently. Use of a cache reduces access time, but might increase memory requirements.

**cardinality.** The number of rows in a database table.

**CGI.** See *Common Gateway Interface*.

**CGI script.** A computer program that runs on a Web server and uses the *Common Gateway Interface (CGI)* to perform tasks that are not usually done by a Web server (for example, database access and form processing). A CGI script is a CGI program that is written in a scripting language such as Perl.

**child component.** Optional second or lower level of a hierarchical *item type*. Each child component is directly associated with the level above it.

**CIF.** See *common interchange file*.

**CIU.** See *common interchange unit*.

**class.** In object-oriented design or programming, a model or template that can be instantiated to create objects with a common definition and therefore, common properties, operations, and behavior. An object is an instance of a class.

**client application.** An application written with the object-oriented or Internet APIs to access *content servers* from Enterprise Information Portal.

**client/server.** In communications, the model of interaction in distributed data processing in which a program at one site sends a request to a program at

another site and awaits a response. The requesting program is called a client; the answering program is called a server.

**collection.** A group of objects with a similar set of management rules.

**combined search.** A query that combines one or more of the following types of searches: *parametric*, text, or image.

**Common Gateway Interface (CGI).** A standard for the exchange of information between a Web server and programs that are external to it. The external programs can be written in any programming language that is supported by the operating system on which the Web server is running. See *CGI script*.

**common interchange file (CIF).** A file that contains one ImagePlus Interchange Architecture (IPIA) data stream.

**common interchange unit (CIU).** The independent unit of transfer for a common interchange file (CIF). It is the part of the CIF that identifies the relationship to the receiving database. A CIF can contain multiple CIUs.

**component.** Generic term for a *root component* or a *child component*.

**connector class.** Object-oriented programming *class* that provides standard access to APIs that are native to specific *content servers*.

**constructor.** In programming languages, a method that has the same name as a class and is used to create and initialize objects of that class.

**content server.** A software system that stores multimedia and business data and the related metadata required for users to work with that data. Content Manager and Content Manager ImagePlus for OS/390 are examples of content servers.

**cursor.** A named control structure used by an application program to point to a specific row within some ordered set of rows. The cursor is used to retrieve rows from the set.

## D

**data format.** See *MIME type*.

**datastore.** (1) Generic term for a place (such as a database system, file, or directory) where data is stored. (2) In an application program, a virtual representation of a *content server*.

**DDO.** See *dynamic data object*.

**document.** An *item* that can be stored, retrieved, and exchanged among Content Manager systems and users

as a separate unit. It can be any multimedia digital object. A single document can include varied types of content, including for example, text, images, and spreadsheets.

**document type definition (DTD).** The rules that specify the structure for a particular class of XML documents. The DTD defines the structure with elements, attributes, and notations, and it establishes constraints for how each element, attribute, and notation can be used within the particular class of documents. A DTD is analogous to a database schema in that the DTD completely describes the structure for a particular markup language.

**DTD.** See *document type definition*.

**dynamic data object (DDO).** In an application program, a generic representation of a stored object that is used to move that object in to, and out of, storage.

## E

**extended data object (XDO).** In an application program, a generic representation of a stored complex multimedia *object* that is used to move that object in to, and out of, storage. XDOs are most often contained within *DDOs*.

**Extensible Markup Language (XML).** A standard metalanguage for defining markup languages that was derived from, and is a subset of, SGML. XML omits the more complex and less-used parts of SGML and makes it much easier to write applications to handle document types, author and manage structured information, and transmit and share structured information across diverse computing systems. The use of XML does not require the robust applications and processing that is necessary for SGML. XML is being developed under the auspices of the World Wide Web Consortium (W3C).

## F

**feature.** The visual content information that is stored in the image search server. Also, the visual traits that image search applications use to determine matches. The four *QBIC* features are average color, histogram color, positional color, and texture.

**federated attribute.** An Enterprise Information Portal metadata category that is mapped to *native attributes* in one or more *content servers*. For example, the federated attribute, policy number, can be mapped to an *attribute*, policy num, in Content Manager and to an attribute, policy ID, in Content Manager ImagePlus for OS/390.

**federated collection.** A grouping of objects that results from a *federated search*.



**federated datastore.** Virtual representation of any number of specific *content servers*, such as Content Manager.

**federated entity.** An Enterprise Information Portal metadata object that is comprised of *federated attributes* and optionally associated with one or more *federated text indexes*.

**federated search.** A query issued from Enterprise Information Portal that simultaneously searches for data in one or more *content servers*, which can be heterogeneous.

**federated text index.** An Enterprise Information Portal metadata object that is mapped to one or more *native text indexes* in one or more *content servers*.

**file system.** In AIX, the method of partitioning a hard drive for storage.

**folder.** A container used to organize *objects*, which can be other folders or *documents*.

**folder manager.** The Content Manager model for managing data as online documents and folders. You can use the folder manager APIs as the primary interface between your applications and the Content Manager content servers.

## H

**handle.** A character string that represents an object, and is used to retrieve the object.

**history log.** A file that keeps a record of activities for a *workflow*.

**HTML.** See *Hypertext Markup Language*.

**Hypertext Markup Language (HTML).** A markup language that conforms to the SGML standard and was designed primarily to support the online display of textual and graphical information that includes hypertext links.

## I

**Image Object Content Architecture (IOCA).** A collection of constructs used to interchange and present images.

**index.** To add or edit the attribute values that identify a specific *item* or *object* so that it can be retrieved later.

**index class.** See *item type*.

**index class subset.** In earlier Content Manager, a view of an *index class* that an application uses to store, retrieve, and display folders and objects.

**index class view.** In earlier Content Manager, the term used in the APIs for *index class subset*.

**information mining.** The automated process of extracting key information from text (summarization), finding predominant themes in a collection of documents (categorization), and searching for relevant documents using powerful and flexible queries.

**interchange.** The capability to import or export an image with its index from one Content Manager ImagePlus for OS/390 system to another ImagePlus system using a *common interchange file* or *common interchange unit*.

**IOCA.** See *Image Object Content Architecture*.

**item.** Generic term for the smallest unit of information that Enterprise Information Portal administers. Each item has an identifier. For example, an item might be a *folder* or a *document*.

**item type.** A template for defining and later locating like *items*, consisting of a *root component*, zero or more *child components*, and a classification.

**item type classification.** A categorization within an *item type* that further identifies the *items* of that item type. All items of the same item type have the same item type classification.

Content Manager supplies the following item type classifications: *folder*, *document*, object, video, image, and text; users can also define their own item type classifications.

**iterator.** A class or construct that you use to step through a collection of objects one at a time.

## J

**JavaBeans.** A platform-independent, software component technology for building reusable Java components called "beans." After they are built, these beans can be made available for use by other software engineers or can be used in Java applications. Using JavaBeans, software engineers can manipulate and assemble beans in a graphical drag-and-drop development environment.

**Joint Photographic Experts Group (JPEG).** (1) A group that worked to establish the standard for the compression of digitized continuous-tone images. (2) The standard for still pictures developed by this group.

**JPEG.** See *Joint Photographic Experts Group*.

## K

**key field.** See *attribute*.

## L

**LAN.** See *local area network*.

**library client.** The component of a Content Manager system that provides a low-level programming interface for the library system. The library client includes APIs that are part of the software developer's kit.

**library server.** The component of a Content Manager system that stores, manages, and handles queries on *items*.

**link.** A directional relationship between two *items*: the parent and the child. You can use a set of links to model one-to-many associations. Contrast with *reference*.

**local area network (LAN).** A network in which a set of devices are connected to one another for communication and that can be connected to a larger network.

## M

**media archiver.** A physical device that is used for storing audio and video stream data. The VideoCharger is a type of media archiver.

**media server.** An AIX-based component of the Content Manager system that is used for storing and accessing video files.

**method.** In Java design or programming, the software that implements the behavior specified by an operation. Synonymous with member function in C++.

**MIME type.** An Internet standard for identifying the type of object being transferred across the Internet. MIME types include several variants of audio, image, and video. Each object has a MIME type.

**multimedia.** Combining different media elements (text, graphics, audio, still image, video, animation) for display and control from a computer.

**multimedia file system.** A *file system* that is optimized for the storage and delivery of video and audio.

**Multipurpose Internet Mail Extensions (MIME)** . See *MIME type*.

## N

**native attribute.** A characteristic of an object that is managed on a specific *content server* and that is specific to that content server. For example, the *key field policy num* might be a native attribute in a Content Manager content server, whereas the field *policy ID* might be a native attribute in an Content Manager OnDemand content server.

**native entity.** An *object* that is managed on a specific *content server* and that is comprised of *native attributes*. For example, Content Manager *index classes* are native entities comprised of Content Manager *key fields*.

**native text index.** An index of the text *items* that are managed on a specific *content server*. For example, a single text search index on a Content Manager content server.

**network table file.** A text file that contains the system-specific configuration information for each node in a Content Manager system. Each node in the system must have a network table file that identifies the node and lists the nodes that it needs to connect to.

The name of a network table is FRNOLINT.TBL.

## O

**object.** Any digital content that a user can store, retrieve and manipulate as a single unit, for example, JPEG images, MP3 audio, AVI video, and a text block from a book.

**Object Linking and Embedding (OLE).** A Microsoft® specification for both linking and embedding applications so that they can be activated from within other applications.

**object server.** See *resource manager*.

**object server cache.** See *resource manager cache*.

**OLE.** See *Object Linking and Embedding*.

**overlay.** A collection of predefined data such as lines, shading, text, boxes, or logos, that can be merged with variable data on a page during printing.

## P

**package.** A collection of related *classes* and interfaces that provides access protection and namespace management.

**parametric search.** A query for *objects* that is based on the *properties* of the objects.

**part.** See *object*.

**persistent identifier (PID).** An identifier that uniquely identifies an *object*, regardless of where it is stored. The PID consists of both an item ID and a location.

**PID.** See *persistent identifier*.

**privilege.** The right to access a specific *object* in a specific way. Privileges includes rights such as creating, deleting, and selecting objects stored in the system. Privileges are assigned by the administrator.

**privilege set.** A collection of *privileges* for working with system components and functions. The administrator assigns privilege sets to users (user IDs) and *user groups*.

**property.** A characteristic of an *object* that describes the object. A property can be changed or modified. Type style is an example of a property.

## Q

**QBIC.** See *query by image content*.

**query by image content (QBIC).** A query technology that enables searches based on visual content, called features, rather than plain text. Using QBIC, you can search for objects based on their visual characteristics, such as color and texture.

**query string.** A character string that specifies the properties and property values for a query. You can create the query string in an application and pass it to the query.

## R

**rank.** An integer value that signifies the relevance of a given part to the results of a query. A higher rank signifies a closer match.

**README file.** A file that should be viewed before the program associated with it is installed or run. A README file typically contains last-minute product information, installation information, or tips for using the product.

**reference.** Single direction, one-to-one association between a root or *child component* and another *root component*. Contrast with *link*.

**release.** To remove suspend criteria from an *item*. A suspended item is released when the criteria have been met, or when a user with proper authority overrides the criteria and manually releases it.

**Remote Method Invocation (RMI).** A set of APIs that enables distributed programming. An object in one Java Virtual Machine (JVM) can invoke methods on objects in other JVMs.

**render.** To take data that is not typically image-oriented and depict or display it as an image. In Content Manager, word-processing documents can be rendered as images for display purposes.

**Resource Interchange File Format (RIFF).** Used for storing sound or graphics for playback on different types of computer equipment.

**resource manager.** The component of a Content Manager system that manages *objects*. These objects are referred to by *items* stored on the *library server*.

**resource manager cache.** The working storage area for the *resource manager*. Also called the *staging area*.

**RIFF.** See *Resource Interchange File Format*.

**RMI server.** A server that implements the Java *Remote Method Invocation (RMI)* distributed object model.

**root component.** The first or only level of a hierarchical *item type*, consisting of related system- and user-defined *attributes*.

## S

**search criteria.** In Enterprise Information Portal, specific fields that an administrator defines for a *search template* that limit or further define choices available to the *users*.

**search template.** A form, consisting of *search criteria* designed by an administrator, for a specific type of federated search. The administrator also identifies the *users* and *user groups* who can access each search template.

**semantic type.** The usage or rules for an *item*. Base, annotation, and note are semantic types supplied by Content Manager; users can also define their own semantic types.

**server definition.** The characteristics of a specific *content server* that uniquely identify it to Enterprise Information Portal.

**server inventory.** The comprehensive list of *native entities* and *native attributes* from specified *content servers*.

**server type definition.** The list of characteristics, as identified by the administrator, required to uniquely identify a custom server of a certain type to Enterprise Information Portal.

**staging.** The process of moving a stored *object* from an offline or low-priority device back to an online or higher priority device, usually on demand of the system or on request of a user. When a user requests an object stored in permanent storage, a working copy is written to the *staging area*.

**staging area.** The working storage area for the *resource manager*. Also referred to as *resource manager cache*.

**streamed data.** Any data sent over a network connection at a specified rate. A stream can be one data type or a combination of types. Data rates, which are expressed in bits per second, vary for different types of streams and networks.

**subclass.** A *class* that is derived from another class. One or more classes might be between the class and subclass.

**superclass.** A *class* from which a class is derived. One or more classes might be between the class and superclass.

**suspend.** To remove an *object* from its *workflow* and define the suspension criteria needed to activate it. Later activating the object enables it to continue processing.

## T

**thin client.** A client that has little or no installed software but has access to software that is managed and delivered by network servers that are attached to it. A thin client is an alternative to a full-function client such as a workstation.

**Tivoli Storage Manager (TSM).** A *client/server* product that provides storage management and data access services in a heterogeneous environment. It supports various communication methods, provides administrative facilities to manage the backup and storage of files, and provides facilities for scheduling backup operations.

**TSM.** See *Tivoli Storage Manager*.

**TSM volume.** A logical area of storage that is managed by *Tivoli Storage Manager*.

## U

**uniform resource locator (URL).** A sequence of characters that represent information resources on a computer or in a network such as the Internet. This sequence of characters 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. For example, in the context of the Internet, these are abbreviated names of some protocols used to access various information resources: http, ftp, gopher, telnet, and news.

**user.** In Enterprise Information Portal, anyone who is identified in the Enterprise Information Portal administration program.

**user exit.** A point in an IBM-supplied program at which a user exit routine can be given control.

**user exit routine.** A user-written routine that receives control at predefined *user exits*.

**user group.** A group consisting of one or more defined individual *users*, identified by a single group name.

**user mapping.** Associating Enterprise Information Portal user IDs and passwords to corresponding user IDs and passwords in one or more content servers. User mapping enables single logon to Enterprise Information Portal and multiple *content servers*.

## V

**volume.** A representation of an actual physical storage device or unit on which the objects in your system are stored.

## W

**wildcard character.** A special character such as an asterisk (\*) or a question mark (?) that can be used to represent one or more characters. Any character or set of characters can replace a wildcard character.

**workflow.** In Enterprise Information Portal, a sequence of *work steps*, and the rules governing those steps, through which a *work packet*, *document*, or *folder* travels while it is being processed.

For example, claims approval would describe the process that an individual insurance claim must follow for approval.

**workflow state.** The status of an entire *workflow*.

**work item.** In earlier Content Manager workflow and Enterprise Information Portal advanced workflow, any work activity that is active within a *workflow*.

**worklist.** A collection of *work items*, *documents*, or *folders* that are assigned to a user.

**work packet.** In Enterprise Information Portal Version 7.1, a collection of *documents* that is routed from one location to another. Users access and work with work packets through *worklists*.

**work state.** The status of an individual *work item*, *document*, or *folder*.

**work step.** A discrete point in a *workflow* or *document routing process* through which an individual *work item*, *document*, or *folder* must pass.

## X

**XDO.** See *extended data object*.

**XML.** See *Extensible Markup Language*.

---

# Index

## A

- access control list
  - moving domains 34
- action list
  - defining 86, 89
  - predefined actions 89
- actions, defining 89
- administration client
  - creating
    - search criteria 24
    - search templates 23
    - workflow 89
  - defining
    - action list 89
    - actions 89
    - worklists 88
- administrative domain 30
- AllPrivSet 31

## C

- catalog
  - adding 56
  - adding training documents 57
  - deleting 56
  - evaluating 60
  - renaming 56
  - training 64
- cmbcc2mime.ini 11
- cmbmime2app 12
- collection
  - assigning to a domain 33
- collections
  - moving domains 34
- configuring image search 98
- connectors 4
- content server
  - defining 15
- content viewer option 5
- customizing MIME types 11

## D

- domain 34
- domains
  - create 31
  - sub administrator privileges 32
  - super administrator privileges 32
  - understanding 31

## E

- EIP
  - administration component 3
  - connector toolkit 5
  - connectors 4
  - content viewer client 5
  - image search client 5
  - information center component 5

- EIP (*continued*)
  - information mining option 4
  - text search client 5
  - Web Crawler option 5
- EIP components
  - administration 3
  - connectors 4
  - content viewer 5
  - image search 5
  - information center 5
  - information mining 4
  - operating system compatibility 3
  - text search 5
  - Web Crawler 5
- Enterprise Information Portal
  - components 2
  - creating
    - search criteria 24
    - search templates 23
    - workflow 89
  - defining
    - action list 89
    - actions 89
    - worklists 88

## G

- grant privilege set 29

## I

- image search
  - assigning an alias 100
  - configuring 98
  - setting up 98
  - verifying the connection 100
- image search option 5
- information mining 35
  - an example 39
  - building a taxonomy 54
  - components of 36
  - defining a taxonomy 56
  - description 35, 54
  - evaluating a taxonomy 60
  - getting started 55
  - installing 55
  - locking mechanism 55
  - selecting training documents 57
  - service 35
  - supported document formats 42
  - supported languages 42
  - target group 38
  - training a taxonomy 64
  - using the information mining 35
  - using WAS 55
  - working in a business environment 38
- ItemAdminPrivSet 31

## L

- LDAP
  - changing to another LDAP server 28
  - configuring 28
  - importing 28
  - LDAP User Registry Import Utility 28
  - loading sample data 100
  - loading text and image search documents 103

## M

- metadata store 35
- MIME type file
  - changing for clients 12
  - changing for servers 11

## P

- planning for
  - Enterprise Information Portal 2
- privilege group 29
- privilege set 27, 29
  - creating 29
  - moving domains 34

## R

- resource manager
  - assigning to a domain 33
  - assigning users to 30
- resource manager, moving domains 34

## S

- sample loader program, running 103
- search criteria
  - defining and mapping 24
- search templates, creating 23
- server inventory 15
- starting
  - workflow builder 89
- SuperAdminPrivSet 31

## T

- taxonomy
  - using the 54
- text search
  - setting up 97
  - XML support 97

## U

- user 27
  - moving domains 33
  - privilege set 29

- user group 30
  - moving domains 34
- user ID 27

## W

- Web Crawler
  - EIP option 5
- work packet, description 86
- workflow
  - concepts 85
  - creating 89
  - planning for 85
- workflow builder
  - creating a workflow 89
  - description 87
  - starting 89
- workflow feature
  - components 87
  - configuring 85
- worklist
  - defining 88
  - description 86





Program Number: 5724-B43



Printed in the United States of America  
on recycled paper containing 10%  
recovered post-consumer fiber.

SC27-1346-00

