

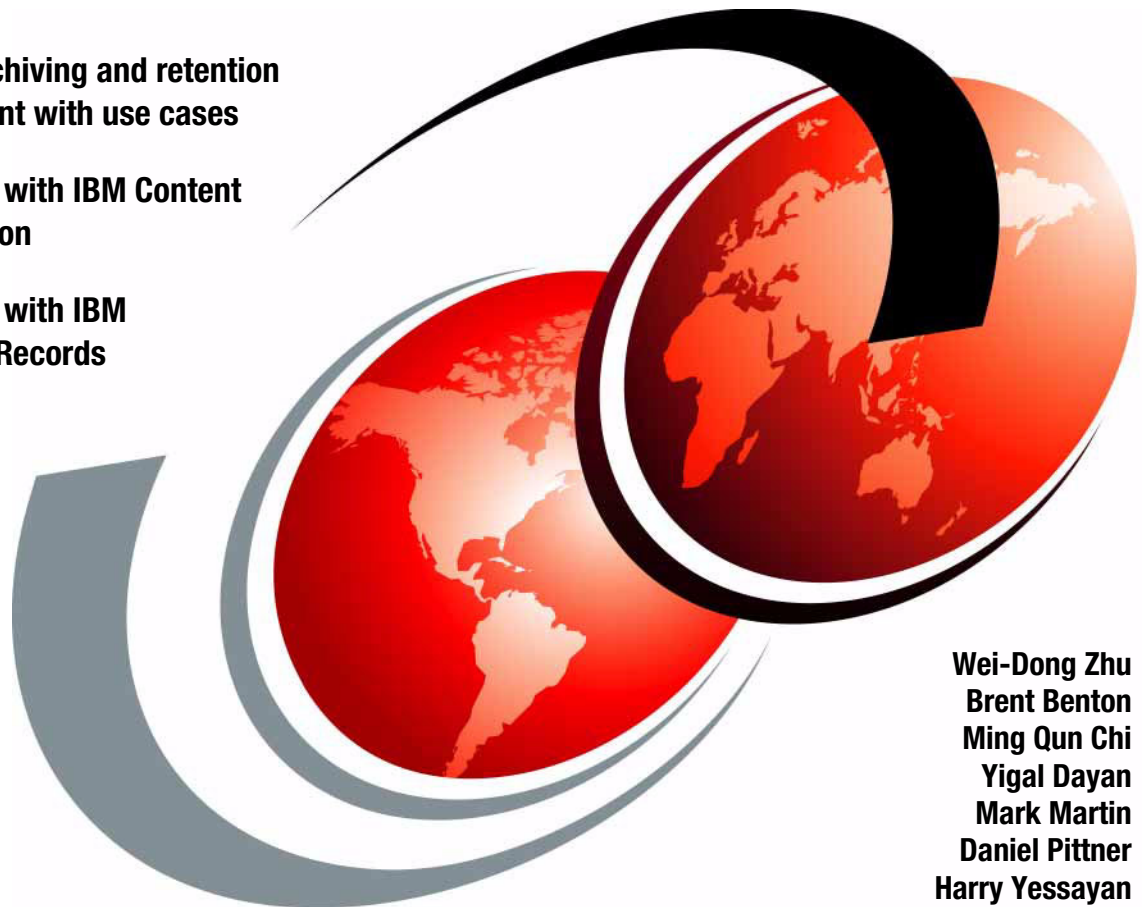


# Creating Value-Based Archiving Solutions with IBM Content Collector

Content archiving and retention management with use cases

Integration with IBM Content Classification

Integration with IBM Enterprise Records



Wei-Dong Zhu  
Brent Benton  
Ming Qun Chi  
Yigal Dayan  
Mark Martin  
Daniel Pittner  
Harry Yessayan

[ibm.com/redbooks](http://ibm.com/redbooks)

**Redbooks**





International Technical Support Organization

**Creating Value-Based Archiving Solutions with IBM  
Content Collector**

January 2013

**Note:** Before using this information and the product it supports, read the information in “Notices” on page ix.

**First Edition (January 2013)**

This edition applies to Version 3, Release 0 of IBM Content Collector (product number 5724-V57) and Version 3, Release 0 of IBM Content Collector for SAP Applications (product number 5725-B46).

**© Copyright International Business Machines Corporation 2013. All rights reserved.**

Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

# Contents

<b>Notices</b> .....	ix
Trademarks .....	x
 <b>Preface</b> .....	 xi
The team who wrote this book .....	xii
Now you can become a published author, too! .....	xiv
Comments welcome .....	xiv
Stay connected to IBM Redbooks .....	xiv
 <b>Chapter 1. Value-based archiving and defensible disposal overview</b> ....	 1
1.1 Business problems .....	2
1.1.1 Introduction .....	2
1.1.2 The growth of archiving .....	3
1.2 IBM solutions .....	4
1.2.1 IBM Information Lifecycle Governance .....	4
1.2.2 Value-based archiving with IBM Content Collector .....	6
1.3 IBM Content Collector overview .....	8
1.3.1 Architectural overview .....	8
1.3.2 IBM Content Collector components .....	9
1.3.3 IBM Content Collector features .....	10
1.4 Typical archiving use cases and scenarios .....	12
1.5 Conclusion .....	15
 <b>Chapter 2. Example use cases</b> .....	 17
2.1 Use case 1: Email archiving for compliance and storage management ..	18
2.2 Use case 2: Email archiving with content classification .....	19
2.3 Use case 3: Email archiving with records declaration .....	20
2.4 Use case 4: File system archiving with records declaration .....	21
2.5 Conclusion .....	21
 <b>Chapter 3. Dimensions of content archiving themes</b> .....	 23
3.1 Dimensions of content archiving with IBM Content Collector .....	24
3.2 Storage management .....	25
3.2.1 Staging rollout for maximizing storage savings .....	26
3.2.2 Document stubbing .....	28
3.2.3 Storage management for file shares .....	30
3.2.4 Storage management for email .....	38
3.2.5 Storage management for Microsoft SharePoint .....	43
3.3 Compliance archiving .....	48

3.3.1 Compliance archiving for email . . . . .	49
3.3.2 Compliance archiving for Microsoft SharePoint . . . . .	56
3.3.3 Compliance archiving for IBM Connections . . . . .	56
3.4 Business process management . . . . .	57
3.4.1 Business process management for file shares . . . . .	58
3.4.2 Business process management for email . . . . .	67
3.5 Use case 1A and 1B: Email archiving for compliance and storage . . . . .	75
3.6 Conclusion. . . . .	76
<b>Chapter 4. Designing, adapting, and deploying task routes . . . . .</b>	<b>77</b>
4.1 Dynamically calculating document retention . . . . .	78
4.1.1 Automatically calculating document retention . . . . .	78
4.1.2 Manually setting document retention . . . . .	83
4.2 Adjusting collectors . . . . .	89
4.2.1 Configuring schedules. . . . .	89
4.2.2 Collector and task route filtering . . . . .	91
4.3 Optimizing task routes for maintainability . . . . .	93
4.3.1 Impact of using multiple connections for Microsoft SharePoint. . . . .	93
4.3.2 Microsoft SharePoint version series task route design . . . . .	95
4.3.3 Creation of user-defined metadata . . . . .	98
4.3.4 Impact of using multiple collection sources . . . . .	98
4.3.5 Impact of using multiple collectors . . . . .	99
4.3.6 Impact of using multiple task routes . . . . .	100
4.3.7 Strategies for minimizing the number of task routes . . . . .	100
4.4 Promoting task routes from development to production systems . . . . .	106
4.4.1 Understanding task route dependencies. . . . .	107
4.4.2 Checklist for task route migration . . . . .	107
4.5 Using the Expression Editor . . . . .	109
4.5.1 Avoiding the need for nested decision points . . . . .	110
4.5.2 Using list lookups . . . . .	110
4.6 Extending IBM Content Collector . . . . .	114
4.6.1 Choosing the correct extension strategy for your scenario . . . . .	114
4.6.2 Extending the source system or target system . . . . .	115
4.6.3 Using the Script Connector . . . . .	115
4.6.4 Using the IBM Content Collector Software Development Kit . . . . .	116
4.7 Conclusion. . . . .	117
<b>Chapter 5. Retention management . . . . .</b>	<b>119</b>
5.1 Retention management overview . . . . .	120
5.2 Stubbing lifecycle . . . . .	121
5.3 Expiration Manager . . . . .	128
5.3.1 Looking up the expiration date of a document . . . . .	128
5.3.2 Working with eDiscovery and records management solutions . . . . .	130

5.3.3	Running multiple instances of Expiration Manager	130
5.3.4	Scheduling Expiration Manager execution	132
5.3.5	Optimizing Expiration Manager for performance	132
5.4	Expired stub management	134
5.4.1	Determine the ID of the repository	134
5.4.2	Email	137
5.4.3	Microsoft SharePoint	142
5.4.4	File system	143
5.5	Use case 1C: Lifecycle stubbing and retention management	145
5.5.1	Create the stubbing lifecycle task route	146
5.5.2	Enable the Expiration Manager	149
5.5.3	Create the audit task route (optional)	152
5.6	Conclusion	153
<b>Chapter 6. Document classification</b>		<b>155</b>
6.1	The business value of using IBM Content Classification	156
6.2	IBM Content Classification overview	156
6.3	Basic content classification integration	158
6.3.1	Setting up Content Classification with Content Collector	160
6.3.2	Configuring task route for automated email archiving example	162
6.3.3	A BPM task route example	163
6.3.4	Working with Content Collector email client integration	165
6.4	Content Classification applied to other scenarios	166
6.4.1	Value-based archiving and defensible disposal	167
6.4.2	Using Content Classification for record declaration	167
6.4.3	Using Content Classification for eDiscovery	167
6.5	Using decision plan for value-based archiving and defensible disposal	168
6.5.1	Setting expiration date using Content Classification calculation	168
6.5.2	Decision plan used in the expiration calculation	172
6.5.3	Loading the decision plan for inspection	179
6.5.4	Reproducing disposal decisions made in the past	180
6.6	Generating new facets with Content Classification	181
6.6.1	Generating facets for multiple taxonomies	182
6.6.2	Generating facets using wordlists	183
6.6.3	Creating facets from regular expressions	187
6.6.4	Creating facets with Content Classification user hooks	189
6.6.5	Creating facets with the UIMA client hooks	193
6.7	Reviewing and auditing archived emails and documents	194
6.7.1	Reviewing results of Content Collector's automatic classification	194
6.7.2	Manual audit and feedback	199
6.7.3	Deferred feedback	200
6.7.4	Pitfalls of feedback	202
6.7.5	Using representative datasets	203

6.7.6	Review and feedback through the Classification Center. . . . .	206
6.7.7	Inspection and feedback through the Email Client integration . . . .	207
6.8	Use case 2: Email archiving with content classification . . . . .	207
6.8.1	The decision plan . . . . .	207
6.8.2	The task route . . . . .	210
6.8.3	Upgrading to use case 3 . . . . .	213
6.9	Considerations and guidelines . . . . .	213
6.9.1	Preferred practices . . . . .	213
6.9.2	Limitation considerations. . . . .	218
6.10	Conclusion. . . . .	219
<b>Chapter 7.</b>	<b>Records management integration. . . . .</b>	<b>221</b>
7.1	Options for classifying and declaring records . . . . .	222
7.1.1	Simple retention management versus record declaration. . . . .	222
7.1.2	Determining classification . . . . .	223
7.1.3	Use cases and examples . . . . .	224
7.2	Record declaration requirements . . . . .	225
7.2.1	Prerequisites for record declaration . . . . .	225
7.2.2	Content must be archived before declaration . . . . .	225
7.2.3	Essential information for record declaration . . . . .	226
7.3	Basic record declaration from a Content Collector task route. . . . .	227
7.3.1	Enabling archived documents for declaration . . . . .	228
7.3.2	Content Collector P8 Declare Record task overview . . . . .	229
7.3.3	Configuring record classification - options . . . . .	234
7.3.4	Configuring property mapping - options . . . . .	236
7.3.5	Which options to use. . . . .	239
7.3.6	Determining record classification . . . . .	239
7.3.7	Task route templates for record declaration . . . . .	241
7.4	Use case 3: Email archiving with records declaration. . . . .	243
7.4.1	Deciding which content to declare . . . . .	243
7.4.2	Archiving the content. . . . .	246
7.4.3	Configuring record declaration . . . . .	248
7.4.4	Results in Enterprise Records. . . . .	255
7.4.5	Including logic to handle intermittent failure . . . . .	256
7.5	Use case 4: File system archiving with records declaration . . . . .	258
7.5.1	Scenario and overview . . . . .	258
7.5.2	Deciding which content to declare . . . . .	260
7.5.3	Separating documents for declaration . . . . .	261
7.5.4	Creating the document in P8. . . . .	262
7.5.5	Configuring record declaration . . . . .	262
7.5.6	Post processing. . . . .	267
7.5.7	Results in Enterprise Records. . . . .	267
7.6	Considerations and guidelines . . . . .	267



7.6.1 Preferred practices . . . . .	268
7.6.2 Limitations . . . . .	268
7.6.3 Declaring a version series . . . . .	269
7.6.4 Using deduplication for files . . . . .	272
7.6.5 Considerations for email . . . . .	272
7.6.6 Considerations for large volumes . . . . .	273
7.6.7 Declaring records after content has been archived . . . . .	274
7.7 Conclusion . . . . .	274
<b>Chapter 8. IBM Connections integration . . . . .</b>	<b>275</b>
8.1 Configuring IBM Connections for IBM Content Collector . . . . .	276
8.1.1 Setting up user permissions . . . . .	276
8.1.2 Scale out and backup considerations . . . . .	278
8.2 Archiving a subset of content . . . . .	278
8.2.1 Content filtering for IBM Connections . . . . .	279
8.2.2 Archiving past content from a specific person . . . . .	280
8.3 Configuring eDiscovery Manager for IBM Connections . . . . .	281
8.3.1 Viewing IBM Connections documents in eDiscovery Manager . . . . .	282
8.3.2 Extending searching capabilities . . . . .	283
8.3.3 Considerations for eDiscovery Manager document export . . . . .	287
8.3.4 Conclusion . . . . .	288
<b>Related publications . . . . .</b>	<b>289</b>
IBM Redbooks . . . . .	289
Online resources . . . . .	289
Help from IBM . . . . .	290



# Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.*

**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 websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites 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.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

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

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at <http://www.ibm.com/legal/copytrade.shtml>

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

DB2®  
developerWorks®  
Domino®  
FileNet®  
IBM®

Lotus Notes®  
Lotus®  
Notes®  
Redbooks®  
Redpaper™

Redbooks (logo) ®  
System Storage®  
WebSphere®

The following terms are trademarks of other companies:

Microsoft, Windows, and the Windows logo are 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 Oracle and/or its affiliates.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

# Preface

This IBM® Redbooks® publication describes how the IBM Content Collector family of products can help companies to create value-based archiving solutions. IBM Content Collector provides enterprise-wide content archiving and retention management capabilities. It also provides IT administrators with a high level of control over the archiving environment. From a common interface, organizations can implement policies that define what gets archived from which source system, make decisions about how content gets archived based on the content or metadata of the information, and determine the retention and governance rules associated with that type of content. Content Collector enables IT staff to implement granular archiving policies to collect and archive specific pieces of information.

IBM Content Collector helps with the following tasks:

- ▶ Eliminating point solutions and lowering costs with a unified collection, management, and governance approach that works effectively across a broad range of source systems and information types
- ▶ Appraising, improving understanding of, culling, and properly selecting the information to archive
- ▶ Retaining, holding, and disposing of archived content efficiently and defensibly
- ▶ Eliminating the costs and risks inherent with over-retention

This book covers the basic concepts of the IBM Content Collector product family. It presents an overview explaining how IBM Content Collector provides value-based archiving and a defensible disposal capability in the archiving solutions. With the integration of IBM Content Classification and IBM Enterprise Records, the book also showcases how these products can be used to add more flexibility, power, and capabilities to archiving solutions.

This book is intended for IT architects and solution designers who need to understand and use IBM Content Collector for archiving solution implementations. Use cases are included to provide specific, step-by-step details about implementing common solutions that fulfill some of the general business requirements.

## The team who wrote this book

This book was produced by a team of specialists from around the world working at the International Technical Support Organization, Rochester Center.

**Wei-Dong Zhu** (Jackie) is an Enterprise Content Management Project Leader with ITSO. Jackie joined IBM in 1996 and has more than 10 years of software development experience in accounting, image workflow processing, and digital media distribution. She is a Certified Solution Designer for IBM Content Manager, and has managed many Enterprise Content Management Redbooks publications. Jackie holds a Master of Science degree in Computer Science from the University of Southern California.

**Brent Benton** is an Advisory Software Engineer at the Vancouver Development Center, IBM Canada. He joined IBM through the FileNet® acquisition in 2006. He has 10 years of experience in the fields of Enterprise Content Management and Information Lifecycle Governance. For the past 5 years, he has led the design and development of social media connectors for IBM Content Collector, and now also the Atlas product. Brent holds Business and Philosophy degrees from Simon Fraser University, and an Applied Information Technology Diploma. He authored the IBM developerWorks® article “Understanding the footprint of IBM Content Collector for Microsoft SharePoint.”

**Ming Qun Chi** is an Enterprise Content Management Staff software engineer with the IBM Software Group in China. He joined IBM in 2008 and has more than four years of experience with IBM Content Collector. Currently, he focuses on IBM Content Collector for Email, including performance testing and IBM Content Collector, \_IER\_ IBM Content Classification integration testing. Ming Qun holds a Master's degree in Computer Science from Beijing University of Posts and Telecommunications.

**Yigal Dayan** is an Advisory Software Engineer with IBM Israel. He has 30 years of experience in software development. Yigal has worked at IBM for six years. His areas of expertise include Natural Language Processing, Classification, and Machine Translation.

**Mark Martin** is the Worldwide Marketing Manager responsible for IBM Value-based Archiving and the IBM Content Collector family of products. This is the second Redbooks publication about the archiving topic that he has worked on. Mark is based in Ottawa, Canada. You can communicate with him and keep current with IBM Value-based Archiving on Twitter @MarkMartin\_ECM.

**Daniel Pittner** is a co-Architect for Client and Server Integration in IBM Germany, working on the IBM Content Collector development team. He has worked at IBM for eight years, and has five years of experience in the field of

Information Lifecycle Governance (ILG). Daniel specializes in developing and designing parts of the IBM Content Collector product, and has applied ILG technology with a broad range of clients. He holds a diploma in Computer Science from the University of Cooperative Education in Stuttgart, Germany.

**Harry Yessayan** is a Solution Architect and Managing Consultant for IBM Software Services in the United States. He has more than 15 years of experience in Enterprise Content Management, most recently focusing on solutions for Records and Retention Management and Defensible Disposal using IBM Enterprise Records. He holds a Master of Science degree in Information and Computer Science from University of California, Irvine. Harry is also an author of the Redbooks publication *Understanding IBM FileNet Records Manager*, SG24-7623 (now known as IBM Enterprise Records).

Special appreciation to **Kevin Carmichael**, who contributed to the topic “Optimizing task routes for maintainability” and provided significant assistance with the review process.

Thank you to the following people for their guidance, commitment, and leadership in getting this book published:

Jerry Gill  
Dieter Schieber  
Dirk Seider  
Silke Wastl

Thank you also to the entire IBM Content Collector development teams in Canada and Germany for their generous support and assistance during the production of the book, especially the following people:

Michael Baessler  
Matthias Bierbrauer  
Jennifer Ewasiuk  
Ralf Hauser  
Vincent Lidou  
Juergen Maletz  
David Mills  
Dan Small  
Tom Smith  
IBM Software Group, Canada and Germany

Andy Stalnecker  
IBM Software Group, US  
International Technical Support Organization, Rochester Center

## Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

[ibm.com/redbooks/residencies.html](http://ibm.com/redbooks/residencies.html)

## Comments welcome

Your comments are important to us!

We want our books to be as helpful as possible. Send us your comments about this book or other IBM Redbooks publications in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:

[ibm.com/redbooks](http://ibm.com/redbooks)

- Send your comments in an email to:

[redbooks@us.ibm.com](mailto:redbooks@us.ibm.com)

- Mail your comments to:

IBM Corporation, International Technical Support Organization  
Dept. HYTD Mail Station P099  
2455 South Road  
Poughkeepsie, NY 12601-5400

## Stay connected to IBM Redbooks

- Find us on Facebook:

<http://www.facebook.com/IBMRedbooks>

- Follow us on Twitter:

<http://twitter.com/ibmredbooks>



- ▶ Look for us on LinkedIn:  
<http://www.linkedin.com/groups?home=&gid=2130806>
- ▶ Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:  
<https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm>
- ▶ Stay current on recent Redbooks publications with RSS Feeds:  
<http://www.redbooks.ibm.com/rss.html>





# Value-based archiving and defensible disposal overview

In this chapter we introduce value-based archiving and defensible disposal concepts. We describe existing problems with growing data volumes, and explain how IBM Content Collector can help solve these problems.

In this chapter we discuss the following topics:

- ▶ Business problems
- ▶ IBM solutions
- ▶ IBM Content Collector overview
- ▶ Typical archiving use cases and scenarios

# 1.1 Business problems

This section describes the current business problems with the growing volume of data, and explains how IBM Content Collector solutions can help solve such problems.

## 1.1.1 Introduction

People who are not directly involved in supporting an IT infrastructure are often surprised to hear that during the last five years there has been an average 50 percent year-over-year growth in the volume of data inside an organization. However, that statistic is not surprising to IT managers, who struggle to find a way to support rapidly growing volumes of data with a flat or only slightly increasing budget.

It is also no surprise to them that much of the data under management is “debris” that is outdated and duplicated many times across multiple systems, with no real value to the business. Such over-retention results in many direct and indirect costs:

- ▶ Overspending on a more complex IT environment with more IT resources required to support larger systems, with more storage
- ▶ Higher costs for eDiscovery processing; with more information, processing requests for information takes longer and results in higher review fees
- ▶ More legal risk inherent with a larger information set

IBM Content Collector is a family of content collection and archiving products designed to help curtail over-retention by attacking the problem at the root and minimizing the amount of unnecessary information (the debris) that IT and other stakeholders must deal with.

This publication explains how IBM Content Collector can be a vital part of an effective Information Lifecycle program, and how it supports efforts to reduce cost and risk associated with over-retention of information. Using typical archiving scenarios as illustrations, the book demonstrates how IBM Content Collector can be deployed to meet many different archiving use cases, and how the solution can be extended through existing awareness of and integration with other IBM products to meet advanced information classification, records and retention management, and eDiscovery requirements.

## 1.1.2 The growth of archiving

Interest in archiving solutions is typically driven by two communities with differing but overlapping concerns:

- ▶ IT teams are concerned with the cost and manageability of systems that continue to show year-over-year increases in the volume of data being generated.
- ▶ Legal or Compliance teams are concerned with requirements to meet regulatory obligations.

Traditionally, archiving solutions were designed and deployed as stand-alone solutions or silos that were intended to solve one problem. eMail archiving, for example, was introduced as a way to combat overloaded email systems. By deploying an archive, administrators were able to push old messages out of the email system and thereby reduce the volume on the mail servers. With a smaller message database after archiving, email servers had improved performance, backups required less time, and maintenance was faster and easier.

As more business communication became electronic, record-keeping regulations were created or adapted to mandate how some industries, especially Financial Services, were to maintain records of their electronic communication. These regulations generated another spike in the adoption cycle of archiving, thus greatly accelerating the deployment of email archiving in regulated industries. A greater awareness of information governance and increased visibility into eDiscovery costs also contributed to the growth of email archiving.

More recently, in addition to email archives, companies are deploying collaboration products such as Microsoft SharePoint and social business platforms such as IBM Connections to communicate internally and externally with employees, partners, and clients. And although the business tools and technology change, the obligation to comply with regulations that govern communications does not.

Meeting compliance obligations with these new content generating systems requires new archiving capabilities. For some organizations, their existing email archive is a silo that cannot capture these new data sources. Instead, the archive has become a barrier to implementing an archiving framework that can grow to meet evolving needs. This results in using multiple tools for archiving multiple data sources, and often entails additional work for collecting and processing data from each system for eDiscovery purposes. This all adds complexity and drives up cost.

Most archiving solutions in market took, and continue to take, an “archive everything” approach. With no way to identify what is important and what is not, they have no way to understand the business value of the content being archived

and therefore the default becomes to keep everything. The net effect is that the growth is transferred out of the source system and into the archive. The archive now becomes a “digital landfill” full of content that is of dubious value to the business, and full of legal risk.

The IBM approach is different and involves a holistic approach to archiving as part of a broader information lifecycle governance program that determines the retention of information based on its value to the organization. This approach is designed to ensure that the debris is jettisoned, and that time and money is spent on managing only information with legal duty, obligation for compliance, or business value.

## 1.2 IBM solutions

This section introduces IBM Value-based archiving and Information Lifecycle Governance solutions.

### 1.2.1 IBM Information Lifecycle Governance

The IBM Content Collector family is part of the unique IBM Information Lifecycle Governance (ILG) portfolio. IBM ILG solutions are modular and are designed to help deal with the ever-growing volume of data within organizations for purposes of IT manageability, cost reduction, and managing legal risk and compliance. These solutions provide clients with instrumentation that helps them automate the long-term management of enterprise information based on its business value, comply more efficiently with litigation and regulatory duties, and dispose of information when it is no longer needed.

IBM ILG solutions are broadly categorized into three solution areas:

- ▶ **Value-based Archiving**  
This provides the ability to identify and collect valuable information from source systems for storage management or compliance.
- ▶ **General Counsel Solutions**  
These provide eDiscovery search, analytics, and legal hold management capabilities to help make responding to legal enquiries more efficient and cost effective.
- ▶ **Records and Retention Management**  
This provides centralized management of retention schedules and policies, and syndication to downstream systems.

These solutions are depicted in Figure 1-1.

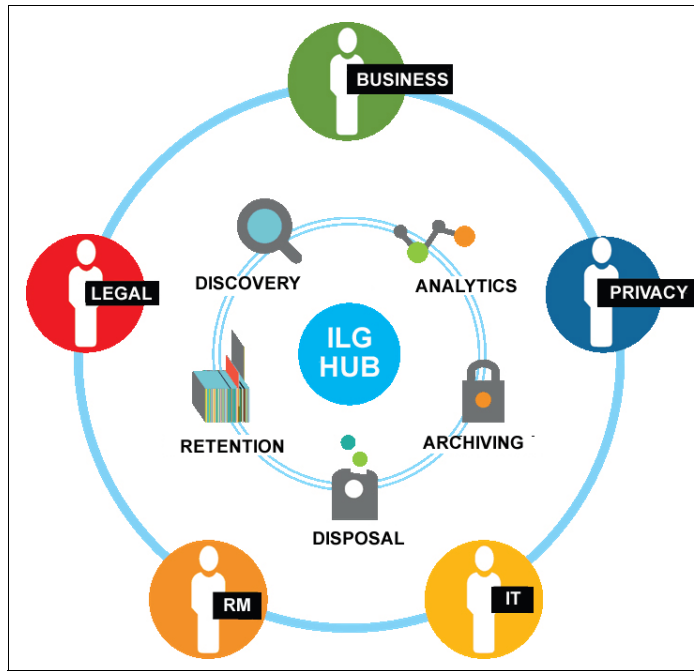


Figure 1-1 IBM ILG solutions

These IBM solutions deliver capabilities that support *defensible disposal*. This term refers to the idea that information that is not required for legal obligation, for compliance reasons, or with no defined business value, is unneeded and should be deleted. However, the information must be deleted in a way that is reasonable and consistent, and that will withstand scrutiny if challenged.

Typically, IT staffs do not have sufficient insight into the nature of the information under their control to determine what has real business value and what has none. Thus, the default in many organizations becomes keep everything, forever leading to runaway storage requirements, higher IT costs, and increased legal risk.

IBM ILG solutions help change that “keep everything” approach by delivering capabilities that the various information stakeholders, including legal departments, IT, records, privacy officers, and business users, can use to address information governance issues in their domain. These solutions help to generate a more complete picture of the information, and make possible information management decisions based on fact and certainty, the core of a defensible disposal program.

These IBM solutions help to drive down the costs and risks associated with over-retention with instrumentation that enables the following outcomes:

- ▶ IT staff are enabled to understand and manage, by system and employee, the content collection, archiving, and retention criteria and procedures established by the organization. Further, they can implement an archiving program that reduces cost by archiving what is important and required, and deleting what is not.
- ▶ Attorneys and paralegals are enabled to automate legal hold processes and coordinate evidence collections across the organization to respond to requests for information more quickly and cost effectively.
- ▶ Records managers are enabled to develop and manage global retention policies and to coordinate compliance and disposition across multiple systems and jurisdictions.
- ▶ Privacy officers are enabled to assess and communicate privacy duty by data subject and data location, including overlapping obligations.
- ▶ IT staff is enabled to determine which systems appear to have the highest cost and risk profiles, and to allow them to address management of systems by information value.

These capabilities generate a more complete picture of the information inside an organization and make possible information management decisions based on fact and certainty. With this confidence comes the ability to implement a defensible disposal program that can have a real impact on the amount of data under management and the associated cost and risk.

The IBM Value-based Archiving solution, and in particular the IBM Content Collector family of products, support defensible disposal efforts with a further set of capabilities that attack the data growth problem at the root (the source system), and thereby immediately reduce the amount of information debris under management inside an organization and the cost and risk associated with over-retention.

- ▶ **Defensible Disposal Library**

More information about the Defensible Disposal Library is available at the following site:

<http://www.ibm.com/software/ecm/disposal-governance/library.html>

## **1.2.2 Value-based archiving with IBM Content Collector**

IBM Content Collector is much more than straight-through archiving. It provides IT administrators with a high level of control over the archiving environment. From a common interface, organizations can implement policies that define what



gets archived from which source system, make decisions on how it gets archived based on the content or metadata of the information, and determine the retention and governance rules associated with that type of content. This enables IT to implement granular archiving policies that collect and archive specific pieces of information. This can help with the following tasks:

- ▶ Eliminate point solutions and lower cost with a unified collection, management, and governance approach that works effectively across a broad range of source systems and information types
- ▶ Appraise, better understand, cull, and properly select information to archive
- ▶ Retain, hold, and dispose of archived content efficiently and defensibly
- ▶ Eliminate the costs and risks inherent with over-retention

The IBM Content Collector v3.0 family is composed of the following products:

- ▶ IBM Content Collector for Email V3.0

This product collects content from IBM Lotus® Domino®, Microsoft Exchange, and SMTP messaging systems and helps ensure that critical email is properly retained and protected for compliance or other reasons.

- ▶ IBM Content Collector for Files V3.0

This product controls documents on network file shares. IBM Content Collector for Files can automatically capture documents as they are placed in a monitored location on the file share, or archive existing content based on age, file size, or other criteria.

- ▶ IBM Content Collector for IBM Connections V3.0

This product supports archiving from the IBM Connections social business platform for compliance or governance reasons. Content is indexed and available to search tools including IBM eDiscovery Manager, and can be used to meet compliance requirements and to respond to legal requests.

- ▶ IBM Content Collector for Microsoft SharePoint 3.0

This product captures content from Microsoft SharePoint for long-term archiving, compliance, or other reasons. All Microsoft SharePoint library and list types are supported.

- ▶ IBM Content Collector for SAP Applications V3.0

This product provides SAP data and document archiving, SAP content-enabling, and complementary process management for SAP environments. IBM Content Collector for SAP Applications is specially optimized for SAP environments and supports both SAP ArchiveLink and SAP NetWeaver Information Lifecycle Management (ILM) protocols.

A discussion of Content Collector for SAP Applications is beyond the scope of this book, but more information is available online at:

<http://www.ibm.com/software/data/content-management/content-collector-sap/>

## 1.3 IBM Content Collector overview

This section introduces IBM Content Collector and discusses its architecture, components, and major features.

### 1.3.1 Architectural overview

All of the IBM Content Collector products previously listed (except for IBM Content Collector for SAP Applications) share a common architecture and use common components. A typical IBM Content Collector infrastructure has three *required* components:

- ▶ The *source system* such as Microsoft Exchange or Lotus Domino.
- ▶ IBM Content Collector.
- ▶ The *target repository* such as FileNet Content Manager that is used to store the archived content. IBM Content Collector leverages the services provided by the repository such as search, content security, and indexing.

A complete and current list of supported repositories and other prerequisites is available at the following site:

<http://www.ibm.com/support/docview.wss?uid=swg27024229>

Other IBM products are *integrated* with IBM Content Collector and can be included in the solution to meet additional requirements:

- ▶ IBM Enterprise Records  
Use this product to manage content according to a corporate file plan.
- ▶ IBM eDiscovery Manager and IBM eDiscovery Analytics for eDiscovery  
Use these products for search, first-pass legal review, matter management, and analytics.
- ▶ IBM Content Classification  
Use this product to automatically classify documents and messages based on their content.

## 1.3.2 IBM Content Collector components

Although each Content Collector product is sold and licensed individually, those previously listed (except for IBM Content Collector for SAP Applications) install into a common application framework. Thus, from one interface, administrators can configure all installed connectors and manage the entire archiving environment. This interface is called the Configuration Manager. You will use the Configuration Manager to implement the archiving use cases presented later in this book.

Configuration Manager is used to define and manage the settings for your archiving environment. This includes:

- ▶ Defining and managing connections to the source systems
- ▶ Defining and managing connections to the back-end archive data store
- ▶ Defining and implementing archiving logic in task routes
- ▶ Defining general system settings

Within the Content Collector Configuration Manager the administrator will work with the following components:

- ▶ Source Connectors

These contain definitions of data sources and the options available for that source. Multiple connectors can be installed and configured. Each connector has a unique set of capabilities specific to the source. Administrators configure the connector to the collection behavior they require.

- ▶ Target Connectors

These contain definitions of the repository that will be used as the archive. The target connectors surface the unique capabilities of the repository that are available for configuration by the administrator.

- ▶ Tasks

These define the processing steps that can be applied while content passes through the archiving process, such as automatic classification, extracting metadata, or declaring as a record.

- ▶ Post-processing Tasks

These define the processing that occurs after content has been added to the repository.

- ▶ Task Routes

These are the principal configuration elements and can be thought of as a workflow. The administrator creates Task Routes that implement the required archiving logic by assembling Source Connectors, Tasks, Post-processing

Tasks and Target Connectors into an end-to end process that defines the archiving logic. Content Collector executes the logic (typically on a scheduled basis) to identify the data that meets the requirement, collect and process it according the defined tasks, and archive it into the defined repository.

The major architectural components of an IBM Content Collector environment and interactions between components are shown in Figure 1-2.

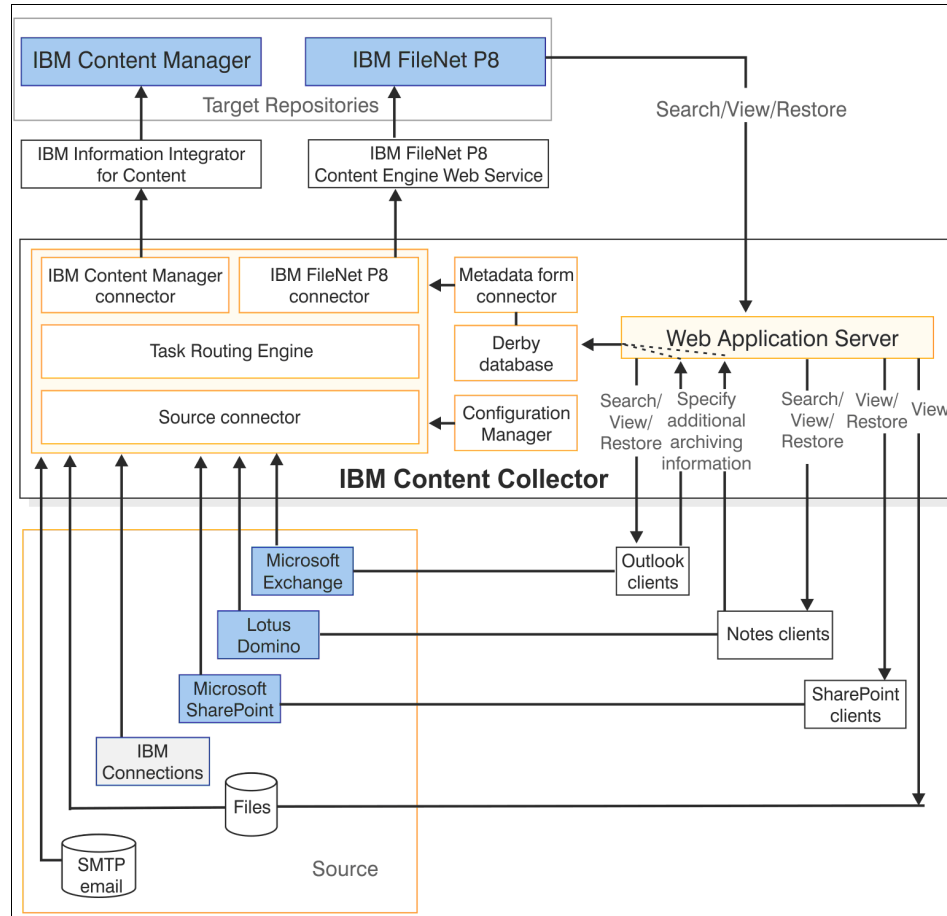


Figure 1-2 IBM Content Collector major components and interactions

### 1.3.3 IBM Content Collector features

Within the Content Collector environment there are many features and capabilities that administrators can draw upon to define a content collection and

archiving environment that can meet their requirements. Many of these are listed here and described in greater detail in other chapters of this book.

## **Lifecycle management**

IBM Content Collector enables you to implement a range of document retention strategies, from simply deleting without archiving for content with no value, to deleting originals after processing, to a gradual removal of content from the source system over time. Control over the lifecycle is quite granular, as illustrated in the following example.

### ***Storage management example***

For storage management, you can remove parts of archived email documents or Notes application documents step-by-step from the original document until finally the entire content is deleted. A typical lifecycle might include the following steps:

1. Stub any attachments to email messages during archive and replace the messages with a link to the archive, perhaps after 14 days.
2. Archive the body of the mail message, leaving just a “stub” in the inbox after 30 days.
3. Finally, remove the entire message from the inbox after 60 days.

This “stubbing” lifecycle is fully controllable, and allows administrators to build a schedule that matches the requirements of their user community.

Document lifecycles are covered in detail in Chapter 5, “Retention management” on page 119.

## **Conditional processing and Expression Editor**

IBM Content Collector makes use of Regular Expressions to provide a flexible method of defining and evaluating metadata attached to the content being archived. The IBM Content Collector Expression Editor can be used to build the rules that will be evaluated during conditional processing, and for mapping or replacing metadata. The Expression Editor is covered in detail in 4.5, “Using the Expression Editor” on page 109.

## **De-duplication of content**

*De-duplication* ensures that only one copy of a document or an embedded attachment is kept in the archive, no matter how many times the same document or attachment was archived by different users. De-duplication depends on the calculation of a unique de-duplication hash key. Each source connector type calculates its hash keys differently, and de-duplication for specific connectors are discussed at relevant points throughout this book.

## **Expiration Manager**

IBM Content Collector can automatically assign a retention date to each document as it is archived, based on instructions in the archiving logic. Expiration dates for each document are typically based on a time period. The dates can be calculated by user name or the LDAP/Active Directory group membership of the recipient, or on a date expression that you can derive from a metadata property, expression, or literal value. Expiration Manager is covered in detail in 5.3, “Expiration Manager” on page 128.

## **Automated declaration of records**

If there is a requirement to classify archived content as records according to a file plan that determines the retention periods of different types of content, IBM Content Collector is integrated with and can use the capabilities of the IBM Enterprise Records product. Archiving task routes can be configured to declare content as records during the ingestion processes. The integration of IBM Content Collector with IBM Enterprise Records is covered in detail in Chapter 7, “Records management integration” on page 221.

## **Content-based classification**

IBM Content Classification categorizes and organizes content by combining multiple methods of context-sensitive analysis. IBM Content Classification can be invoked from an Content Collector Task Routes, thereby enabling automatic classification of content during the archiving process. This is described in greater detail in Chapter 6, “Document classification” on page 155.

# **1.4 Typical archiving use cases and scenarios**

The IBM Content Collector components and features discussed previously can be applied to meet various use cases. In some cases, their functionality can be used “as is” with no modification required to the task routes and other configurable components. To meet other requirements, more configuration might be required. Following are typical use cases that can be addressed with IBM Content Collector.

## **1. eMail archiving for compliance**

In a compliance email archiving scenario, a client must collect every email sent to or by a particular set of defined users. It is important that the collection be handled so that no message can be deleted by the user before it is archived.

To meet this requirement both IBM Lotus Notes® and Microsoft Exchange support the use of journals to perform the capture of the messages before they

are delivered to the user. IBM Content Collector processes and archives the messages directly from the mail system journals.

Additionally, many SMTP/MIME mail systems can be configured to automatically forward a copy of each email to an alternate address for archiving. Content Collector includes an SMTP listener to capture and archive these redirected messages.

## **2. Multi-source compliance and eDiscovery platform**

Clients who are deploying social business and collaboration tools as their “next generation” communication platforms have an archiving requirement that spans existing email systems, plus the new social platforms. Content Collector with IBM eDiscovery Manager can meet the requirement to provide a single archive and eDiscovery platform that spans email, IBM Connections, and Microsoft SharePoint source systems. This provides legal users with the ability to search across data collected and archived from all three source systems with a single search and with no further collection required. Integration with IBM Enterprise Records further enhances the compliance platform by providing DOD 5015 certified records management capabilities.

## **3. User-driven or “interactive” email management**

Content Collector for Email integrates directly into the Microsoft Outlook and Lotus Notes email clients. If installed, these components create an “interactive” mode, enabling users to interact directly with the archive from their email client so they can decide what to archive and when. Users can also search the archive for their archived content, and retrieve it back into the email application. The client components are delivered as an Outlook Extension for Microsoft Exchange and as wizard-driven template modifications for Lotus Notes.

## **4. Automatic archiving based on profiles**

For storage savings or other reasons, there might be a requirement to automate the collection of content that meets certain parameters such as size, age, or file type. All Content Collector products can be configured to collect and archive content automatically based on profile settings appropriate to the source connectors being defined. This allows administrators to define what is archived and how it is archived based on profiles and metadata.

The following list includes a few examples of the many possibilities:

- ▶ Content Collector for Files Systems supports creating a profile that processes content based on file properties such as file type, file size, created date, or author. Files meeting the profile definition are moved to the archive. The file is typically replaced with a URL link that leads to the content in the archive.

- ▶ Content Collector for Email can be configured to use a document lifecycle that removes email attachments or even full messages and replaces them with links.
- ▶ Content Collector for Microsoft SharePoint can be configured to archive content from Microsoft SharePoint sites based on metadata values such as document status or age.

## **5. Archiving application output with associated metadata**

IBM Content Collector is often used as an information management tool for moving important content from a generating system into a long-term archive or Business Process Management tool.

Imagine a scenario where a web application collects information from a client, perhaps as part of an online application, and the requirement is to maintain a copy of this application as a record for long-term retention or to launch a workflow around this piece of content.

A solution to this requirement is to configure the source system to export the application form in a common format, such as pdf, and associate various metadata with that file either as properties or in separate index definition file. Content Collector for Files Systems can be configured to watch a specific location for these files, and process them per the appropriate task route. This task includes reading the associated metadata and storing to the archive or launching the workflow.

## **6. File Server management**

For server consolidation or storage management requirements, a client has a requirement to capture the content of network file shares. As part of the archiving process, the client also wants to capture the associated file system properties to use as metadata, and ensure that user access to the archived files is maintained.

Content Collector for Files Systems can be configured to capture file properties and file systems properties and pass that information to the archive to replicate settings in the file system, including access control lists (ACLs) and file “ownership” information.

## **7. Archiving Microsoft SharePoint content**

For Microsoft SharePoint environments, there is often a requirement to move content to an archive repository for long-term retention or performance management reasons. IBM Content Collector for Microsoft SharePoint supports all Microsoft SharePoint library and list types. It also provides rich post-processing options to define what happens to the content in Microsoft SharePoint after archiving. Options include delete the original from Microsoft



SharePoint; leave in Microsoft SharePoint but prohibit editing; or leave stubs in place.

Content Collector for Microsoft SharePoint also has the ability to intelligently archive updates to previously collected content, creating new versions on target repository documents.

## **8. SAP data archiving and content enabling**

Customers with SAP often have the requirement for an SAP-certified data archiving tool that can control the growth of the SAP environment. IBM Content Collector for SAP Applications can meet this requirement with an SAP-certified connector that supports the SAP ArchiveLink and NetWeaver ILM protocols.

Content Collector for SAP Applications also support “content-enabling” SAP applications. This improves the efficiency of SAP business processes and users by linking relevant content, such as incoming scanned invoices, to SAP workflow.

A discussion of Content Collector for SAP Applications is beyond the scope of this book, but more information is available online at:

<http://www.ibm.com/software/data/content-management/content-collector-sap/>

## **1.5 Conclusion**

The remainder of this book examines implementing several of these use cases in greater detail, and describes the configuration options available to meet the requirements of the individual use case.





## Example use cases

In this chapter we provide specific use cases that we utilize throughout this book to explain the usage of various Content Collector features and function, the usage of IBM Content Classification within the solution, and the usage of Enterprise Records in the solution.

We implemented four specific use cases to help illustrate various aspects of using Content Collector for value-based archiving. These use cases build on each other, showing that a value-based archiving solution can evolve over time as the needs and maturity of an organization grow.

In this chapter we discuss the following topics:

- ▶ Use case 1: Email archiving for compliance and storage management
- ▶ Use case 2: Email archiving with content classification
- ▶ Use case 3: Email archiving with records declaration
- ▶ Use case 4: File system archiving with records declaration

## 2.1 Use case 1: Email archiving for compliance and storage management

In this use case, an organization is initially motivated by an overburdened email server and has an immediate need to offload email messages to P8 to relieve that burden. In addition, the organization is looking to start implementing compliance archiving for their emails and will be looking to integrate with eDiscovery capabilities supported by a P8 ECM system.

To achieve these goals, we implement three task routes for this use case:

### 1A - Compliance archiving of email from the Journal

The first task route will collect and delete all email from the Journal mailbox on the email server to ensure that all emails are archived to P8 for compliance and eDiscovery purposes.

### 1B - Archiving email from mailboxes for storage management

All emails will be collected and archived from the server mailboxes (exclusive of the journal) after one month to relieve the storage burden on the email server. Having a delay before archiving from user mailboxes gives users a chance to delete unwanted, junk, and transient email messages that might uselessly fill up the archive.

Because we are also collecting all journal email, there will already be a preserved copy of all messages for compliance purposes, so there is no risk of losing important emails. The email is not stubbed at that point in time, so that the email clients can create an offline copy to be used when a user is working disconnected.

### 1C - Lifecycle stubbing and retention management

A separate task route is used to perform lifecycle stubbing on the emails archived from server mailboxes. Attachments will be stubbed after two months or when a offline copy has been created. The remainder of the email message itself will be stubbed after one year. This lifecycle stubbing will reduce the storage requirements on the email server.

During this initial implementation, the organization decides to set a blanket default expiration date on all email collected to three years from the Date Received. Content Collector Expiration Manager will be used to delete all email from the archive after three years unless the email has been placed on hold or otherwise locked down by subsequent record declaration.

See 3.5, “Use case 1A and 1B: Email archiving for compliance and storage” on page 75 for a detailed explanation about how to configure the task routes for this use case.

See 5.5, “Use case 1C: Lifecycle stubbing and retention management” on page 145 for a detailed explanation about how to configure and use Content Collector Expiration Manager for this use case.

## 2.2 Use case 2: Email archiving with content classification

After using simple email archiving for some time, our example organization decides to become more effective in their email retention by keeping email for varying amounts of time based on the nature of the email. Based on the content of an email, they can identify the email to be one of four Mail Types: Personal, Business, Sensitive, or Critical. Instead of retaining all archived email for three years, the organization wants follow the retention policy for email shown in Table 2-1 based the type of email.

Table 2-1 Retention policy for email based on type of email

Mail type	Retention period
Personal	2 months
Business	2 years (24 months)
Sensitive	5 years (60 months)
Critical	10 years (120 months)

Instead of having users identify which mail type applies to each email, the organization decides to use IBM Content Classification to classify all email automatically. Content Classification analyzes the email body and attachments, and matches them to a knowledge base that contains statistical profiles for each mail type. It also executes rules that scan the email fields for specific keywords and phrases. When properly configured and trained, Content Classification can identify classification categories for content that can be used to determine retention or make other decisions about what to do with a document.

This use case will extend the previous use case by adding to the task route for Journal archiving. Instead of having the task route set the expiration date on all emails to three years from the Sent Date, we include the IBM Content Classification task to process each email and determine which of the four mail types apply. We will then add a task in the task route to calculate the expiration date for each archived email according to the mail type determined by Content Classification.

See 6.8, “Use case 2: Email archiving with content classification” on page 207 for a detailed explanation about how to configure the task route for this use case.

## 2.3 Use case 3: Email archiving with records declaration

After continuing with email archiving where all Journal email is sorted into the four categories for various retention periods, our example organization decides to enhance their solution by declaring only some of these emails as records into Enterprise Records, to have more comprehensive records management capabilities for the more important emails. This use case illustrates how to use a task route to decide which emails to declare and how to determine the record classification based on the results from Content Classification that were used previously to set the retention.

For this use case, we continue to archive Personal and Business email as before, by setting the retention in Content Collector and archiving without record declaration. The rationale for this might be that personal and routine business email can continue to be managed by simple retention and avoid the overhead of record declaration, letting Content Collector continue to dispose of these email types based on the simple expiration date. However, Sensitive or Critical email will now be declared as records and the disposition will be managed by Enterprise Records.

Table 2-2 shows the logic for deciding which email to declare and how to set the expiration date for email that is not declared.

*Table 2-2 Logic for declaration and retention based on type of email*

Mail type	Declare as Record	Content Collector expiration date
Personal	No	Two months from Date Sent
Business	No	Two years from Date Sent
Sensitive	Yes	Do not set - retention managed by Enterprise Records file plan
Critical	Yes	Do not set - retention managed by Enterprise Records file plan

See 7.4, “Use case 3: Email archiving with records declaration” on page 243 for a detailed explanation about how to configure the task route for this use case.

## 2.4 Use case 4: File system archiving with records declaration

This use case illustrates a simple automated collection process where documents are placed on a file share by an automated business process in an organized manner, and can be ingested and declared automatically based on the file path and other attributes found on the ingested documents. As long as there is enough information from the incoming metadata to determine all the required information for declaring records with the P8 Declare Record task, there is no need for an external classification mechanism like Content Classification.

For this scenario, we collect two different types of records from a file system where the type of record is identified by the name of the parent folder where the file is located: Invoices and Contracts.

Table 2-3 shows the configuration to be used for record declaration.

*Table 2-3 Configuration requirements for declaring records from the file system*

File System folder	Record category	Record class	Property mapping
Invoices	Invoices	Invoice Record	Set Invoice Date to the File Created Date
Contracts	Contracts	Contract Record	Set the Contract Number to the 8 digits in the file name that identify the contract

See 7.5, “Use case 4: File system archiving with records declaration” on page 258 for a detailed explanation about how to configure the task route for this use case.

## 2.5 Conclusion

In this chapter we provided specific use cases that are used throughout the book to explain the use of various IBM Content Collector features and function, the use of IBM Content Classification within the solution, and the use of IBM Enterprise Records in the solution. In the remaining chapters of the book we illustrate, with these use cases, content archiving and retention management and integration with IBM Content Classification and IBM Enterprise Records.







## Dimensions of content archiving themes

In this chapter we discuss content archiving with IBM Content Collector for Email, Microsoft SharePoint, Files, and IBM Connections solutions. It covers the three main themes behind archiving and provide requirement, approach, and benefits for each theme.

**Configuration assumption:** Throughout this chapter and subsequent chapters, we assume the installation and configuration of IBM Content Collector using Initial Configuration has already been done. For more information about these steps, refer to the product information center.

In this chapter we discuss the following topics:

- ▶ Dimensions of content archiving with IBM Content Collector
- ▶ Storage management
- ▶ Compliance archiving
- ▶ Business process management
- ▶ Use case 1A and 1B: Email archiving for compliance and storage

## 3.1 Dimensions of content archiving with IBM Content Collector

There are three main themes for archiving content into a Enterprise Content Management (ECM) system. Each use case that is to be implemented within a archival solution is driven by one or more of these themes. IBM Content Collector provides preconfigured task routes in the form of templates to address these use cases.

- ▶ **Storage management**

Cost savings is a major driver for storage management, along with system performance and manageability. There are two aspects of cost savings when using IBM Content Collector to archive content that entails migrating content from a higher storage tier to a lower storage tier:

- The first aspect is lower costs from migrating content from a high storage tier (Mail/File/Microsoft SharePoint) to a lower storage tier attached to the ECM system.
- The second aspect is eliminating redundant information when possible, so that only one instance of a document is stored.

- ▶ **Compliance**

Minimizing the risk of legal exposure is the main driver for the compliance theme. Ensuring compliance can be complex because laws might vary by country and thus the requirements for storing electronic communications might also vary. Moreover, requirements imposed by different laws might be conflicting. Thus special care must be given to ensure legal compliance when designing archival solutions.

- ▶ **Business process management**

Time and cost savings motivate companies to automate their business processes. And although the execution of a business process might be managed by a business process management system such as IBM Business Process Manager, the workflow might actually be started manually. For example, a user receives an email containing an invoice. After detaching the attachment containing the invoice, the user adds the invoice and additional metadata, such as a customer number, to the ECM system through a repository client.

In the following sections, we expand upon these main themes for archiving content into an ECM system.

## 3.2 Storage management

In addition to cost savings realized through migration to lower cost archive storage, deduplication provides another huge opportunity for cost savings. Multiple copies of each document can exist in different sources. A document that has been placed on a file share might also have been placed into Microsoft SharePoint or attached to an email.

Because of this, IBM Content Collector allows for cross-source deduplication, ensuring that only one instance of each unique document is kept. This is done by creating a unique hash of the document that allows for identifying potential duplicates. If using IBM FileNet P8, Content Collector can also make use of the Suppress Duplicate Content Elements feature to have the ECM system deduplicate binary identical objects.

Further storage reduction can be achieved by using solutions for storage level deduplication on the ECM system, such as IBM System Storage® N series.

A company archiving policy must be established and be enforced by IBM Content Collector. The policy should provide answers to several questions pertaining to storage management and its implications on users, as listed here:

- When should the content be archived?

Archiving content too early negatively impacts the user experience when working with the content. Even a slight delay when accessing content might lead to less user acceptance and loss of user productivity. Conversely, archiving content too late does not maximize storage savings. Thus, an informed decision is needed that brings both requirements into balance, based on the nature of content.

Additional aspects might also arise, depending on the nature of the source. For example, Blackberry devices used to access the content of a mailbox are not able to restore an email back to the mailbox or search the archive. Thus, content should not be removed from email as long as it is used actively by Blackberry devices.

- How long should the content be kept?

Retention time should be assigned depending on the type of content, so that the content is guaranteed to be disposed after a certain time when it is no longer of value to the business. An additional consideration is any legal regulation that requires content to be kept for a certain amount of time, depending on the type of content. Contract-relevant data requires a different retention period than personal communication that is not directly related to business.

- How should users search and access archived content?

Because storage management is about removing content from the source system, the question arises how users can access, retrieve, and find the archived content. Ideally, users do not realize that content has been archived and continue to work as before. IBM Content Collector also provides email integration for searching the archive for email owned by the user of a mailbox.

- How should valuable content be identified and protected?

There are different approaches to identifying valuable content and assigning appropriate retention periods. Users can manually declare content as valuable by dragging the content to a folder or flagging it by other means. Because this approach requires user interaction, it is limited and sometimes not desirable or sufficient. If there are known structures or patterns, you can implement content categorization using rules in a task route or IBM Content Classification, depending on the complexity of the categorization.

### **3.2.1 Staging rollout for maximizing storage savings**

To effectively roll out storage management task routes for a source system, a staged approach should be taken depending on the volume of the source data. Initially there is usually a huge backlog to be processed when the task route is deployed, so the initial ingestion will run for a longer period to process the backlog.

If you gather knowledge about the source data size distribution first, you will be able to segment the backlog processing by source document size. This allows you to process the largest documents first and maximize the immediate storage

savings, thereby accelerating the return on investment (ROI) compared to the non-staged approach (Figure 3-1).

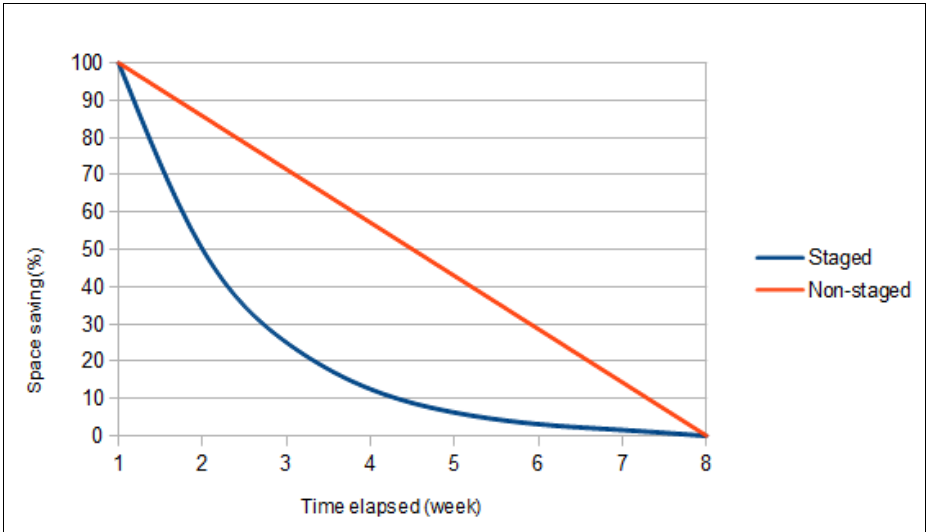


Figure 3-1 Storage saving comparison between staged versus non-staged rollout

An example of rollout staging is given in Table 3-1. The target criteria is “older than 30 days” with no size restriction. Based on the content size distribution generated using statistics task routes, the staging uses 50 MB, 10 MB, and 1 MB as intermediate size criteria to ensure that the most storage is saved in the first stage of the rollout.

As shown, although the percentage of storage saving decreases, the number of documents processed per stage increases. Thus, the date criteria is increased to 90 days after the size criteria is removed to ensure that processing is spread evenly on all sources (mailboxes, file shares or Microsoft SharePoint sites), due to the number of small documents that are to be processed in the backlog. Starting with the fifth week, the target criteria is in place and steady state has been reached.

Table 3-1 Example rollout stages for accelerating immediate storage savings

Step	Date criteria	Size criteria	% of expected storage saving
First week	Older than 30 days	Bigger than 50 MB	40%
Second week	Older than 30 days	Bigger than 10 MB	25%
Third week	Older than 30 days	Bigger than 1 MB	20%

Step	Date criteria	Size criteria	% of expected storage saving
Fourth week	Older than 90 days	-	5%
Starting fifth week	Older than 30 days	-	10%

### 3.2.2 Document stubbing

The majority of storage saving scenarios require users to be able to access documents archived by IBM Content Collector in a convenient way. So ideally the document location should not change. To achieve this goal, IBM Content Collector supports stubbing. What kind of stubbing options are available depends on the source system. However, the general concept is always to either replace the original content with a shortcut that will retrieve the original document, or to partially remove content from the original document. For email this typically means removing attachments after archiving, because attachments account for the majority of storage occupied by mailboxes.

#### File system stubbing

The most commonly used post-processing option for file system is to delete the file and replace it with a shortcut. Depending on the configuration of the repository task, the shortcut uniform resource locator (URL) can be set to two different kinds of targets:

- ▶ The IBM Content Collector web application can provide transparent access to a document, so a user will launch the associated program when clicking a shortcut.
- ▶ A custom shortcut URL will launch a user-defined web application, like Workplace XT, for viewing the document in the repository.

All other post-processing options do not delete the file from the source system and are used for non-storage saving scenarios.

#### Email stubbing

The email connector of IBM Content Collector provides for fine-grained stubbing options in the EC Create Email Stub task. Here we focus on the stubbing options most commonly used for storage saving scenarios. For a detailed description, refer to the information center:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/task\\_reference/r\\_afu\\_ec\\_create\\_email\\_stub.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/task_reference/r_afu_ec_create_email_stub.htm)

The most commonly used type of initial stubbing for email is attachment stubbing. Because attachments account for the majority of the size of a mailbox, removing them from the email system is desirable. IBM Content Collector can remove attachments from email and replace them with links in the email body. Thus, the attachments will be accessible with one click after their removal.

From the user perspective, the difference is that a link needs to be clicked instead of an attachment icon. If this is not desired, transparent retrieval can be used to retrieve the attachment on demand, leading to no visual difference between stubbed and non-archived email in the client. However, be aware that transparent retrieval might add to the load on the archive system because users are not aware that when they open a document, the attachment is retrieved from the archive.

### In-place attachment link support for Lotus Domino

For Lotus domino email and application documents, IBM Content Collector will create in-place links for removed attachments in addition to a list of links to the attachments that are attached to the end of the message body (Figure 3-2).

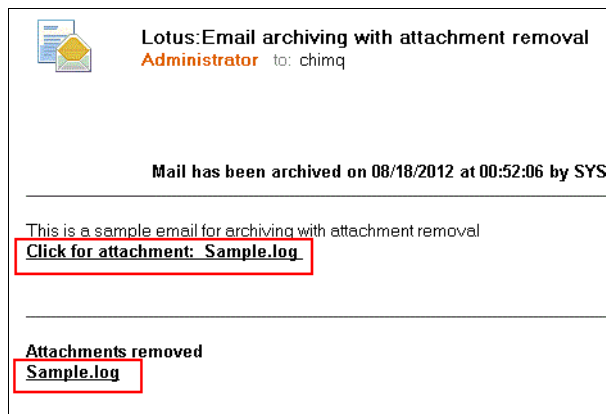


Figure 3-2 Lotus email with in-place links

In-place links are helpful if the order of attachments is important for reading an email. Especially when multiple attachments with similar names are attached, it can be difficult to determine the correct attachment to open from the list. It is advisable to keep the default and have in-place links created.

## Microsoft SharePoint stubbing

IBM Content Collector provides three different post-processing options for Microsoft SharePoint content. The options provide different degrees of storage saving:

- ▶ Leave items in place with the option to change access control list (ACL) information to make the item read only.
- ▶ Replace items with links.
- ▶ Delete the item.

Additionally, you can select whether the post processing applies to all versions of an item or simply to the recent version. For information about how to select the appropriate post-processing option for a scenario, see 3.2.5, “Storage management for Microsoft SharePoint” on page 43.

For more information about the post-processing options for Microsoft SharePoint content, see the following site:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/task\\_reference/r\\_afu\\_sp\\_post-processing.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/task_reference/r_afu_sp_post-processing.htm)

## 3.2.3 Storage management for file shares

In our example scenario, a company wants to perform storage management on its file servers and archive all files that have not been accessed within a certain amount of time. During archiving, files are replaced with shortcuts that preserve the original icon so that users can still access the content of the file. The goal is to reduce the size, and therefore the associated direct and indirect cost, of file servers.

Before deploying the task route, the company first needs to decide at what time inactive files are to be archived. Files should not be archived too early because this causes slight delays when users open a file that is retrieved from the archive. Conversely, the company wants to minimize the amount of storage needed, so files should be archived as soon as they become inactive and are no longer frequently used.

To help make an informed decision, a statistics task route is used to build a histogram showing the last accessed time distribution of a representative file share.



## Using histograms to decide the last accessed date threshold

IBM Content Collector ships different statistics task route templates, which you can use to assess the nature of the content to be archived in the source systems. To collect information regarding last access times, complete these steps:

1. Deploy the “FS Statistics Collection” task route template using the New Task Route option.
2. Select the **Schedule** tab.  
Set the schedule to run once, starting in the past. This way, the collector will only run once each time the task route engine is started.
3. Select the **Collection Sources** tab.  
Add the file share for which you want to gather statistics.
4. Select the **TitleAuditLog** audit task.  
Check the Access property option under the File metadata to be collected.
5. Start the task route engine to collect the information.

**Performance consideration:** Scanning a complete file share might impact performance for those who are using the file share. Consider running this task route during an off-hour period.

6. Launch the Windows Event viewer.

7. Monitor the Event Log for this task route for the event indicating collection has completed (Event ID 129, as shown in Figure 3-3).

CTMS - FS Statistics Collection 2,334 Events				
Level	Date and Time	Source	Event ID	Task Category
Information	8/16/2012 1:22:14 AM	CTMS - FS Statistics Collection	129	Collector
Information	8/16/2012 1:22:14 AM	CTMS - FS Statistics Collection	145	Collector
Information	8/16/2012 1:22:14 AM	CTMS - FS Statistics Collection	160	Task Routing
Information	8/16/2012 1:22:14 AM	CTMS - FS Statistics Collection	160	Task Routing
Information	8/16/2012 1:22:14 AM	CTMS - FS Statistics Collection	160	Task Routing
Information	8/16/2012 1:22:14 AM	CTMS - FS Statistics Collection	160	Task Routing
Information	8/16/2012 1:22:14 AM	CTMS - FS Statistics Collection	160	Task Routing
Information	8/16/2012 1:22:14 AM	CTMS - FS Statistics Collection	160	Task Routing
Information	8/16/2012 1:22:14 AM	CTMS - FS Statistics Collection	160	Task Routing
Information	8/16/2012 1:22:14 AM	CTMS - FS Statistics Collection	160	Task Routing
Information	8/16/2012 1:22:14 AM	CTMS - FS Statistics Collection	160	Task Routing
Information	8/16/2012 1:22:14 AM	CTMS - FS Statistics Collection	160	Task Routing

Event 129, CTMS - FS Statistics Collection	
General	Details
Collector "FSC Collector" finished search started on +2012-08-15T23:09:25:759+00:00-UTC.	
Log Name:	CTMS - FS Statistics Collection
Source:	CTMS - FS Statistics Collection
Event ID:	129
Level:	Information
User:	SYSTEM
OpCode:	
More Information:	<a href="#">Event Log Online Help</a>
Logged:	8/16/2012 1:22:14 AM
Task Category:	Collector
Keywords:	Classic
Computer:	martin.iccex2010.bb

Figure 3-3 Example event log for the FS Statistics task route

8. Stop the Task Route engine.
9. Deactivate the FS Statistics Collection task route to avoid running it accidentally.
10. Load the audit log files generated by the run into a spreadsheet software of your choice. These files can be found in the configured folder for audit log files and will be prefixed with fs-statistics, as configured in the audit task.

11. Build a last accessed date histogram (see Figure 3-4) to assess the appropriate time for archiving, shortcutting, and removing files.

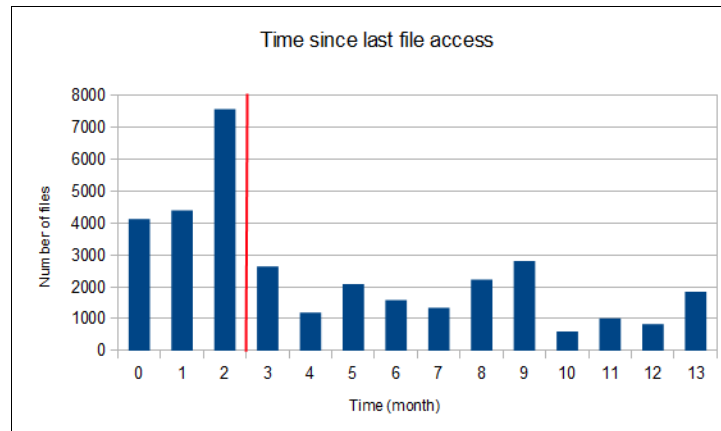


Figure 3-4 File share histogram with marker for optimal time for archival

Based on the information presented in the file access histogram, the company decides to archive all files that have not been accessed within the last 90 days. This threshold provides a useful balance between storage savings and not impacting users wanting to access files they work with on a regular basis. In addition, the statistics gathered are valuable input for the sizing of both the IBM Content Collector system and the ECM repository to calculate expected storage requirements, savings, and throughput.

### Deploying file system task routes for storage management

Archiving small files is not likely to provide storage savings, unless we assume these files are likely to be duplicates. This is because the ECM system holds metadata in addition to the file. The size of the metadata has to be weighed against the size of the file itself. So the decision is made that only files larger than 1 KB are archived. Based on the scenario, the following parameters are configured at the task route:

- ▶ Archive all files that are bigger than 1 KB and have not been accessed within the last 90 days.
- ▶ Set an expiration date of seven years for these files.

The “FS to P8 Archiving (Shortcut)” task route template best fits the storage management scenario. Deploy this template using the New Task Route option, as shown in Figure 3-5.

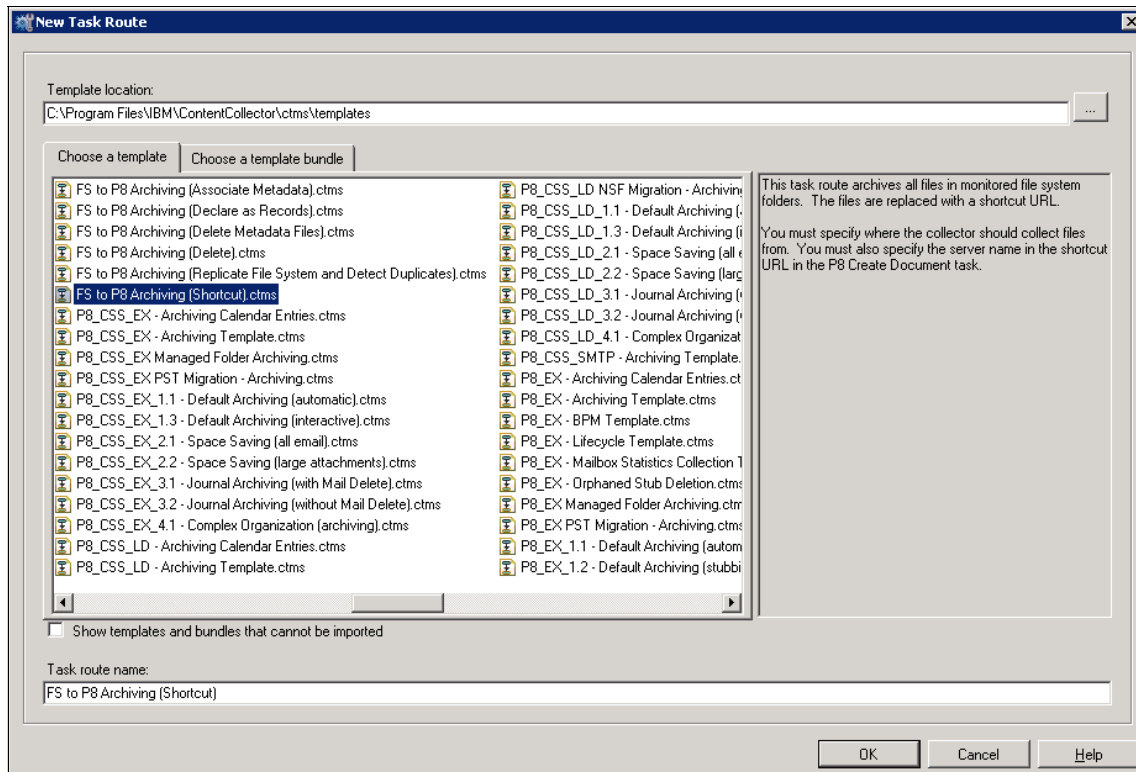


Figure 3-5 New Task Route window

After the import completes, follow these steps to customize the task route for the scenario:

1. Select the collector; the General tab will be selected.

Choose the option to collect content type information from the source system or the local system. This ensures that file icons will be retained on shortcut files if the icon information can be found on either the IBM Content Collector server or the server that hosts the file share.

2. Select the **Schedule** tab.

Set the schedule to run weekly on Saturday, starting at 12:00 a.m. and stopping at the latest at 11:45 p.m. Based on the archiving criteria that will be used (file not accessed within the last 90 days) running the collection once a week will be sufficient.

3. Select the **Collection Sources** tab.
  - a. Add the path of the file shares to be archived in Uniform Naming Convention (UNC). Ensure that the service logon user of the IBM Content Collector File System Source Connector has sufficient access rights to perform the required operations on the file share.
  - b. Check the option to monitor subfolders and set the maximum folder depth to 1024<sup>1</sup>.
  - c. Use the New Technology File System (NTFS) post-processing option to allow IBM Content Collector to use alternate data streams (ADS) for marking files as processed.

**NTFS consideration:** Alternate data streams are a unique feature of the NTFS, which is part of Microsoft operating systems. When using Samba (<http://samba.org>) as a file server running on the UNIX platform, make sure that support for alternate data streams is enabled in the `smb.conf` file.

Alternatively, use control folder post processing as described in “Storage management for Novell NetWare file systems” on page 37.

4. Select the **Filter** tab (see Figure 3-6 on page 36).
  - a. Set the last access date filter to a relative age of 90 days.
  - b. Check the **Minimum file size in bytes** option and set the value to 1024 bytes.

---

<sup>1</sup> It is not possible to configure an unlimited folder depth, so choose a sufficiently high number for the file shares to be archived.

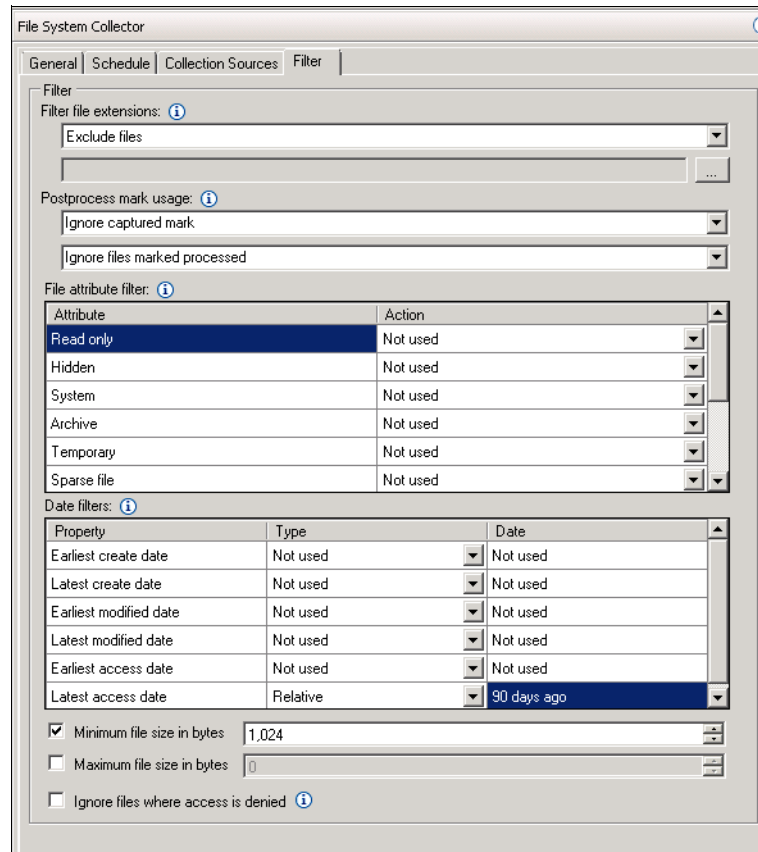


Figure 3-6 File System Collector Filter tab

5. Select the **P8 Create Document Task**.
  - a. Change host and port in the Shortcut link field to match your system.
  - b. Edit the ICCExpirationDate property from the property mappings section.
  - c. Use the expression editor to set the expiration date to the last access date plus 7 years (Figure 3-7 on page 37).

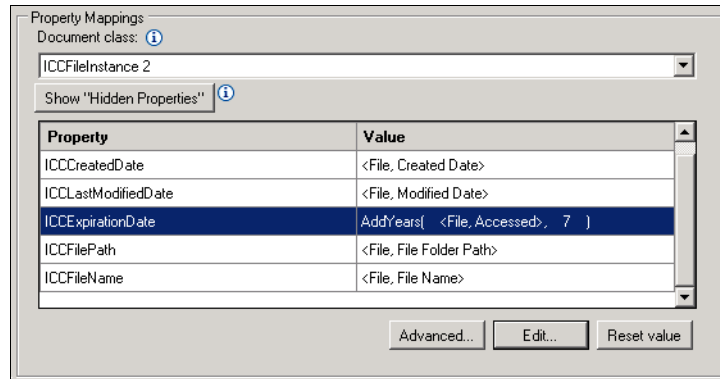


Figure 3-7 ICCExpirationDate expression for P8 Create Document task

The task route is now set up correctly and can be promoted to production after verification.

## Storage management for Novell NetWare file systems

After successfully rolling out storage management for Common Internet File System (CIFS)-based file servers, the company in this scenario wants to extend storage management to the Novell NetWare file servers of a recently acquired competitor. The “FS to P8 Archiving (Shortcut)” task route instantiated in “Deploying file system task routes for storage management” on page 33 can be extended to Novell NetWare file shares by following these steps:

1. Select the collector; the General tab will be selected.
2. Select the **Collection Sources** tab.
  - a. Add the path of the file shares to be archived in Uniform Naming Convention (UNC). Ensure that the service logon user of the IBM Content Collector File System Source Connector has sufficient access rights to perform the required operations on the file share.
  - b. Check the option to Monitor subfolders and set the maximum folder depth to 1024<sup>2</sup>.
  - c. Use the control folder post-processing option to have IBM Content Collector write a status file in a folder ICC\_PostProcessing.afu. for each file that is processed. This is necessary because alternate data streams are not available on Novell NetWare file systems.
3. Save the task route.

<sup>2</sup> It is not possible to configure a unlimited folder depth, so choose a sufficiently high number for the file shares to be archived.

4. Set the environment variable IBM\_CTMS\_NETWARE\_FILESYSTEM\_NAMES as described in the information center<sup>3</sup>.

The task route is now set up correctly and can be promoted to production after verification.

### 3.2.4 Storage management for email

Within any company, email server storage requirements continually increase. This is because users tend to retain a significant number of emails and treat them as personal assets, when in reality these emails are corporate assets. In fact, emails might contain regulated content or corporate records. However, increasing email server storage indefinitely will cause various problems:

- ▶ Degraded email server performance

The more documents accumulate in a mailbox, the slower response time will be. The limit and impact will vary, based on the characteristics of your email system. However, there are several preferred practices available to guide email retention, as listed here:

<http://blogs.technet.com/b/exchange/archive/2005/03/14/395229.aspx>

<http://www.ibm.com/developerworks/lotus/library/notes-mail-files/>

<http://msdn.microsoft.com/en-us/library/ms954401.aspx>

The conclusion of these practices is that both the number of documents and the size of documents in a mailbox affect performance of the email server and thus the mailbox size should be kept under control.

- ▶ Elongated backup and restore windows

Backups of mission-critical systems, such as email servers, are created on a regular basis. Because backup time is proportional to the amount of data in the mailboxes, disaster recovery time, which includes the restore of a backup, will increase because old, unneeded documents are included in the backup.

- ▶ Creation of local archives by users

Almost all companies impose a quota on the size of a mailbox. This leads to users performing frequent maintenance on their mailbox when they reach the quota. Frequently, however, users do not delete content, but move documents to local archives.

For legal discovery, these local archives must be located, indexed, and searched, which can be a challenge. Furthermore, there is no storage saved,

---

<sup>3</sup> [http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/environment\\_and\\_tools/r\\_afu\\_environment\\_variables.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/environment_and_tools/r_afu_environment_variables.htm)



from a high-level perspective, because documents are still included in the backups of the local user workstations.

IBM Content Collector can address all of these problems by imposing a lifecycle on all documents in an email system. This results in improved better email server performance, shortened backup times, and avoidance of local archives and user action to remove outdated documents from their mailbox.

For information about the stubbing lifecycle and expiration-related aspects, see Chapter 5, “Retention management” on page 119.

## Choosing effective filter criteria

Storage management for email systems is based on the received date of a message. Similar to what was done for file systems in “Using histograms to decide the last accessed date threshold” on page 31, we can perform an analysis on the email system. However, several differences arise:

- ▶ Users are deleting email that they do not need anymore.

Users can and often do delete email, such as spam, that is of no value to the company. Because we do not want to archive email that the user deletes afterwards, we want to provide a certain time span where users can delete email before it is archived. Based on this, established practice is to archive email that has been received 14 days ago. Further attachments should be removed during the archival phase, unless you are planning to use the offline repository features of the IBM Content Collector. When using offline repository functions, the document lifecycle task route should be in charge of removing attachments.

- ▶ Users are using other third-party systems (BlackBerry) to access their mailboxes.

Experience shows that the majority of email accessed with BlackBerry devices was received within the last 30 days. So to avoid impacting the BlackBerry user experience, documents may not be stubbed before they are older than 30 days.

- ▶ Users have a mailbox quota in place.

The choice of the criteria that is used for archiving and stubbing will influence the mailbox quota needed by a user. The typical amount of email received within the configured age limit should fit into the quota, so users will not need to perform manual delete operations.

Given these differences, a histogram build from email system data is still valuable for IBM Content Collector performance and storage sizing, but might not help you decide on the actual date criteria for archiving.

In this scenario, the company currently allows for a mailbox quota of 100 MB, because this matches the average email load of 20 days archiving and stubbing will occur after 14 days.

A smaller group of users use BlackBerry devices. Because they want to access their email from BlackBerry devices for 30 days, a separate policy is deployed for those users and the quota for those users is increased to 150 MB to avoid manual cleanup.

## Deploying email task routes for storage management

Based on the filter criteria the company created two groups and assigned all users to one of them:

- ▶ MailArchive\_Default
- ▶ MailArchive\_BlackBerry

Furthermore, an object store has been created and set up for indexing with IBM Content Search Services. IBM Content Collector Content Search Services Support has been installed and configured and the appropriate document classes have been deployed using Initial Configuration. Each night there is a backup window for both the email and the ECM system. Between 7:00 PM and 9:00 PM, no archiving can occur.

The “P8\_CSS\_EX - Archiving” task route template best fits the storage management scenario. Deploy this template using the New Taskroute option. To fulfill both scenarios, multiple collectors will be attached to the task route.

After the import completed, perform the following steps to customize the task route for the scenario:

1. Select the collector; the General tab will be selected.
  - a. Change the collector name to EC Collect All Email - Default.
  - b. Mark the collector as active.
2. Select the **Schedule** tab.

Set the schedule to run daily, with running endlessly, starting today, running each day at 9:00 p.m., stopping at the latest at 5:00 a.m. Based on the archiving criteria that will be used (older than 14 days), running the collection once a day will be sufficient.
3. Select the **Collection Sources** tab.

Add the group MailArchive\_Default as collection source.
4. Select the **Filter** tab.
  - a. Check filter email by age.

- b. Select to filter email by received date.
  - c. Change the filter to relative.
  - d. Set the criteria to older than 14 days.
- 5. Select the collector in the task route designer and right-click.
  - a. Choose copy from the context menu.
  - b. Right-click free storage at the top of the task route.
  - c. Choose paste to create a copy of the collector.
- 6. Select the collector copy; the general tab will be selected.  
Change the name to EC Collect All Email - BlackBerry.
- 7. Select the **Collections Sources** tab.  
Change the group to MailArchive\_BlackBerry.
- 8. Select the **Filter** tab.  
Change the criteria to older than 30 days.
- 9. Right-click the task route in the task route explorer and make the task route active.
- 10. Save the task route.

The task route is now set up correctly and can be promoted to production after verification (Figure 3-8).

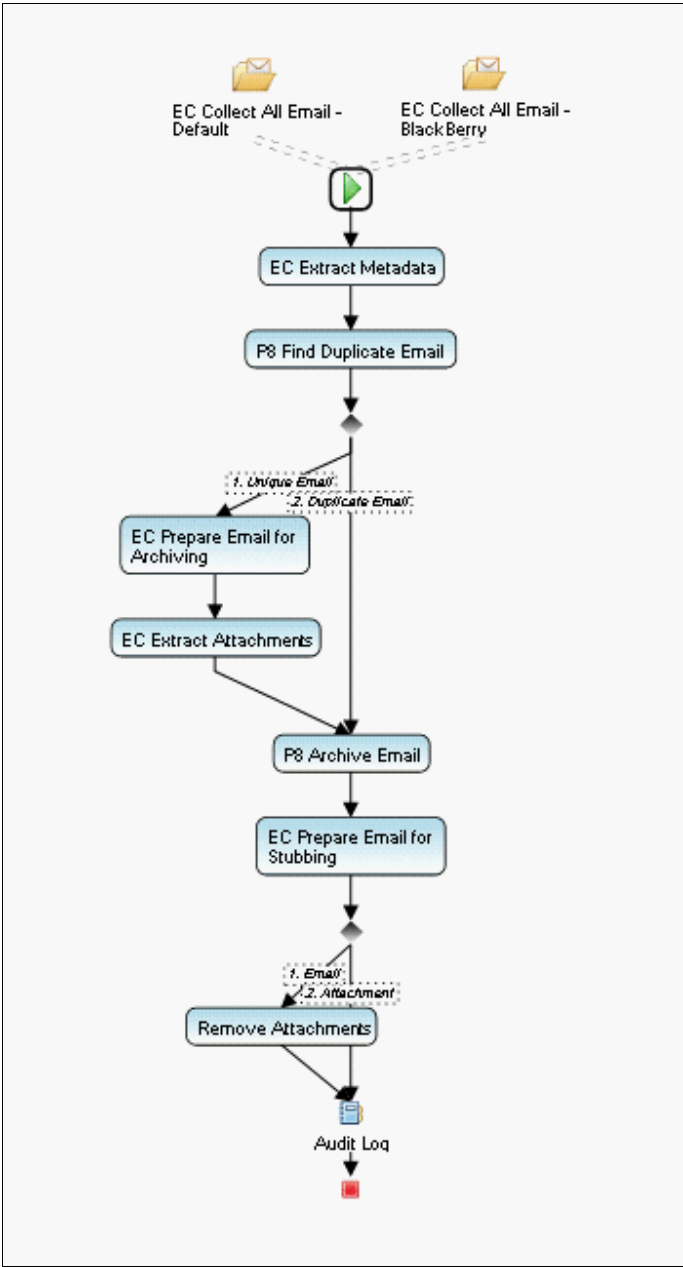


Figure 3-8 P8\_CSS\_EX - Archiving Template with two collectors for both policies

### 3.2.5 Storage management for Microsoft SharePoint

Because of the organic and user-driven nature of Microsoft SharePoint, the topology and taxonomy found across an entire Microsoft SharePoint farm is often uncontrolled and not guaranteed to be consistent. This conflicts with the controlled nature of ECM repositories, and conflicts even more so with records management.

IBM Content Collector for Microsoft SharePoint does not provide a magical solution to this conflict. Implementing a targeted solution for specific Microsoft SharePoint user groups should engage content owners to ensure appropriate metadata is mapped and captured as part of the archival process. However, the same level of engagement might not be possible or productive when implementing a broad solution across a broader area within Microsoft SharePoint, so there will normally be less metadata mapped and captured. The ideal scenario is a Microsoft SharePoint environment where there are strong standards and controls exist on user customization; but in practice the ideal does not exist.

When a particular area in Microsoft SharePoint is at or approaching a storage limit, the prescribed solution is to offload content. When an overall Microsoft SharePoint implementation is struggling with scale and performance issues, the same prescription applies. But the similarities between the two scenarios ends there.

The first scenario deals with a specific area and likely has the attention of content owners who understand the issue. The scope of the area is known and often covers a fairly homogenous area of the topology and taxonomy. If there are special requirements regarding the offloading of content in these cases (such as custom metadata, security, or dynamic behavior when populating the target repository), the requirements will be revealed more quickly, if not understood initially.

However, the second scenario deals with a broad area and content owners who are unaware of the issue. The scope of the area is large and might be unknown, and range over a diverse area of topology and taxonomy. Special requirements regarding the offloading of content in these cases will be difficult and time-consuming to discover and include in the solution. Additionally, the more special requirements there are, the more difficult it is to design a solution that applies and can be implemented broadly.

There are trade-offs that must be analyzed and weighed in every Content Collector for Microsoft SharePoint implementation. In a narrow scenario, metadata mapping is quickly and well defined. In a broad scenario, though, metadata mapping can be an enormous and perhaps insurmountable challenge.

Attempts to design multiple narrow scenario solutions across a large Microsoft SharePoint environment will encounter other difficulties, such as burdensome configuration and maintenance.

When archiving to FileNet P8, the P8 connector has the ability to assign document class dynamically. The ability can reduce the configuration burden in complicated metadata mapping scenarios, but has limitations as well. For more information about this topic, visit the following site:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/property\\_values/t\\_afu\\_dynamic\\_classes\\_or\\_values.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/property_values/t_afu_dynamic_classes_or_values.htm)

When analyzing Microsoft SharePoint for archival scenarios, ask the following questions:

- ▶ What is the Microsoft SharePoint topology? How many farms, web applications, site collections, sites, libraries are involved?
- ▶ What is the Microsoft SharePoint taxonomy? How many content types, custom site columns, custom list columns are involved? Are custom columns actually being populated (excluding default values)?
- ▶ What are the use cases for metadata (built-in and custom columns), both inside and outside of Microsoft SharePoint?

With the answers to those questions you can start to develop a picture of where commonalities exist and can be leveraged broadly; where differences exist and have requirements that must be handled uniquely; and most importantly learn what you do not know about your Microsoft SharePoint environment and need to discover.

The first specific configuration point to determine for any Microsoft SharePoint content areas to undergo archival is the appropriate type of post processing. Ask the following questions to drive discussion about the post-processing options (Figure 3-9 on page 45):

- ▶ Should the content no longer be retrievable in Microsoft SharePoint? If yes, then “Delete” post processing is appropriate.
- ▶ Will there be authoring or collaboration on content after archival? If no, the “Replace with link” post processing might be appropriate.
- ▶ Should future authoring/collaboration be restricted to specific users or groups? If yes, then “Leave item” and “Make item read-only (with exceptions)” options are appropriate.
- ▶ Is storage savings in Microsoft SharePoint desired? If yes, then either use the “Leave item” in combination with the version retention option of “Most recent version only”, or use the “Replace with link” option.

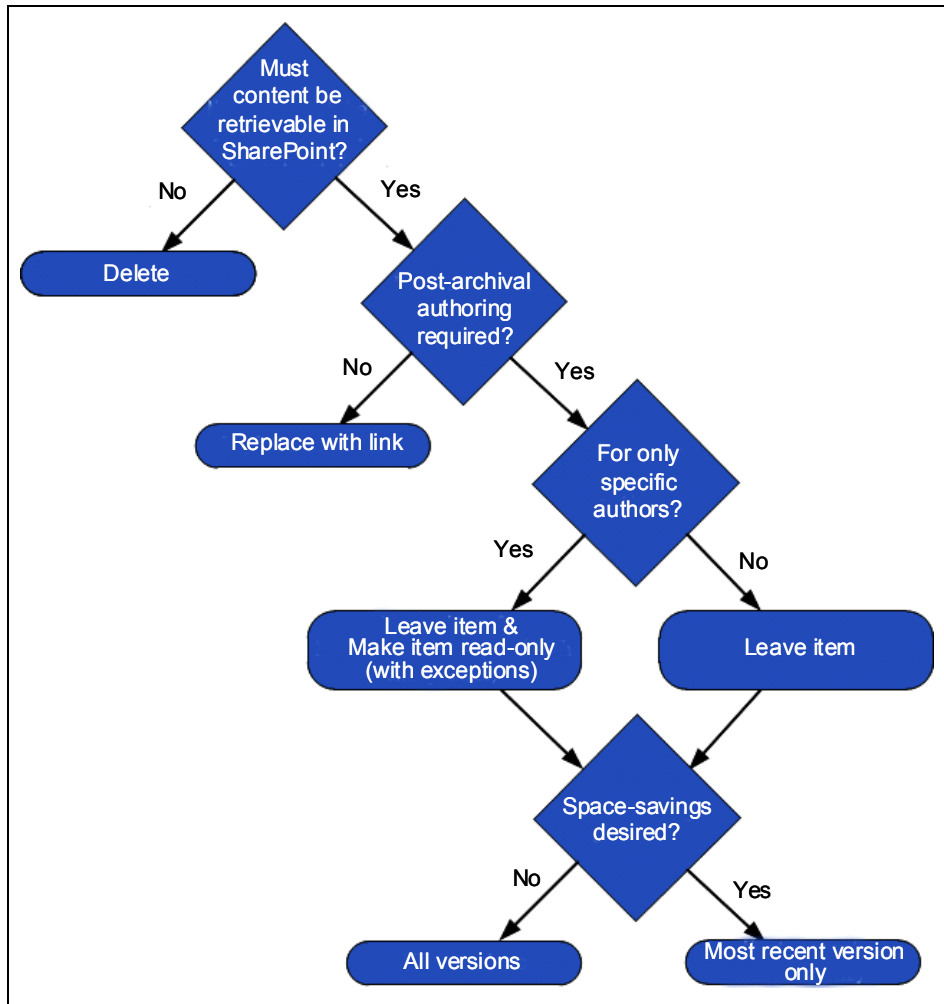


Figure 3-9 Question flow to determine the appropriate post-processing option

When different post-processing options are appropriate depending on Microsoft SharePoint factors, there are two alternatives for configuration. Either create additional task routes, each with a different post-processing option selected for the SP post-processing task, or add multiple SP post-processing tasks to a single task route and add a decision point before those tasks with rules and criteria that evaluate the Microsoft SharePoint factors through metadata.

If the "Replace with link" post-processing option is used, then the relevant Microsoft SharePoint connector link management templates (for example, SP Audio/Manage P8 Links) should be configured for future use. These link

management task routes facilitate the updating and expiring of links in Microsoft SharePoint.

### Deploying storage saving and compliance for Microsoft SharePoint

In our scenario, the company wants to ingest documents from Microsoft SharePoint servers to make them available for eDiscovery. The Microsoft SharePoint sites Human Resources (HR) and Finance have been identified to contain content of interest for legal discovery. Thus, the company decides to archive the content of these sites. Documents archived from the Finance site need to be locked down after collection, so no further modification is possible. Documents that are part of the HR site should be replaced with links, so that the storage occupied by the documents is freed up.

To implement this scenario, deploy the “SP to P8 - With Versions” task route template using the New Task Route option.

After the import completes, perform the following steps to customize the task route for the scenario:

1. Select the collector; the general tab will be selected.
2. Select the **Schedule** tab.
3. Set the collector to run at intervals, starting today, repeating every 15 minutes, running endlessly, and have the collector stop when the task completes.
4. Select the **Collection Sources** tab:
  - a. Edit the collection source.
  - b. Add Finance and HR as site-level collection sources as shown in Figure 3-10.

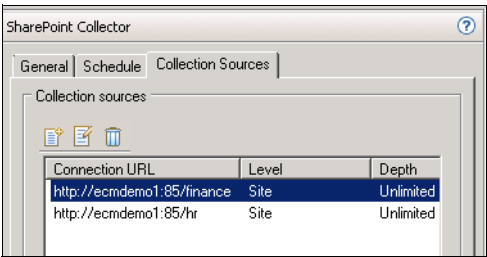


Figure 3-10 Collection sources for HR and Finance departments

5. Add a decision point after the P8 File Document in Folder task.
  - a. Configure the rule to match all documents from the HR site (Figure 3-11 on page 47).



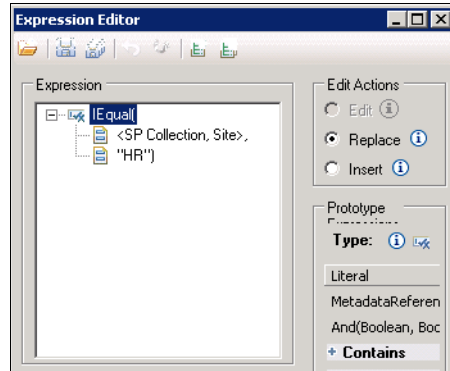


Figure 3-11 Rule to match all documents collected from HR site

- b. Add a SP post-processing task to the branch.
  - c. Configure the task to replace documents with links.
6. Add a second branch for the Finance department.
    - a. Configure the rule to match all documents from the Finance site (Figure 3-12).

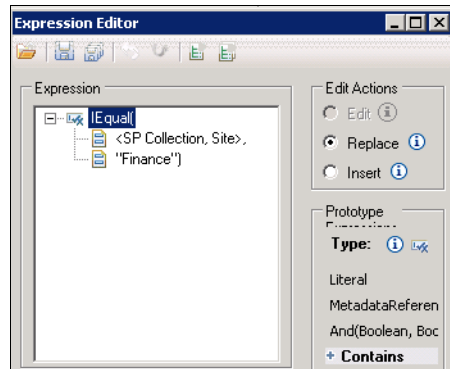


Figure 3-12 Rule to match all documents collected from Finance site

- b. Add a SP post-processing task to the branch.
  - c. Configure the task to mark the item as read only with a version retention of most recent version.
7. Save the task route.

The task route is now set up correctly and can be promoted to production after verification (Figure 3-13 shows the complete task route).

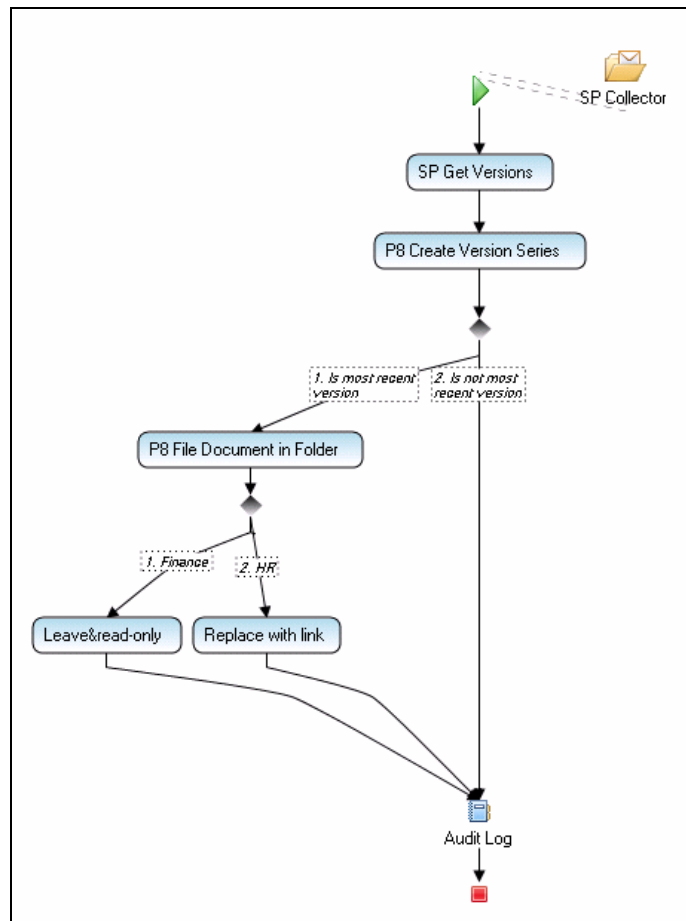


Figure 3-13 Task route for archiving from the Finance and HR Microsoft SharePoint site

### 3.3 Compliance archiving

As stated in 3.1, “Dimensions of content archiving with IBM Content Collector” on page 24, archiving for compliance reasons is primarily driven by minimizing the risk of legal exposure. The mechanisms used to ensure legal compliance are mainly driven by the applicable laws in the countries in which a company does business. The main requirement imposed by laws are centered around enablement of an audit trail to guarantee and prove the correctness of the information that has to be preserved.

### 3.3.1 Compliance archiving for email

Based on audit trail requirements, enterprise email systems such as IBM Lotus Domino and Microsoft Exchange provide for journaling all incoming and outgoing messages. This enables key aspects for ensuring legal compliance:

- Making sure an unmodified copy is always available

Email systems place a copy of the original message in a journal mailbox during delivery of the message to the recipients. If the message cannot be inserted into the journal, the message will not be delivered to the recipients. Because the journal is not accessible to users, there is no way for them to modify the message and thus the originality of a message can be guaranteed.

- Complete recipient information needs to be available

Depending on the email system, this is done either by adding a special property to each message that indicates the list of resolved recipients (Lotus Domino), or by creating an envelope message that contains the same information in the body and has the original message attached (Microsoft Exchange).

Because each incoming or outgoing email is journaled, journals contain by far the most messages compared to other mailboxes. Thus it is highly desirable to archive the content of journals to minimize email server storage requirements and optimize email system performance (see 3.2.4, “Storage management for email” on page 38 for more information about this topic).

IBM Content Collector can archive journal mailboxes, archive their content, and preserve and index their additional distribution information. It can also provide the benefits of deduplication with copies that have been delivered to individual user mailboxes; for more information, see:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/deduplication/c\\_afu\\_deduplication.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/deduplication/c_afu_deduplication.htm)

Because of sheer volume, assign a retention period during archival so that the content can be disposed of at the earliest point allowed by law.

Determining the minimal retention period can be complex, depending on the number of laws and countries a company is doing business in or with. In these times, scenarios might become even more complex because we live in a interconnected world. For example, assume an email is sent from a user in Israel to users in China and Germany, with the email servers hosted in the United States. Finding the applicable laws and retention periods for such a situation is a task that will involve the Legal department of a company and may finally reveal that there is no way to fulfill all applicable laws at the same time.

## **Role of journaling for compliance**

To understand why archiving the copy of an email from a mailbox is not sufficient for legal compliance, we need to take a look at how a message is delivered in a typical email system. When an email is sent to an email server for delivery, different address headers (for example: to; carbon copy (cc); blind carbon copy (bcc)) are inspected by the server. The server will deliver a copy of the message to each recipient that is identified in one of these fields.

If the email is to be delivered to a group, the server will use its directory to expand the group members at that point in time and place a copy of the email in each group member's mailbox. Because group membership is dynamic, any expansion of the group at a later point in time might yield a different set of group members and thus lead to incorrect assumptions about who received a copy of the email.

To address this problem, email systems contain a journaling feature. By enabling journaling, the system ensures that the original message and information about who received a copy will be retained in a distinct journal mailbox. Because users have no access to this mailbox, it is also guaranteed that the email is not modified previous to archiving.

IBM Content Collector can access this journal mailbox and store the distribution information along with the message in the archive. The distribution information is also added to the index and thus available for later eDiscovery in IBM eDiscovery Manager.

If a message is archived from a mailbox, IBM Content Collector will not resolve groups that are part of any address header because there can be no guarantee the information is correct. Thus, the archive and index will only contain the recipient information that is displayed in a user's email client.

Based on the details and reasons for journaling, it is advisable to use the journaling feature of your email system because distribution information is only guaranteed to be preserved in that case.

## **Using SMTP for compliance archiving**

The SMTP Connector might provide an alternative to directly collecting from source systems such as IBM Lotus Domino or Microsoft Exchange. Additionally, it enables you to use any email system that is capable of forwarding journal reports to be used by IBM Content Collector for compliance archiving. For more information about this topic, see:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/connectors/c\\_afu\\_smtp\\_conn.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/connectors/c_afu_smtp_conn.htm)

The following considerations can help guide you to an informed decision regarding whether to use the SMTP Connector:

- ▶ If used together with storage management task routes, attachments will still be deduplicated, but actual messages will not be deduplicated because different storage formats are used. Email Connector uses CSN for Lotus Domino, MSG for Microsoft Exchange. SMTP Connector uses EML for storing Multipurpose Internet Mail Extensions (MIME) mails. This leads to slightly higher storage requirements in the ECM system.
- ▶ When deploying compliance archiving to a geographically distributed environment, performance is considered to be better when using the SMTP connector, compared to Email Connector. This is due to network latency having a major influence on Email Connector performance. Because the SMTP Receiver receives email from the servers and stores them in a queue directory, the SMTP Connector archiving performance is not influenced by network delay, but by queue directory performance.

**Lotus Domino consideration:** For Lotus Domino, you may use replication to replicate the journals of remote servers to a server local to the IBM Content Collector server and have the Email Connector archive from the local Lotus Domino server. So this option is preferred in most cases, compared to using the SMTP Connector for high latency networks.

- ▶ The SMTP Connector and Receiver use a queue directory for storing messages. Set up this directory to be highly available. Furthermore, backups of the directory need to be created on a regular basis. Using a mechanism like Microsoft Distributed File Systems (DFS) is advisable.

**Lotus Domino consideration:** If you plan to forward journal reports from Lotus Domino to the Content Collector SMTP Connector, ensure that your Lotus Domino server supports forwarding encrypted documents over SMTP. If necessary, contact IBM Software Support to determine whether your Lotus Domino installation contains the required fix for APAR LO58538. If the required fix is not installed, the body of email documents that were encrypted by the sender is not forwarded properly, and the content of the journal email body is lost.

## Deploying task routes for email compliance archiving

The company in our scenario decides to use Email Connector for journal archiving. The company decided against using the SMTP Connector because all email servers are local, so network latency is low and slightly better deduplication is rated as an important benefit.

The company used the following parameters:

- ▶ All email will be immediately removed from the journal `journal@company.com` after archiving.
- ▶ The rate of incoming email is currently 120 k mails per day. This is estimated to grow to 150 k mails per day within the next two years, with an average email size of 80 KB.

Based on these numbers, it is advisable to size the email server storage to be able to handle a backlog of one week. This allows for a buffer in case spikes in the rate of email received are higher than the throughput that the archiving solution has been sized for.

In addition, this allows the email system to be unaffected if either the ECM system or IBM Content Collector become unavailable. With higher volumes of processing, a smaller buffer may be used because of associated storage requirements.

The “P8\_CSS\_EX\_3.1 - Journal Archiving (with Mail Delete)” task route template best fits the described scenario. Deploy this template using the New Task Route option.

After the import completes, perform the following steps to customize the task route for the scenario:

1. Mark the task route as active.
2. Click the collector; the General tab will be selected.

Mark the collector as active.

3. Select the **Schedule** tab.

Set the schedule to run daily, with running endlessly, starting today, running each day at 9:00 p.m., stopping at the latest at 7:00 a.m. Based on the estimated volume to be archived (120 k mails per day) running the collector once an hour should be sufficient.

The schedule should be configured so that the number of the items in the journal stays *significantly* below the recommended threshold. Reaching the threshold is known to have a negative performance impact, as listed in Table 3-2 on page 53.

Table 3-2 Folder item limit for different versions of Microsoft Exchange<sup>4</sup>

Exchange version	Folder item limit	Reference
2000/2003	5000	<a href="http://technet.microsoft.com/en-us/library/cc535025.aspx">http://technet.microsoft.com/en-us/library/cc535025.aspx</a>
2007	20000	<a href="http://technet.microsoft.com/en-us/library/cc535025.aspx">http://technet.microsoft.com/en-us/library/cc535025.aspx</a>
2010	100000	<a href="http://technet.microsoft.com/en-us/library/ee832791(EXCHG.140).aspx">http://technet.microsoft.com/en-us/library/ee832791(EXCHG.140).aspx</a>

4. Select the **Collection Sources** tab.

Add journal@company.com as the collection source of type journal.

5. Save the task route.

The task route is now set up correctly and can be promoted to production after verification.

## Deploying an SMTP task route for compliance archiving

The task route suitable for the scenario is the P8\_CSS\_SMTP - Archiving Template. Deploy it using the New Task Route option. To set up the task route, the task route and the collector have to be marked as active. All other options are already populated by the template.

## Migrating local archives

Local archives might also represent a legal exposure because the information contained is not available for eDiscovery. To make this information discoverable, IBM Content Collector is used to ingest the content into the ECM system.

In addition to local archives from user workstations, local archives might need to be archived from backups to have all necessary documents available for discovery. Deduplication provides essential relief in this situation. It ensures that duplicates are saved in a storage-efficient manner, and that binary identical duplicates are not stored at all and thus do not occupy any storage in the ECM system.

<sup>4</sup> Also see <http://blogs.technet.com/b/exchange/archive/2009/12/07/3408973.aspx>

For Microsoft Exchange, it is advisable to disallow further local archive creation. For computers with Microsoft Outlook installed, select the **Disable PST file creation** option on the Edit Collection Source window (Figure 3-14).

This figure shows the collector options. Storage management policies have eliminated the need for users to swap out the content of a mailbox to a local archive.

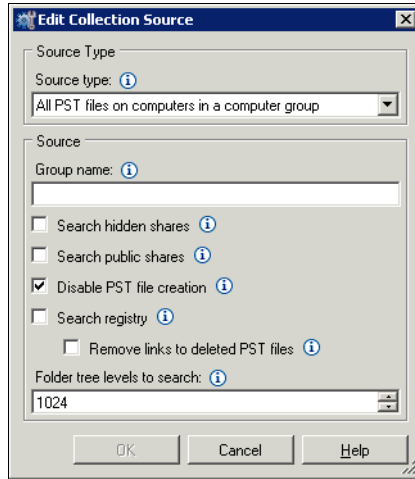


Figure 3-14 Collector option to disable PST file creation

Local archives will be removed from user workstations during the ingestion process, but users still need to have access to these local archives, both during and after migration. Thus, we will make sure that content archived from local archives that are in use by users is searchable using the IBM Content Collector search, as though it had been archived from users' mailboxes.

Because of the sheer number of local archives that are to be processed (typically terabytes to petabytes of content), it is advisable to set up a separate IBM Content Collector server cluster. This way, processing local archives does not impact the storage management task route, or the journal compliance task routes.

## Planning for migration of local archives

The general process of local archive migration is described in the information center:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/local\\_files/t\\_afu\\_archiving\\_local\\_files.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/local_files/t_afu_archiving_local_files.htm)



However, there are further points to consider when performing enterprise local archive migration including local archives from backups.

- ▶ Although local archives that are in use will be tagged with the owning user, local archives that are not in use, or located backups, are not tagged with the owning user. Because IBM Content Collector will only archive local archives that are assigned to an owner, two possibilities exist:
  - If users still need access to the content of the local archives, each local archive needs to be assigned to the correct owner before archival.
  - If users do not need access to the content of the local archives, which is the typical case for local archives that are located in backups, assign all local archives to the IBM Content Collector administrative user.

For details about how to assign owners, see:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/local\\_files/r\\_afu\\_listpsts.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/local_files/r_afu_listpsts.htm)

- ▶ Put additional file system task routes in place to manage error handling and post processing. Because of the number of local archives to migrate, use a file system task route to streamline processing. Depending on the result of processing:
  - Remove local archives that have been processed successfully.
  - Move local archives that had processing errors to a special folder that is monitored. The documents might need to be inspected manually and appropriate action has to be taken (for example, involve IBM support, and repair the local archive).
- ▶ If your email system is Microsoft Exchange, consider moving local archives from workstations to a central file share before beginning migration. PST files may only be accessed by one process at a time. Thus, if the user has Outlook open and is accessing the PST file, IBM Content Collector cannot access the PST file and vice versa. Also, network latency has a significance influence on archiving performance. Thus, it is advisable to move the PST files to a central share that cannot be accessed by users.

To guarantee that PST files are not modified after copying to the central share, PST modification should be disabled on all workstations; for guidance see:

<http://support.microsoft.com/kb/954268>

- ▶ If your email system is Lotus Domino and you will be collecting from different sites, set up distributed file shares with a replication to the central archiving site. For guidance, see:

<http://www.ibm.com/support/docview.wss?uid=swg21455675>

### 3.3.2 Compliance archiving for Microsoft SharePoint

In the Microsoft SharePoint context, compliance has two impacts, depending on how strict your view of compliance is.

The first impact depends on whether any version of content may exist in two systems at the same time. If permissible, then the “Leave item” post-processing option may be used. If not permissible, then either the “Replace with link” or the “Delete” post-processing option must be used.

The second impact is metadata preservation. Ask and discuss the following questions to help determine what metadata must be preserved to achieve compliance:

- ▶ What “built-in” and custom metadata has business value, is not found within the content, and facilitates repository use cases?
- ▶ Do user-created metadata fields and content types, which content area owners might be unaware of, have business value?
- ▶ Does metadata need to have a repository use case to have business value?

### 3.3.3 Compliance archiving for IBM Connections

This section briefly describes how to deploy compliance archiving for IBM Connections using IBM Content Collector. For more detailed information, see Chapter 8, “IBM Connections integration” on page 275.

With the rise of social software such as IBM Connections, communication that previously might have taken place through email is now moving to IBM Connections. As a result, there is increasing focus on archiving IBM Connections content for ensuring legal compliance and to allow for legal discovery.

IBM Content Collector can be used to capture all content stored in IBM Connections. If content is changed (for example, a comment is added to a blog post), IBM Content Collector will capture a new version<sup>5</sup> of this blog post that includes the comment.

In our scenario, the company recently deployed IBM Connections 3.0.1 to the enterprise and wants to extend the compliance archiving that was in place for email to IBM Connections. Content captured from IBM Connections should be kept for 10 years.

To implement this scenario, deploy the “CX to P8 - Calculate Expiration Date” task route template using the New Task Route option.

---

<sup>5</sup> The native versioning feature of the ECM system is not used.

After the import completed, perform the following steps to customize the task route for the scenario:

1. Select the collector; the General tab will be selected.
2. Select the **Schedule** tab.  
Set the collector to run at intervals, starting today, repeating every 15 minutes, running endlessly, and have the collector stop when the task completes.
3. Select the **Collection Sources** tab.
  - a. Edit the collection source.
  - b. Check all applications to archive content from all applications.
- ▶ Select the **Calculate expiration** task.
  - a. Open the expression editor in the date set by expression section.
  - b. Launch the expression editor.
  - c. Select the second parameter of the AddYears operator.
  - d. Increase the value to 10 years.
  - e. Press **OK** two times to close the edit screens.
4. Save the task route.
5. Right-click the task route explorer and mark the task route as active.

The task route is now set up correctly and can be promoted to production after verification.

## 3.4 Business process management

Although the execution of a business process is managed by a business process management system such as IBM Business Process Manager, the workflow might be started manually. IBM Content Collector allows for adding documents automatically to an ECM system to start a business process. By eliminating the manual process, several enhancements can be realized:

- ▶ Shorter turnaround times  
The time needed to start a business process is decreased.
- ▶ Decreased error rate  
When additional metadata is to be provided by users upon adding the document, free text fields will yield a higher error input rate than selection dialogs, which simply allow for entering correct values. For example, entering

a customer number manually is more error prone than selecting a customer number from a list of valid customer numbers.

- ▶ Increased processing volume

Because no user interaction is required, the processing rate can be scaled up to thousands of documents per hour or day.

To address these opportunities for enhancement, IBM Content Collector supports the following methods of adding documents to an ECM system:

- ▶ Adding a document automatically
- ▶ Allowing users to add a document from within the context of the source system
- ▶ Adding a document that has additional metadata associated
- ▶ Allowing users to add a document from within the context of the source system, while collecting additional associated metadata

### 3.4.1 Business process management for file shares

The major business process scenario for file shares is the automated ingestion of documents to the ECM system. This might include additional metadata presented in the form of a file with the same name that is placed beside of the actual content file. The supported format for these metadata files are extensible markup language (XML) and coma-separated value (CSV).

#### **Adding documents to FileNet P8 automatically**

In our scenario, the company wants to automate ingestion for one of the business processes. Once a day a supplier uploads all invoices (PDF files) to be paid by the company to a file share through File Transfer Protocol (FTP). The files are placed in subfolders named by the department to charge. As of today, a user takes the individual files and adds them to FileNet P8 through workplaceXT. That user also enters the department identifier given by the folder name as a property for each document. The company wants to automate this process using IBM Content Collector.

To do this, deploy the FS to P8 Archiving (Delete) template using the New Task Route option.

After the import completed, perform the following steps to customize the task route for the scenario:

1. Select the collector; the General tab will be selected.

2. Select the **Schedule** tab.

Configure the schedule to run Daily, starting today, in endless mode, stopping collection when the task completes.

3. Select the **Collection source** tab.

- a. Add the location `\\staging.company.com\invoicesToIngest` as the UNC path.
- b. Monitor subfolders with a folder depth of two. Invoices will be placed in a subfolder named by the department, so the collector needs to descend one level.
- c. Choose NTFS post processing.

4. Select the **Filter** tab.

Change filter file extensions to include and specify PDF as included extension.

5. Select the **P8 Create Document** task.

- a. Select the **Invoice** document class in the property mappings section of the task configuration.
- b. Leave the document `title` property to be mapped to the file name without extension.
- c. Double-click **DepartmentID** to open the Edit Expression window.
- d. Select **advanced**.
- e. Launch the Expression Editor by clicking the icon next to it.

Use a regular expression substitute to obtain the name of the parent folder of the file that is processed (see Figure 3-15 on page 60). We use regex `(.*)\\` and replace all matches with a blank. The regular expression will match all characters before, including the last backslash of the path. For example, the file path `\\staging.company.com\invoicesToIngest\3591` will be transformed to `3591`, which is the department number we want to extract.

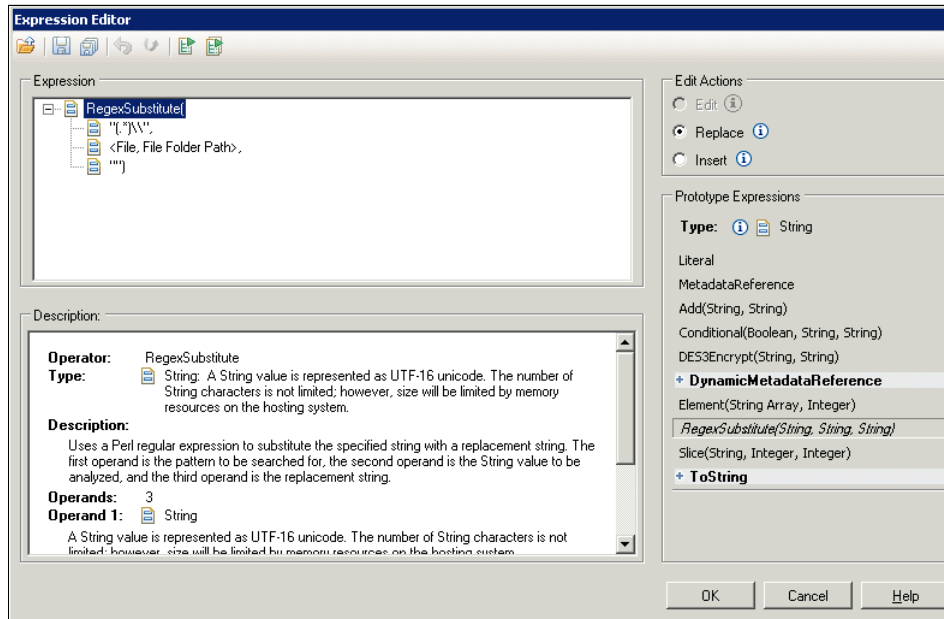


Figure 3-15 Expression to extract the parent folder of the processed file

6. Select the **P8 File Document in Folder** task. Choose **/invoices** as the folder path.
7. Save the task route.

The task route is now set up correctly and can be promoted to production after verification.

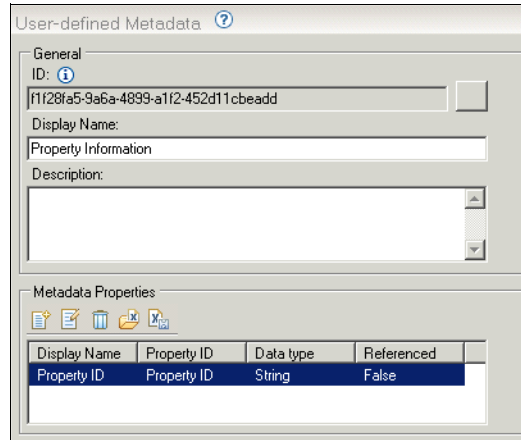
## Associating additional metadata

After automating ingestion for the invoice scenario, the company decides to automate ingestion of another process. All properties of the company are managed centrally. All written communication that is received for a property is scanned by another application and sent to a central file share for processing.

An additional CSV file with the same name contains metadata that is not contained in the document. In this case the CSV file contains the unique property identifier that marks the property for which the document was received. This information is used by the business process to assign the document to the correct knowledge worker for this property.

To automate ingestion of these documents, complete the following steps:

1. Select **User-defined Metadata** under Metadata and lists.
2. Create a user-defined metadata named **Property Information** to hold the additional metadata (Figure 3-16). Add a property named **PropertyID** of type string.

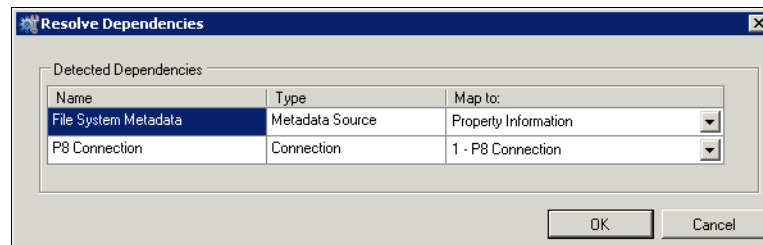


The 'User-defined Metadata' dialog box is shown. It has a 'General' tab and a 'Metadata Properties' section. In the 'General' tab, the 'ID' field contains 'f1f28fa5-9a6a-4899-a1f2-452d11cbeadd', the 'Display Name' is 'Property Information', and the 'Description' is empty. The 'Metadata Properties' section contains a table with the following data:

Display Name	Property ID	Data type	Referenced
Property ID	Property ID	String	False

Figure 3-16 User-defined Metadata to hold property information

3. Use the **New task** route option to import the **FS to P8 Archiving (Delete Metadata Files)** template.
4. In the **Resolve Dependencies** window, select **Property Information** to be used for **File System Metadata** (Figure 3-17) during import.



The 'Resolve Dependencies' dialog box is shown. It has a 'Detected Dependencies' section with a table containing the following data:

Name	Type	Map to:
File System Metadata	Metadata Source	Property Information
P8 Connection	Connection	1 - P8 Connection

At the bottom of the dialog are 'OK' and 'Cancel' buttons.

Figure 3-17 Map File System Metadata to Property Information during import

5. Select the collector; the **General** tab will be selected.
6. Switch to the **Schedule** tab. Set the collector to run daily, starting today, running endlessly, and complete when the task completes.

7. Select the **Collection sources** tab.
  - a. Add the location `//staging.company.com/propertyCommunication` as the UNC path.
  - b. Do not include subfolders.
8. Select the **FSC Associate Metadata** task.
  - a. Switch from the metadata file name tab to the metadata mapping tab.
  - b. Map Property ID to column 1 in the delimited files metadata mapping section.
9. Select the **P8 Create Document** task.
  - a. Choose the **PropertyCommunication** document class.
  - b. Double-click the **Property ID** property to open the Edit Expression window.
    - i. Choose the metadata option.
    - ii. Select **Property Information** as the metadata source.
    - iii. Choose **Property ID** as the property.
    - iv. Click **OK** to close the window.
10. Select the **P8 File document in Folder** task.

Set the folder path to be used to `/propertyCommunications`.
11. Select the **FSC Post Processing - replace files with shortcuts** task of the left branch.

Uncheck the replace file with shortcut option to make sure the file is deleted after archival.
12. Save the task route.

The task route is now set up correctly and can be promoted to production after verification.

### **Adding recovery functions using error task routes**

Finally, the company decides to automate ingestion of another, more complex process. The company has an existing reporting application that will generate a set of reports. In addition, an XML file consisting of information about the reports referencing the report files for this run is generated (Figure 3-18 on page 63).



The content and format of the XML file is controlled by the reporting application, so it is not specifically generated for IBM Content Collector.

```
<?xml version="1.0" encoding="UTF-8"?>
<reportexecution xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <report product="981" file="rp_23984.pdf" />
  <report product="856" file="rp_24584.pdf" />
  <report product="881" file="rp_23974.pdf" />
  <report product="951" file="rp_22984.pdf" />
</reportexecution>
```

Figure 3-18 Example XML metadata file

The product attribute of the report is mapped to the productID property of the report document class that is used. If one of the reports cannot be added to FileNet P8, the metadata should be preserved for analysis and for rerunning ingestion of the problematic reports, after the underlying problem has been solved.

To automate ingestion of these documents, complete the following steps:

1. Select **User-defined Metadata** under Metadata and lists.
2. Create a user-defined metadata named Report Information to hold the additional metadata (similar to what is shown in Figure 3-16 on page 61).
  - a. Add a property named ProductID of type string.
  - b. Add a property named Report file of type string.
3. Use the **New task route** option to import the FS to P8 Archiving (Delete Metadata Files) template.
4. In the “Resolve Dependencies” window, select **Report information** to be used for File System Metadata (similar to what is shown in Figure 3-17 on page 61).
5. Select the collector; the General tab will be selected.
6. Select the **Schedule** tab.

Set the collector to run daily, starting today, running endlessly, and complete when the task completes.
7. Select the **Collection Sources** tab.
  - a. Add the location //staging.company.com/reports as the UNC path.
  - b. Do not include subfolders.
8. Select the **Filter** tab.

Change the file extension filter to include files with a .XML extension.
9. Select the **FSC Associate Metadata** task.
  - a. Switch the input file type to metadata file.

- b. Select to base the name of the document on values of the metadata file.
      - i. Choose **report file** as the property that contains document file names.
      - ii. Check the ignore missing files option.
10. Switch from the document name tab to the metadata mapping tab.
  - a. Change the format type to XML.
  - b. Launch the wizard (see Figure 3-19):
    - i. Map Report ID to /reportexecution/report/@product.
    - ii. Map Report file to /reportexecution/report/@file.

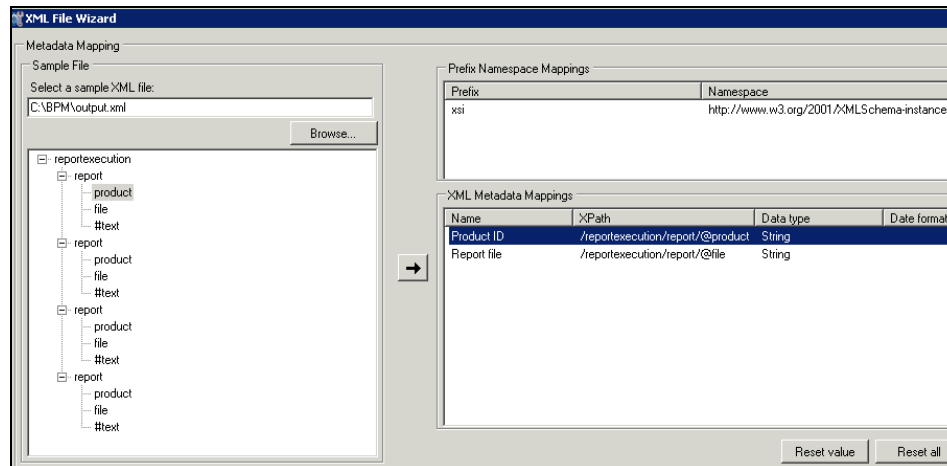


Figure 3-19 XML metadata mapping wizard

- iii. Press **OK**.
11. Select the **P8 Create Document** task.
  - a. Choose the **Report** document class.
  - b. Double-click the **Product ID** property to open the Edit Expression window.
    - i. Choose the metadata option.
    - ii. Select **Report Information** as the metadata source.
    - iii. Choose **Product ID** as property.
    - iv. Click **OK** to close the window.
12. Select the **P8 File document in Folder** task.
 

Set the folder path to be used to /reports.

13. Select the **FSC Post Processing - replace files with shortcuts** task of the right branch.
  - a. Right-click the task and select **Delete**.
  - b. Confirm the delete operation.
  - c. Connect the “is metadata” branch to the remaining FSC Post Processing task (Figure 3-20).

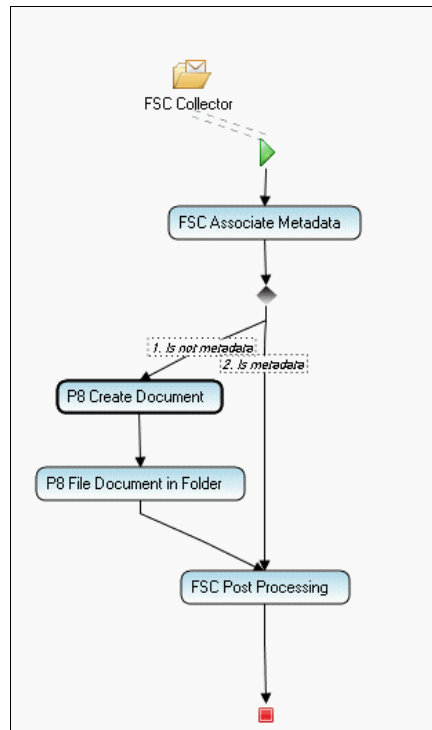


Figure 3-20 Task route after deleting the second FSC Post Processing task

14. Select the **FSC Post Processing** task of the left branch.

Uncheck the replace file with shortcut option to ensure the file is deleted after archival.
15. Switch to the **Error** task route.
16. Add a decision point.
17. Add an **FSC Post processing** task in a new branch.

Set the FSC Post processing task to rename the file and add a .error extension.

18. Edit the rule for the new branch and set the rule to be evaluated when the FSC Metadata indicates this is the metadata file (Figure 3-21).

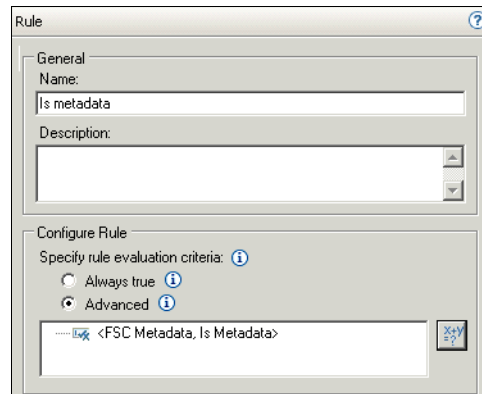


Figure 3-21 The “Is metadata” rule

19. Name the branch “Is metadata.”
20. Name the branch that goes straight to the end of the task route “Is not metadata.” The task route should look as shown in Figure 3-22.

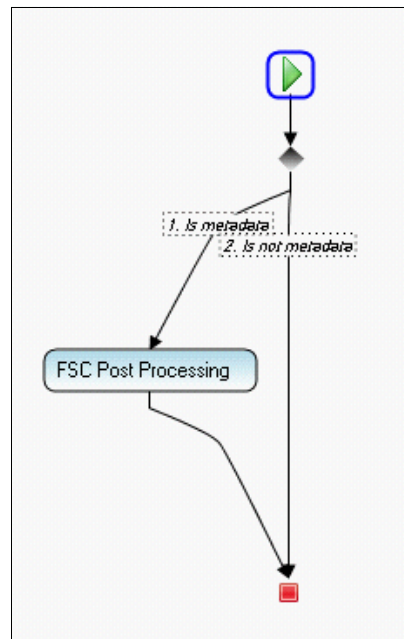


Figure 3-22 Error task route for preserving the metadata file

21. Save the task route.

If an error occurs during evaluation of the task route, all entities will be routed through the error task route. The error task route will add an extension to the file to indicate that the file and its referenced files could not be processed. Because we create an individual P8 document per report, some reports might already be archived into P8.

The company wants to handle a set of reports as atomic pieces, so the task route is modified to add all reports as content elements to one P8 document. If adding one of the content elements fails, the P8 document will not be created. Thus atomicity is guaranteed.

To perform this modification, check the `Group documents by metadata file` option of the `FSC Associate Metadata` task.

The task route is now set up correctly and can be promoted to production after verification.

### **3.4.2 Business process management for email**

The major business process scenario for email is the user-driven ingestion of documents to the ECM system. This might include additional metadata added by the user after a document is moved to a folder. In addition, email can be processed from a designated mailbox to move the email or attached documents into the ECM system. To view email after it has been added to the ECM system, the document viewer component can be integrated into the viewing application.

For interactive scenarios, IBM Content Collector is configured to poll specific folders on a regular basis. However, as the number of mailboxes increases, so does the turnaround time for crawling the folders. When planning for deploying folder monitoring, it is important to think about the turnaround time to be achieved. If a high number of users need to add documents to the ECM system for business processes and a short turnaround time within minutes is required, an automatic approach is preferable.

#### **Interactively driven by folders**

Commonly, users receive email that is used to start a business process. Because these emails are intermixed into the other email that users receive, moving them to a folder is the natural choice to indicate these emails are to be added to the ECM system.

In our scenario, the company wants to store all documents regarding communication with a specific customer in FileNet P8. The email and each attachment are to be saved as a separate object. However, all documents should

be filed in a folder that has the customer name. This is the preferred data model for the customer relationship management (CRM) system that is deployed at the company.

In this scenario, users will assign emails to a folder structure in their mailbox. This structure consists of a folder “customers” and user-created subfolders for each customer. About 100 knowledge workers, who are part of a customer-facing department, will use this function.

A group named `BPM_customerCommunication` has been set up by the domain administrators. The email will be archived separately for storage management purposes and handled according to the configured lifecycle for email documents.

Deploy the `P8_EX - BPM Template` template using the `New Task Route` option.

After the import completed, perform the following steps to customize the task route for the scenario:

1. Select the collector; the **General** tab will be selected. Mark the collector as **active**.
2. Select the **Schedule** tab. Set the schedule to run at intervals, with running endlessly, starting today, running each day starting at 7:00 PM, stopping at latest at 5:00 PM. Based on the scenario running the collection once an hour will be sufficient.
3. Select the **Collection Sources** tab.
  - a. Add the `BPM_customerCommunication` group as a collection source.
  - b. Remove all monitored folders.
  - c. Add a new monitored folder named `/customers`.

**Folder information:** If IBM Content Collector finds the folder is missing in the mailbox during crawling, it will create the folder.

4. Select the **P8 File Documents in Folder** task.
  - a. Press **Edit** in the folder path section.
  - b. Choose **regular expression**.
  - c. Set **Email** as metadata and **Folder** as property.
  - d. Use `.*` as the replacement regular expression.
  - e. Use `/customers/$2` as the replacement string.

**Expression information:** Note that `.*` will match the complete path of a message, except for the parent folder of the message. This match is replaced with `/customers/$2`, where `$2` refers to the remaining text, not matched by the regular expression. For example, `/customers/2010/Bolts` and `More` will be transformed to `/customers/Bolts` and `More`.

5. Save the task route.
6. Mark the task route as active in the task route explorer.

The task route is now set up correctly and can be promoted to production after verification.

### Interactively driven by folders with additional metadata

Sometimes additional information is required to correctly process a document. Examples for this include assignment of project-specific information, manual assignment of a retention period, or data that requires complex validation or lookup.

For such scenarios, IBM Content Collector provides the ability to present a customizable form to the user to fill in additional metadata before an email is processed. The form is highly customizable and can be connected to any back-end web service to fetch information (for example, fetch customer information from the CRM system). This is highly preferred because there is no way to enter malformed customer information. For more information about customizable forms, see:

<http://www.ibm.com/developerworks/data/library/techarticle/dm-0912contentcollectorform/index.html>

In our case, the company stores email related to projects in FileNet P8 and retains them for a specific time. The retention time depends the category assigned to an email by the user (Table 3-3).

*Table 3-3 Categories and assigned retention periods*

Category	Retention period	Description
Discussion	2 years	Used for documents that contain discussion pertaining to a certain project
Reference	5 years	Used for email that contains reference material
Confidential	10 years	Used for email containing

Information about projects is available through a web service. The form will consume this information through a custom web service wrapper that has been built to expose the web service as ItemFileReadStore, as shown in Figure 3-23. For more information, see:

<http://dojotoolkit.org/reference-guide/1.8/dojo/data/ItemFileReadStore.html>

After it is archived, users use IBM Content Navigator to access the content; see:

<http://publib.boulder.ibm.com/infocenter/cmgmt/v8r4m0/topic/com.ibm.developingeuc.doc/eucdv003.htm>

To be able to view email in IBM Content Navigator, the email is stored as one document in P8.

Valid	Subject
✓	Order
✓	RE: Customer meeting
✗	FYI: IBM Content Collector 2.1.11
✗	coffee?<eom>

\*Project: Project Moon

\*Retention Category: Reference

Buttons: Submit All Documents, Submit Selected Documents, Cancel, Apply to Selected Documents

Figure 3-23 Customized metadata form to obtain project name and communication category

To deploy this scenario, perform the steps necessary as described in the information center:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/connectors/t\\_afu\\_enable\\_add\\_info.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/connectors/t_afu_enable_add_info.htm)

Use the following information to complete the steps:

- The project name will be used to assign the email to a folder, and the category will be used to assign a expiration date. Therefore, adding these properties to the project document class is not necessary. However, it is necessary to add the ICCExpirationDate property to the document class to



allow the expiration management tool to delete the documents after they expire.

- ▶ Name the user-defined metadata Project information and add the following properties:
  - Project Name of type string
  - Category of type string
- ▶ In step 9 (described in the information center), deploy the P8\_EX - BPM Template using the New Task Route option.
  - a. Select the collector; the General tab will be selected.
    - i. Mark the collector as active.
    - ii. Check the Collect email with additional archiving information option.
  - b. Select the **Schedule** tab. Set the Schedule to run at intervals, with running endlessly, starting today, running each day starting at 7:00 PM, stopping at latest at 5:00 PM. Based on the scenario, running the collection once an hour will be sufficient.
  - c. Select the **Collection Sources** tab.
    - i. Add the BPM\_projectCommunication group as a collection source.
    - ii. Remove all monitored folders.
    - iii. Add a new monitored folder /Project Communication.
  - d. Add a **MC Retrieve Additional Metadata** task after the EC Extract Metadata task. Configure the task to extract project information.
  - e. Select the **EC Prepare Email for Archiving** task. Uncheck the save document without attachment option.
  - f. Remove the following items:
    - i. EC Extract Attachments task.
    - ii. P8 Create Document for Attachment task.
    - iii. Attachment rule.
    - iv. First decision point.
    - v. P8 Link documents.
    - vi. Has no attachments rule.
    - vii. Second decision point.
  - g. Select the audit task at the end of the task route. Remove all properties marked as invalid.

- h. Select the **P8 Create Document for Email** task.
  - i. Select the Project document class.
  - ii. Map all desired properties.
  - iii. Edit the expression for ICCExpirationDate property.
  - iv. Set the expression to use the Retention Category from Project Information and add it as years to the email received date, as shown in Figure 3-24.

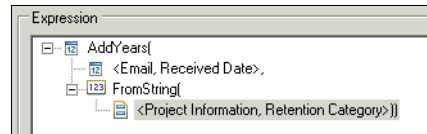


Figure 3-24 Calculate retention date from Project information category and email received date

- a. Select the **P8 File Document in Folder** task.
  - i. Edit the folder path.
  - ii. Select the calculated value option.
  - iii. Add /projects/ as a literal.
  - iv. Add a metadata reference to the project Project Name property of Project Information (Figure 3-25).

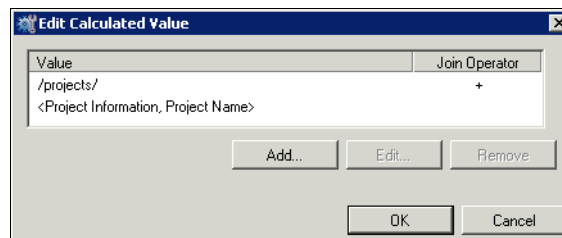


Figure 3-25 Final expression for mapping the folder path

- b. Mark the task route as active and save it.

The task route is now set up correctly and can be promoted to production after verification (Figure 3-26).

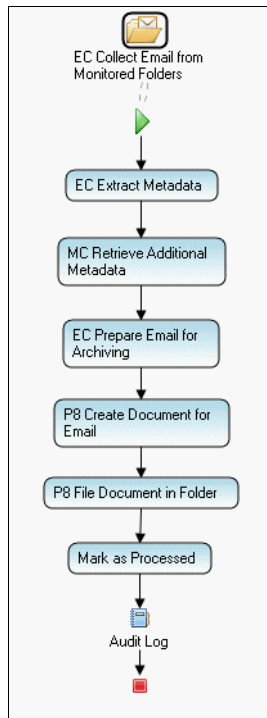


Figure 3-26 Final task route for collecting project documents with setting user-defined expiration

### Automatically without separating attachments using SMTP

The storage format of an email can be important for business process-related scenarios if the whole email needs to be archived and it is not possible to embed the document viewer in the viewing application. If the viewing application supports viewing MIME email, using the SMTP Connector is the ideal choice.

In our scenario, the company creates an email address to which customers can send comments or complaints. The business goal is to improve turnaround time and customer satisfaction by automating the process of adding these emails to FileNet P8 on which the CRM system is based. Knowledge workers review the emails after they are added and assign them to a specific customer account.

To implement this scenario, create a new task route named SMTP - BPM (Business Process Management) using the New Task Route option.

Perform the following steps to customize the task route for the scenario:

1. Add an SC Collect All Email collector to the task route.
2. Click the **Schedule** tab.  
Select the collector to run every 15 minutes, starting today, stopping when the task completes.
3. Add an **SC Extract Metadata** task.
4. Add an **SC Prepare Email for Archiving** task.  
Uncheck the Save document without attachment option.
5. Add a **P8 Create Document** task.
  - a. Select the **Complaint document** class.
  - b. Map the desired properties.
6. Add an **SC Prepare Email for deletion** task.
7. Add an **SC Delete Email** task.
8. Add an **Audit** task. Select the following properties to be audited:
  - a. Email, Subject
  - b. P8 Create Document, Object ID

The task route is now set up correctly and can be promoted to production after verification (Figure 3-27).

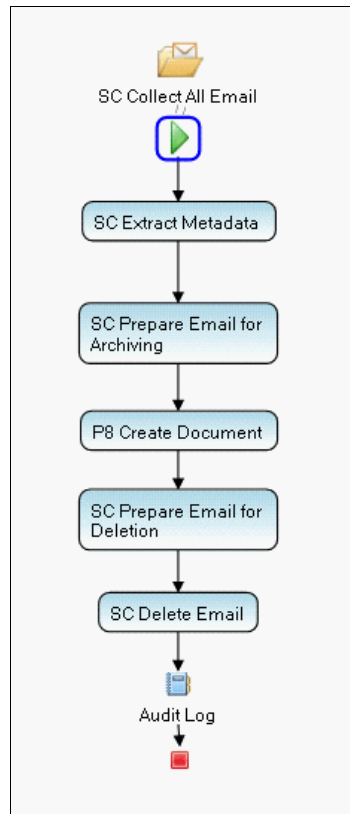


Figure 3-27 Task route for storing MIME messages for business process-related scenarios

### 3.5 Use case 1A and 1B: Email archiving for compliance and storage

Implementation details for use case 1 A and 1 B that are described in 2.1, “Use case 1: Email archiving for compliance and storage management” on page 18 can be found in the following sections:

- ▶ 3.2.4, “Storage management for email” on page 38
- ▶ 3.3.1, “Compliance archiving for email” on page 49

## 3.6 Conclusion

In this chapter we provided an overview of the three major themes that drive archiving solutions such as that described in use case 1 of 2.1, “Use case 1: Email archiving for compliance and storage management” on page 18. We explained the specific requirements and benefits associated with each of the themes. We translated these themes into a broad range of use cases and specific deployment examples. In the next chapter, we examine details of deploying and extending the task routes to satisfy specific archiving goals and requirements.



## Designing, adapting, and deploying task routes

In this chapter we explain how to design archival-related aspects of IBM Content Collector for Email, Microsoft SharePoint, Files or IBM Connections solutions. It covers the various approaches, alternatives available, and key considerations. We use 2.1, “Use case 1: Email archiving for compliance and storage management” on page 18 to illustrate the process of selecting, designing, or importing task routes to fulfill the requirements imposed by specific use cases.

**Configuration assumption:** Throughout this chapter and the subsequent chapters in the book, we assume the installation and configuration of IBM Content Collector using Initial Configuration has already been done. For more information about these steps, refer to the product information center.

In this chapter we discuss the following topics:

- ▶ Dynamically calculating document retention
- ▶ Adjusting collectors
- ▶ Optimizing task routes for maintainability
- ▶ Promoting task routes from development to production systems
- ▶ Using the Expression Editor
- ▶ Extending IBM Content Collector

## 4.1 Dynamically calculating document retention

Assigning a static retention to all documents archived does not account for the different value that content has to the enterprise. To maintain legal conformance, static retention has to be chosen based on the least common denominator, leading to a huge portion of the documents being over-retained. Over-retaining, however, might cause legal exposure too. All documents, even though deletion might have been allowed according to law prior to a lawsuit, need to be provided during a trial. Besides that, eliminating over-retaining decreases storage costs because documents are not stored longer than necessary based on their value.

There are different ways to control document retention in an archiving solution. IBM Content Collector itself provides a mechanism to set an expiration date during archival of a document. In contrast to records management, this requires the disposal date of the document to be fixed during archival. If the disposal date for a document is not fixed, use IBM enterprise records for declaring a record during archiving.

### 4.1.1 Automatically calculating document retention

IBM Content Collector provides the calculate expiration task to perform different kind of dynamic retention date calculations based on document metadata. For complex scenarios that are based on the content of a document, use the IBM Content Classification integration. Decision plan or knowledge base results are made available as metadata in a task route, so they can be used for expiration date calculation.

#### Setting retention based on LDAP groups

One scenario that the calculate expiration task can support is the assignment of retention based on group membership. To determine the document retention, the task performs multiple lookups in LDAP to determine whether users contained in a given set of metadata are members of the groups that have a specific retention period assigned. If users are members of multiple groups, the highest retention period is returned and the expiration date is calculated.

Assume the company in our scenario wants to set the retention for email based on group membership of the recipients. Per default, all email will have a retention period of 3 years. The following extended retention times are to be used:

- ▶ Email with recipients from the Human Resources department will have a 7-year retention period assigned.
- ▶ Email with recipients from the Finance department will have a 10-year retention period assigned.



The copy of email present in mailboxes does not contain complete information regarding to which recipients an email has been delivered (see “Role of journaling for compliance” on page 50). The appropriate task route to modify is the compliance archiving task route that archived email from the journal.

Based on the task route deployed in “Deploying task routes for email compliance archiving” on page 51, perform the following modifications:

1. Select **Tools** → **Task Route Service Configuration** and change to the LDAP tab.
2. Fill in the necessary details and test the LDAP settings. For details about how to configure this panel, see the information center:  
[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/task\\_route\\_service/t\\_afu\\_troute\\_service\\_ldap.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/task_route_service/t_afu_troute_service_ldap.htm)
3. Select the **Set Retention to 3 Years** task of the task route and change the name to Set Retention based on LDAP groups (see Figure 4-1 on page 80 for final configuration).
4. Choose **Date set by metadata matching**.
  - a. Click the Expression Editor for setting the document data to match.
  - b. Create an expression that uses the Email Recipients Addresses metadata.

**Group membership consideration:** The **Recipients Addresses** field contains all recipients of a journal email, including the members of groups. Because the task will perform a lookup for group memberships during archival, a person needs to be member of a group at the point when the document is archived (in contrast to being member of a group when the email is received by the email server).

- c. Set the **Email Received Date** as the metadata that contains the base date.
- d. Enter the retention periods for the various groups as shown in Table 4-1.

*Table 4-1 Retention periods used in the Calculate Expiration task*

Match value	Retention period in days
human resources	2555 days (7 years)
finance	3650 days (10 years)
users	1095 days (3 years)

5. Save the task route.

Calculate Expiration Date

General

Name:  
Set Retention to 3 Years

Description:  
This task sets the retention to three years. All email documents that were archived more than three years ago can be deleted from the archive using the Content Collector Expiration Manager.

Configuration

☒ Date set by metadata matching  
Document data to match: [i](#)

<Email, Recipients Addresses (multi)>

Metadata that contains the base date: [i](#)

Source:  Property:

Retention periods: [i](#)

Match Value	Retention Period (days)
human resources	2555
finance	3560
users	1095

Figure 4-1 Calculate expiration task with three different retention periods

6. The task route is now set up correctly and can be promoted to production after verification.

### Setting retention based on user-defined metadata

Regulations for different countries might require retention to be assigned based on the country an email was received in. Deploying an archive solution per country might be undesirable due to increased costs for hardware and maintenance. In a scenario where journal email needs to be captured from Microsoft Exchange servers in a geographically dispersed setup, using the SMTP connector is the appropriate option. In such a setup, email servers are configured to forward journal email to a central IBM Content Collector SMTP receiver service. Because email is actively forwarded by email servers, network latency has only a minor impact on performance in this scenario.

Information about the sending email server is part of the received email, so user-defined metadata can be used to extract that information and use it for deciding the retention period of an email.

Looking at the Received header of an email, we identify the received header to contain the information about the journal that was used for this email:

```
Received: from sue78fk.prod.ibm.com ([29.2.16.212])  
by iccNode1 (JAMES SMTP Server 2.3.2) with SMTP ID 148  
for <USJournal@sue7rk.prod.ibm.com>;  
Wed, 1 Aug 2012 13:01:27 -0400 (EDT)
```

Based on the for portion of the received header, a decision can be made regarding the retention period to be assigned. Similar, this information can be used to store the email in different object stores.

In our scenario, the company has already deployed SMTP-based journal archiving to its US infrastructure. For email received in the US, a 3-year retention is used. The company now wants to roll out compliance archiving to the UK infrastructure. However, legal regulations in the UK require a 5-year retention to be assigned.

To implement this scenario, the company extends the SMTP configuration done for the US infrastructure by using the following steps:

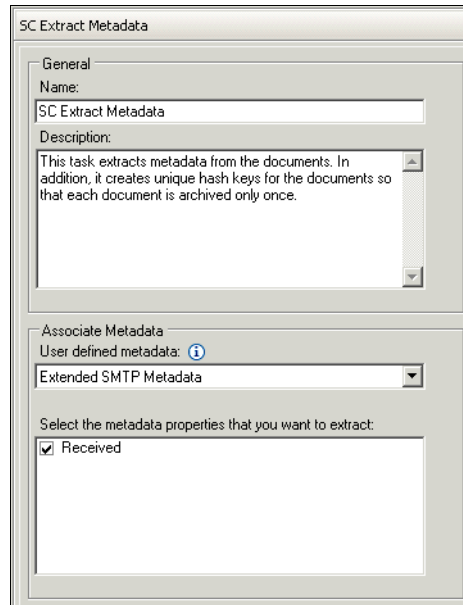
1. Configure the UK email server to forward journal email to the IBM Content Collector server, similar to the US email server configuration.
2. Select **Metadata and Lists**.
3. Create a User-defined Metadata named Extended SMTP Metadata. Add a property as listed in Table 4-2.

*Table 4-2 User-defined metadata property for received header extraction*

Property ID	Display name	Data type
afuSmtplibJournalEnvelope-Received	Received	String multi-value

**Header extraction:** Using the prefix afuSmtplibJournalEnvelope- indicates to the SMTP connector that the header is to be extracted from the envelope message, but not from the actual journal message.

4. Open the task route deployed in “Using SMTP for compliance archiving” on page 50.
5. Select the **SC Extract Metadata** task.
  - a. Configure the task to extract the Extended SMTP Metadata.
  - b. Check the **Received** property to be extracted (Figure 4-2).



The screenshot shows the 'SC Extract Metadata' configuration window. It has two main sections: 'General' and 'Associate Metadata'. In the 'General' section, the 'Name' field is set to 'SC Extract Metadata' and the 'Description' field contains the text: 'This task extracts metadata from the documents. In addition, it creates unique hash keys for the documents so that each document is archived only once.' In the 'Associate Metadata' section, the 'User defined metadata' dropdown is set to 'Extended SMTP Metadata'. Below this, there is a checkbox labeled 'Received' which is checked.

*Figure 4-2 SC Extract Metadata task configured to extract received header*

6. Select the **Set Retention to 3 years** task.
  - a. Change the name to Set Retention based on Geography.
  - b. Choose the date set by expression option and launch the expression editor.
  - c. Configure a expression as shown in Figure 4-3 on page 83 to calculate a 3-year retention for email received from the US journal and 5-year retention for email received from the UK journal.

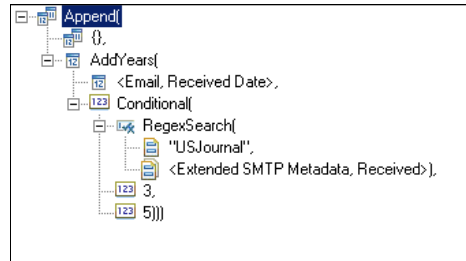


Figure 4-3 Expression to calculate retention based on received header

#### 7. Save the task route.

The task route is now set up correctly and can be promoted to production after verification.

### 4.1.2 Manually setting document retention

When automatic calculation of retention is not desired, users can manually set the retention period for a document. Because retention is calculated from metadata, all metadata that can be influenced by a user can be used:

#### ► Email metadata

Users are changing metadata of the email by moving it to a different folder, using custom Lotus Notes agents, or assigning an email category.

**Custom metadata extraction properties:** Configure custom metadata extraction to use properties set by custom Lotus Notes agents or the email category for expiration date calculation.

#### ► Additional metadata

Users are explicitly specifying the retention category similar as shown in “Interactively driven by folders with additional metadata” on page 69.

### Impact of deduplication on manual retention

Interactive declaration of a retention period always involves user action. The user will assign a retention to the copy of the email archived from the mailbox, but not to the copy archived from the journal.

However, because deduplication happens across mailbox and journal copies, the highest assigned retention will be set in the repository. For more information about this topic, visit the following site:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/deduplication/c\\_afu\\_deduplication.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/deduplication/c_afu_deduplication.htm)

If the mailbox copy and the journal copy are not deduplicated, two separate objects will exist in the repository and the journal object might be purged from the repository earlier than the mailbox object. Thus, journal information might not be available for discovery. Typical reasons why a mailbox copy of an email does not deduplicate with the journal copy are listed here:

- ▶ BCC copies will not be deduplicated with the journal copy or each other.
- ▶ Lotus Notes MIME email will not be deduplicated with the Rich Text Format version of the same email. You may configure the Domino server to convert email to a specific format upon receiving.
- ▶ The Microsoft Exchange sender copy of email will not be deduplicated if it contains bcc or tracking information.

### **Setting retention for email by folder structure**

Moving email to specific folders is a task that users perform on a daily basis. The folder structure can be leveraged for calculating document expiration. To allow users to effectively assign retention, a predefined file plan can be pushed to user mailboxes by IBM Content Collector. With each run of a collector, the necessary folders are created.

In our scenario, the company realizes that further cost savings can be realized by having users assign retention categories by dragging email to folders. From a legal perspective, storing business-relevant email for 3 years is required and will be the default retention if no other category applies. The company identified the following categories:

- ▶ **Customer-related email**

This refers to email that has been received from customers or is sent to customers. It has a high value for the business and should be kept for 5 years.

- ▶ **Confidential email**

This refers to email containing sensitive information that should not be disclosed to people who do not have a need to know. Because of the intimate nature, these emails should be kept as reference for 10 years.

► Personal email

This refers to email that is not relevant to the business. Email of this nature should only be kept for 1 year.

To implement this scenario, we extend the storage management task route deployed in “Deploying email task routes for storage management” on page 40, by using the following steps:

1. Select **Metadata and Lists**.
2. Click **User-defined metadata** and create a new list.
3. Assign the name Email Retention Classes to the list.
4. Add the values listed in Table 4-3 to the list (see Figure 4-4).

Table 4-3 Email retention classes list

Name	Value
/archive/customers	5
/archive/confidential	10
/archive/personal	1

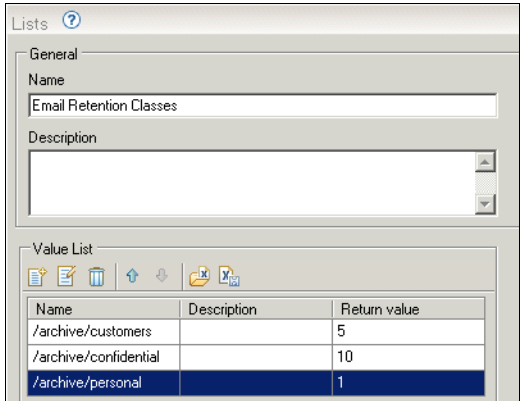


Figure 4-4 Email retention classes in Configuration Manager

5. Add a new **EC Collect Email by User Selection** collector to the task route.
  - a. Add the suffix US to the collector name.
  - b. Choose the **Collect from Folders** option.

6. Select the **Schedule** tab.

Set the schedule to run daily, with running endlessly, starting today, running each day at 9:00 p.m., stopping at the latest at 5:00 a.m. Running the collection once a day will be sufficient.

7. Select the **Collection Sources** tab (Figure 4-5).

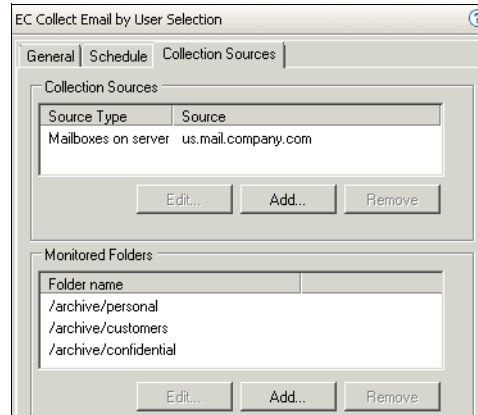


Figure 4-5 Collection Sources configuration for folder monitoring

- a. Add the us email server as a collection source.
  - b. Add the content of Table 4-3 on page 85 as monitored folders, all subfolders will also be monitored for email.
8. Select the **EC Collect All Email - US** collector.
- a. Select the **Filter** tab.
  - b. Add /archive to the list of excluded folders for this collector (Figure 4-6).

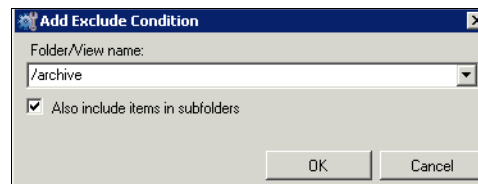


Figure 4-6 Excluding the /archive folder

9. Repeat the steps for the DE collector.
10. Add a Calculate expiration date task after the EC Extract Metadata task.
- a. Select the Calculate expiration date task.



- b. Configure the expiration date to be calculated based on the folder that the email is located in. Use a dynamic list lookup to retrieve the value from the previously created email retention classes list. For email not located in a folder that is part of the list, use a default retention of 3 years (Figure 4-7).

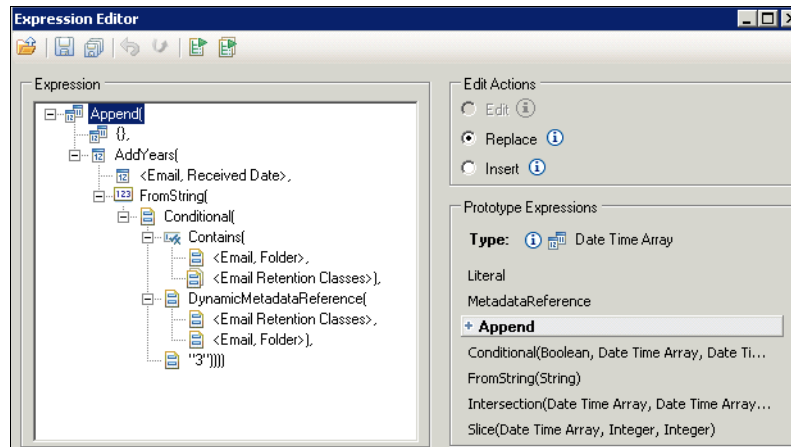


Figure 4-7 Expression to assign retention based on folder

11. Save the task route.

The task route is now set up correctly and can be promoted to production after verification (Figure 4-8).

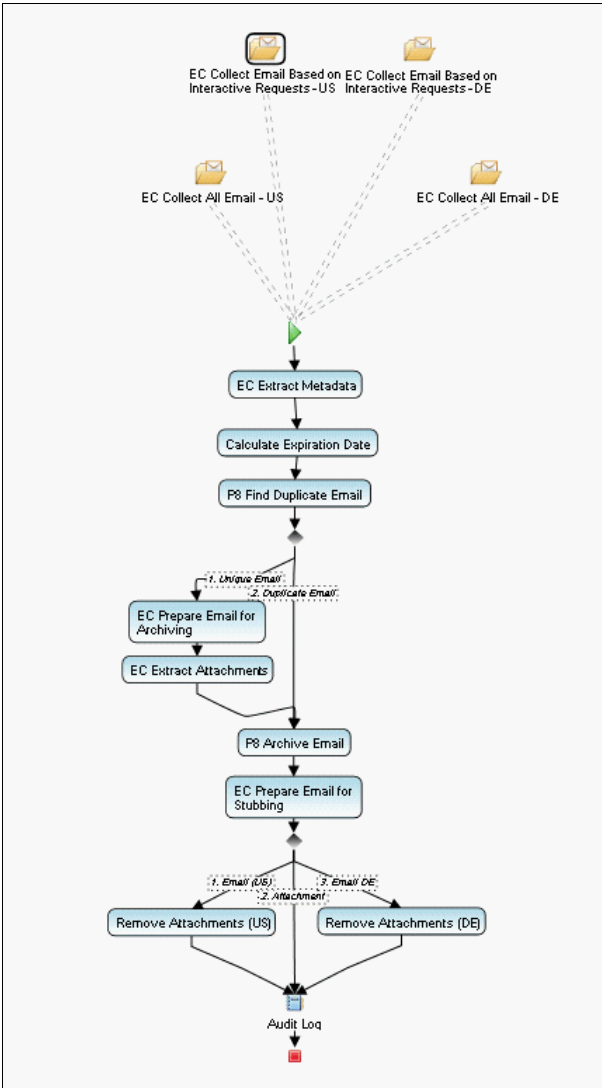


Figure 4-8 Task route to assign retention for specific folders

## 4.2 Adjusting collectors

Collectors crawl source systems for eligible documents to be processed. Because of this, their configuration has significant influence on the impact that IBM Content Collector has on source system performance. Typical changes to collectors include:

- ▶ Adjusting collection sources
- ▶ Configuring schedules
- ▶ Adjusting filter options

In this section we discuss configuring schedules. Collection sources and filter options are highly dependent on the type of connector used, thus it is advisable to refer to the Information Center for details.

### 4.2.1 Configuring schedules

Schedules have a significant impact on how your IBM Content Collector system performs. They also impact the overhead put on the source system that is monitored. So it is important to use balanced schedules that are reasonable for the specific task they are to perform.

A typical example is the lifecycle collector of the email connector. As explained in Chapter 5, “Retention management” on page 119, the lifecycle collector is configured with a table of stubbing operations that should occur after a specific time. The schedule assigned to the lifecycle collector should be aligned with the criteria used in the lifecycle.

Given the data listed in Table 4-4, running the collector once a week should be sufficient. Running the collector more frequently will yield only a small number of documents to be processed, while still putting the overhead of a mailbox search on the source system.

*Table 4-4 Example configuration for an email lifecycle*

After	Stubbing operation
30 days	Remove attachments
60 days	Remove attachments and body
90 days	Delete entire email
7 days	Select documents to re-create stubs

However running the collector less frequently will affect the effective dates for the lifecycle (Table 4-5).

**Frequency considerations:** Assume an email was restored 7 days ago, exactly at the moment when the collector finishes a run. The collector will finally pick up the email during next collection. So the effective criteria is the sum of the criteria plus the schedule interval, which is also 7 days in our example. That means that in a worst-case scenario, a document might be restubbed 14 days after it has been restored.

*Table 4-5 Example configuration with column for worst-case dates*

After	Worst-case	Stubbing operation
60 days	67 days	Remove body
90 days	97 days	Delete email
7 days	14 days	Restub restored email

In this case, the deviation is accepted. If the deviation is deemed too high, the collector can be run more frequently, but at the cost of additional mailbox searches.

## Planning for balanced collector schedules

IBM Content Collector systems run most effectively when the schedules are balanced across all collectors used. Each task route might have different requirements with regard to scheduling. It is generally advisable to distribute collector runs throughout the day evenly, if possible. Starting all collectors at the same time might create resource conflicts and lead to suboptimal performance.

A typical example is a storage management and compliance scenario for email that incorporates three task routes:

- Journal archiving

This task route archives and deletes email from one or more journals. Because of the high volume, the task route needs to run with a high frequency.

- Storage management

This task route archives and stubs email from a high number of mailboxes. With each run, significant storage is freed up on the email servers. The volume is moderate and so is the run frequency.

► Email lifecycle

This task route stubs the aging email incrementally until it is completely removed from the system. With each run, some storage is freed up on the email servers. The volume is moderate and so is the run frequency.

Based on the volume of email that is to be processed on a day forward base, the journal task route will be scheduled to collect email every 15 minutes for the whole day, unless there are scheduled maintenance windows. Taking into account the archiving criteria of email older than 30 days, the storage management task route will be scheduled to run once a day, in the off-hours. The email lifecycle task route will be run even less frequently because it will be most efficient when a high number of email are to be processed. Thus, it will only run once a week on the weekend.

In some rare cases, the always schedule may be used. Examples for such a case would be interactive archiving for email or archiving from IBM Connections. for more information about the always schedule, see:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/collectors/r\\_afu\\_collector\\_schedule.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/collectors/r_afu_collector_schedule.htm)

**Restart note:** IBM Content Collector does not preserve the time when a collector was run after a restart. Thus, a collector might appear to run unexpectedly after a restart.

For more information, refer to the following Technote:

<http://www.ibm.com/support/docview.wss?uid=swg21430225>

## 4.2.2 Collector and task route filtering

There are two types of filtering in IBM Content Collector. The first is filtering done in the collector. Filtering items at this stage avoids submitting metadata and data about items into the task route. The second type of filtering can be done during the remainder of the task route, using decision points and rules to route items to particular branches or to stop processing an item entirely.

### Collector-based filtering

Collector-based filtering provides for the most effective filtering and should always be preferred. Ideally when using collector-based filtering the criteria can be pushed down to the source system, where it is handled most effectively. For example, in the case of Lotus Domino, the collector builds a formula that is used for running a native Domino search against the Domino database. This is far

more efficient than using task route-based filtering. However, collector-based filtering depends on the capabilities of the source system and the connector.

If it is not possible to use collector-based filtering for determining whether an item should be processed, task route-based filtering must be used.

## Task route-based filtering

Task route-based filtering uses decision points that are built using expressions. Such filtering is more powerful than collector-based filtering.

When using decision points, it is important to consider the evaluation order of rules. Depending on the order of evaluation, the result might be different.

Furthermore, it is inadvisable to use the `always true` option in a rule. It is more obvious to put the criteria into a expression; for example, if you have two branches and the left branch is checking whether a document is an attachment (Figure 4-9).

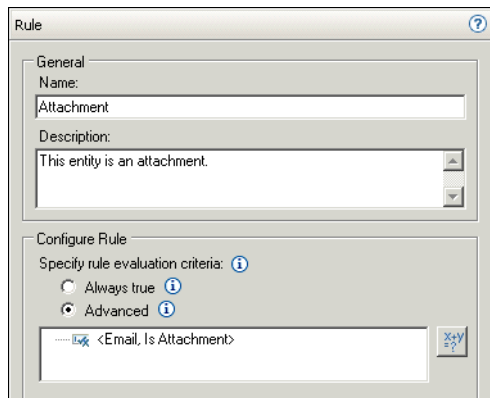


Figure 4-9 Rule to check whether a document is an attachment

A second branch should check whether the remainder of the solution storage is, for example, an email (Figure 4-10). This is to ensure that there is no situation where no rules match. If no rules match, a warning is logged.

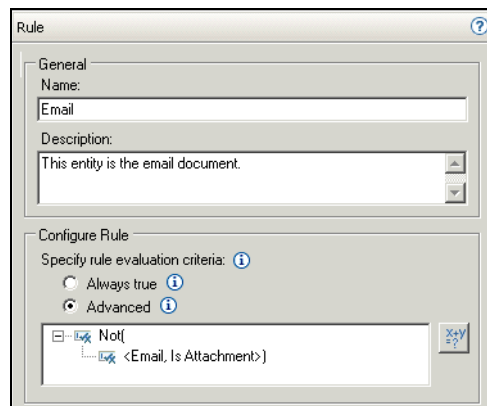


Figure 4-10 Rule to check whether a document is an email

## 4.3 Optimizing task routes for maintainability

When deploying different scenarios using IBM Content Collector, the question arises as to how many task routes collectors are to use to implement the scenario. The design and configuration is quite flexible, namely one or more connections; one or more user-defined metadata definitions; one or more task routes; one or more collectors per task route; and one or more collection sources per collector are possible.

The general advice is that instead of creating a new task route per variation of a scenario, keep the number of task routes and collectors minimal. This helps clarify the configuration and ease maintenance tasks. Additionally, it helps to minimize IBM Content Collector resource consumption.

Throughout this section we inspect each point of flexibility and explain its purpose, strengths, and weaknesses, to help determine what is best for your situation.

### 4.3.1 Impact of using multiple connections for Microsoft SharePoint

The Microsoft SharePoint connector configuration can have multiple connections. There are a few reasons for having multiple connections:

- Distinct Microsoft SharePoint farm implementations

- ▶ Different requirements (filtering, column mapping, post-processing) in areas (web applications, site collections, sites, or lists)
- ▶ Different credentials required to access different areas

Nothing prevents configuration of connections to areas that have complete or partial overlap, such as duplicate URLs or a top-level site for one connection and subsite for another connection. In these cases there are three strategies to prevent collection and archival duplication, in order of preference:

- ▶ Configure the collection source filtering options to be mutually exclusive. For example, select Library A for one collection source, and Library B for another.
- ▶ Use decision points and rule criteria to filter out undesired overlap. This strategy is easiest to configure when using separate task routes.
- ▶ Configure Microsoft SharePoint permissions for different credentials to be mutually exclusive, and use those different credentials in the connection configuration. This strategy applies only to broad collection scenarios such as farm or web application levels.

#### **Connection considerations:**

- ▶ Avoid creating a connection for every use case, solution, or set of requirements.
- ▶ Using peer connections is preferable to using connections that will have overlap depending on collection source configuration. *Peer connections* are connections on the same level of depth, for example, connections to two sites that are siblings.
- ▶ Give each connection a display name that conveys its intended scope and, when appropriate, include the port number of the Microsoft SharePoint web application enclosed in parentheses ( ) for ease of use.
- ▶ Broad collection scenarios should use library or list type filtering rather than specific library or list selection, except when there is an enterprise standard for library or list creation and naming.
- ▶ Content type filtering is helpful in broad and narrow scenarios. Broad scenarios would normally be using either built-in content types or the content type hub in Microsoft SharePoint 2010, thus ensuring that the same content types exist across the broad area.
- ▶ The folder filtering capability is expected to be used in narrow collection scenarios. However, it can apply in a broad scenario if there is a standard folder path that all libraries had for collectible content.



### 4.3.2 Microsoft SharePoint version series task route design

When processing version series documents, such as documents collected through the Microsoft SharePoint collector, additional considerations need to be taken when implementing decision points. Not all tasks deal with an entire version series, but with each individual version. Tasks that support handling an entire version series are listed here:

- ▶ CM 8.x Store Version Series
- ▶ P8 Create Version Series
- ▶ P8 Declare Record
- ▶ SP Post-processing
- ▶ SP Create File
- ▶ SP Manage Link

Appropriate version routing can be seen in the Microsoft SharePoint task route templates. For example, with the “SP to P8 - With Versions” task route template, additional handling is provided to deal with the P8 File Document in Folder task. Because only one filing reference is expected, execution of this task is required for only one version of the document version series being processed.

If every version was routed for processing within the P8 File Document in Folder task, multiple filing references would result. It would appear in a FileNet P8 client as though a copy of the document was created for every version.

**Decision point note:** Content Manager task route processing does not require a decision point because the CM file in folder functionality is provided directly within the CM 8.x Store Version Series task.

Similarly, the “SP to P8 - Declare as Record” task route template is configured to have only the last version of the document version series declared as a record (Figure 4-12), in addition to the single version being filed.

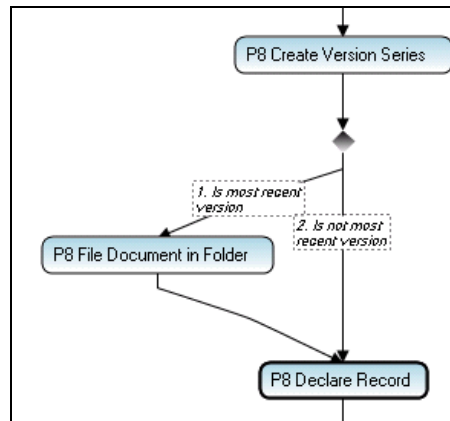


Figure 4-11 Task route to declare all versions

If the entire version series is to be declared as a record, then the task route would need to be modified as shown in Figure 4-11.

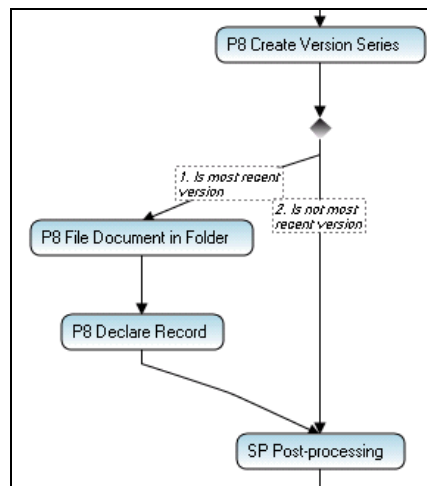


Figure 4-12 Task route to declare last version

## Microsoft SharePoint version series and decision points

Additional considerations are required with version series processing and decision point placement.

The scenario is that storage space is an issue in a Microsoft SharePoint implementation and the business has decided that all documents that have not been modified for 90 days can be archived and replaced with a link.

To accomplish this use case, a decision point must be added to a task route created from the “SP to P8 - With Versions” task route template. A single rule must be configured to evaluate each Microsoft SharePoint document's Last Modified Date. Only if the date value is greater than 90 days in the past will the document continue through the task route and be processed. The placement of the decision point is important.

The decision point must be placed prior to the “SP Get Versions” task (Figure 4-13). The “SP Collector” task only provides information about the most current version, and that information can and should be used to make decisions on behalf of all versions of a document.

The “SP Get Versions” task provides information about all versions of a document to the task route. If the decision point is placed after the “SP Get Versions” task, then the rules are evaluated for each version.

In this scenario it is possible that some versions would meet the rule but others would not, thereby sending only a subset of the versions to the remainder of the task route. This would have undesired results for archiving by not capturing all versions. As well, when the “SP Post-processing” task replaces the entire Microsoft SharePoint document with a link, the uncaptured versions would be lost.

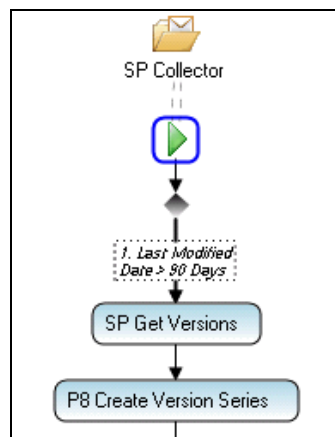


Figure 4-13 Correct usage of age criteria for document selection in Microsoft SharePoint

### 4.3.3 Creation of user-defined metadata

User-defined metadata definitions are necessary when your scenario requires the use of Microsoft SharePoint columns or associating metadata from metadata files in the task route, typically for target repository property mapping or for rule criteria. There is no limit to the number of user-defined metadata definitions. Preferred practice however, is to create as few as necessary to fulfill your requirements. This means using one or a few definitions to encompass all the Microsoft SharePoint columns or file system metadata you require, potentially from many different use cases and solutions. There are several reasons for this practice:

- ▶ It requires less configuration effort.
- ▶ It is less prone to configuration errors.
- ▶ Each Microsoft SharePoint collector task allows only one metadata definition for output.

Thus, the ability to map the same but differently named Microsoft SharePoint columns (for example, Name, Last name, Surname) or file system metadata into a single user-defined metadata definition property can help to simplify configuration. This is preferable to using complex expressions downstream in the task route to determine which one of several properties to use.

### 4.3.4 Impact of using multiple collection sources

The preferred option is to add additional collection sources to already existing collectors. This way the overhead introduced by the additional scenario is minimal and complexity is not increased.

The performance characteristics of adding another collection source to a collector depends on the type of connector. All connectors except the email connector process collections sources in the order you add them. The email connector uses a two-staged approach to allow for optimal efficiency.

First, all collection sources are resolved into individual mailboxes (for example, all members of a group are resolved). Then resulting list of mailboxes is shuffled and passed to a set of crawler threads. The crawler threads are in charge of searching the mailboxes for eligible email.

This set of crawler threads is globally used by the connector, and allows for controlling the load that the email connector will put on the email system. The size of the crawler pool can be set in the connector settings.

For more information about this topic, refer to the following sites:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/connectors/r\\_afu\\_ex\\_conn\\_settings.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/connectors/r_afu_ex_conn_settings.htm)

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/connectors/r\\_afu\\_ln\\_conn\\_settings.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/connectors/r_afu_ln_conn_settings.htm)

Reasons that might prevent the use of different collection sources include the following examples:

- ▶ Different schedule requirements - such as, archiving should occur during off-hours in each time zone.  
  
In this case, creating multiple collectors attached to the same task route is the best choice)
- ▶ Different filter requirements - such as, different user groups have different criterias for archiving.  
  
For an illustration, see “Deploying file system task routes for storage management” on page 33.
- ▶ Different output requirements (Microsoft SharePoint/File system)  
  
Perhaps you are required to use multiple user-defined metadata definitions because a Microsoft SharePoint or File system collector only allows one user-defined metadata to be output.
- ▶ Different processing requirements  
  
Perhaps you want to generate a hashkey or collect file content information for some files but not for others.

### 4.3.5 Impact of using multiple collectors

If adding additional collection sources to collectors is not sufficient, adding additional collectors to existing task routes might be a viable option. Adding a additional collector will increase the work that needs to be done by IBM Content Collector. For all connectors except the email connector (see 4.3.4, “Impact of using multiple collection sources” on page 98 for an explanation), adding additional collectors might result in improved performance. However, this will only be the case when collection throughput was the limiting factor and the source system can handle the load caused by the additional collector. This will only be the case in some rare scenarios.

Reasons that might prevent the use of multiple collectors attached to the same task route include the following examples:

- ▶ Different mappings for metadata  
You have metadata files with varying formats and it is not possible to create a single metadata source that presents a union of all of the information that is to be captured from the different metadata files.
- ▶ Different order or sequence of tasks  
Depending on the group a mailbox is assigned to during archiving, IBM Content Classification should either be invoked or not be invoked.

### 4.3.6 Impact of using multiple task routes

Each additional task route increases the complexity of configuration and makes maintenance more time consuming. Besides that an additional collector is created, which effectively puts more load on the system and creates more overhead. For each collector a separate thread will be used to run collection. If you are deploying hundreds of task routes and schedule the collectors to run at the same time, the system will be overloaded and it will lead to degraded performance. Thus, it is advisable to spread the run time of all collectors throughout the available time window.

However, in most cases the tasks and configurations for variations of the same scenario are the same or can be unified, thereby allowing for a lower number of task routes.

### 4.3.7 Strategies for minimizing the number of task routes

The major strategy for keeping the number of task routes low is unification. This allows for combining multiple scenarios into one task route. However, this results in additional complexity introduced by additional rules or expressions. Some tasks provide options to make unification easier. Here are a few examples of such tasks:

- ▶ The P8 Connector Create Document, Create Version Series, Declare Record tasks allow for dynamic document class selection.
- ▶ The File System Connector Metadata Collector and Associate Metadata tasks can be configured with xpaths that pull values from disparate file formats and populate a single metadata source that presents a union of the information that is available from the different files.

Despite the additional complexity, it is advisable to use unification and adaptive features as much as possible to keep the number of task routes low. A separate task route is appropriate in the following cases:

- ▶ If decision points would be used and the majority of the tasks would be duplicated and put into separate branches, depending on the kind of scenario.
- ▶ If scenarios vary based on the source of user-defined metadata property values (Microsoft SharePoint sites or File system metadata files), and you cannot build a user-defined metadata source that contains the union of all properties

## Using decision points for using multiple object stores

In this example we highlight how decision points can be used to implement two scenarios with one task route.

In this case, a company wants to deploy storage management to their users in Germany (DE) and in the United States (US). During archival, email will be stubbed and different text fragments will be inserted. These text fragments will be in the English language for US mailboxes and in the German language for DE mailboxes. Instead of deploying two separate task routes, the company uses decision points to allow a single task route to implement the requirement. This leads to easier maintenance.

Similar to the task routes deployed in “Deploying email task routes for storage management” on page 40, in this case we make the following modifications:

1. Create one collector for each country and the respective mail servers.
2. Copy the remove attachments stubbing task.
3. Attach another rule to the decision point.
4. Link this rule to the copy of the remove attachments stubbing task.
5. Change the rule in the branch for the first remove attachments stubbing task to be only true for email that has been collected by the US collector (Figure 4-14).

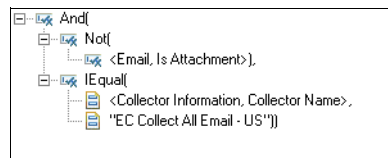


Figure 4-14 Decision for matching email collected by the US collector

6. Change the rule in the branch for the second remove attachments stubbing task be only true for email that has been collected by the DE collector.
7. Adjust the stubbing text of both tasks to match the desired wording and language.

The task route is now set up correctly and can be promoted to production after verification (Figure 4-15).

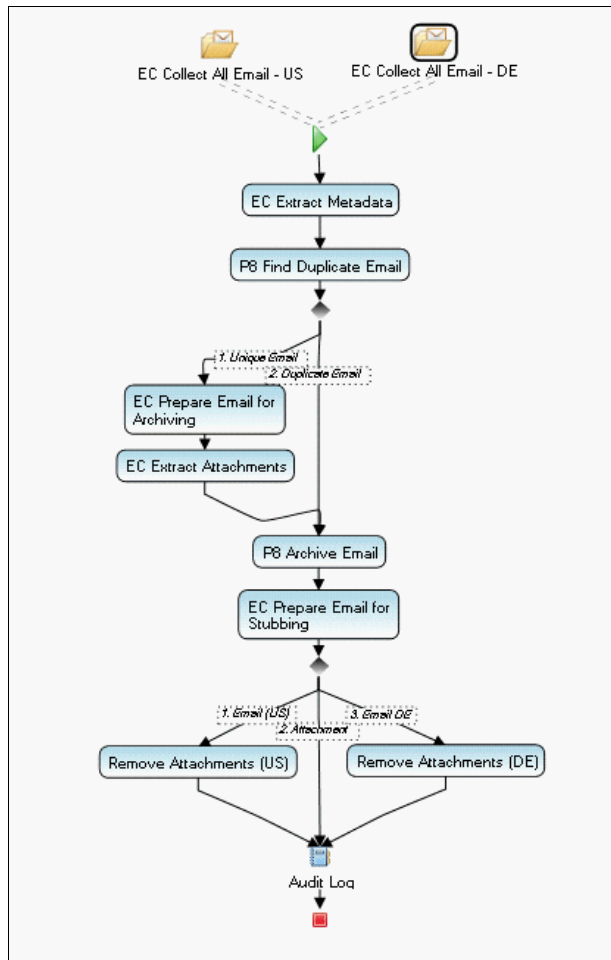


Figure 4-15 Task route illustrating the use of different stubbing depending on the collector that collected a document



## Metadata unification and dynamic document class selection

In this example we highlight how metadata unification and dynamic P8 document class selection can be used to handle two different, but similar, kinds of files with one task route. For more information about this topic, refer to the following site:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/property\\_values/t\\_afu\\_dynamic\\_classes\\_or\\_values.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/property_values/t_afu_dynamic_classes_or_values.htm)

Based on the scenario deployed in “Adding recovery functions using error task routes” on page 62, the company wants to ingest another kind of report generated by a similar reporting application.

Comparing the two metadata files to be processed, it is determined that they are quite similar. However, when comparing Figure 4-16 with Figure 4-17, a few differences are visible. Figure 4-16 displays the information produced by the first application.

```
<reportexecution xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <report product="981" file="rp_23984.pdf" />
  <report product="951" file="rp_22984.pdf" />
</reportexecution>
```

Figure 4-16 XML fragment depicting the information produced by the first application

Figure 4-17 displays the information produced by the new application. Note the following points:

- ▶ Additional attributes *StartDate*, *EndDate*, and *Geography* are introduced.
- ▶ The product id is contained in properties with different names (*product* and *productID*).

```
<reportexecution xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <report productID="856" startDate="01-01-2012" endDate="10-08-2012" geography="EMEA" file="rp_23584.pdf" />
  <report productID="881" startDate="01-01-2012" endDate="10-08-2012" geography="EMEA" file="rp_13974.pdf" />
</reportexecution>
```

Figure 4-17 XML fragment depicting the information produced by the new application

To archive these new documents to the repository, a new document class *ReportByGeo* inheriting from the *Report* document class has been created. The *ReportByGeo* document class has the following additional properties:

- ▶ *StartDate* of type *DateTime*
- ▶ *EndDate* of type *DateTime*
- ▶ *Geography* of type *String*

Based on the task route already deployed to handle this first kind of report, we make the following modifications:

1. Add new properties to the *Report information* user-defined metadata:
  - a. *StartDate* of type DateTime
  - b. *EndDate* of type DateTime
  - c. *Geography* of type String
2. Switch to the task route.
3. Select the **FS Associate Metadata** task.
4. Switch to the **Metadata mapping** tab.
  - a. Map *StartDate*, *EndDate*, and *Geography* as shown in Figure 4-18.

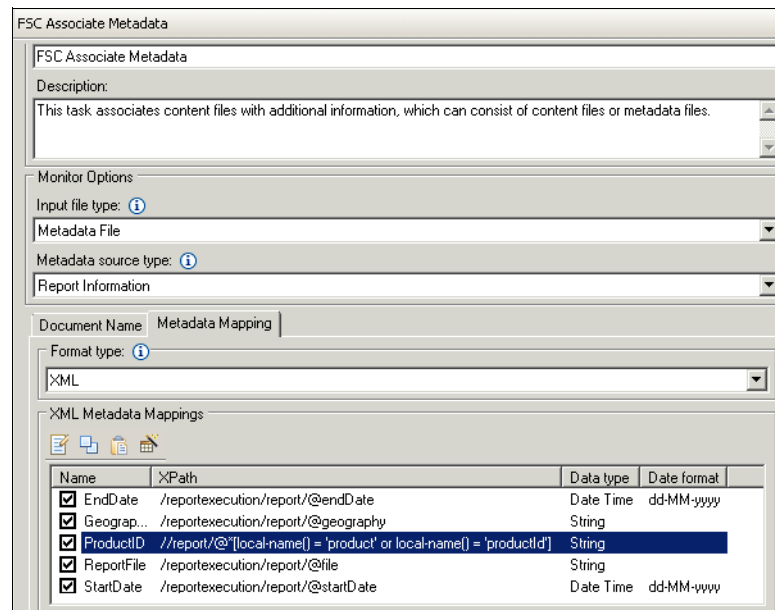


Figure 4-18 Metadata mappings for mapping information for both types of reports

- b. Change the mapping for the ProductID parameter to select both the *product* and the *productID* attribute. Use the expression `//report/@*[local-name() = 'product' or local-name() = 'productId']` as shown in Figure 4-19 on page 105.

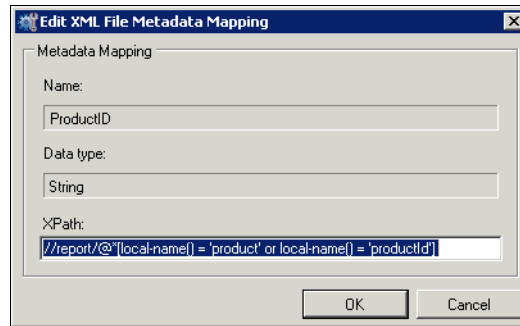


Figure 4-19 Xpath expression to select both attributes

With these changes we allow for both XML metadata files to be extracted into a common user-defined metadata.

Next, we use this metadata to choose the P8 document class dynamically:

1. Select the **P8 create document** task.
2. Press the **Advanced** button in Property mappings.
  - a. Check the **use an expression to determine the class** option.
  - b. Launch the **Expression Editor**.
  - c. Configure the expression to return Report if the geography attribute is not set, and to return ReportByGeo otherwise (Figure 4-20).

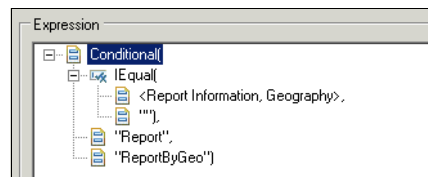


Figure 4-20 Expression to choose the document class depending on the existence of the geography property

- d. Configure Advanced Property Mappings to map all report information properties (Figure 4-21 on page 106).

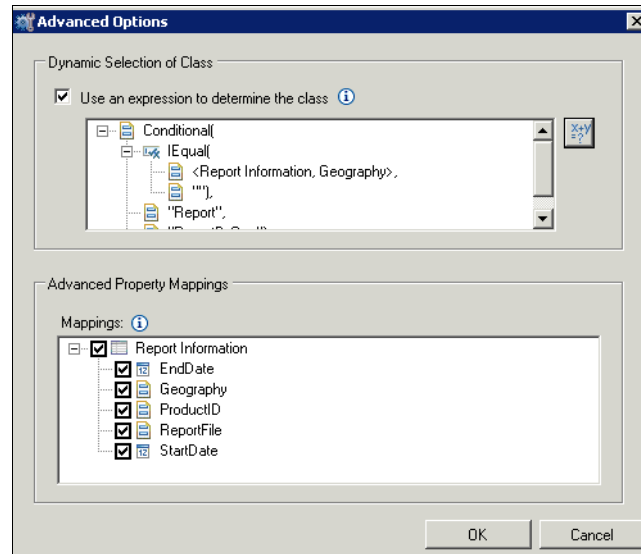


Figure 4-21 Final configuration for Advanced Options

3. Save the task route.

The task route is now set up correctly and can be promoted to production after verification.

## 4.4 Promoting task routes from development to production systems

A typical setup for IBM Content Collector includes a development system that is used for planning, designing, and testing task routes or other modifications of the system. After a task route is designed and tested, it can be promoted to the production system.

Because task routes can have various direct and indirect dependences, in this section we provide a step-by-step guide for moving task routes. Further, we explain the differences between the types of dependencies and discuss how to deal with them.

## 4.4.1 Understanding task route dependencies

IBM Content Collector provides for the export and import of task routes, including error task routes. This feature can be used to move task routes between different environments.

A task route can have two kinds of dependencies:

- ▶ **Internal**

These are dependencies referring to internal resources such as user-defined metadata, archived data access settings, or connections to specific systems. Use IBM Content Collector to migrate dependencies of this kind.

- ▶ **External**

These are dependencies referring to external resources, such as a document class, IBM Content Classification knowledge base, or IBM Content Manager item type. Dependencies of this kind are created by specific connectors. Use the system-specific utilities such as Classification Workbench for Content Classification knowledge bases to migrate the necessary data.

You must resolve dependencies before you migrate the task route. Otherwise, importing the task route might fail or yield unresolvable dependencies, or the system might behave differently than intended.

## 4.4.2 Checklist for task route migration

Use the following checklist to ensure that you export the necessary configuration objects for your task route. Repeat the steps in the same order to import and duplicate the configuration objects.

### **Metadata and lists**

- ☐ Export/Import the content of lists.
- ☐ Record the name of the lists.
- ☐ Export/Import the definition of user-defined metadata.
- ☐ Record the display name of the user-defined metadata.

### **Repository connectors**

#### **P8 Connector**

- ☐ Ensure that you migrate customized document classes to production before importing a task route template that references these document classes.
- ☐ Record any modifications done to the MIME type mappings section of the connector.

- ☐ Record the connection details to match them with the target system.

### **Content Manager Connector**

- ☐ Ensure that you migrate customized item types to production before importing a task route template that references these item types.
- ☐ Use the Content Manager Administration client to carry over any customizations done to MIME types.
- ☐ Record the connection details to match them with the target system.

### **Task Connector**

### **IBM Content Classification Connector**

- ☐ Export the decision plan and knowledge base using the Classification Workbench.

### **Metadata Form Connector**

- ☐ Export/Import the form template and definition.

## **General settings**

### **Archived data access**

- ☐ Export/Import the configuration and mapping if you made modifications.
- ☐ Export/Import customized search and result fields<sup>1</sup>.

### **Client Configuration**

- ☐ Record the configuration done in this section.

### **Metadata Form Template**

- ☐ Export/Import the template if you modified it.

### **Metadata Form Definition**

- ☐ Record the settings in this section.

Use the information collected to manually recreate the necessary configuration objects in the production system.

---

<sup>1</sup> [http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/data\\_access/t\\_afu\\_customize\\_fields.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/data_access/t_afu_customize_fields.htm)

## 4.5 Using the Expression Editor

The Expression Editor is an essential part of IBM Content Collector. Expressions are used in various places to map metadata or to decide on processing flow in a rule.

An *expression* is a tree formed by operators that are evaluated to a final result. For a full list of all supported operators, see the information center:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/expression\\_editor/r\\_afu\\_expression\\_prototypes.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/expression_editor/r_afu_expression_prototypes.htm)

The list of operators include regular expression matching and replacement operators:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/expression\\_editor/c\\_afu\\_about\\_regex.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/expression_editor/c_afu_about_regex.htm)

Regular expressions can be simple but powerful. For example, assume you want to extract a contract number from an email subject. Further assume that a contract number consists of an uppercase letter, followed by a separator, followed by five letters. Then here is an example for a regular expression to find contract numbers in email subjects:

```
[A-Z]-\d{5}
```

Other examples for regular expression matching and replacement can be found in the information center:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/expression\\_editor/r\\_afu\\_regular\\_expression\\_samples.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/expression_editor/r_afu_regular_expression_samples.htm)

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/expression\\_editor/r\\_afu\\_regex\\_reference.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/expression_editor/r_afu_regex_reference.htm)

For more detailed information about regular expressions, refer to the following publications:

- ▶ *Mastering Regular Expressions*<sup>2</sup>
- ▶ *Regular Expressions Cookbook*<sup>3</sup>

---

<sup>2</sup> Mastering Regular Expressions, Jeffrey E.F. Friedl, 2006, third edition, 978-0596528126

<sup>3</sup> Regular Expressions Cookbook, Jan Goyvaerts & Steven Levithan, 2009, 978-0596520687

## 4.5.1 Avoiding the need for nested decision points

One or more rules are attached to a decision point. However, decision points cannot be nested, thus forcing you to collapse all decisions into a single rule within a single decision point. The following example illustrates how to rewrite rules to avoid nested decision points.

Assume that you want to process email based on size and attachment count:

- ▶ If the email size is above 1 MB, process the document.
- ▶ If the email has attachments, process the document.

Directly modelling these rules would require a nested decision point. However, the rules can be combined into one rule that can be contained within a single decision point:

- ▶ If the email size is above 1 MB or the email has attachments, process the document (Figure 4-22).

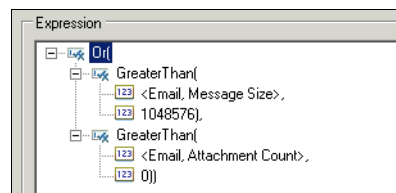


Figure 4-22 Combining two expressions into one

Similarly, any other set of rules can be combined into a single rule.

## 4.5.2 Using list lookups

In scenarios that require dynamic evaluation or matching of metadata against a list, list lookups are quite helpful. An example of such a scenario is the dynamic assignment of retention periods.

Assume we want to assign retention to files based on the folder in which they are located. Retention categories and folder names are listed in the following tables.

Although the retention categories list is quite small, the folder names list might contain multiple hundred entries to cover all cases and possible typos. By using two lists, the cost of maintaining the lists is reduced.

Figure 4-6 on page 86 lists the example retention classes.



Table 4-6 Example retention classes

Retention class	Description	Retention
Confidential	Confidential documents	10 years
Invoice	Used for bills or documents that have to be threaded as bills	7 years
Reference	Documents containing plans or other documents that act as reference	5 years
Default	All other documents	10 years

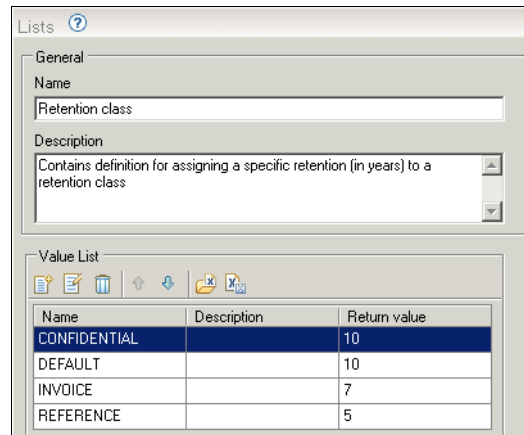
Table 4-7 lists the example folder mappings.

Table 4-7 Example folder mappings

Folder	Retention class
discussion	REFERENCE
contract	CONFIDENTIAL
...	...

This example is based on the task route provided in “Deploying file system task routes for storage management” on page 33. Complete these steps:

1. Create a list named `Retention class`. Fill in the values to map retention categories to a retention time span (Figure 4-23). You may import the values from a CSV file.

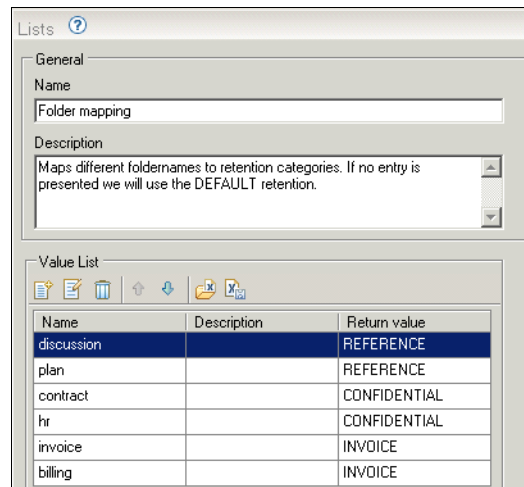


The screenshot shows the 'Lists' configuration window in the Configuration Manager. The 'General' tab is active, showing the list name 'Retention class' and a description: 'Contains definition for assigning a specific retention (in years) to a retention class'. Below this is the 'Value List' section, which contains a table with three columns: 'Name', 'Description', and 'Return value'.

Name	Description	Return value
CONFIDENTIAL		10
DEFAULT		10
INVOICE		7
REFERENCE		5

Figure 4-23 Retention class list created in Configuration Manager

2. Create a list named `Folder mapping`. Fill in the values to map folder names to retention categories (Figure 4-24). You may import the values from a CSV file.



The screenshot shows the 'Lists' configuration window in the Configuration Manager. The 'General' tab is active, showing the list name 'Folder mapping' and a description: 'Maps different folder names to retention categories. If no entry is presented we will use the DEFAULT retention.' Below this is the 'Value List' section, which contains a table with three columns: 'Name', 'Description', and 'Return value'.

Name	Description	Return value
discussion		REFERENCE
plan		REFERENCE
contract		CONFIDENTIAL
hr		CONFIDENTIAL
invoice		INVOICE
billing		INVOICE

Figure 4-24 Folder mapping list created in Configuration Manager



## 4.6 Extending IBM Content Collector

IBM Content Collector provides a broad range of configuration options to implement different scenarios. Sometimes, however, it might be necessary to programmatically extend IBM Content Collector or the source system. This section discusses different extension points that are available, describes their strengths and limitations, and advises how to choose the appropriate extension point for your scenario.

### 4.6.1 Choosing the correct extension strategy for your scenario

Before implementing an extension for IBM Content Collector, carefully evaluate whether the scenario really requires an extension to be implemented. If a scenario cannot be implemented with the IBM Content Collector functionality that is ready for immediate use, then two different extension points are provided to add custom functionality to a task route:

- ▶ **Script Connector**

Script Connector is an IBM Content Collector connector that enables customers and service providers to implement custom tasks and collectors that extend standard IBM Content Collector functionality. This makes the Script Connector useful for integrating IBM Content Collector with existing business processes, prototyping and proof of concept studies, and for customization by implementers and customers.

- ▶ **Connector Development Kit**

If volume and required throughput for the extension is high, choosing the Connector Development Kit (CDK) is the appropriate option. The CDK provides APIs for implementing connector services and user interfaces in Visual C++ or .NET-based languages.

Developing an extension using the CDK is more complex than using the script connector, but usually yields better performance.

Depending on the source and target systems that are used in a scenario, additional extensions might be possible; for example:

- ▶ Writing a Notes agent that adds a specific property to a document before it can be archived.
- ▶ Writing an IBM FileNet P8 Event Handler that performs specific modifications, such as changing the document class, to a document after it has been added to the Object Store.
- ▶ Writing an application that extends Microsoft SharePoint.

- Having an application write out data files that are consumed by the file system connector. Typically this kind of extension provides additional metadata, but it can also generate additional data files, or interoperate with a third party system.

## 4.6.2 Extending the source system or target system

Archiving solutions can and sometimes should include custom development in the source system or target system. A distinct benefit of customizing a source or target system is that those involved (administrators, developers) have intimate knowledge of the system and can more directly relate the business requirements to the customization work. Another benefit can be that those responsible do not require permission from IBM Content Collector or other systems to make changes.

For this scenario, assume there are too many unknown Microsoft SharePoint custom columns to be discovered and mapped to target repository properties, most of which have unknown use cases in the target repository as well.

In such a scenario, creating Microsoft SharePoint scripts (for existing documents) and an event handler (for future documents) that will package all custom column values into a single or few hidden Microsoft SharePoint columns that you create across the Microsoft SharePoint areas to be collected is a appropriate solution. Configure those hidden columns to be mapped to a single or a few target repository properties. Appropriately configured searching in the repository will work against the properties. Where specific use cases are determined for custom columns, develop target repository customizations that extract the appropriate values and store them in distinct properties.

## 4.6.3 Using the Script Connector

The Script Connector uses popular scripting languages to provide a straightforward but flexible way of implementing custom collectors and tasks. Using scripting enables rapid delivery of comparatively simple and self-contained functionality. However, the use of scripting also means that the Script Connector is not suitable for all scenarios, especially those that have tightly constrained performance requirements.

The Script Connector is based on Windows COM technology. To perform work, the Script Connector creates a COM object using information in the Windows registry, then uses COM late binding to call a method named `execute` on that object.

To use the Script Connector to provide custom functionality, you must provide a COM component that implements the execute method and also provide the registry information to call it. The Script Connector is specifically intended to work with the scripting languages provided by the Windows Scripting Runtime (JScript and VBScript).

To interface with script code, the Script Connector uses Windows Scripting Components, a bridging technology that allows script code to be registered and called as COM objects. The script code can then use other COM components developed in any language, including those developed in other scripting languages. The Script Connector also provides Python libraries to work with Python scripts.

The Script Connector also provides an API to allow interaction with the IBM Content Collector infrastructure. This allows scripts to retrieve passed configuration and input parameters; to add metadata to existing entities; to submit new files and other entities; to return task status; to log messages; to update performance counters; and to register the script code so that it can be used. The Script Connector also supports the creation and registration of custom metadata sources.

The Script Connector provides a simple plug-in user interface to the IBM Content Collector Configuration Manager to add tasks and collectors to task routes, provide basic task and collector configuration, and to specify task input parameters. The text labels and tooltip help shown to users can be customized by updating a localization file.

For further details refer to the *IBM Content Collector Script Connector Implementation Guide*.

<http://www.ibm.com/support/docview.wss?uid=swg27024906>

#### **4.6.4 Using the IBM Content Collector Software Development Kit**

The Content Collector Software Development Kit (SDK) is the basis for all connectors that are available in IBM Content Collector. It provides the most scalable and flexible way to build a connector. To effectively work with the SDK requires C# or Visual C++ programming skills.

The SDK is not publicly available. Contact your IBM representative to obtain a copy of the SDK.

## 4.7 Conclusion

In this chapter we explained how to calculate document retention, and described preferred practices for scheduling collectors and deciding on the appropriate stubbing options. We also described how to work with task routes and decision points, and how to extend IBM Content Collector. Advanced expression examples that include working with list lookups were covered, along with explanations about how to migrate task routes from development to production systems and how to optimize your task routes for easy maintenance.







# Retention management

In this chapter we explain how to use IBM Content Collector to manage a document's retention after it is archived. Using a typical use case scenario as introduced in Chapter 2, "Example use cases" on page 17, we provide detailed steps demonstrating how to implement the scenario.

In this chapter we discuss the following topics:

- ▶ Retention management overview
- ▶ Stubbing lifecycle
- ▶ Expiration Manager
- ▶ Expired stub management
- ▶ Use case 1C: Lifecycle stubbing and retention management

## 5.1 Retention management overview

A document's full lifecycle covers the period from the document's creation through to its destruction. In this period, documents are subject to changing requirements for capture, storage, index, access, and timely retrieval and deletion. An effective retention policy ensures the removal of useless and expired documents in the repository in time to achieve cost-effective maintenance. Otherwise, the explosive growth of document volume can overwhelm system resources, depreciate performance, and bury valuable information beneath unnecessary and over-retained information.

IBM Content Collector provides both document archiving and document retention management. Retention management is applied to documents after they have been archived. Proper retention management enhances efficiency and reduces maintenance cost for the lifecycle management of documents.

In most organizations, various kinds of documents have different commercial use. Organizations want to set respective rules for managing document content so that non-valuable material is kept for a minimum amount of time, and valuable information is indexed and protected for future use.

When managing document retention, use the following guidance:

- ▶ To retain documents for litigation: The retention period must comply with corresponding laws and regulations.
- ▶ To retain documents for special business purposes: Set the retention period based on different business rules.
- ▶ To retain general documents: A default retention period must follow internal document controls. Retention periods for non-essential documents, such as junk emails, can have a much shorter retention period.

IBM Content Collector provides flexible and value-based task routes. Task routes can set a retention period for each incoming document based on various policy and document stubs for immediate storage saving.

Typically, organizations retain documents for a minimum period of time before they delete the documents. In IBM Content Collector, you can assign a retention date to each document. The Expiration Manager tool that ships with IBM Content Collector can then check the retention period of each document and delete expired documents for defensible disposal. You can run Expiration Manager regularly to dispose of expired documents.

When Expiration Manager deletes expired documents in the repository, it leaves the stubs in the source system. If the stub deletion policy does not align with the

retention policy in the repository, you need to synchronize the source system and repository. Otherwise, an error occurs when the user tries to retrieve content from the repository. IBM Content Collector also provides a special task route to delete expired stubs in source system automatically for data consistency.

The use case described in this chapter illustrates an example of retention management using an archive based on an expiration date. The archive was created in Chapter 3, “Dimensions of content archiving themes” on page 23.

This chapter focuses on the retention management of content after it is archived.

## 5.2 Stubbing lifecycle

Stubbing documents immediately after archiving is the fastest way to free storage. However, this can impact the user experience for accessing the content.

Because older emails are less likely to be accessed, IBM Content Collector for Email provides a stubbing lifecycle, which minimizes the impact on the user experience. This feature reduces the size of emails as they age by removing parts of the emails at different points in time. See 4.2.1, “Configuring schedules” on page 89 for an example of stubbing operations performed over a period of time for different parts of emails.

A stubbing lifecycle can minimize the impact on the user experience, but can also reduce the rate at which storage space is freed, so here is a guideline:

- ▶ If you have a backlog of email that is ready to be deleted, use immediate stubbing.
- ▶ For recently delivered email, use a stubbing lifecycle.

Before planning for the stubbing lifecycle, define the time line of email retention. Figure 5-1 illustrates a typical email retention time line.

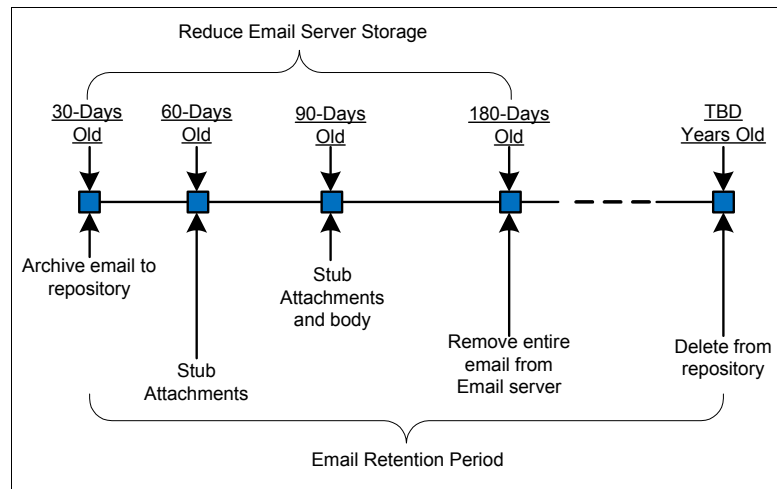


Figure 5-1 Email retention time line

You can add or remove stub options in the Reduce Email Storage phase based on your requirements. To configure an email lifecycle, you must set up a task route that contains the EC Process Stubbing Life Cycle collector. Modify a task route based on the template that matches your requirement, for example “P8\_LD\_1.2 - Default Archiving(stubbing).ctms.” The stubbing schedule needs to comply with the email retention time line definition. If you need multiple email retention time lines, for example different time lines for different departments, you can either create one task route for each time line or use one task route with multiple EC Process Stubbing Life Cycle collectors.

Table 5-1 lists email retention time for two departments: Finance and HR.

Table 5-1 Email retention time lines for two departments

Stub option/stub schedule	Finance	HR
Remove nothing and add text	N/A	N/A
Remove attachments	1 month	15 days
Remove attachments and cut body	N/A	N/A
Remove attachments and body	3 months	N/A

Stub option/stub schedule	Finance	HR
Delete entire email	1 year	6 months
Select documents to re-create stubs	7 days	15 days
Select restored documents for deletion	3 months	1 month

In this scenario, to minimize the workload, especially on the email server, the departments want to execute lifecycle management when the email server is idle or has a small workload. After consulting with the IT department, the department choose the following schedule:

- ▶ Finance: 1:00 a.m. to 8:00 a.m. every Saturday
- ▶ HR: 1:00 a.m. to 8:00 a.m. every Sunday

The sample stubbing task route template that ships with IBM Content Collector has one EC Process Stubbing Life Cycle collector. For this scenario, you can use the following process to add an additional collector and configure each collector to be responsible for one group's email:

1. Import the "P8\_LD\_1.2 - Default Archiving(stubbing).ctms" task route template. Then change the name of the EC Process Email Stubbing Life Cycle collector to Finance Dept.

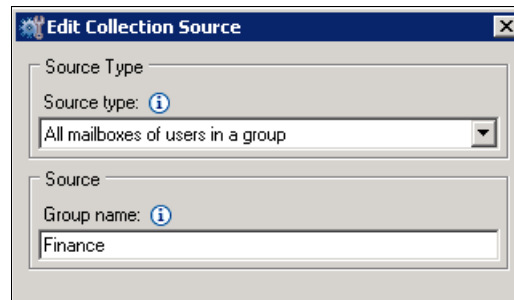
2. On the **Schedule** tab, enter the parameters as shown in Figure 5-2.

The screenshot shows the 'EC Process Email Stubbing Life Cycle' configuration window with the 'Schedule' tab selected. The window has five tabs: 'General', 'Schedule', 'Collection Sources', 'Life Cycle', and 'CommonStore'. The 'Schedule' tab contains the following settings:

- This collector runs:** A dropdown menu set to 'Weekly'.
- Time Frame:**
  - Start date:** A date picker set to '8/21/2012'.
  - End date:** Radio buttons for 'Run endlessly' (selected), 'Until' (with a date picker set to '8/20/2012'), and 'End after running' (with a spinner set to '1' times).
- Run Time:**
  - Start first collection at:** A time picker set to '1:00 AM'.
  - Stop collection:** Radio buttons for 'When task completes' and 'At:' (selected, with a time picker set to '8:00 AM').
- Repeat Collection Every:** A spinner set to '1' weeks on:
  - Days of the week: Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, and Saturday. The 'Saturday' checkbox is checked.

Figure 5-2 Task route schedule for the Finance department

3. Go to the **Collection Sources** tab and enter Finance as the group name, as shown in Figure 5-3.



The image shows a Windows-style dialog box titled "Edit Collection Source". It has two main sections: "Source Type" and "Source". In the "Source Type" section, there is a label "Source type:" followed by a dropdown menu that currently displays "All mailboxes of users in a group". In the "Source" section, there is a label "Group name:" followed by a text input field that contains the word "Finance". Both labels have a small blue information icon (i) next to them. The dialog box has a standard Windows title bar with a close button (X) in the top right corner.

*Figure 5-3 Collection source for the Finance department*

4. Go to the **Life Cycle** tab and enter the lifecycle schedule (listed in Table 5-1 on page 122). Refer to Figure 5-4.

The screenshot shows the 'EC Process Email Stubbing Life Cycle' configuration window with the 'Life Cycle' tab selected. The window has a title bar with a question mark icon. Below the title bar are five tabs: 'General', 'Schedule', 'Collection Sources', 'Life Cycle', and 'CommonStore'. The 'Life Cycle' tab is active. The configuration is organized into four sections, each with a title bar and an information icon (i):

- Perform Stubbing**: Contains a 'Relative to:' label and a dropdown menu set to 'archived date'.
- Select Documents To**: Contains four options, each with a checkbox, a label, and a duration selector:
  - ☐ Remove nothing and add text (i): Duration is 2 months.
  - ☒ Remove attachments (i): Duration is 1 month.
  - ☐ Remove attachments and cut body (i): Duration is 1 month.
  - ☒ Remove attachments and body (i): Duration is 3 months.
  - ☒ Delete entire email (i): Duration is 1 year.
- Re-create Stubs**: Contains a checkbox 'Select documents to re-create stubs (i)' which is checked, followed by a duration selector set to 7 days after restoring.
- Delete Documents Restored From Search Results**: Contains a checkbox 'Select restored documents for deletion (i)' which is checked, followed by a duration selector set to 3 months after restoring.

Figure 5-4 Time line definition for the Finance department

5. Add a new EC Process Email stubbing lifecycle collector. Then, repeat steps 2 through step 4 for the new lifecycle collector with the following parameters:
  - a. Enter the name of HR Dept.
  - b. Enter a schedule as shown in Table 5-2 on page 127.



Table 5-2 Schedule definition of HR department

Fields	Value
This collector runs	Weekly
Start date	8/22/2012
End date	Run endlessly
Start first collection at	1:00 a.m.
Stop collection at	8:00 a.m.
Repeat Collection Every	1 weeks
Day	Sunday

- c. Set the collection source as “All mailboxes of users in a group” with a group name of HR.
- d. For the lifecycle, enter the values listed in Table 5-1 on page 122.

Now, the task route supports two different email retention time lines. Figure 5-5 show the new task route.

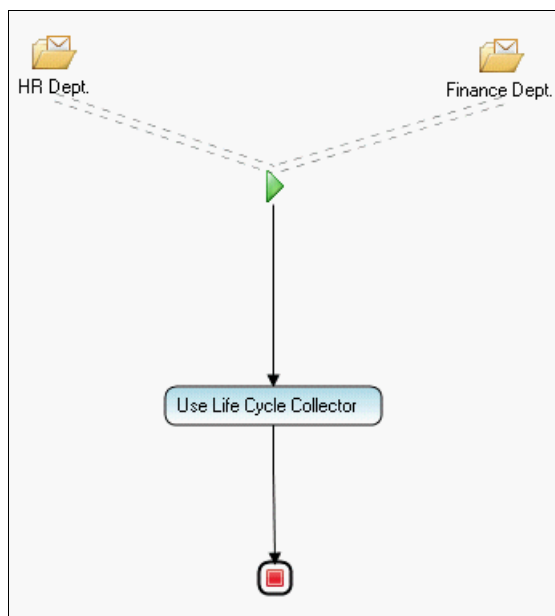


Figure 5-5 Overview of the new task route

Validate the task route in a test environment before you put it into production. To speed testing, change the collectors so that they start immediately, and change the units of the age criteria from *months* to *minutes*. After testing, be sure to restore the collector settings to the values that you want to use in production.

**About the Email stubbing lifecycle:** Email stubbing lifecycle is provided only for Lotus Domino and Microsoft Exchange.

## 5.3 Expiration Manager

Over time, documents accumulate and occupy storage. Although documents expire over time, you need to dispose the expired documents in time to alleviate the burden of storing them and reduce maintenance cost. IBM Content Collector provides Expiration Manager, which can perform the following tasks:

- ▶ Check the expiration date of each document in the repository
- ▶ Perform statistics on expired documents
- ▶ Delete the expired documents to free up storage and achieve defensible disposal

The Expiration Manager is a stand-alone tool that can be run on any workstation in addition to the IBM Content Collector servers. It mainly includes two components for IBM FileNet P8 and IBM Content Manager, respectively. The IBM Content Collector package provides .bat files for the Windows operating system platform by default. For information about running IBM Content Collector on UNIX, refer to the IBM Content Collector Information Center:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/index.jsp?topic=%2Fcom.ibm.content.collector.doc%2Fretention%2Ft\\_afu\\_running\\_expirationmgr\\_unix.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/index.jsp?topic=%2Fcom.ibm.content.collector.doc%2Fretention%2Ft_afu_running_expirationmgr_unix.htm)

### 5.3.1 Looking up the expiration date of a document

Expiration Manager looks up the expiration date of a document based on the repository type and the corresponding data model. For documents that do not have an expiration date set or have an invalid expiration date, Expiration Manager can specify a general expiration date for them.

#### **Expiration date for documents in IBM Content Manager**

For documents in IBM Content Manager, Expiration Manager can look up multiple item types (document types) and use wildcard characters. In IBM Content Manager, IBM Content Collector uses the default data model, the

EXPIRATIONDATE system attribute, to store the expiration date of a document. To see the expiration date of documents at the database level, run a query at the database level and check the “EXPIRATIONDATE” column.

## Expiration date for documents in IBM FileNet P8

For documents in IBM FileNet P8, Expiration Manager uses the IBM Content Collector Email data model, which has three versions. Table 5-3 lists the configuration differences among data source types and data model versions.

*Table 5-3 Configuration for different source types and data model versions*

Source type	Data model	Target class type	Symbolic ClassName
Email	Bundled (V1) for LCSE	EMAIL	ICCDEIClassName=ICCMail ICCXITClassName=ICCMailSearch
Email	Compound (V2) for LCSE	EMAIL	ICCDEIClassName=ICCMail2 ICCXITClassName=ICCMailSearch2
Email	V3 for IBM Content Search Services	EMAILCSS	ICCDEIClassName=ICCMail3
File System, Microsoft SharePoint, Connections	N/A	NON-EMAIL	ICCNonEmailClass=corresponding symbolic class name of each source type

For simplicity, you can create subclasses of the default email data model instead of creating your own classes for email. For example, you can set the `ExcludeSubclasses = No` parameter in the configuration file to enable the Expiration Manager to look up subclasses.

Because other non-email data sources use normal document classes to store content, you can create your own classes for them. In addition, IBM Content Collector uses the non-system *ICCExpirationDate* property to save the expiration date on a FileNet P8 repository. Thus, you need to add the *ICCExpirationDate* property to your own classes to save the expiration date.

Some parameters of the `afu-P8ExpirationMgr-config-sample.properties` file depend on the data source type and data model version. Set these parameters carefully based on your requirements.

## Documents without a valid expiration date

Sometimes documents do not have a valid expiration date for the following reasons:

- ▶ You forgot to set an expiration date while archiving the document, so that the value of the expiration date in the repository is NULL.
- ▶ The document was archived into P8 prior to IBM Content Collector V2.1.1. The IBM Content Collector data model for P8 did not have ICCExpirationDate fields at that time.
- ▶ You did not add the ICCExpirationDate property to your own classes to save the expiration date for non-email archiving.

If you have documents without a valid expiration date, use the Expiration Manager to calculate a general virtual expiration date by setting the following parameters in the configuration files:

```
UseArchiveDate=ifExpirationDateIsNull|ifExpirationDateFieldDoesNotExist  
ExpireDays = <number of days until expiration>
```

With this calculation, the expiration date becomes ArchiveDate + ExpireDays.

### 5.3.2 Working with eDiscovery and records management solutions

If your system has another data protection mechanism for compliance, for example eDiscovery hold and records management, Expiration Manager excludes expiring the documents that are held by IBM eDiscovery Manager or managed by IBM Enterprise Records even though the expiration date has occurred. Thus, no conflict exists with either eDiscovery Manager or Enterprise Records.

### 5.3.3 Running multiple instances of Expiration Manager

As mentioned previously, Expiration Manager for IBM Content Manager can process multiple item types at the same time. Therefore, you do not need to run multiple instances of the Expiration Manager.

**Tip:** The item types in IBM Content Manager are handled on a one-by-one basis instead of in a multi-thread way.

With FileNet P8, often organizations use multiple object stores to store different kinds of documents. Because the Expiration Manager for P8 does not support multiple object stores, you have to run multiple instances of the Expiration Manager to process documents for different object stores respectively. In this

scenario, you can run multiple instances of Expiration Manager on one machine instead of deploying it on multiple machines.

In general, this tool has minimal impact on the host machine. However, if you want to run multiple instances on one machine, use a machine that does not run other applications and set it up as follows:

1. Create multiple configuration files for each object store respectively.
2. Set a different port number in the **TCPIPPortForShutdown** field in each configuration file.
3. Specify the appropriate log and report directories.

Now you can start the instances with the corresponding configuration files and shut down instances with the corresponding shutdown port number.

For example, assume you have two object stores named OS1 and OS2. The OS1 object store stores historical emails based on a version 2 data model. The OS2 object store is a new object store that stores emails based on a version 3 data model. For this example, follow these steps to run multiple instances of the Expiration Manager on one machine:

1. Create two copies of the `afu-P8ExpirationMgr-config-sample.properties` file. Name the files `OS1.properties` and `OS2.properties`. Modify these configuration files with the key parameters listed in Table 5-4.

*Table 5-4 Configuration for each object store*

Parameter	OS1.properties	OS2.properties
ObjectStoreName	OS1	OS2
ICCDelClassName	ICCMail2	ICCMail3
ICCXITClassName	ICCMailSearch2	N/A
TargetClassType	EMAIL	EMAILCSS
LogFileDir	C:\\OS1\\log	C:\\OS2\\log
ReportFilePath	C:\\OS1\\report	C:\\OS2\\report
TCPIPPortFortShutdown	8100	8200

2. Use the following commands to start each of these files:

```
P8ExpirationMgr-sample.bat -propFile OS1.properties -delete  
-password xxxxxx
```

```
P8ExpirationMgr-sample.bat -propFile OS2.properties -delete  
-password xxxxxx
```

Now, two instances are running on the same machine.

3. Shut down one of the instances by specifying a port number. For example, use the following command to shut down the OS1 instance:

```
P8ExpirationMgr-sample.bat -shutdown -p 8100
```

Alternatively, use the following command to shut down the OS2 instance:

```
P8ExpirationMgr-sample.bat -shutdown -p 8200
```

### 5.3.4 Scheduling Expiration Manager execution

Because the Expiration Manager is a stand-alone tool that is launched by a script, you can use third-party tools to implement a schedule easily. For example, you can use Task Scheduler on a Windows operating system platform and the **cron** utility on a UNIX platform.

Additionally, the Expiration Manager provides the following parameter options in the configuration file to control how long it runs:

- **MaxRunTime**: Specify the maximum time that the Expiration Manager can run before it is stopped.
- **MaxNumOfBatchesInOneRun**: Control the maximum number of documents that can be deleted in one run.

Otherwise, you can create two tasks in a third-party scheduler, one task to start and another task to stop.

### 5.3.5 Optimizing Expiration Manager for performance

The Expiration Manager creates a high workload on the repository, especially the disk I/O of the database. To balance the load of the repository, plan the execution schedule carefully. Launch it when the repository has a light workload.

To improve performance of Expiration Manager, also consider the following tuning tips.

## Index

Two database indexes are usually needed on the database when the delete load is added to an object store. For example, with an IBM DB2® system:

- ▶ If the expiration date is saved in ICCExpirationDate, the following index can improve query performance greatly. A slight adjustment is needed for different field names on ICCEXPIRATIONDATE.

For efficiency reasons, add this index only when deleting is necessary.

```
db2 "create index ceuser.expirationmanager on ceuser.docversion  
("OBJECT_CLASS_ID" ASC,"HOME_ID" ASC, "OBJECT_ID" ASC,  
"UXXXX_ICCEXPIRATIONDATE" ASC,"CREATE_DATE" ASC) ALLOW REVERSE  
SCANS"
```

- ▶ An index on the ICCMailReference property of the ICCMailInstanceX object needs to be created to avoid a table scan on the GENRIC table. This index is normally automatically created when creating the IBM Content Collector data model. Be sure it exists before starting the Expiration Manager.

If the index does not exist, use IBM FileNet Enterprise Manager to create it with the following steps:

1. Right-click the class **ICCMailInstanceX** and select **Properties**.
2. Select the **Property Definition** tab.
3. Select the **ICCMailReference** property and click **Edit**.
4. In the Properties window, select **Set/Remove**.
5. In the Set/Remove Index window, select the **Set** option and check **Single Indexed**.
6. Click **OK** to close all the windows and apply the changes.

## Parameters for optimization

The afu-P8ExpirationMgr-config-sample.properties configuration file provides the following options for performance tuning:

- ▶ DeleteBatchSize
- ▶ NumberOfDeletionThreads
- ▶ QueryPageSize

These options have detail annotation in the configuration file. Table 5-5 on page 134 lists the preferred values for these parameters.

Table 5-5 Preferred values for common servers

Parameters	Recommend value	Upper limit
DeleteBatchSize	50-100	<1000
NumberOfDeleteThreads	10-20	<50
QueryPageSize	500	<50000

If your system hardware is powerful and the performance of this tool is not at a level where you need it, you can increase the values. However, do not exceed the upper limit.

## 5.4 Expired stub management

A *document stub* is a document that has its content removed and replaced with hotlinks. An *orphaned* document stub, or also known just as an orphaned stub, is a document where the content referenced by the hotlinks is deleted. Typically, this situation occurs because the Expiration Manager deleted the content in accordance with the company retention policy. This section focuses on the management of expired stubs.

### 5.4.1 Determine the ID of the repository

You can use task route templates to create task routes for managing stubs that were generated by IBM Content Collector and that have become orphaned. IBM Content Collector V3.0 ships with templates for Email and Microsoft SharePoint stubs.

Stubs generated by versions of IBM Content Collector prior to V3.0 do not contain the ID of the repository where content was archived. Thus, if you point the *Confirm Document task* in those task route templates to the wrong repository, IBM Content Collector might delete stubs that you want to keep.

To avoid this occurrence, the task requires that you provide the correct ID of the repository for old stubs, as explained here:

- ▶ In the case of FileNet P8, this ID is the GUID that is generated automatically by FileNet P8 when you create an object store.
- ▶ In the case of IBM Content Manager, this ID is the value of the *ICCRepositoryGUID* in *ICCRepositoryInformation* item type.



**Tip:** The **Shortcut link** field in the P8 and CM8 Confirm Document task is required only on FileSystem and Microsoft SharePoint.

## Object store ID for FileNet P8

As mentioned, this ID is the GUID that is generated automatically by FileNet P8 when you create an object store. You can open FEM (Content Engine Enterprise Manager) and find it in object store properties pane as shown in Figure 5-6.

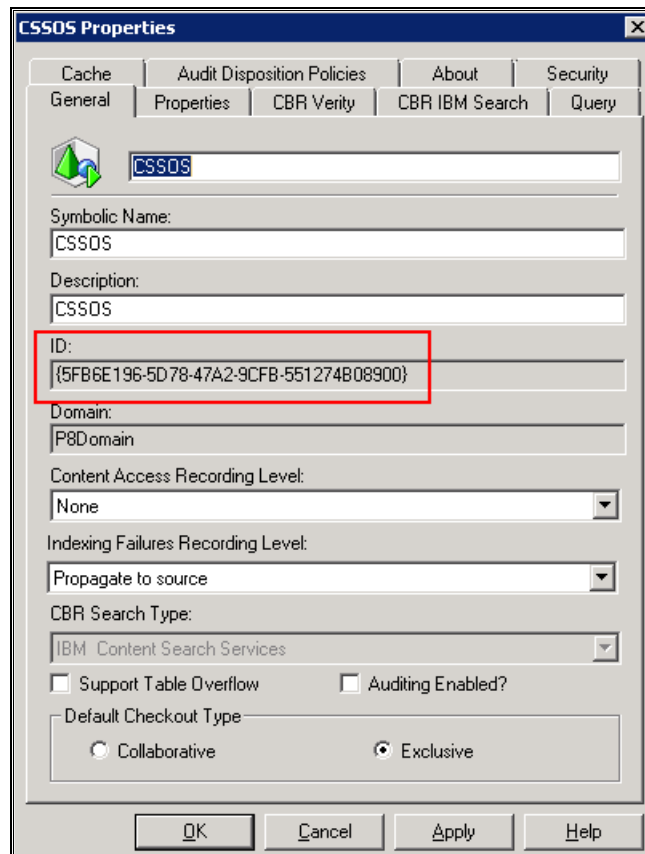


Figure 5-6 Object store ID in properties pane

## Repository ID for IBM Content Manager

IBM Content Manager does not provide a repository ID. Therefore, IBM Content Collector assigns a repository ID the first time that it connects. It does this by creating an item type named *ICCRRepositoryInformation* and an attribute named *ICCRRepositoryGUID*. It then creates an *ICCRRepositoryInformation* item with a freshly generated GUID as the value for the *ICCRRepositoryGUID* attribute.

You can find the repository ID using the client of IBM Content Manager (for example, IBM Content Navigator as shown in Figure 5-7), or by querying the database if you do not have the client.

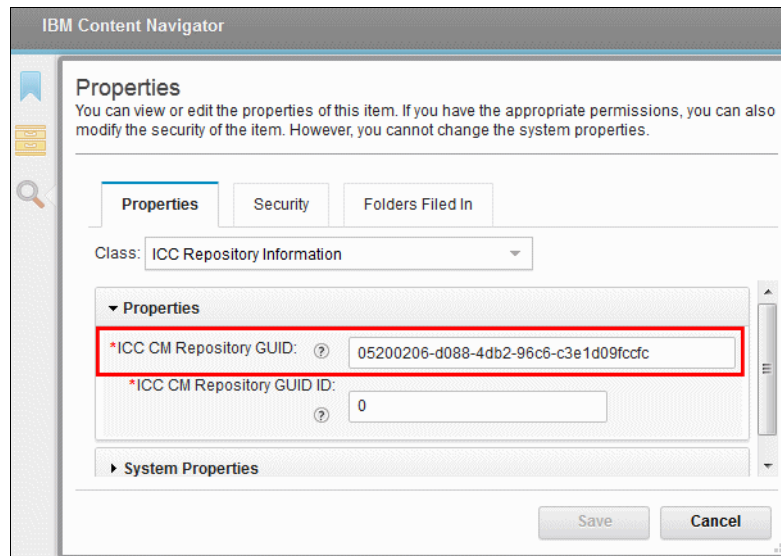


Figure 5-7 Repository ID displayed in IBM Content Navigator

To discover the table or view of the ICCRepositoryInformation in the database level, use the following query:

```
select COMPONENTVIEWID, COMPONENTTYPEID, ITEMTYPEID,  
COMPONENTVIEWNAME, CONCAT (CONCAT('ICMUTO', CAST(COMPONENTTYPEID as  
char(4))), '001') TABLE from icmstcompviewdefs where itemtypeid in  
(select keywordcode from icmstnlkeywords where keywordclass=2 and  
keywordname = 'ICCRepositoryInformation') with ur
```

Then, go to corresponding view or table to find the value of ICCRepositoryGUID, as shown in Figure 5-8.

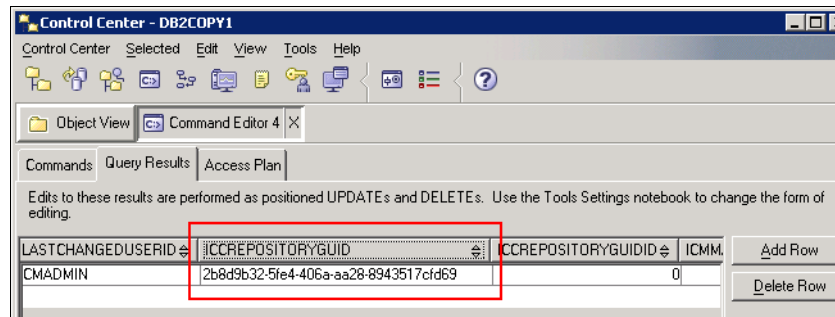


Figure 5-8 Value of ICCRepositoryGUID

## 5.4.2 Email

Unlike Microsoft SharePoint and File System, email has a stubbing lifecycle feature. This feature provides an alternative method to delay stub deletion. This method is preferred over using the special task route for expired stub deletion because it uses fewer resources. However, if you want to keep stubs as long as possible until the expiration date in an email system, or if you want to double-check whether an expired stub exists after a stubbing lifecycle, configure orphaned stub deletion task routes to delete stubs or archived messages that refer to messages that no longer exist in the repository.

IBM Content Collector provides four orphaned stub deletion task routes that cover Lotus Domino, Microsoft Exchange, FileNet P8, and CM8. Select one of these task routes according to your system type. These task routes search the mail system for stubbed or restored email. Then, for each hot link that the stub email contains, the task route searches the repository for the corresponding document.

Understand that these searches can impact the performance of both the mail system and the repository. Therefore, schedule the collectors in the task routes to run, at most, once a month. Additionally, schedule the run at a time of day when the repository has a smaller workload.

The “P8\_LD - Orphaned Stub Deletion.ctms” template, shown in Figure 5-9, is an example of a clean-up task route.

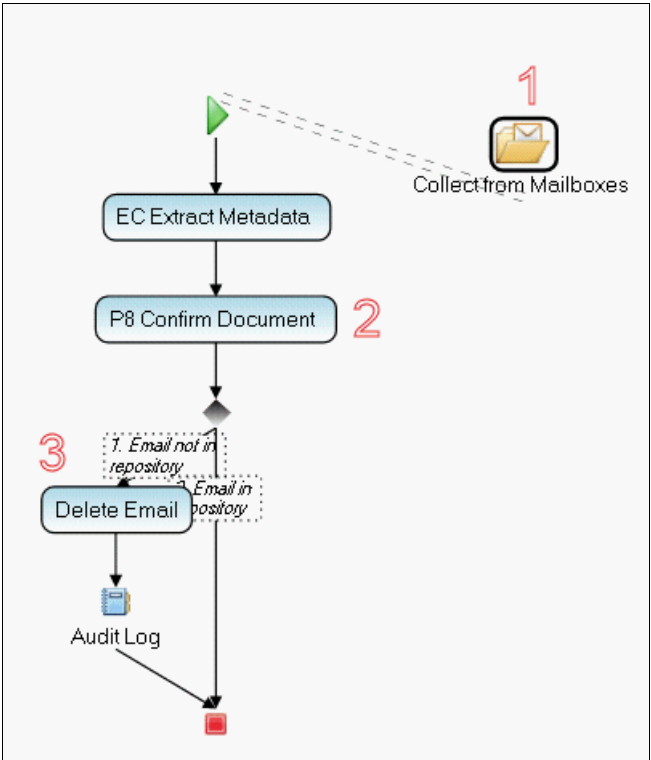


Figure 5-9 P8\_LD - Orphaned Stub Deletion template

This template uses the following tasks:

1. Collect from Mailboxes

This task is an EC Process Email Stubbing Life Cycle collector. The default settings are shown in Figure 5-10.

The screenshot shows the 'EC Process Email Stubbing Life Cycle' configuration window with the 'Life Cycle' tab selected. The window has five tabs: 'General', 'Schedule', 'Collection Sources', 'Life Cycle', and 'CommonStore'. The 'Life Cycle' tab contains three main sections: 'Perform Stubbing', 'Select Documents To', and 'Re-create Stubs'. In the 'Perform Stubbing' section, 'Relative to:' is set to 'archived date'. The 'Select Documents To' section has four options: 'Remove nothing and add text' (2 months), 'Remove attachments' (1 month), 'Remove attachments and cut body' (1 month), and 'Delete entire email' (1 minute), with the last one being checked. The 'Re-create Stubs' section has 'Select documents to re-create stubs' checked, set to '1 minutes after restoring'. The 'Delete Documents Restored From Search Results' section has 'Select restored documents for deletion' unchecked, set to '1 months after restoring'.

EC Process Email Stubbing Life Cycle

General | Schedule | Collection Sources | Life Cycle | CommonStore

Perform Stubbing

Relative to: [i](#)

archived date

Select Documents To

☐ Remove nothing and add text [i](#)

2 months

☐ Remove attachments [i](#)

1 months

☐ Remove attachments and cut body [i](#)

1 months

☐ Remove attachments and body [i](#)

1 months

☒ Delete entire email [i](#)

1 minutes

Re-create Stubs

☒ Select documents to re-create stubs [i](#)

1 minutes after restoring

Delete Documents Restored From Search Results

☐ Select restored documents for deletion [i](#)

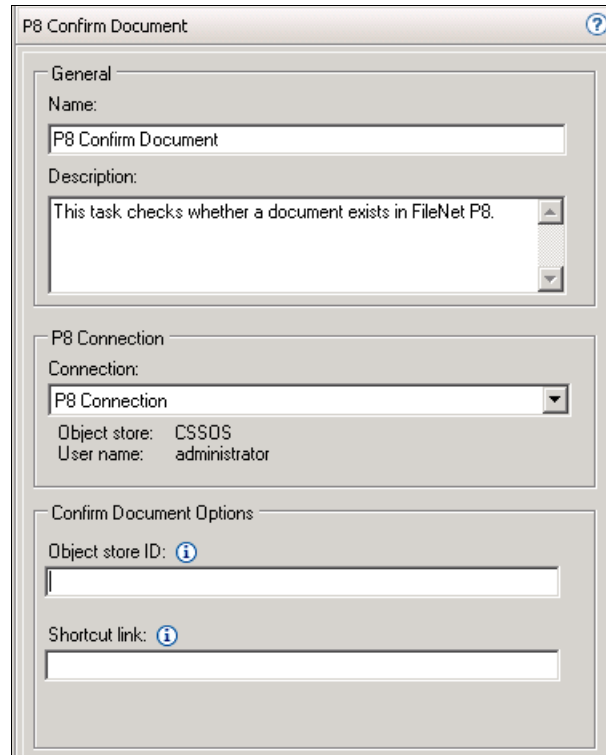
1 months after restoring

Figure 5-10 Configuration of the Collect from Mailboxes task

This collector collects stubs by default. If you do not want to delete the stubs, clear the **Delete entire email** option. Restored documents are excluded from the search by default, because they typically have a short retention period in the mail server. You can also include them if you want to delete them.

## 2. P8 Confirm Document

This task includes the **Object store ID** and **Shortcut link** options, as shown Figure 5-11. Refer to 5.4.1, “Determine the ID of the repository” on page 134 to ensure the correct value.



The screenshot shows a window titled "P8 Confirm Document" with a help icon in the top right corner. The window is divided into three sections: "General", "P8 Connection", and "Confirm Document Options".

- General**: Contains a "Name:" field with the text "P8 Confirm Document" and a "Description:" text area with the text "This task checks whether a document exists in FileNet P8."
- P8 Connection**: Contains a "Connection:" dropdown menu with "P8 Connection" selected, and "Object store:" (CSSOS) and "User name:" (administrator) fields.
- Confirm Document Options**: Contains an "Object store ID:" field with an information icon and a "Shortcut link:" field with an information icon.

Figure 5-11 The P8 Confirm Document task

## 3. Delete Email

This task is an EC Create Email Stub task with the **Delete entire email** option. If you are processing stubs that are generated by versions of IBM Content Collector before V3, run this task route without this task in advance to pre-check whether orphaned stubs exist. If orphaned stubs exist, add the task again to execute the deletion operation.

If you confirm email in multiple object stores, you need a task route with a decision point that routes the stubs to the appropriate P8 Confirm Document task. The example in Figure 5-12 shows a task route that handles stubs where the email received date is used to determine in which object store the email is archived.

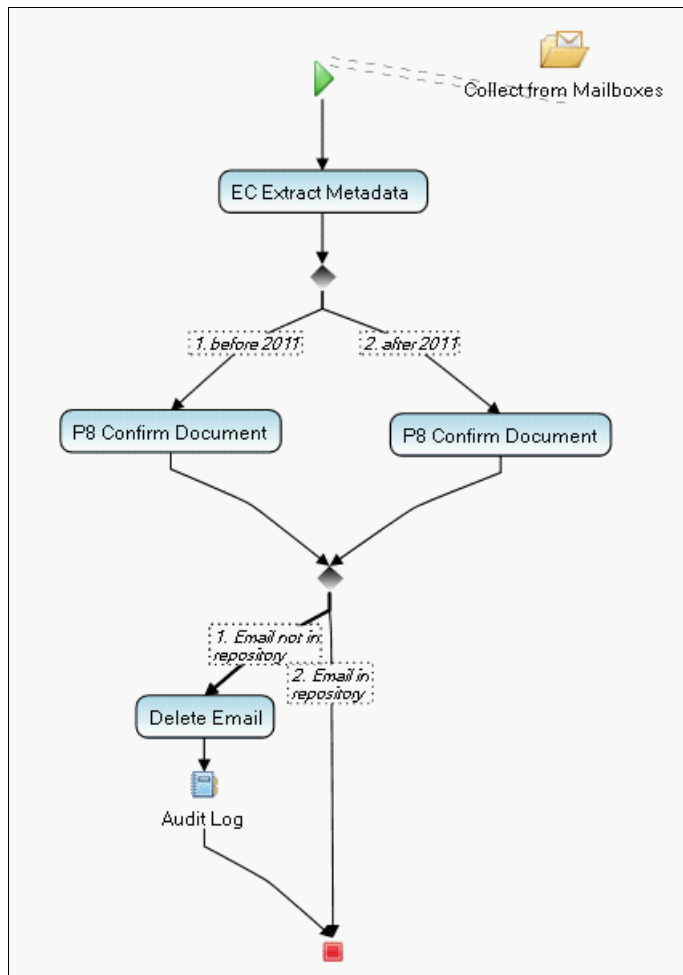


Figure 5-12 Sample taskroute for multiple object stores

### 5.4.3 Microsoft SharePoint

If you configured IBM Content Collector to generate document stubs in Microsoft SharePoint, use the link management stubs task route to update and expire them. IBM Content Collector for Microsoft SharePoint can update links, and not simply delete expired links. Link update is essential in the following conditions:

- ▶ If you previously used the **Replace with link** option in IBM Content Collector V2.2 or later, and now IBM Content Collector is upgraded to V3, use link update. Because IBM Content Collector did not write a repository ID to the links prior to V3.0, you must supply a correct repository ID in a clean-up task route (for example SP Manage P8/CM Links.ctms), to prevent the unintentional deletion of links.
- ▶ The URL for the web application server that is responsible for handling the requests that are generated when users try to access content through a stub has changed. For example, the host name or port has changed.

To use a link management template to update stubs after the URL is changed, complete these steps:

1. Use the “SP Audit P8/CM Links.ctms” task route template to identify and report broken or expired links from the Microsoft SharePoint server to the repository. By using this, you validate the repository ID and get a list of the unresolved links. The unresolved links are recorded in an audit log.
2. If unresolved links are reported in audit log, use the “SP Manage P8/CM Links.ctms” template to delete or repair them. If you renamed your web application server or changed the port on which it listens, be sure to update the **Shortcut link** field in the P8/CM Confirm Document task.
3. After the link update, run “SP Audit P8/CM links.ctms” again for verification.

**Tip:** Make sure the repository ID is correct for the P8/CM Confirm Document task.

If you confirm documents from Microsoft SharePoint in multiple object stores, you need a task route with a decision point that routes the stubs to the appropriate P8/CM Confirm Document task.



## 5.4.4 File system

IBM Content Collector does not provide a template for cleaning up orphaned file system stubs. You need to build one manually. Figure 5-13 shows an example of a clean-up task route.

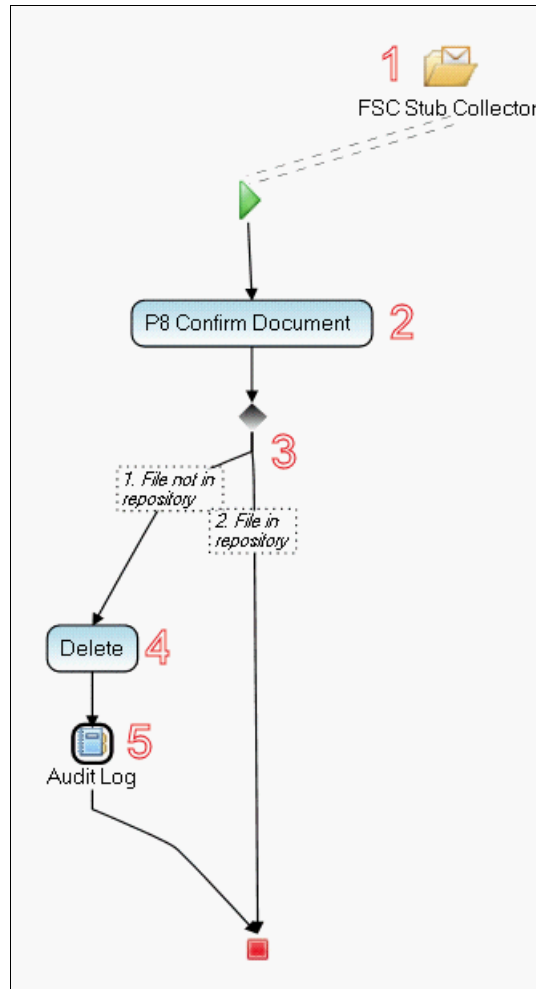


Figure 5-13 Task route overview

Here is a description of the task route shown in Figure 5-13:

### 1. FSC Stub Collector

The collector uses a monthly schedule because the stub validation can put a significant load on the source and target system. Because stubs are not

marked as processed or archived by default, the collector ignores these flags by default. Ignoring the flags can provide better performance. Similarly, no special attributes are set on stubs by default; therefore, these attributes can generally be ignored.

## 2. P8 Confirm Document

The object store ID is optional, because FSC Stub Collector only collects stubs that are generated by IBM Content Collector V3.0. If your repository is IBM Content Manager, replace the P8 Confirm Document task with the CM Confirm Document task. Be sure to input the correct URL in the **Shortcut** link field.

## 3. Decision point and rules

The decision point has the following branches:

- File not in repository
- File in repository

Figure 5-14 displays the file not in repository expression.

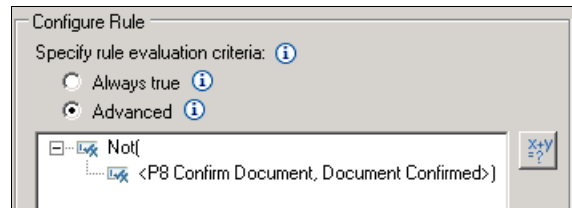


Figure 5-14 File not in repository

Figure 5-15 displays the file in repository expression.

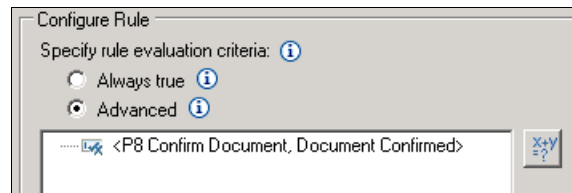


Figure 5-15 File in repository

#### 4. FSC Post Processing task

To delete orphaned stubs, select the **Delete file** option without the Replace file with shortcut created from metadata property option, as shown in Figure 5-16.

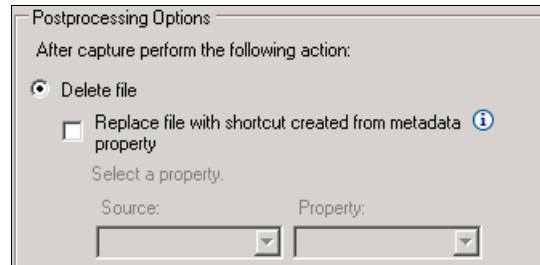


Figure 5-16 Post-processing option for clean up taskroute

To simply confirm whether orphaned stubs exist before deletion, remove this task from the task route.

#### 5. Audit Log

Add an audit log to record results.

Because there is no risk of unintentional deletion of document stubs, you can run this task route directly without a pre-check. Similar to Email, you can add multiple P8/CM Confirm Document tasks and add a decision point before those tasks with rules and criteria through metadata for multiple object stores or library servers support.

## 5.5 Use case 1C: Lifecycle stubbing and retention management

**Before you begin:** The retention management scenario described in this section builds upon the use case 1 that is described in Chapter 3, “Dimensions of content archiving themes” on page 23. To run this scenario, emails are archived with an expiration date setting. Thus, this scenario begins with a stubbing lifecycle and describes how to use the Expiration Manager to delete expired emails in the repository and to clean up expired stubs in an email server.

In this scenario, the company prefers to use stubbing lifecycle to minimize any impact to the user experience introduced by stubbing. The company uses the following timeline for stubbing:

- ▶ Stub attachments 30 days after archiving
- ▶ Stub all content after 90 days after archiving
- ▶ Remove entire email 180 days after archiving

Regarding restored emails during the lifecycle, the company wants to recreate the stub 15 days later. Some people might restore a document for search results. Because the volume and retention period in the mailbox is short, the company wants to delete the documents manually instead of automatically.

Part of the emails are expired based on the retention policy in the repository. The company does not want to waste resource for over-retention and wants to use the Expiration Manager to remove these emails in time.

The IT department wants to double-check whether all expired stubs are deleted from the mail system in their monthly system health check. Thus, an audit task route is configured for this requirement.

The following steps are involved in implementing this concrete scenario:

1. Create the stubbing lifecycle task route.
2. Enable the Expiration Manager.
3. Create the audit task route (optional).

### 5.5.1 Create the stubbing lifecycle task route

The task route template that best fits this scenario is the “P8\_EX\_2.3 - Space Saving(stubbing).ctms” template. After you import the task route template, configure it by completing the following steps:

1. Select the start node (the green triangle). Then mark the task route as active.
2. Select the **Collect from Mailboxes** collector. On the **General** tab, mark the collector as active.

3. Go to the **Schedule** tab. Set the collector to run daily, running endlessly. Set the start date for today. Set the first collection to start at 10:00 p.m. and to stop at 7:00 a.m. Repeat the collection every day.

Figure 5-17 shows these settings.

The screenshot shows the 'EC Process Email Stubbing Life Cycle' dialog box with the 'Schedule' tab selected. The 'General' tab is also visible. The 'Schedule' tab contains the following settings:

- This collector runs:** A dropdown menu set to 'Daily'.
- Time Frame:**
  - Start date:** A dropdown menu set to '8/30/2012'.
  - End date:** A dropdown menu set to '8/29/2012'.
  - Run endlessly:** A radio button that is selected.
  - Until:** A radio button that is not selected.
  - End after running:** A radio button that is not selected.
  - 1** times (with a spinner box).
- Run Time:**
  - Start first collection at:** A dropdown menu set to '10:00 PM'.
  - Stop collection:**
    - When task completes:** A radio button that is not selected.
    - At:** A radio button that is selected.
    - 7:00 AM** (with a dropdown menu).
- Repeat collection every:** A spinner box set to '1' days.

Figure 5-17 Task route schedule settings

4. Go to the **Collection Sources** tab and edit the collection source.

5. Go to the **Life Cycle** tab, and set the time line based on requirements (Figure 5-18).

The screenshot shows the 'EC Process Email Stubbing Life Cycle' configuration window with the 'Life Cycle' tab selected. The window has a title bar with a question mark icon. Below the title bar are tabs: 'General', 'Schedule', 'Collection Sources', 'Life Cycle', and 'CommonStore'. The 'Life Cycle' tab is active. The configuration is organized into several sections:

- Perform Stubbing**:
  - Relative to: [i](#)
  - archived date
- Select Documents To**:
  - ☐ Remove nothing and add text [i](#)
    - 2 months
  - ☒ Remove attachments [i](#)
    - 30 days
  - ☐ Remove attachments and cut body [i](#)
    - 1 months
  - ☒ Remove attachments and body [i](#)
    - 90 days
  - ☒ Delete entire email [i](#)
    - 180 days
- Re-create Stubs**:
  - ☒ Select documents to re-create stubs [i](#)
    - 15 days after restoring
- Delete Documents Restored From Search Results**:
  - ☐ Select restored documents for deletion [i](#)
    - 1 months after restoring

Figure 5-18 Life Cycle time line setting

6. Save the task route.

The task route is now set up correctly and can be promoted to production after verification.

## 5.5.2 Enable the Expiration Manager

For this scenario, an email archived based on the V3 data model and the FileNet P8 CE application are running on IBM WebSphere® Application Server. To enable the Expiration Manager for FileNet P8:

1. Navigate to the <ICC\_HOME>\tools\ExpirationManager directory.
2. Edit the P8ExpirationMgr-sample.bat file as follows:
  - a. Locate the line set AppServer\_Type= "<AppServer\_Type>" and replace "<AppServer\_Type>" with WAS.
  - b. Locate the line set ConnProtocol\_Type= "<ConnProtocol\_Type>" and replace "<ConnProtocol\_Type>" with http.
  - c. Locate the line set AppServer\_Port= "<AppServer\_Port>" and replace "<AppServer\_Port>" with 9080.
3. Edit the afu-P8ExpirationMgr-config-sample.properties file.

You do not need to change all the parameters in the configuration file. Change only the parameters listed in Table 5-6 for this scenario. Save your changes and give the file a meaningful name, such as EmailOS.properties.

Table 5-6 Mandatory changes in the configuration file for this scenario

Parameters	Default value	Modification result
UserName	ceadmin	real user name
CEServerHost	P8server1	real host name
P8DomainName	P8Domain1	real p8 domain
ObjectStoreName	OStore1	Symbolic name of target object store
ICCDelClassName	ICCMail2	ICCMail3
ICCXITClassName	ICCMailSearch2	Ignore it, because version3 datamodel is being used
TargetClassType	EMAIL	EMAILCSS
NumberOfDeleteThreads	1	5

4. Verify the configuration.

Execute the following command in command window:

```
P8ExpirationMgr-sample.bat -propFile EmailOS.properties -count  
-passw0rd xxxxxx
```

If the configuration is correct, the Expiration Manager lists the configuration information, connects to the FileNet P8 CE server, and starts the query to count the expired items, as shown in Figure 5-19.

```
C:\IBM\ContentCollector\tools\ExpirationManager>P8ExpirationMgr-sample.bat
t -propFile EmailOS.properties -count -password xxxxxx

-----
Licensed Materials - Property of IBM
IBM Content Collector
Copyright IBM Corp. 2009, 2012
IBM Content Collector FileNet P8 Expiration Manager
Version: 3.0.0.0-20120509-1446

-----
Log file directory: C:\IBM\ContentCollector\tools\ExpirationManager\log
Logging was initialized successfully.

-----

Options
Content Engine server host name:-----P8CE.icc.com
P8 domain name:-----P8Domain
Object store name:-----EmailOS
User name:-----p8admin
Action:-----list
IBM Content Collector email class name:-----ICCMail3

-----
Connected to the Content Engine server

-----
Domain: P8Domain

-----
Object store: EmailOS

-----
Report file directory:
C:\IBM\ContentCollector\tools\ExpirationManager\report

-----
The tool listens for shutdown requests on port 8,100.

-----
2012-08-30 16:07:47 Start query
2012-08-30 16:08:02 End query
2012-08-30 16:08:02 Query time: 15907 ms
```

*Figure 5-19 Output of Expiration Manager to list expired items*

5. If expired items exist, execute a delete operation with the following command:

```
P8ExpirationMgr-sample.bat -propFile EmailOS.properties -delete
-passw0rd xxxxxx
```



Next, the Expiration Manager looks up and deletes expired items, and the console outputs current progress and throughput during deletion, as shown in Figure 5-20. You can also find the overall progress in the report file.

```
2012-08-30 16:26:36 Number of documents that qualify for deletion:--5000
                    Number of documents that were deleted:-----4000
                    Number of documents that could not be deleted:--0
                    Number of query tasks that are running:-----0
                    Number of delete tasks that are running:-----5
                    Number of delete tasks that are idle:-----0
                    Batches of documents in the data queue:-----14
                    Deletion rate:-----69
documents/second
```

*Figure 5-20 Output during deletion operation*

6. By default configuration, the Expiration Manager runs 9 hours. You can interrupt it with the following command in another command window.

```
P8ExpirationMgr-sample.bat -shutdown -p 8100
```

When the deletion instance receives the shutdown request, it finishes any pending work and stops with a statistic summary of the current round, as shown in Figure 5-21.

```
2012-08-30 18:44:59 Received a shutdown request. Completing pending work
...

2012-08-30 18:45:27 Number of documents that qualify for
deletion:--16000
                    Number of documents that were
deleted:-----15250
                    Number of documents that could not be deleted:--0
                    Number of query tasks that are running:-----0
                    Number of delete tasks that are running:-----0
                    Number of delete tasks that are idle:-----0
                    Batches of documents in the data queue:-----0
                    Deletion rate:-----1
documents
/second
2012-08-30 18:46:27 Number of documents that were deleted: 15250
                    Number of documents that could not be deleted: 0
                    Deletion rate: 68 documents/second
2012-08-30 18:46:27 Operation time: 224 seconds
```

*Figure 5-21 Received shutdown request and stop*

### 5.5.3 Create the audit task route (optional)

This step is optional. You do not need to audit the task route frequently because the stubs are deleted by stubbing lifecycle. You can use an audit task route if you want to confirm whether expired stub exists in the email system.

The task route template that best fits this requirement is the “P8\_EX - Orphaned Stub Deletion” task route. After you import the task route template, configure it using the following steps:

1. Select the **Collect from Mailboxes** collector. On the **General** tab, mark the collector as active.
2. Go to the **Schedule** tab. Set the schedule to run montly, running endlessly. Start with the current month and run until the last day of the month at 1:00 a.m. Stop the schedule when the task completes.

**Task completion time:** This task can take a long time to complete, because all emails in the collection source must be checked. Thus, it is possibly that the task will not complete in one night.

3. Go to the **Collection Sources** tab and edit the collection source.
4. Select the **P8 Confirm Document** task, and set the Object store ID if part of stubs were generated before IBM Content Collector V3.0.
5. Remove the **Delete Email** task.

The task route is now set up correctly, as shown in Figure 5-22 on page 153. It can be promoted to production after verification. If expired stubs exist, the

information is recorded in the audit log. You can add the **Delete Email** task again to clean up expired stubs.

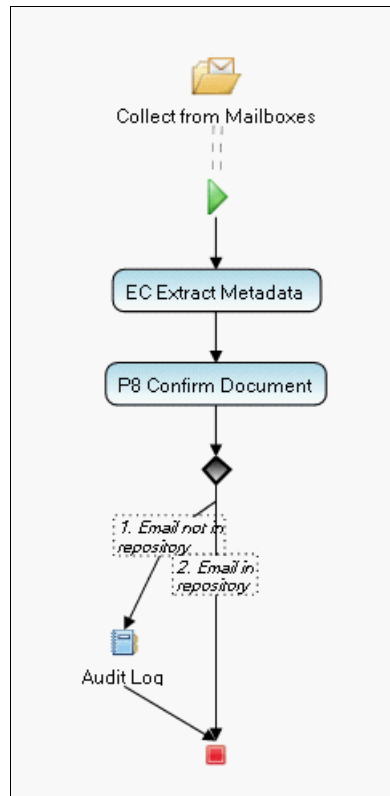


Figure 5-22 Audit task route for expired stubs

## 5.6 Conclusion

In this chapter we described the basic retention management provided by IBM Content Collector after the documents are archived. Specifically, we covered stub lifecycle, Expiration Manager, and expired stub cleanup. We also demonstrated the use case described in Chapter 2, “Example use cases” on page 17 and provided detailed steps explaining how to implement retention management. If you have more complex retention requirements, such as the application of variable retention periods and disposition options, use the Declare Record task to with the Enterprise Records capability as described in Chapter 7, “Records management integration” on page 221.





## Document classification

IBM Content Classification is a tool for classifying unstructured content. Integrating Content Classification with Content Collector provides Content Collector with the ability to make routing decisions based on free-text content.

In this chapter we discuss the following topics:

- ▶ The business value of using IBM Content Classification
- ▶ IBM Content Classification overview
- ▶ Content Classification applied to other scenarios
- ▶ Setting up Content Classification with Content Collector
- ▶ Using decision plan for value-based archiving and defensible disposal
- ▶ Generating new facets with Content Classification
- ▶ Reviewing and auditing archived emails and documents
- ▶ Use case 2: Email archiving with content classification
- ▶ Considerations and guidelines

## 6.1 The business value of using IBM Content Classification

Given the explosion in volume of unstructured, free-form texts saved in email servers, file systems and clouds, data departments find it increasingly necessary to distill the essence of these documents automatically, so that a quick decision can be made about how to handle them.

Content Classification can analyze free form texts statistically to recognize their underlying business-defined categories. These categories are not generic, but are trained from examples of relevant data for each specific deployment, which allows a higher degree of accuracy.

Content Classification can also apply a workflow consisting of manual rules to arrive at a classification decision. Each rule can be as complex as necessary, and the final result can be a combination of statistical analysis and manual rules.

Content Classification adds value to Content Collector by enhancing Content Collector routing capabilities, allowing it to make routing decisions based on statistical analysis of free-form text, word proximity, and other content-based criteria. When a Content Collector task route uses Content Classification, the task route can be kept simple by leaving the complex workflow logic to Content Classification and merely acting on its results.

The results returned from Content Classification can help determine whether or not a document should be declared as a record in IBM Enterprise Records. If not declared as a record, then what retention period should be given to it, what FileNet folder it belongs in, and similar decisions. Content Classification can also enrich the document with new metadata as facets to allow for better eDiscovery.

## 6.2 IBM Content Classification overview

IBM Content Classification classifies unstructured and structured content using a combination of rules and statistics. The rules reside in a decision plan (DP) and the statistics reside in a knowledge base (KB), as explained here.

### Decision plan

A decision plan contains various rules and actions that might trigger when checked against the fields of a document. A rule is *triggered* if applying the rule's conditions to the document returns true. When a rule is triggered, its actions are executed one by one.

Decision plan rules are created manually in the Classification Workbench. They can scan the document fields for words and regular expressions, and generate new fields.

Decision plan rules can also pass the document to one or more knowledge bases to match the document statistically to the category profiles. The results of a match can be exported from the decision plan without change, or they can be overridden by subsequent rules.

Decision plans do not have to contain references to knowledge bases. In this case the rules act directly on the content fields and do not perform statistical matching.

To obtain match results from a knowledge base, Content Collector can connect to the knowledge base directly, or it can connect to a decision plan whose rules connect to the knowledge base.

## Knowledge base

A knowledge base contains statistical profiles for each category in its domain. A knowledge base is created by training it in the Classification Workbench from a set of preclassified documents. Matching a document against the knowledge base yields suggested categories and their scores.

## Additional references

Detailed information about Content Classification is beyond the scope of this book. For more details, you can refer to the following documentation:

- ▶ Content Classification Information Center  
<http://pic.dhe.ibm.com/infocenter/classify/v8r8>
- ▶ IBM Redbooks publication *IBM Classification Module: Make It Work for You*, SG24-7707 (IBM Content Classification was formerly known as IBM Classification Module.)  
<http://www.redbooks.ibm.com/abstracts/sg247707.html?Open>
- ▶ White paper “Decision Plan best practices”  
<http://www.ibm.com/support/docview.wss?uid=swg27023248>
- ▶ White paper “Achieve compliance and control costs with automatic categorization of email for records management”  
<http://public.dhe.ibm.com/common/ssi/ecm/en/zzw03051usen/ZZW03051USEN.PDF>
- ▶ Defensible Disposal Library  
<http://www.ibm.com/software/ecm/disposal-governance/library.html>

- ▶ Integrating Content Classification into Content Collector  
[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/index.jsp?topic=%2Fcom.ibm.content.collector.doc%2Ficm%2Ft\\_afu\\_enabling\\_rc\\_icm\\_integration.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/index.jsp?topic=%2Fcom.ibm.content.collector.doc%2Ficm%2Ft_afu_enabling_rc_icm_integration.htm)
- ▶ Technote: Email archiving  
<http://www.ibm.com/developerworks/data/library/techarticle/dm-0812chitive1/index.html>
- ▶ IBM Redpaper™ publication *Content Collector/Content Classification*  
<http://www.redbooks.ibm.com/redpapers/pdfs/redp4705.pdf>

## 6.3 Basic content classification integration

From Content Collector, you can include the IBM Content Classification task in a task route to add content classification capability in the archiving solution. Figure 6-1 on page 159 illustrates a task route using the IBM Content Classification task.



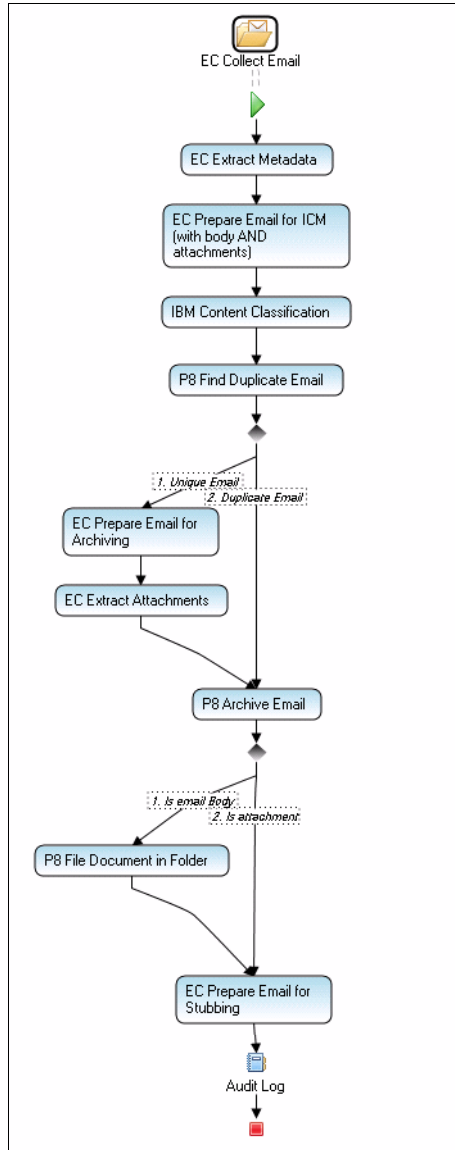


Figure 6-1 Basic email archiving task route

This task route archives emails from a Microsoft Exchange server to FileNet. The task route uses IBM Content Classification to classify contents, and saves each email under a folder determined by a Content Classification knowledge base.

To use the IBM Content Classification task, you need to set up IBM Content Classification for Content Collector.

### 6.3.1 Setting up Content Classification with Content Collector

Follow the Content Collector Information Center instructions for integrating Content Classification with Content Collector. These instructions are available at the following site:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/index.jsp?topic=%2Fcom.ibm.content.collector.doc%2Ficm%2Ft\\_afu\\_enabling\\_rc\\_icm\\_integration.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/index.jsp?topic=%2Fcom.ibm.content.collector.doc%2Ficm%2Ft_afu_enabling_rc_icm_integration.htm)

After the integration, the Content Collector configuration manager will include the Content Classification task in its toolbox (Figure 6-2).

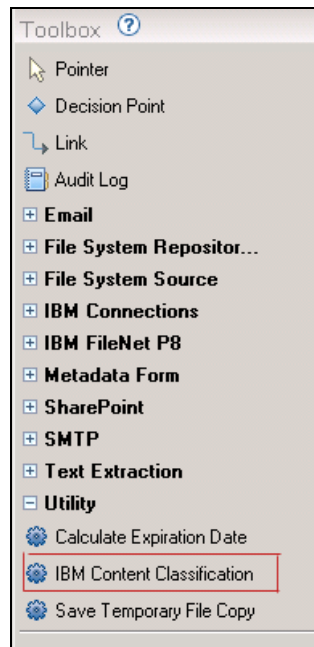
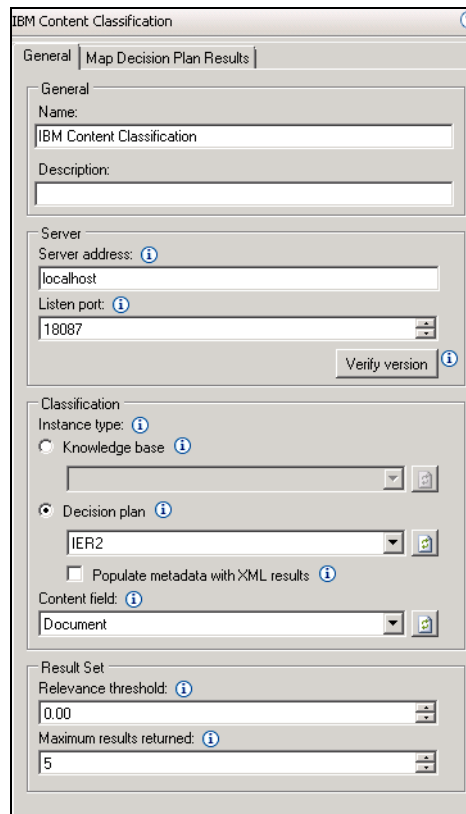


Figure 6-2 Content Classification task in Content Collector toolbox

The Content Classification properties dialog has two tabs. The first tab contains Content Classification connection parameters (Figure 6-3).



The image shows the 'IBM Content Classification' dialog box with the 'General' tab selected. The dialog has two tabs: 'General' and 'Map Decision Plan Results'. The 'General' tab contains several sections: 'General' with 'Name' (IBM Content Classification) and 'Description' (empty); 'Server' with 'Server address' (localhost), 'Listen port' (18087), and a 'Verify version' button; 'Classification' with 'Instance type' (radio buttons for 'Knowledge base' and 'Decision plan', with 'Decision plan' selected), a dropdown for 'Knowledge base' (empty), a dropdown for 'Decision plan' (IER2), and a checkbox for 'Populate metadata with XML results' (unchecked); 'Content field' (Document); and 'Result Set' with 'Relevance threshold' (0.00) and 'Maximum results returned' (5). Information icons (i) are present next to several fields.

Figure 6-3 Content Classification connection parameters

After specifying the Content Classification host and port, test the connection with the **Verify version** button. Then refresh the knowledge base or decision plan drop-down list, and select one knowledge base or decision plan. The content field selected for Content Collector is usually the “Document” field.

The second tab in this dialog (Map Decision Plan results) contains a mapping from decision plan output fields to Content Collector metadata and will be shown later. This tab is not used when Content Collector connects directly to a knowledge base, because the results of the knowledge base match are mapped automatically to predefined Content Collector metadata fields.

## 6.3.2 Configuring task route for automated email archiving example

As shown in Figure 6-1 on page 159, the example task route uses the IBM Content Classification task to automatically archive Microsoft Exchange emails to FileNet.

The task route is created using the email archiving template P8\_CSS\_EX\_1.1 from the available templates with a few modifications.

The changes from the original template are as follows:

1. Add the EC Prepare Email for ICM task.

Configure it to extract the attachments with the body, because they will be needed by Content Classification in its classification decision. See “Use care when defining task routes that send email to Content Classification” on page 216 for more details about this configuration.

2. Add the IBM Content Classification task.

Configure it to match against a knowledge base or run through a decision plan.

3. After the P8 Archive Email task, add a decision point to check for attachments.

You will file the email body to the folder, but not its attachments. Attachments do not contain the Content Classification metadata that is necessary for filing. In any case, they will be filed automatically into the body's folder.

4. Add the P8 File in Folder task.

The task uses the knowledge base classification result, or some metadata created by a decision plan, to generate a folder path. Figure 6-4 on page 163

shows a possible configuration for this task, where the found category name forms a folder under a Classification folder.

The screenshot shows a configuration window titled "P8 File Document In Folder". It has three main sections: "General", "P8 Connection", and "File in Folder Options".

- General:** Contains a "Name:" field with the text "P8 File Document in Folder" and a "Description:" text area with the text "This task saves document objects to specific repository folders to facilitate search and retrieval."
- P8 Connection:** Contains a "Connection:" dropdown menu set to "P8 Connection", and fields for "Object store:" (ICCEXDS) and "User name:" (p8admin).
- File in Folder Options:** Contains a "Folder path:" text area with the text "Add('Classification/' <IBM Content Classification, Most Relevant Category>)". Below this are three buttons: "Edit...", "Add...", and "Remove".

At the bottom of the window, there are two checkboxes: "Create folder if it does not exist" (checked) and "Inherit folder security:" (unchecked). Below the "Inherit folder security:" checkbox are two radio buttons: "Set security parent of document to folder" (selected) and "Add folder security to document security".

Figure 6-4 Generating path for a P8 folder

### 6.3.3 A BPM task route example

In a somewhat different scenario, you can file all mail bodies in the folder "email." The email attachments would each be classified separately by Content Classification, and filed into different folders based on Content Classification's

suggested category. The email body would contain references to the location of its attachments in the repository. This is shown in Figure 6-5.

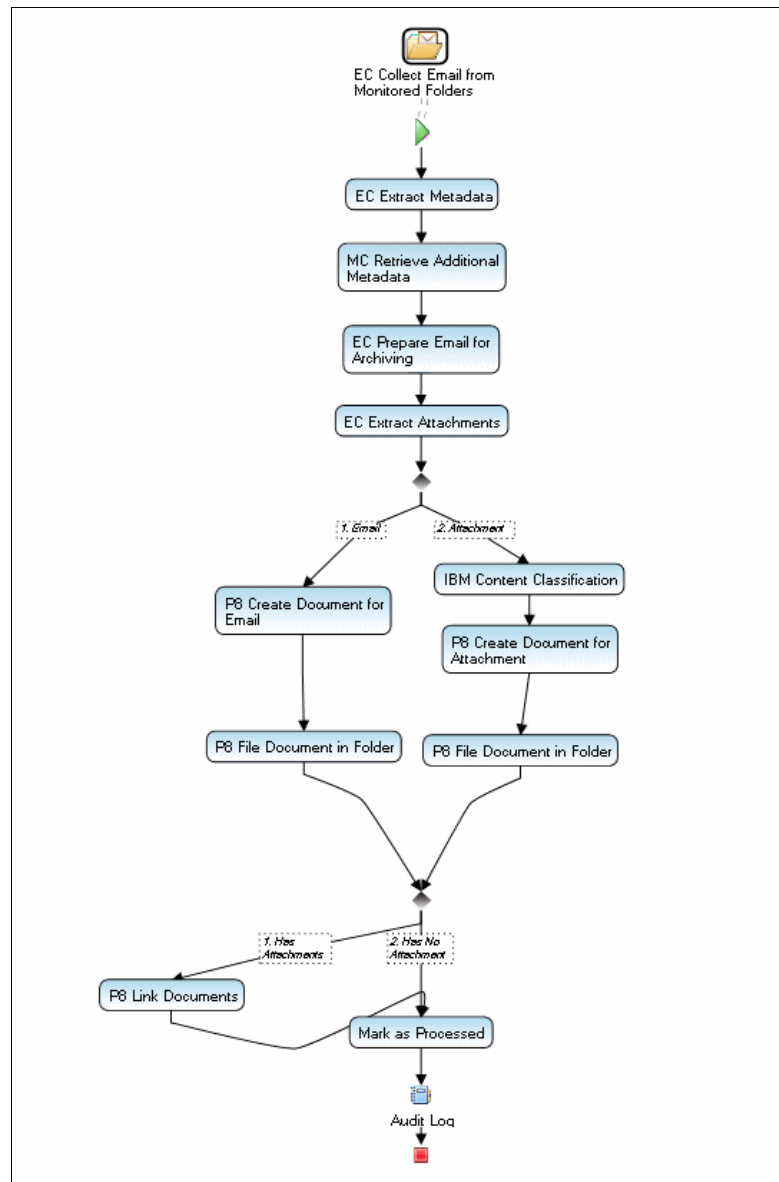


Figure 6-5 Alternative BPM task route

In this task route, the attachments extracted by Content Collector are sent one by one to classification, and the suggested category name is appended to a folder

path by the task “P8 File Document in Folder.” The original body is not classified but always filed under “email.”

### 6.3.4 Working with Content Collector email client integration

The Content Collector email client integration is a new feature in Content Collector 3.0 that allows users to specify metadata for an item *before* Content Collector processes the item. Review, audit and control can take place before any Content Collector action is executed.

Consider a BPM application that requires users to inspect incoming documents and set certain fields such as document type and author. The Content Collector email client integration can plug into this process by presenting a form to users with some fields already filled in with suggestions obtained from a Content Classification decision plan.

The decision plan can suggest a document type by matching to a knowledge base and taking the top category, or populating a drop-down list with likely categories. The decision plan might also search for regular expressions in the document fields, and this can help fill some of the form’s fields. In addition, the decision plan might look for specific words and patterns in proximity, and this can help fill other fields.

After the user reviews Content Classification’s suggestions, the document ID and the user-approved metadata are cached in a temporary database by Content Collector. When Content Collector starts to process the item, it appends this metadata to the item and the metadata takes part in Content Collector’s routing decisions.

Such a system ties into workers’ familiar user interface and can reduce their online time, speed up the handling of documents, and increase consistency in the filling of forms.

In addition, a user’s correction or acceptance of Content Classification’s suggestions can provide seamless feedback benefits. When configured properly, the normal work done by the user will automatically supply a steady stream of information back to the knowledge base, which can improve its subsequent suggestion of categories.

Figure 6-6 shows a customizable Content Collector form that allows the user to select a customer number suggested by Content Classification, before Content Collector processes the email.

Please select a customer number

Valid	Subject	From	Received
✓	Order	...	...
✓	RE: Customer meeting	...	...
	FYI: IBM InfoSphere Content Collector 2.1.11	...	...
✓	coffee?<ec...	...	...

Submit All Documents

Submit Selected Documents

Cancel

customer number

customer a

customer b

Apply to Selected Documents

Figure 6-6 A browser form used in email client integration

For more information, see the developerWorks article from the following URL:

[https://www.ibm.com/developerworks/mydeveloperworks/groups/service/html/communityview?communityUuid=e8206aad-10e2-4c49-b00c-fee572815374#fullpageWidgetId=Wf2c4e43b120c\\_4ac7\\_80ae\\_2695b8e6d46d&file=f61b7a03-752a-4810-8322-8e918d3d980b](https://www.ibm.com/developerworks/mydeveloperworks/groups/service/html/communityview?communityUuid=e8206aad-10e2-4c49-b00c-fee572815374#fullpageWidgetId=Wf2c4e43b120c_4ac7_80ae_2695b8e6d46d&file=f61b7a03-752a-4810-8322-8e918d3d980b)

An implementation sample is available here:

<ICM\_SERVER\_HOME>\Samples\ICCEmailClient\_Classification\_Integration

## 6.4 Content Classification applied to other scenarios

Aside from the standard email archival scenarios discussed in 6.3, “Basic content classification integration” on page 158, Content Classification can provide value in other use cases. Several of the more common ones are detailed in this section.



## 6.4.1 Value-based archiving and defensible disposal

Value-based archiving implies that users have a well-defined procedure to determine what goes into the archive and what and when it can be discarded. Defensible disposal is often an important component of value-based archiving.

Machines and humans both make mistakes when deciding whether to keep a document or dispose of it. The difference is that machine mistakes are systematic and reproducible. Paradoxically, this can sometimes make machine mistakes easier to defend in court than human mistakes.

In 6.5, “Using decision plan for value-based archiving and defensible disposal” on page 168, we explain one way of using a decision plan from Content Classification to calculate a document’s expiration date automatically for value-based archiving and defensible disposal.

## 6.4.2 Using Content Classification for record declaration

In task routes that declare documents as records, Content Classification can assist the decision process by classifying the document content and searching it for special words. It can create fields specifying record type and other information necessary for its declaration. See 7.4, “Use case 3: Email archiving with records declaration” on page 243 for an example.

**Records declaration consideration:** The Classification Workbench’s decision plan editor has an action template for records declaration. However, this template is designed for use by the Classification Center and uses a specific format that only this application understands.

When creating rules to help Content Collector declare records, use the generic `set_content_field` actions instead of using this template.

## 6.4.3 Using Content Classification for eDiscovery

When a document is processed by a decision plan, the decision plan might tag it with new metadata or facet fields. Before the document is added to the repository, Content Collector can use these metadata fields in its routing decisions. After the document has been added to the repository by Content Collector, users can employ these facets as query filters to narrow down searches and obtain a useful set of results.

In 6.6, “Generating new facets with Content Classification” on page 181, we illustrate how to generate new metadata from Content Classification to be used

for eDiscovery. The new metadata can also be used for records management or defensible disposal.

## **6.5 Using decision plan for value-based archiving and defensible disposal**

Content Classification Decision Plans lend themselves well to value-based archiving and defensible disposal scenarios. If required, you can run the document through a long workflow and write as many rules as it takes to get the job done. In this section, we show how Content Classification can be used to calculate the expiration date for documents.

### **6.5.1 Setting expiration date using Content Classification calculation**

Figure 6-7 on page 170 shows a task route that sets an expiration date on the archived mail. In the IBM Content Classification task, a decision plan is called to calculate an expiration date, and the result is mapped to Content Collector

metadata. In the P8 Archive Email task, the expiration date is inserted into the repository.

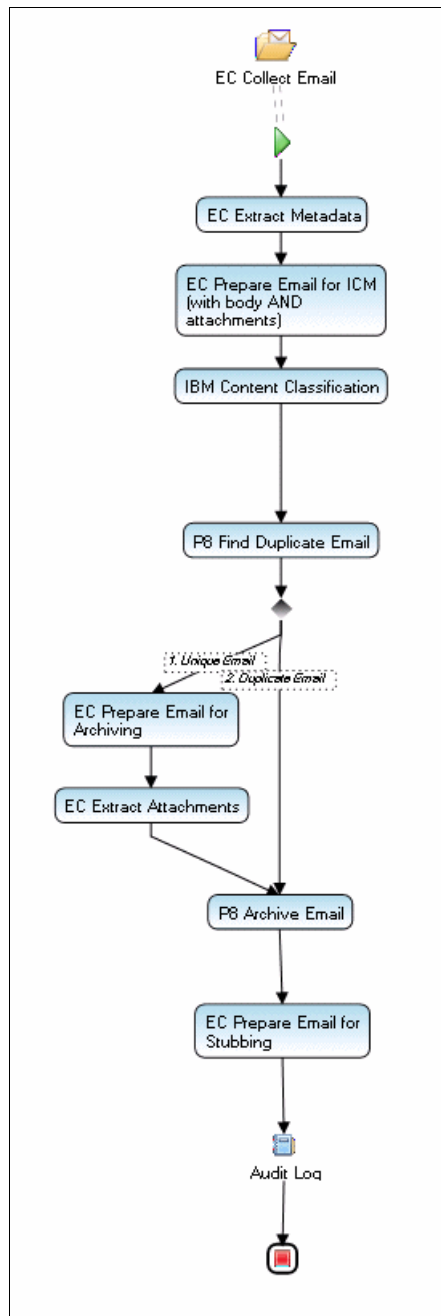


Figure 6-7 Archival with a calculated expiration date

Figure 6-8 shows the second tab in the properties of the IBM Content Classification task. Only one field of interest (ICC\_ExpirationDate) is created by the decision plan, and it is mapped to an Content Collector metadata field called Expiration Date.

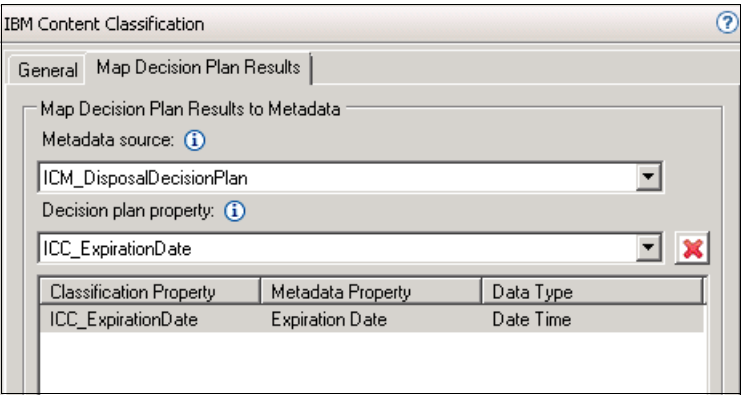


Figure 6-8 Mapping decision plan field to Content Collector field

Figure 6-9 on page 171 shows the properties of the P8 Archive Email task. The Content Collector metadata field Expiration Date is used for setting the email's expiration date in FileNet.

P8 Archive Email

Common Settings | Instance Settings

General

Name:  
P8 Archive Email

Description:  
Archive an email into IBM FileNet P8 that has IBM Content Search Services configured as its indexing server.

P8 Connection

Connection:  
P8 Connection

Object store: ICCEXOS

User name: p8admin

Define the settings for creating a distinct email instance (DEI) object to store data that is shared among all copies of an email document.

Expiration Metadata Mapping Options

☒ Set an expiration date [i](#)

Expiration date metadata mapping [i](#)

Source: ICM\_DisposalDecisionPlan Property: Expiration Date

Data Correction

☒ Truncate strings [i](#)

☐ Ignore choice list properties on error [i](#)

Property Mappings

Document class: [i](#)

ICCMail 3

Show "Hidden Properties" [i](#)

Property	Value
Document Title	<Email, Subject>
ICM_action	
ICM_creationDate	
ICM_DecisionResults	
ICM_details	
ICM_expirationDate	

Edit... Reset value

Figure 6-9 Setting FileNet expiration date from metadata

A CM8 repository has a similar setup, except that the calculated date cannot be used directly, and a Calculate Expiration Date task must be used as a thin wrapper.

## 6.5.2 Decision plan used in the expiration calculation

In this section we demonstrate the decision plan used in this example.

When creating the decision plan, you will typically organize the rules in *groups*. Each group is in charge of discovering whether a document meets a business criterion that implies a specific retention period. At the beginning of each group, you check whether a retention period has already been determined for the document. If the answer is yes, the rest of the group is skipped. The last group calculates a purge date based on the creation date and the retention period.

The following implementation of this scheme shows a decision plan composed of three parts:

- ▶ Group 1 checks for a retention period of ten years.
- ▶ Group 2 checks for a retention period of five years.
- ▶ Group 3 calculates the expiration date.

### Group 1 rules: Checking for a retention period of ten years

The first rule of decision plan group 1 rules is shown in Figure 6-10. It assumes the senders of the email appear in a field called **ICM\_From** (which is Content Classification's default field name for mail senders). It scans that field to see if it contains any words found in a wordlist called **LitigatorNames**. (See 6.6.2, "Generating facets using wordlists" on page 183 for an example of wordlist creation). If the rule is triggered, an action sets the field **retention\_years** to the value 10.

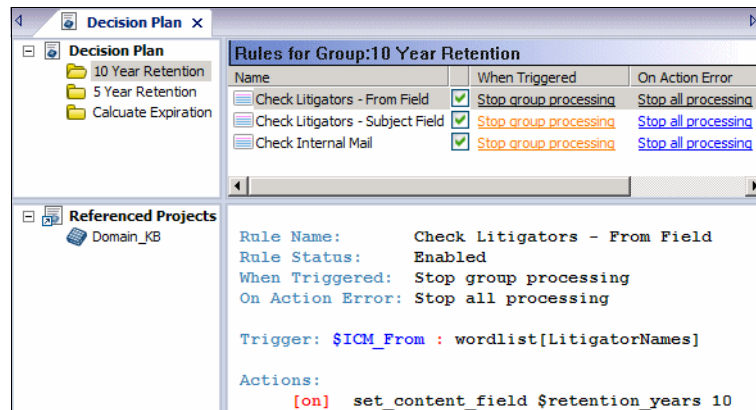


Figure 6-10 Disposal decision plan, group 1, rule 1

Note that the column “When Triggered” is set to Stop group processing. It means that if the rule is successfully triggered, the rest of the rules in the group are skipped (after the rule’s actions are executed).

If the first rule was unsuccessful, the second rule is executed, as shown in Figure 6-11. It assumes the subject of the email appears in the field **ICM\_Subject**, and scans that field to see if it contains any words found in the LitigatorSubjects wordlist. On success we set the **retention\_years** field to 10 and skip the rest of the group.

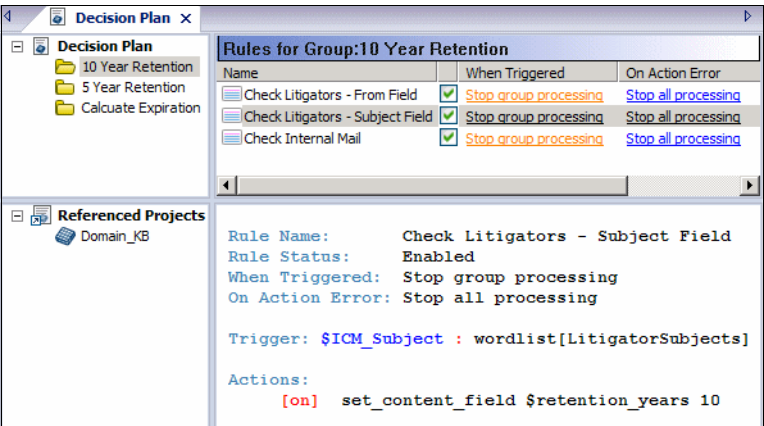


Figure 6-11 Disposal decision plan, group 1, rule 2]

If the second rule was unsuccessful, we execute the third rule, as shown in Figure 6-12 on page 174.

The rule checks whether the mail originated in the Legal Services department (The **ICM\_From** field contains the phrase legal services). It then checks that the body either contains the word complaint, or that it contains the words claim and legal in close proximity (a distance of 5 words or less). The word claim is followed by the wildcard character \* because we also want to check for claims or claiming.

A word surrounded by a tilde (~) character, it means we do not care if it is uppercase or lowercase. If the search is case sensitive, put the word or phrase in single ( ' ) quotes.

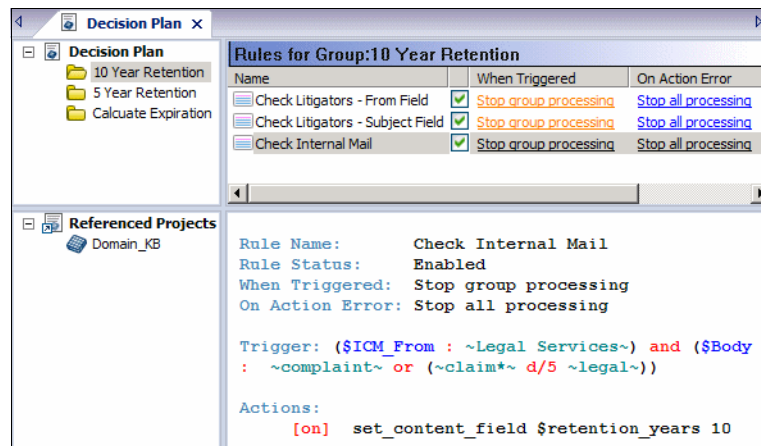


Figure 6-12 Disposal decision plan, group 1, rule 3

## Group 2 rules: Checking for retention period of 5 years

We now move on to execute the first rule of decision plan group 2 rules, as shown in Figure 6-13.

This rule is just an escape trigger with no actions. That is, if a retention period has already been found in part one of the decision plan, skip the rest of group 2 rules.

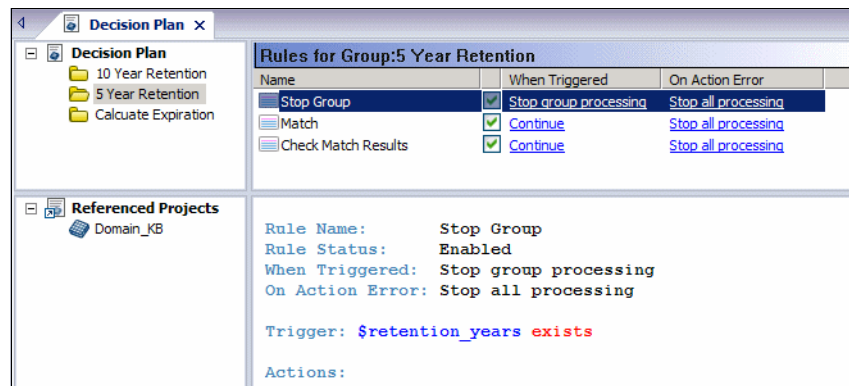


Figure 6-13 Disposal decision plan, group 2, rule 1

Otherwise, proceed to execute the next rule, as shown in Figure 6-14 on page 175.



This rule executes unconditionally (the trigger is true). It sends the email document to a knowledge base associated with the decision plan. The knowledge base matches the document to its statistical profiles, and returns a set of matching categories, sorted by score.

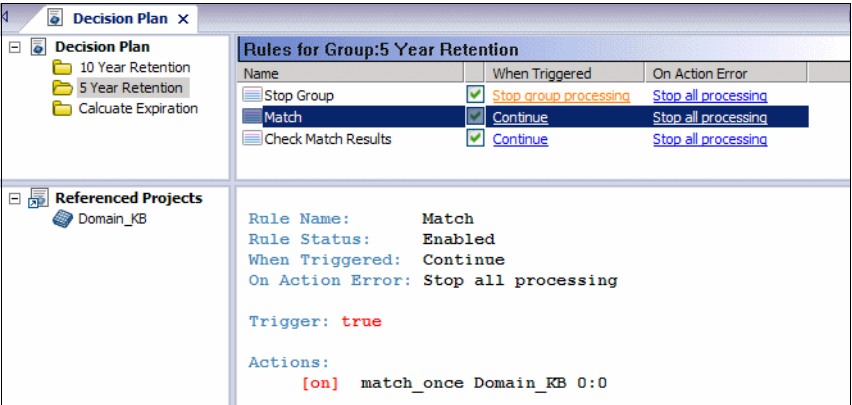


Figure 6-14 Disposal decision plan, group 2, rule 2

There are several options for matching a document. The user can send all document fields for matching, or just a subset. The user can tell the matching to return all results, or ignore low-scoring categories. Under the “Referenced Projects” window, the decision plan might refer to more than one knowledge base, and the document could be matched by each of the referenced knowledge bases.

The results of the statistical matching are inspected in the next rule, as shown in Figure 6-15 on page 176.

The trigger checks that the top-matching category is called *Litigation* and that its score is higher than 0.7 (the highest possible score is 1.0).

The function `cat('Domain_KB', 1)` returns the top-matched category for the knowledge base named `Domain_KB`, and the function `score('Domain_KB', 1)` returns its score.

If the rule is triggered, `retention_years` is set to 5 years.

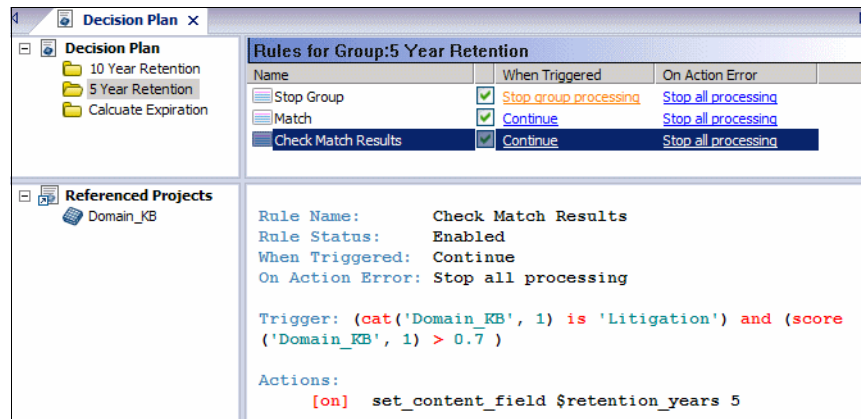


Figure 6-15 Disposal decision plan, group 2, rule 3

### Group 3 rules: Calculating the expiration date

The final group sets an expiration date based on the retention period.

First, if no retention period was found in the previous rules, it sets a default retention period of one year, as shown in Figure 6-16.

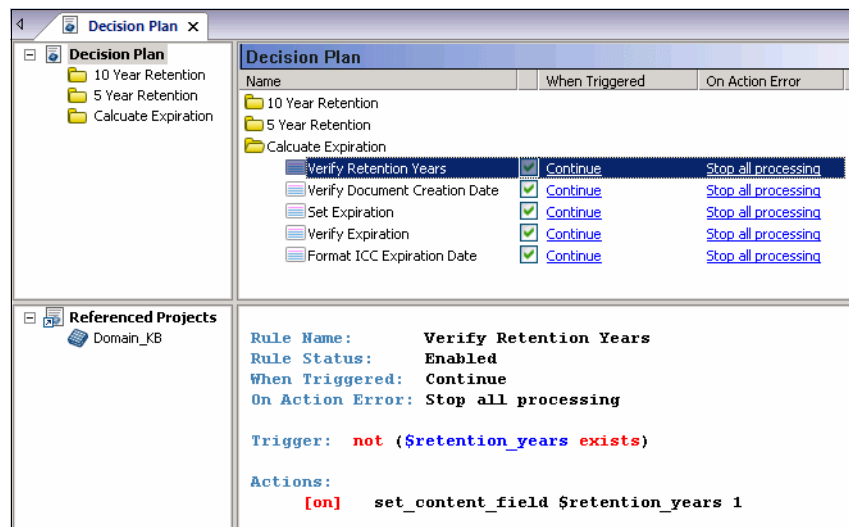


Figure 6-16 Disposal decision plan, group 3, rule 1

Second, if the email document's creation date (ICM\_SentDate) is missing, it sets a dummy creation date equal to today's date, as shown in Figure 6-17.

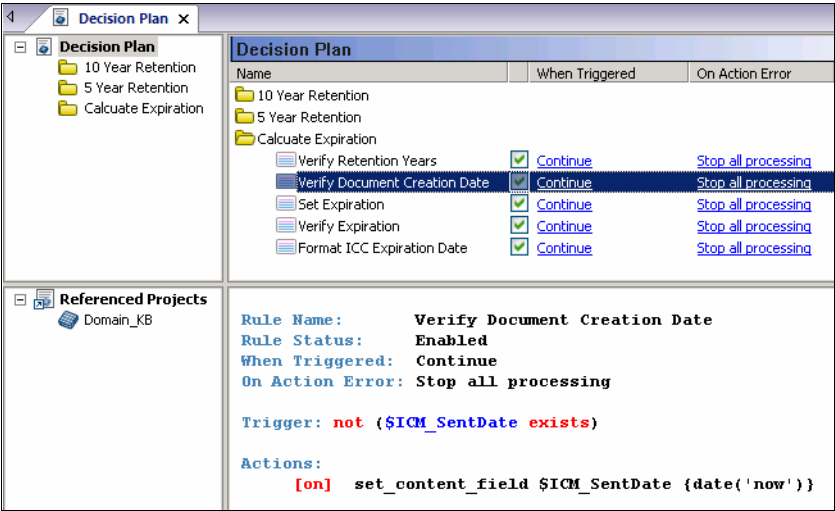


Figure 6-17 Disposal decision plan, group 3, rule 2

Third, it converts retention years to retention days, and calculates an expiration date by adding the retention period to the creation date, as shown in Figure 6-18 on page 178.

In the example, the function `date($ICM_SentDate)` takes a string field that contains a date, and converts it to an internal structure capable of performing date arithmetic. Another variant of this function allows you to specify the date string's format.

Similarly, the function `period($retention_days)` takes a string containing days (and optionally hours, minutes, and seconds), and converts it to a number suitable to performing date operations.

The calculated date is converted back to a string and stored in the field **\$expiration\_date**.

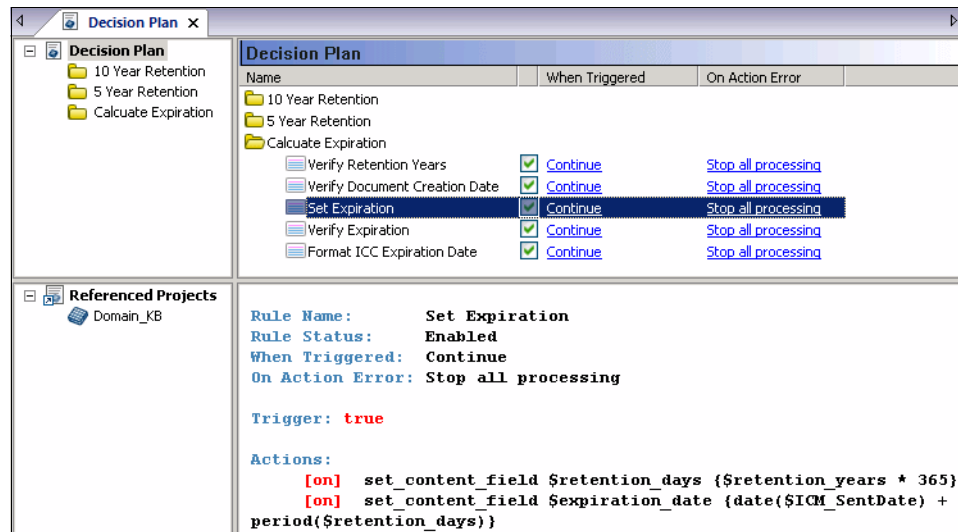


Figure 6-18 [Disposal decision plan, group 3, rule 3]

Next, we verify that the expiration date is valid. Some implementations will refuse to accept an expiration date set in the past. We check for this condition and reset the expiration date to tomorrow, as shown in Figure 6-19.

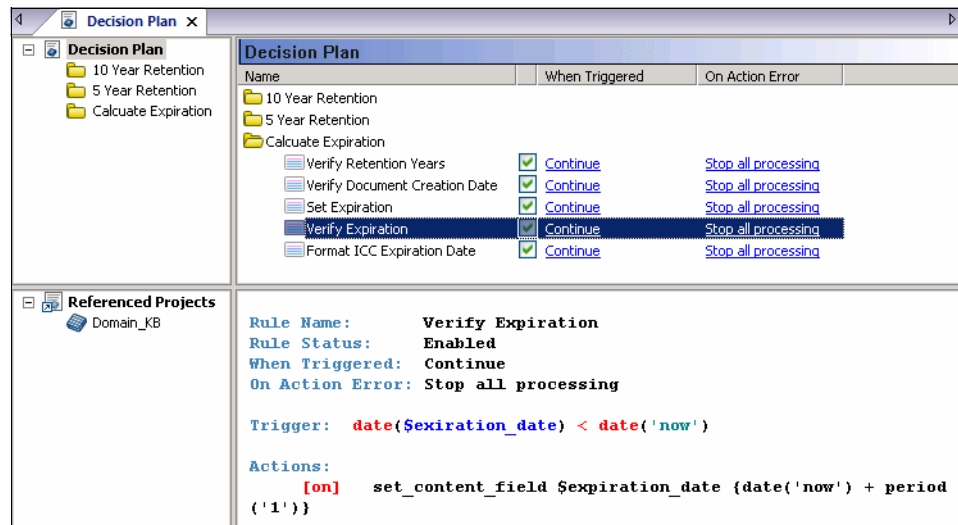


Figure 6-19 [Disposal decision plan, group 3, rule 4]



```

end_group

group on 5 Year Retention
rule on stop_group/stop_all Stop Group
trigger $retention_years exists
rule on continue/stop_all Match
trigger true
action on match_once Domain_KB 0:0
rule on continue/stop_all Check Match Results
trigger (cat('Domain_KB', 1) is 'Litigation') and (score('Domain_KB', 1) > 0.7 )
action on set_content_field $retention_years 5
end_group

group on Calculate Expiration
rule on continue/stop_all Verify Retention Years
trigger not ($retention_years exists)
action on set_content_field $retention_years 1
rule on continue/stop_all Verify Document Creation Date
trigger not ($ICM_SentDate exists)
action on set_content_field $ICM_SentDate {date('now')}
rule on continue/stop_all Set Expiration
trigger true
action on set_content_field $retention_days {$retention_years * 365}
action on set_content_field $expiration_date {date($ICM_SentDate) + period($retention_days)}
rule on continue/stop_all Verify Expiration
trigger date($expiration_date) < date('now')
action on set_content_field $expiration_date {date('now') + period('1')}
rule on continue/stop_all Format ICC Expiration Date
trigger true
action on set_content_field $ICC_ExpirationDate {export_date(date($expiration_date),
'+yyyy-MM-dd\\'T\\'hh:mm:ss:SSS-00:00-')}
end_group

```

---

## 6.5.4 Reproducing disposal decisions made in the past

Decision plans and knowledge bases can evolve over time. In decision plans, new rules can be inserted and others deleted. In knowledge bases, categories can be added and removed. Even when the categories themselves do not change, the behavior of the knowledge base can alter after it receives new feedback.

Such modifications are problematic in the defensible disposal scenario. When running a document through a decision plan or a knowledge base, the disposal recommendation could change if the decision plan or knowledge base had changed. Defensible disposal might require us to reproduce a former disposal decision faithfully. To do that, we must have access to the decision plan or knowledge base as they were when the original decision was made.

The Content Classification server can be configured to keep back versions of all the decision plans and knowledge bases it controls. It is possible to import the correct version into the Classification Workbench and demonstrate the disposal decision it made for a particular document at a particular point in time.

To enable keeping back copies of a Decision Plan, select **back up automatically** in the decision plan properties dialog of the Content Classification

Management Console, as shown in Figure 6-21. The knowledge base properties dialog has a similar option.

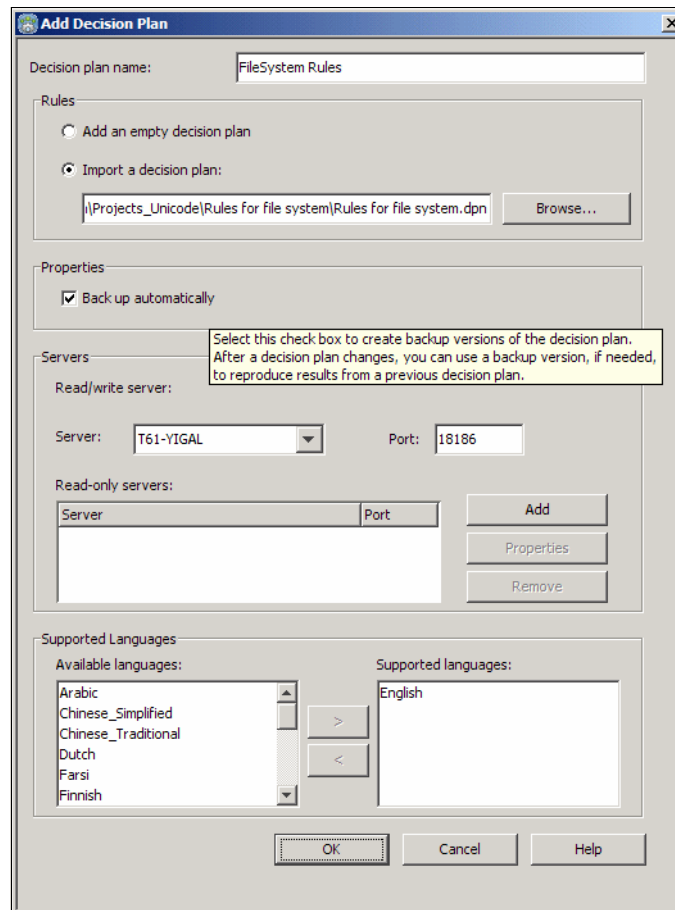


Figure 6-21 Keeping back versions of a decision plan

See the following Technote for practices useful in Content Classification document retention:

<http://www.ibm.com/support/docview.wss?uid=swg27022095>

## 6.6 Generating new facets with Content Classification

Content Classification has many tools and methods for generating new metadata, often known as facets. Several examples are listed in this section. In

the first example we use knowledge bases. The other examples use decision plan rules.

### 6.6.1 Generating facets for multiple taxonomies

Consider a set of documents that can be broken down by product: a bank loan, a mortgage, and a checking account. There might be many other ways to categorize the same data. For example, you can break the documents down by geographic location, by sentiment analysis, and so on. Each grouping of the documents uses an independent criterion unrelated to other groupings.

Content Classification can capture these different dimensions by training several different knowledge bases from the same set of documents, each time splitting and categorizing the documents according to a different criterion. An Content Classification decision plan can match a single document against all the knowledge bases, and store the match results of each knowledge base into a separate facet or metadata field. An example is shown in Figure 6-22 on page 183.

The first three actions match the document to the department, geography and sentiment knowledge bases.


The fourth action copies the best results returned from the Department\_KB (scoring above 0.7) to the multivalued **Department** field.

The fifth action copies the top category returned from the Geography\_KB to the **Location** field.



The sixth action sets the **Sentiment** field. If the top category of the Sentiment\_KB is **OK**, this field is set to **Positive**; otherwise it is set to **Negative**.

Rules for Group:Taxonomies

Name	When Triggered	On Action E...
 Match 3 KBs	<input checked="" type="checkbox"/> Continue	Stop all pro...

Rule Name:

Match 3 KBs

Rule Status:

Enabled

When Triggered:

Continue

On Action Error:

Stop all processing

Trigger:

true

Actions:

[on]

match\_once Department\_KB 0:0

[on]

match\_once Geography\_KB 0:0

[on]

match\_once Sentiment\_KB 0:0

[on]

copy\_categories \$Department Department\_KB 0:0.7

[on]

set\_content\_field \$Location {cat('Geography\_KB', 1)}

[on]

set\_content\_field \$Sentiment {if ( cat('Sentiment\_KB', 1) is 'OK' ) then ('Positive') else ('Negative')}

Figure 6-22 Multiple taxonomies example

### 6.6.2 Generating facets using wordlists

Another preferred way to add facets to a document involves the use of wordlists in the decision plan. The generated facet will typically contain the subset of words from the wordlist that were found inside the document.

A wordlist is just a utf-8 encoded text file where each line contains a single word or phrase. Such lists are easy to create and maintain.

To create a new wordlist in the Classification Workbench:

1. Click **Word and string list files** in the Project Explorer, as shown in Figure 6-23 on page 184.

2. Select **New** and type the wordlist's name.

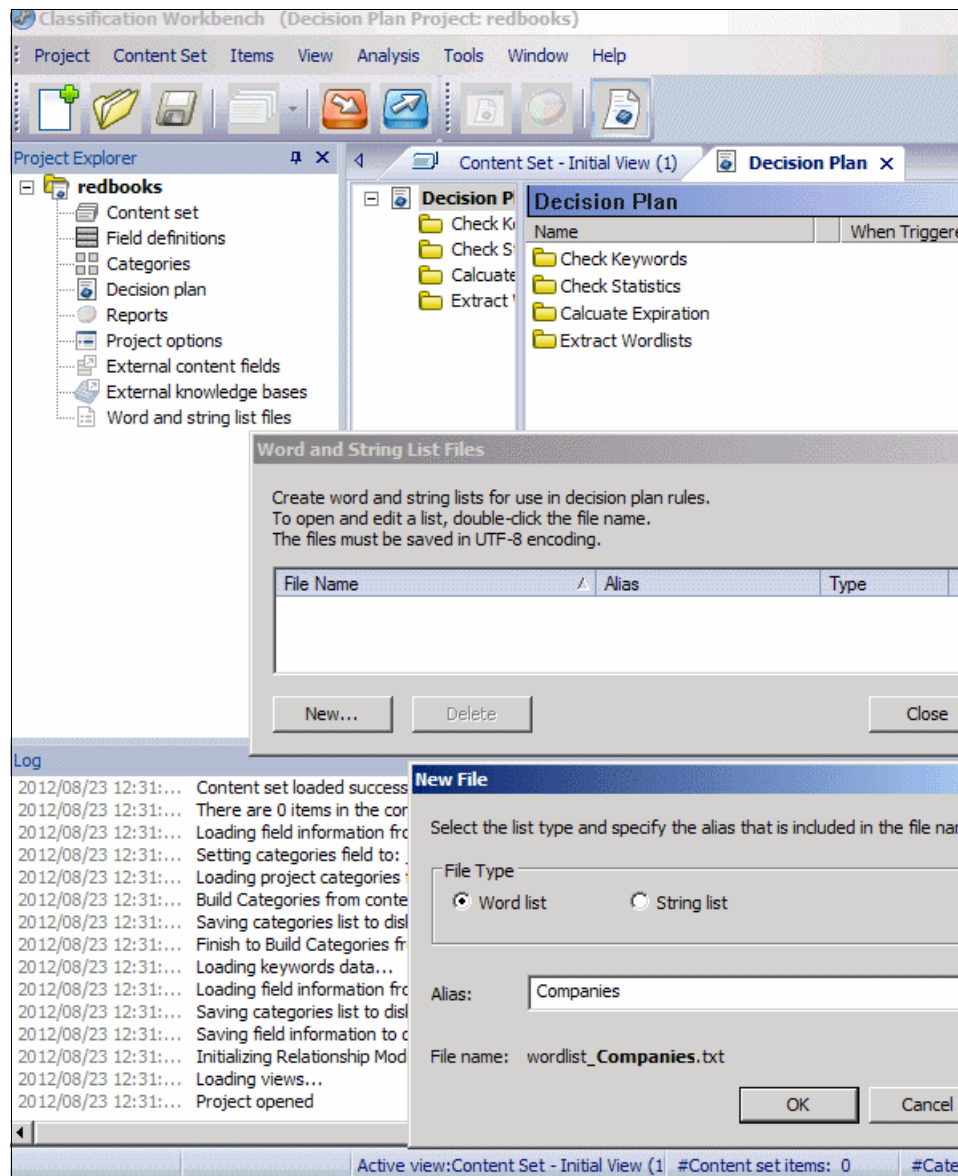


Figure 6-23 Adding a new wordlist

To edit a wordlist:

1. Click **Word and string list files** in the Project Explorer.

2. Double-click the wordlist name you want to edit, as shown in Figure 6-24. This will open the wordlist in an editor.
3. Enter the word list, one per line.

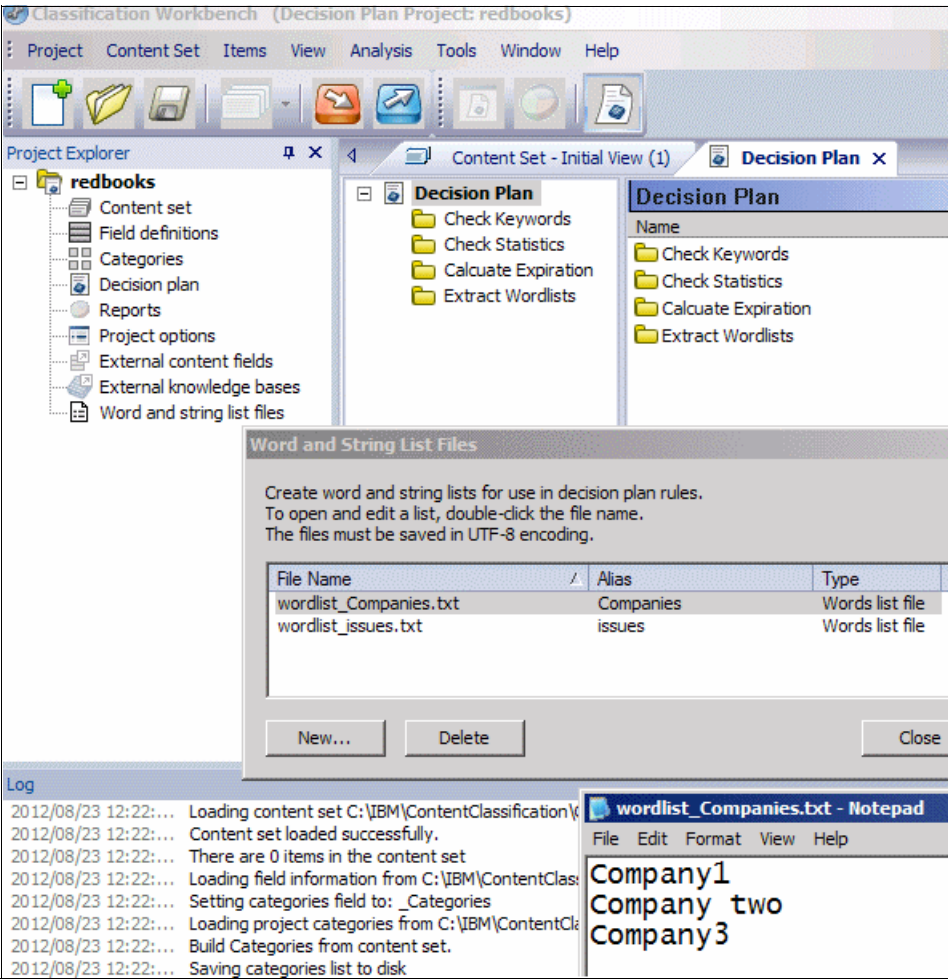


Figure 6-24 Editing a wordlist

The following rule triggers if the word `confidential` and any of the words in the wordlist appears in the same sentence:

```
$Body : ~confidential~ d/s wordlist[Companies]
```

The alternative without a wordlist would look as shown:

```
$Body : ~confidential~ d/s (~Company1~ or ~Company two~ or ~Company3~ )
```

This can be tedious to type when the list is long, and cannot be reused in other rules without retyping.

Wordlists can be used in both triggers and actions, as shown in Figure 6-25. In this example the **extract\_word\_list** action generates a field called **Found\_Companies** containing all the words from the Companies wordlist that were found in the document's **Body** field.

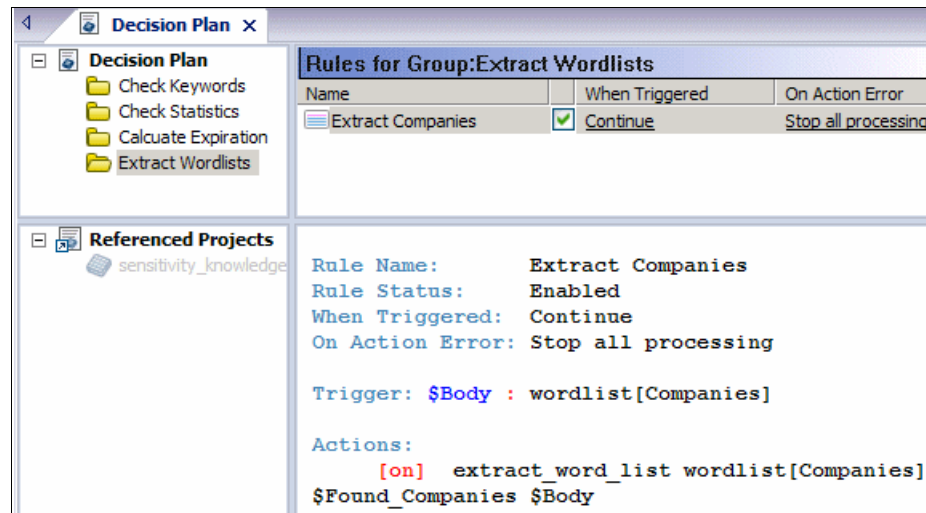


Figure 6-25 Word extraction using a wordlist

## Scalability of wordlist

Wordlists are designed for scalability. A wordlist can contain hundreds or thousands of words (or more, in a 64-bit application). The only limitation is memory size. A wordlist runs much faster than a comparable trigger that uses “a or b or c...” instead.

## Automatic deployment of wordlist

Wordlists allow for automatic deployment of changes, because changes are confined to a textual list of words, avoiding updates to rules and triggers.

Consider this example: A decision plan scans documents for any mention of new chemicals, but the chemicals list changes frequently, necessitating a daily modification and redeployment of the decision plan. Using wordlists, the procedure is easy to automate:

1. Create a wordlist of new chemicals and refer to it in the decision plan.
2. Export the decision plan to an ASCII serialization, a folder where each wordlist resides in a separate file.

3. Write a script and schedule it to run daily. The script overlays the “NewChemicals” wordlist with an updated list obtained from an external source. It compiles the decision plan's ASCII serialization into a single binary (\*.dpn) file using the ExtractDP program. It then deploys the new decision plan to the Content Classification server from a command line program using the Content Classification *import\_decision\_plan* API.
4. Typically a decision plan is configured as a Read-Write server controlling several Read-Only servers. After the script deploys the new decision plan to Content Classification, the Read-Write server propagates the change to the Read-Only servers one by one, thus ensuring that the response time for classification requests does not suffer during the update.

### 6.6.3 Creating facets from regular expressions

Regular expressions allow a fine degree of control over what is extracted to a facet, but they do not scale well. Aside from extracting the regular expression, you can extract internal groups within it. This allows you to embed the expression you want in a specific context; the context is searched for, but will be discarded from the extracted expression.

The Decision Plan action shown in Figure 6-26 on page 188 scans the **Body** field and extracts all numbers surrounded by square brackets. The results are placed into the **Numbers** field, which is possibly multivalued.

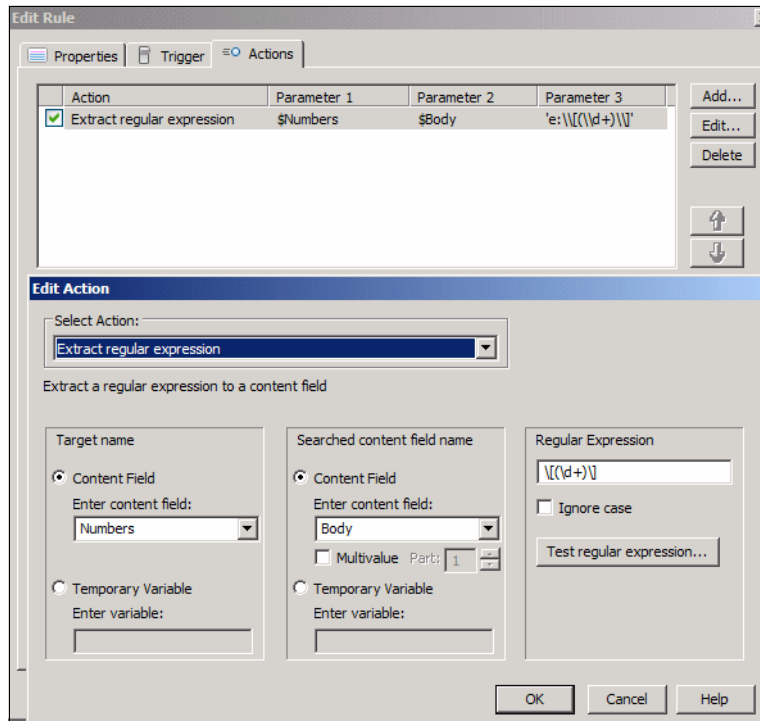


Figure 6-26 Extracting a regular expression

When the regular expression contains internal () grouping elements, the decision plan generates an extra output field for each group. In our example, one extra field is generated.

Assuming the **Body** field contains 123 [1234] abc [12345], then the following fields will be generated:

Numbers: [1234] [12345]

Numbers\_1: 1234 12345

**Numbers** contains the full results. **Numbers\_1** contains the results for group 1 (that is, without the enclosing brackets).

The two fields are synchronized, and it is possible to inspect their items one by one in tandem, using the decision plan's advanced looping feature. For more information, refer to this section of the Content Classification Information Center:

[http://pic.dhe.ibm.com/infocenter/classify/v8r8/index.jsp?topic=%2Fcom.ibm.classify.workbench.doc%2Ft\\_wbg\\_looping\\_rule\\_groups.htm](http://pic.dhe.ibm.com/infocenter/classify/v8r8/index.jsp?topic=%2Fcom.ibm.classify.workbench.doc%2Ft_wbg_looping_rule_groups.htm)

## 6.6.4 Creating facets with Content Classification user hooks

Content Classification decision plans allow users to define and deploy their own decision plan actions through a hook mechanism. A triggered hook receives a set of document fields from the decision plan, and returns another set of fields. The results are added to the document by the decision plan as new fields, or used for updating existing fields.

There are two types of hooks:

- ▶ In-process hooks written in C or Java  
These hooks are deployed by adding a shared library or Java class to the Content Classification **AddOn** folder.
- ▶ Out-of-process hooks that invoke a script or a program  
The script or program resides in the **AddOn** folder.

Figure 6-27 on page 190 illustrates a simple Java hook that does a map lookup. The decision plan sends a name to the Demo hook and the hook returns an address, which the decision plan adds to the document as a field called **Address**. The Demo class implements methods with the standard names of **Init** to initialize the hook and **Run** to run a document through it.

```
// Demo hook for map lookup.
import java.util.Vector;
import java.util.HashMap;

public class Demo {

    private static HashMap<String,String> s_map = new HashMap<String,String>();

    // Initialize the hook
    public static String Init(String addOnDir)
    {
        // ToDo: read the map from a file in the addOnDir folder
        s_map.put("John", "24 Mayfair Drive");
        s_map.put("Mary", "100 Ocean Parkway");
        return ""; // no error message
    }

    // run the hook for a document
    public static int Run(String context, Vector in_key, Vector in_val,
        Vector out_key, Vector out_val, Vector error)
    {
        // if using more than one map, give its name in the context parameter

        // for now, accept one name and return one address
        if (in_val.size() != 1)
            return 0;

        String name = (String) in_val.get(0);
        String address = s_map.get(name);
        if (address != null)
        {
            // return the found address in a field called "Address"
            out_key.add("Address");
            out_val.add(address);
        }
        return 0;
    }

    // declare the hook as thread-safe
    public static void ThreadSafe()
    {}
}
```

Figure 6-27 A simple Java hook

After adding the compiled Demo.class to the Content Classification **AddOn** folder and restarting Content Classification, you can invoke the hook from the decision



plan as shown in Figure 6-28. The decision plan passes the document's **Name** field to the Demo hook, and an (unused) context string.

Rules for Group:Extract Wordlists

Name	When Triggered	On Action Error
Extract Companies	<input checked="" type="checkbox"/> Continue	<a href="#">Stop all processing</a>
User Hook	<input checked="" type="checkbox"/> Continue	<a href="#">Stop all processing</a>

Rule Name: User Hook

Rule Status: Enabled

When Triggered: Continue

On Action Error: Stop all processing

Trigger: \$Name exists

Actions:  
[on] use\_callback J:Demo context "Name";

Edit Rule

PropertiesTriggerActions

Action	Parameter 1	Parameter 2	Parameter 3	
<input checked="" type="checkbox"/> Use callback	J:Demo	context	"Name";	<div>Add... Edit... Delete</div>

Select Action:  
Use callback

Invoke callback function in memory

Hook name

Type: Java API

Name: Demo

Context string

☐ Content field  
Enter content field:  

Multivalue Part: 1

☒ Other data sources  
Type: String  
Value: context  

Edit value...

(Optional) fields to pass

Name

Select additional fields

OKCancelHelp

Figure 6-28 Calling a hook from the decision plan

To test the hook, you add a document to the Classification Workbench project and run it through the decision plan, as shown in Figure 6-29.

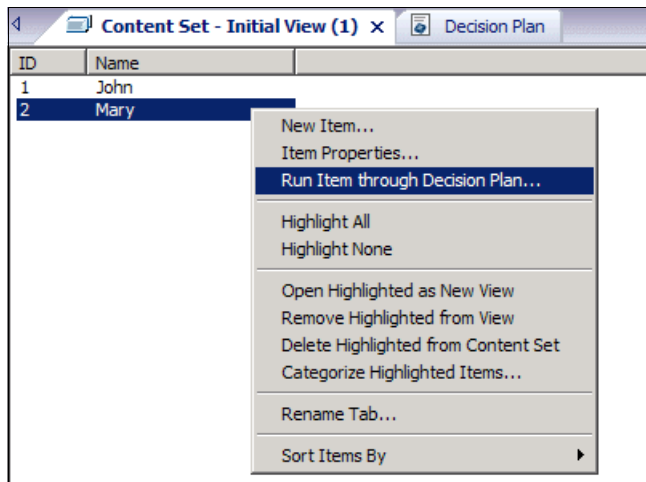


Figure 6-29 Testing the hook

The Analyzed Item dialog in Figure 6-30 shows the new **Address** field that was added by the hook.

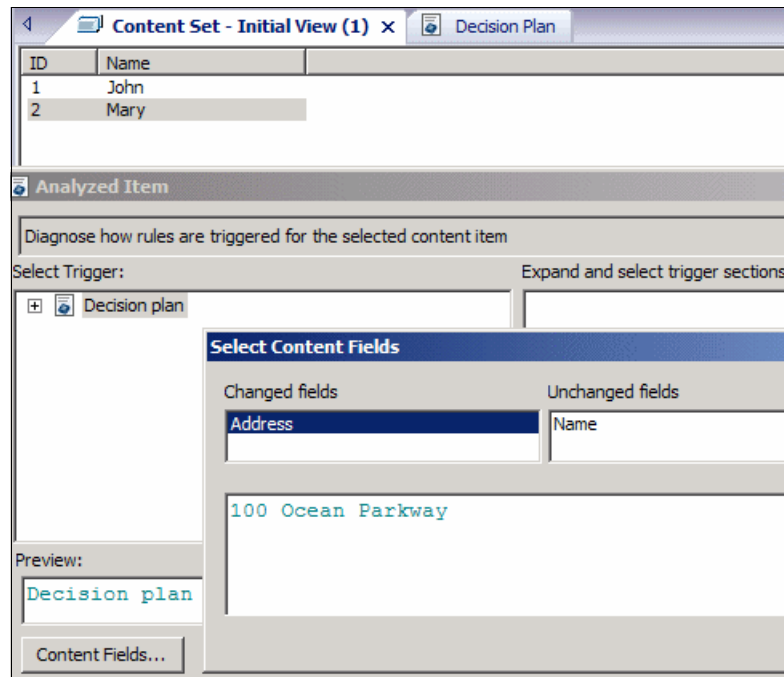


Figure 6-30 Test results

### 6.6.5 Creating facets with the UIMA client hooks

Content Classification has an in-process user hook available ready for immediate use that connects to UIMA servers. You can use it to pass document fields to UIMA annotators and import the results into the document as new facets.

Note that if your solution includes IBM eDiscovery Analyzer, you might prefer to run the annotation process from the repository. Running the annotation process from Content Classification can slow down the process of classification and injection into the repository. However, if you do not call UIMA from Content Classification, the annotations discovered by UIMA will not be able take part in Content Classification's classification decision, which can impact routing accuracy.

## **6.7 Reviewing and auditing archived emails and documents**

The following section explains how to inspect the repository and list Content Classification results, and how to accept or reject Content Classification results.

### **6.7.1 Reviewing results of Content Collector's automatic classification**

The main tool for reviewing the output of Content Collector's Content Classification task is the Content Classification's Classification Center. This tool

connects directly to a FileNet or CM8 repository and allows the user to inspect the classification results. Figure 6-31 shows an example of the review window.

IBM Content Classification - Classification Center

Help | About | Log Off

Configuration Dashboard Review Add Document

Review classification decisions and either accept the suggested classification actions or reclassify content, if necessary. [Learn more...](#)

Total number of reviewed documents: 0 since 1/1/00 12:00 PM  
Page 1 out of 346  
Total number of documents: 3456

[Content to Review](#)

Status	File Name	Applied Actions
<input type="checkbox"/>	RTF_essential	File the document into IBM FileNet Content Manager folder "Classified/Prop65".
<input type="checkbox"/>	RTF_ruse muskellunge inking makeup venture oil Andover sphinx	File the document into IBM FileNet Content Manager folder "Classified/Prop65".
<input type="checkbox"/>	RTF_Xhosa rivalry possible audiovisual Christensen Logan evanescent shipshape	File the document into IBM FileNet Content Manager folder "Classified/Prop65".
<input type="checkbox"/>	RTF_Wesleyan Hamal Methodist daisy Roentgen prow risk stalk Jackman	File the document into IBM FileNet Content Manager folder "Classified/Prop65".
<input type="checkbox"/>	RTF_	File the document into IBM FileNet Content Manager folder "Classified/Prop65".
<input type="checkbox"/>	RTF_bondsmen impracticable rotund AR enforce AK	File the document into IBM FileNet Content Manager folder "Classified/Prop65".
<input type="checkbox"/>	RTF_Hudson blurring watchmen boldface upslon clothesbrush chamfer crewcut	File the document into IBM FileNet Content Manager folder "Classified/Prop65".
<input type="checkbox"/>	RTF_Livemore enchantress metaphor	File the document into IBM FileNet Content Manager folder "Classified/Prop65".
<input type="checkbox"/>	RTF_OSHA tenuous mimetic playwrighting	File the document into IBM FileNet Content Manager folder "Classified/Prop65".
<input type="checkbox"/>	RTF_	File the document into IBM FileNet Content Manager folder "Classified/Prop65".

**RTF\_essential (23 KB)**  
Please see attachment for the list of my questions From emcryu1@phoenix.Princeton.EDU\lncancyamm Thu Feb 3 23:26:26 1994 remot.... [View full document](#)  
[Hide details](#)

IBM FileNet Content Manager document ID: {5CC0DB58-55D6-5589-A6A6-02F03FCA466A}  
File version: 1.0  
Input folders: /NewCSN; /Classified/Prop65;

**Applied classification actions**  
The following actions were taken during the classification process:  
File the document into IBM FileNet Content Manager folder "Classified/Prop65".

**Classification Details**  
Classification date:  
06/09/2012 12:10 PM  
Triggered rules:  
P8 actions / File to folder  
Added and modified content fields:  
myCat: Prop65;  
Knowledge bases and categories  
**online retail**  
Prop65 (1.09%)  
Return Policy (0.955%)  
Shipping Charges (0.927%)  
Account Info (0.907%)  
Selection (0.856%)  
Order Cancellation (0.818%)  
Secure Order (0.81%)  
Product Search (0.8%)  
Bridal Registry (0.726%)  
Gift Wrap (0.709%)

Figure 6-31 Review window

The user selects which items to review by setting options in the Classification Center filter dialog. For example, when Content Collector uses Content Classification to archive emails to a FileNet folder, you can set the filter to display specific FileNet folders as shown in Figure 6-32.

**Content to Review**

Specify options to limit the documents that you review. You must specify at least one start folder. If you save the settings as the default filter, the same settings are used for reviewing documents when the Classification Center is restarted. [Learn more...](#)

▼ Start folders —

Include documents in these folders and subfolders.

Classified Browse... Add Start Folder

► Skip folders —

► Document classes —

► Document class properties —

► Document class property values —

► Modification Date —

► Expiration date —

► Number of documents —

► Classification status —

☐ Do not use the classification status to filter documents

☒ Include only documents that were not reviewed after the last classification.

☐ Include only documents that were not previously classified or reviewed

☐ Save as the default filter

Clear Form Save Cancel

Figure 6-32 Classification Center filtering results by FileNet folder

The classification status flag allows you to exclude documents that have already been reviewed.

There are many other filtering options. In the defensible disposal scenario where Content Classification sets an expiration date, you can filter by this date, as shown in Figure on page 198

**Content to Review**

▼ Start folders —  
Include documents in these folders and subfolders. Add Start Folder  
 Browse...

▶ Skip folders —

▶ Document classes —

▶ Document class properties —

▶ Document class property values —

▶ Modification Date —

▼ Expiration date —  
☐ Do not use the document date to filter documents  
☒ Include only documents that will expire (date format: dd/MM/yyyy)  
Between  and

▶ Number of documents —

Clear Form Save Cancel

*Figure 6-33 Classification Center filtering results by expiration date*

In the records declaration scenario, you can filter by record class as shown in Figure 6-34.

The image shows a 'Content to Review' dialog box with a title bar containing a close button. The dialog is divided into several sections, each with a blue arrow icon and a label: 'Start folders', 'Skip folders', 'Document classes', 'Document class properties', 'Document class property values', 'Modification Date', 'Expiration date', 'Number of documents', and 'Classification status'. The 'Document classes' section is expanded, showing a text input field containing 'INVOICE', a 'Browse...' button, and an 'Add Document Class' button. At the bottom right of the dialog are three buttons: 'Clear Form', 'Save', and 'Cancel'.

Figure 6-34 Filtering results by record class



In CM8 repositories, the selection criteria are somewhat different, as shown in Figure 6-35.

The screenshot shows a dialog box titled "Content to Review" with a close button (X) in the top right corner. The dialog contains several sections for filtering content:

- Item types:** A section with three radio button options:
  - ☐ Review only documents of the following item types:
  - ☐ Do not review documents of the following item types:
  - ☒ Do not filter documents based on their item type.
- Include Category Folder:** A text input field.
- Exclude Category Folder:** A text input field.
- Document attributes:** A text input field.
- Attribute values:** A text input field.
- Modification Date:** A text input field.
- Expiration date:** A text input field.
- Number of documents:** A text input field.
- Checked out items:** A text input field.
- Classification status:** A text input field.
- Save as the default filter:** A checkbox.

At the bottom right of the dialog, there are three buttons: "Clear Form", "Save", and "Cancel".

Figure 6-35 Classification Center filtering options for CM8

## 6.7.2 Manual audit and feedback

Manual audit is a process where a human corrects or confirms the results of automatic classification. Such an audit can change the category of the reviewed document, but more importantly, it can be reflected back to the knowledge base as feedback, resulting in a modification of the statistical profiles, so that future classification will be more accurate.

In many deployments, the nature of the classified data changes gradually over time, and feedback is one method of incorporating this change into the knowledge base. Another method is to explicitly add new content examples in the Classification Workbench and retrain the knowledge base from scratch.

### **6.7.3 Deferred feedback**

In some use cases, users prefer to provide feedback at their leisure, but want it to take effect at a designated time and not immediately. This is particularly true in the defensible disposal use case, which requires tight control of change schedules. In these scenarios it is advisable to use the “deferred feedback” feature, which will collect the online feedbacks but postpone applying them to the knowledge base until a time of the user’s choosing.

To enable deferred feedback, set the knowledge base's feedback option to “defer processing” in the Content Classification Management Console, as shown in Figure 6-36.

The screenshot shows the 'Add Knowledge Base' dialog box with the following sections:

- Knowledge base name:** A text field containing 'Online Retail'.
- Statistics:** A group box containing three radio buttons:
  - ☐ Add an empty knowledge base
  - ☒ Import a knowledge base (KB file):
    - ☒ Import from the Management Console computer: A text field containing 'ion Workbench\Projects\_Unicode\online retail\online retail.kb' and a 'Browse...' button.
    - ☐ Import from the Data Server computer (recommended for large files): An empty text field.
- Properties:** A group box containing:
  - ☐ Use a cache: A 'Cache size:' text field.
  - ☐ Back up automatically
  - ☐ Associate learning data (SARC file): A 'Retrain frequency:' text field.
  - A label 'Select learning data to import with the knowledge base:' above an empty text field and a 'Browse...' button.
  - Feedback:** A dropdown menu currently showing 'Defer processing'.
- Servers:** A group box containing:
  - Read/write server:** A 'Server:' dropdown menu showing 'T61-YIGAL' and a 'Port:' text field showing '18285'.
  - Read-only servers:** A table with 'Server' and 'Port' columns, currently empty, and buttons 'Add', 'Properties', and 'Remove' to its right.
- Supported Languages:** A group box containing:
  - Available languages:** A list box with 'Arabic', 'Chinese\_Simplified', 'Chinese\_Traditional', 'Dutch', 'Farsi', and 'Finnish'.
  - Navigation buttons '>' and '<' between the two language lists.
  - Supported languages:** A list box currently containing 'English'.

At the bottom are 'OK', 'Cancel', and 'Help' buttons.

Figure 6-36 Defer processing of knowledge base feedback

## 6.7.4 Pitfalls of feedback

Content Classification applications can gather feedback as a side result of user actions. This feedback updates knowledge base statistics with new data. Giving feedback to the knowledge base is a sensitive operation, however. Use care to ensure the performance of the knowledge base is enhanced.

To provide useful feedback, inspect a random subset of the documents processed by the Content Classification task. It is important that you do not simply provide corrections to wrongly classified documents, but that you also confirm correctly classified documents.

Make the order of feedback as random as possible. If you give feedback to a large batch of documents, they should be from various categories. The reason for this is because new feedbacks count more heavily than old ones, so feedback for a single category will give it undue prominence.

Ideally, select at random a percentage of all the documents being classified by the Content Classification task, and copy them to an audit folder sorted by time of arrival. You can then proceed to give feedback from that.

The decision plan can maintain this audit folder for you. A sample rule to do that is displayed in Figure 6-37. The decision plan rule shows how to set aside a percentage of the classified documents into a special audit folder.

```
Rule Name:      Audit 2 Percent
Rule Status:    Enabled
When Triggered: Continue
On Action Error: Stop all processing

Trigger: ( true/2 ) and (counter[AuditCount] < 500)

Actions:
  [on] File to this IBM FileNet Content Manager folder: 'AuditedDocuments'
  [on] add_to_counter AuditCount daily
```

Figure 6-37 Auditing rule

The rule says this: “If the number of documents audited today has not exceeded 500, then in 2 out of 100 cases add the current document to the list of audited documents.”

The trigger in this rule consists of two parts. The first one, `true/2`, evaluates as true 2 percent of the time (`true` by itself evaluates as true 100 percent of the time). The second part of this trigger checks the value of a global counter that the creator of the decision plan named `AuditCount`.

For 98 percent of the time, the rule will fail immediately due to the first condition. In the remaining 2 percent, the counter `AuditCount` will be checked. If it is less than 500, the trigger succeeds and two actions are executed:

- ▶ First action: The **FileNetP8:File** field is set to the value `AuditedDocuments`, which is an indication that Content Collector should move the document to a FileNet folder by that name.
- ▶ Second action: The global counter **AuditCount** is incremented by one. The other parameter on that line, **daily**, indicates that this counter is reset to zero at midnight, so a new count of audited documents is started each day. The alternative is to use **unlimited** instead of **daily**, in which case the counter is not reset until the system is brought down.

### 6.7.5 Using representative datasets

Another way to avoid bias in feedback is to use a new option available in Content Classification 8.8 that maintains a representative dataset in the background. In this method, Content Classification will copy the feedback to a hidden dataset, rather than provide the feedback directly to the active knowledge base. Periodically, the representative dataset automatically trains the knowledge base from scratch.

The dataset maintains a balance between old and new documents, and also between common and uncommon categories. It will automatically purge itself of old feedbacks, unless they represent rare (underrepresented) categories, or have been marked by to be kept indefinitely by the Classification Workbench.

To enable this option, associate a learning data \*.sarc file with the knowledge base when training it in the Classification Workbench, as shown in Figure 6-38.

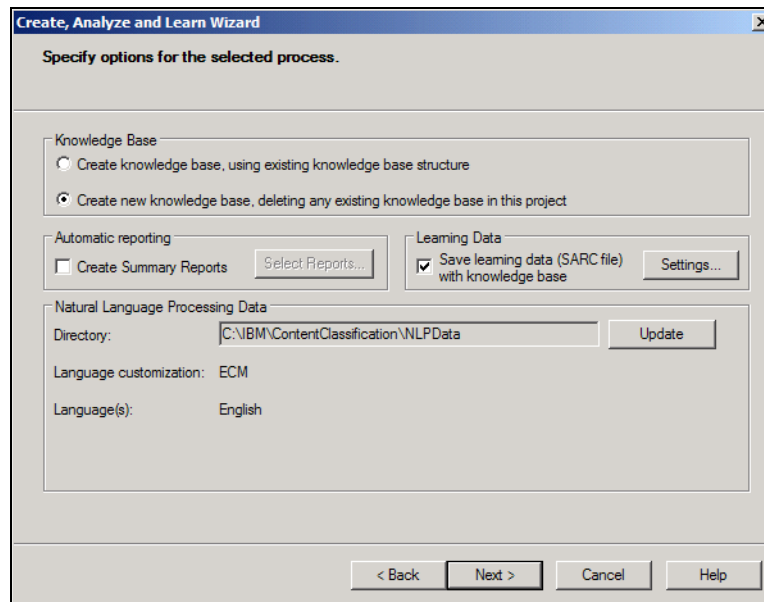


Figure 6-38 Associate knowledge base with SARC file in Classification Workbench

In the Content Classification Management Console, you can deploy a knowledge base with its associated learning data sarc file, as shown in Figure 6-39.

**Add Knowledge Base**

Knowledge base name:

**Statistics**

☐ Add an empty knowledge base

☒ Import a knowledge base (KB file):

☒ Import from the Management Console computer

☐ Import from the Data Server computer (recommended for large files)

**Properties**

☐ Use a cache Cache size:

☐ Back up automatically

☒ Associate learning data (SARC file) Retrain frequency:

Select learning data to import with the knowledge base:

Feedback:

**Servers**

Read/write server:

Server:  Port:

Read-only servers:

Server	Port

**Supported Languages**

Available languages:

- Arabic
- Chinese\_Simplified
- Chinese\_Traditional
- Dutch
- Farsi
- Finnish

Supported languages:

- English

Figure 6-39 Associate knowledge base with SARC file in Content Classification Management Console

For more information about this topic, refer to the Content Classification Information Center sections:

[http://pic.dhe.ibm.com/infocenter/classify/v8r8/index.jsp?topic=%2Fcom.ibm.classify.workbench.doc%2Fc\\_WBG\\_Saving\\_Learning\\_Data.htm](http://pic.dhe.ibm.com/infocenter/classify/v8r8/index.jsp?topic=%2Fcom.ibm.classify.workbench.doc%2Fc_WBG_Saving_Learning_Data.htm)  
[http://pic.dhe.ibm.com/infocenter/classify/v8r8/index.jsp?topic=%2Fcom.ibm.classify.admin.doc%2Fr\\_AG\\_Knowledge\\_Base\\_Properties.htm](http://pic.dhe.ibm.com/infocenter/classify/v8r8/index.jsp?topic=%2Fcom.ibm.classify.admin.doc%2Fr_AG_Knowledge_Base_Properties.htm)

## 6.7.6 Review and feedback through the Classification Center

The main tool for providing feedback to the knowledge base is the Classification Center.

This feedback is given after the Content Collector task route has run, by reviewing the documents injected into the repository by Content Collector.

As you review repository documents in the Classification Center, you can mark some of them for feedback, confirming or changing their classification manually. If Content Classification is configured for feedback, these documents will be sent to the Content Classification server and modify the Knowledge Base. To ensure unbiased feedback, it is advisable to review documents in the special audit folder as explained in 6.7.4, “Pitfalls of feedback” on page 202.

The Classification Center dialog shown in Figure 6-40 is used for reclassifying documents in the repository.

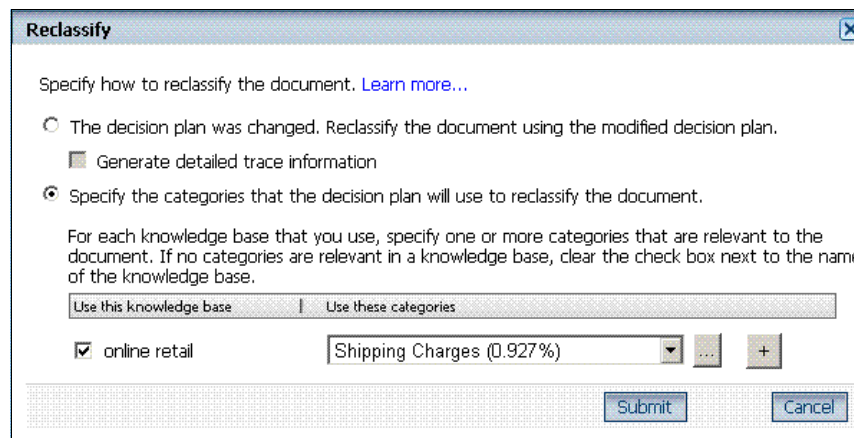
The image shows a 'Reclassify' dialog box with a title bar and a close button. The main text says 'Specify how to reclassify the document. [Learn more...](#)'. There are two radio buttons: the first is 'The decision plan was changed. Reclassify the document using the modified decision plan.' with a checkbox 'Generate detailed trace information' below it; the second is 'Specify the categories that the decision plan will use to reclassify the document.' which is selected. Below this, a text box explains: 'For each knowledge base that you use, specify one or more categories that are relevant to the document. If no categories are relevant in a knowledge base, clear the check box next to the name of the knowledge base.' There is a table with two columns: 'Use this knowledge base' and 'Use these categories'. The first row has a checked checkbox for 'online retail' and a dropdown menu showing 'Shipping Charges (0.927%)'. To the right of the dropdown are three buttons: an ellipsis (...), a plus (+), and a minus (-). At the bottom right are 'Submit' and 'Cancel' buttons.

Figure 6-40 Classification Center reclassification and feedback

Classification Center has an option to provide a random sample of documents for review. See the Content Classification Information Center for more information.



### 6.7.7 Inspection and feedback through the Email Client integration

The feedback in 6.7.6, “Review and feedback through the Classification Center” on page 206 is provided *after* Content Collector has processed the document. The process mentioned in 6.3.4, “Working with Content Collector email client integration” on page 165, is an alternative method to provide the feedback *before* Content Collector processes the document.

The system can be configured so that a percentage of the inspected documents are selected for feedback. Because all documents must pass through the user-inspection stage, a useful sample is obtained and the potential for feedback bias is reduced. However, if users can select the data they choose to work on, then bias is not completely eliminated. In such cases you can elect to implement feedback through a representative dataset as described in 6.7.5, “Using representative datasets” on page 203.

## 6.8 Use case 2: Email archiving with content classification

Use case 2 sets an expiration date on all emails that are injected into the repository, based on unstructured content. To analyze this content, it deploys an Content Classification decision plan.

In 6.4.1, “Value-based archiving and defensible disposal” on page 167, we demonstrate how a decision plan can calculate the expiration date directly. In use case 2 the approach is somewhat different: the decision plan does not return an expiration date, but a MailType category. It is up to Content Collector to match that category with a suitable retention period, and calculate the expiration date based on that period.

### 6.8.1 The decision plan

The following decision plan finds the email type by using a combination of rules and statistics. There are four possible email types. In ascending order of importance they are: personal mail; non-sensitive business mail; sensitive business mail; and critical business mail.

Rule 1 matches the email document against a knowledge base that contains two categories: Business and Personal (see Example 6-2 on page 208).

**Tip:** Content Classification 8.8 is available for immediate use, with a demonstration knowledge base project that distinguishes various business categories from personal categories (located at `<ICM_SERVER_HOME>\Classification Workbench\Projects_Unicode\Personal vs Business Content`).

However, for better accuracy you might prefer to train a new one based on your specific content.

---

#### *Example 6-2 Rule 1*

---

```
Rule 1: Match
Trigger: true
Actions:
    [on] match_once MailType_KB 0:0
```

---

If the best matched category indicates a personal mail, set the MailType accordingly and stop processing the document (Example 6-3).

---

#### *Example 6-3 Rule 2*

---

```
Rule 2: Check for Personal
When Triggered: Stop all processing
Trigger: cat('MailType_KB', 1) is 'Personal'
Actions:
    [on] set_content_field $MailType Personal
```

---

Otherwise, check whether the email body or subject contain sensitive keywords. If none are found, set the email type to regular Business and stop processing the mail (Example 6-4).

---

#### *Example 6-4 Rule 3*

---

```
Rule 3: Check Sensitive Words
When Triggered: Stop all processing
Trigger: not (($Body : wordlist[SensitiveWords]) or ($ICM_Subject :
    wordlist[SensitiveWords]))
Actions:
    [on] set_content_field $MailType Business
```

---

If sensitive keywords are found, control passes to the next rule. Check whether, in addition to the sensitive keywords, the sender or recipient contain sensitive email addresses. If found, set the email type to Critical and stop processing the mail (Example 6-5 on page 209).

#### *Example 6-5 Rule 4*

---

Rule 4: Check Sensitive Address

When Triggered: Stop all processing

Trigger: (\$ICM\_To contains stringlist[SensitiveAddress]) or (\$ICM\_From contains stringlist[SensitiveAddress])

Actions:

[on] set\_content\_field \$MailType Critical

---

Otherwise, set the email type to Sensitive, meaning it has sensitive keywords but no sensitive addresses (Example 6-6).

#### *Example 6-6 Rule 5*

---

Rule 5: Set SensitiveContent

Trigger: true

Actions:

[on] set\_content\_field \$MailType Sensitive

---

This decision plan is simply an example, not a recommendation. In your implementation you might prefer to look for keywords first, and fall back to statistical matching if the keyword search fails.

### 6.8.2 The task route

The Use case 2 task route is shown in Figure 6-41. Content Collector connects to a decision plan named IER2.

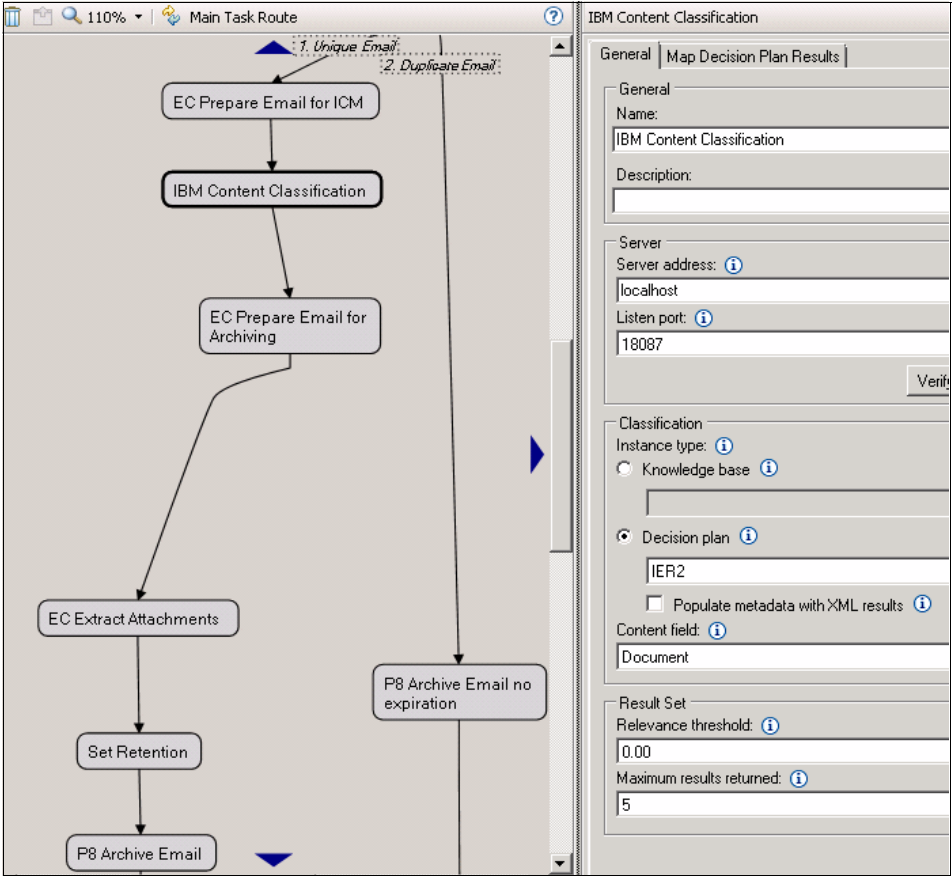


Figure 6-41 Use case 2 - Document classification task

The decision plan **MailType** field is mapped to the Content Collector **Mailtype** field as shown in Figure 6-42.

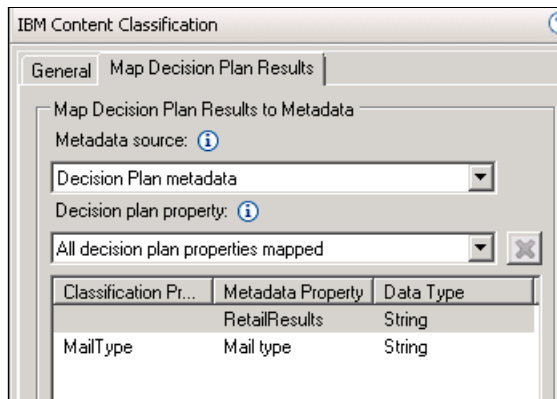


Figure 6-42 Map decision plan output to Content Collector metadata

We define a list in Content Collector to map each mailtype category to the number of months it should be retained, as shown in Figure 6-43.

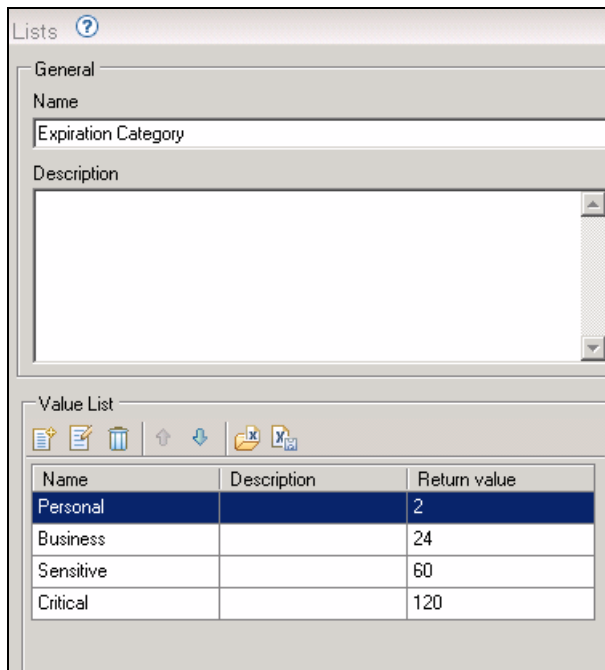


Figure 6-43 Define a retention period for each category

We calculate the expiration in the **Calculate Expiration Date** task as shown in Figure 6-44.

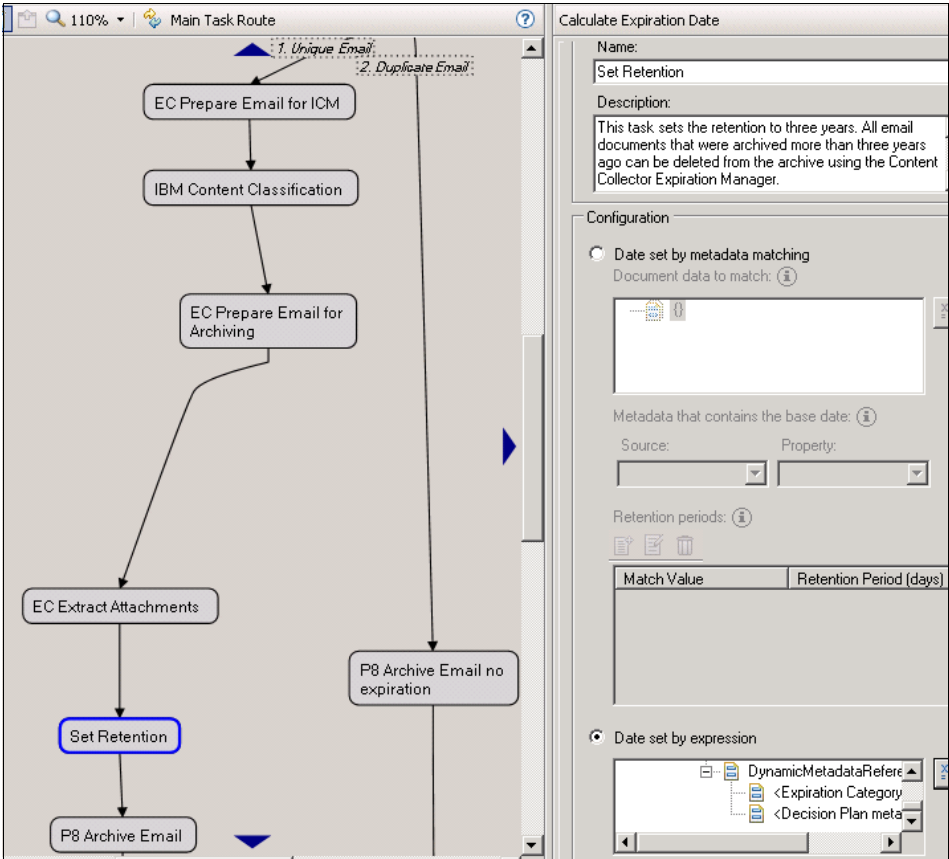


Figure 6-44 Use case 2 - Set retention task

This task looks up the mailtype category in the list and obtains a retention period, which is then added to the email received date to arrive at an expiration date. The expression that performs this calculation is shown in full in Figure 6-45.

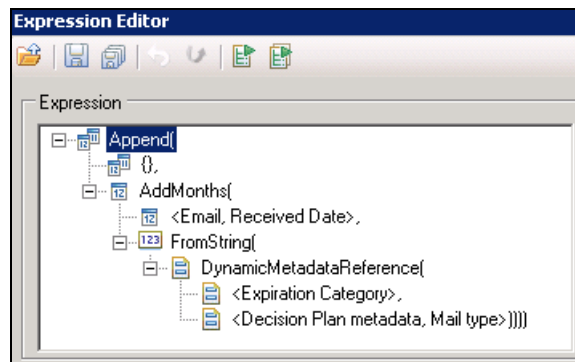


Figure 6-45 Expression to calculate expiration date based on mail type

### 6.8.3 Upgrading to use case 3

The decision plan described in this section is shared by use cases 2 and 3. It is reused in Chapter 7, “Records management integration” on page 221, to illustrate how the classification results provided by Content Classification can be leveraged when integrating record declaration with email archiving.

In 7.4, “Use case 3: Email archiving with records declaration” on page 243, use case 3 illustrates how the same classification results can be used to decide whether to declare records and to determine the record classification.

## 6.9 Considerations and guidelines

This section highlights several common issues that users should be aware of when deploying the Content Classification task route.

### 6.9.1 Preferred practices

Several preferred practices are discussed here.

## Export the necessary fields from Content Classification to Content Collector

The Classification Workbench contains templates for the easy creation of decision plan actions. For example, a user can choose the action template “Move document to FileNet folder”. After the user enters various dialog parameters, Classification Workbench adds the action to the decision plan.

A common misperception is that creating a decision plan action such as “Move document to FileNet folder” will cause the decision plan to execute the move. This is not the case. The decision plan is not a FileNet client, but a generic tool, and the only real action it performs is to add and modify fields in its local copy of the document.

For the document to be updated in FileNet, a naming convention must be established and followed throughout the workflow. In this specific case, the decision plan adds a field called **FileNetP8:File** to the document.

In the Content Collector’s Content Classification task dialog, the user must specify fields that were generated by the decision plan, and map them to Content Collector fields. Further down in the task route, Content Collector will update the document in FileNet using the contents of these fields.

See Figure 6-46 on page 215 as an example of mapping the decision plan’s **FileNetP8:File** field to a Content Collector metadata field called **FileNet\_File**. Later in the task route, the task “P8 File Document in Folder” will use this property to construct a FileNet path for the document.



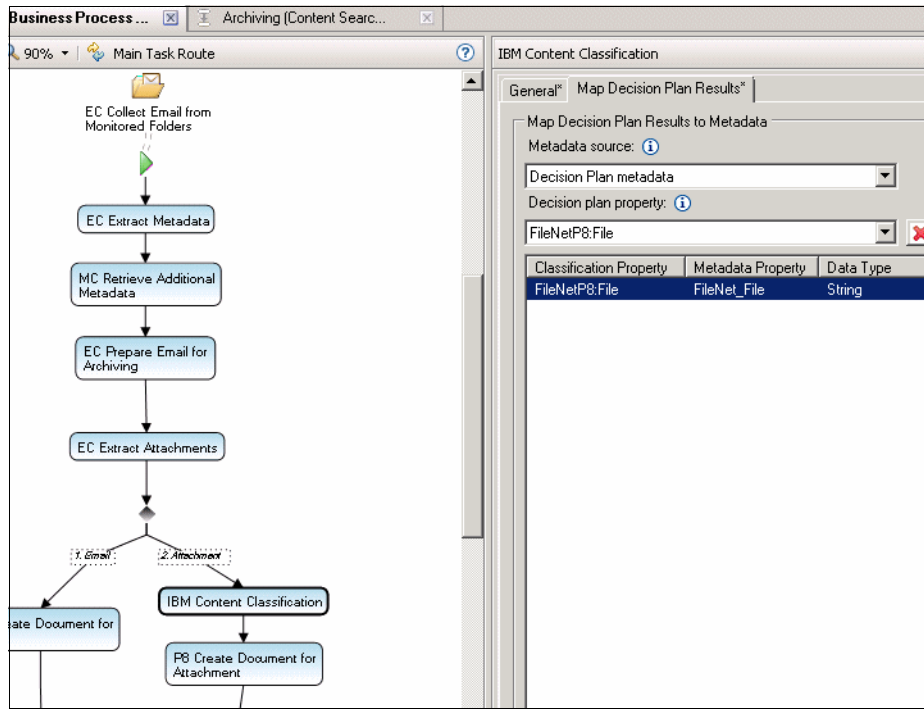


Figure 6-46 Mapping decision plan fields to Content Collector fields

For more information about mapping the decision plan's output to Content Collector fields, refer to this section of the Content Collector Information Center:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/icm/t\\_afu\\_mapping\\_icm\\_properties.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/icm/t_afu_mapping_icm_properties.htm)

## Import the necessary fields from Content Collector to Content Classification

In some cases it will not be sufficient to map the output of Content Classification to Content Collector fields. Instead, you will also need to perform the complementary action of mapping Content Collector fields as input to Content Classification.

In previous versions of Content Collector, a filesystem document was passed from Content Collector to Content Classification as a single buffer without accompanying metadata such as filename and filepath.

Starting with Content Collector 3.0, there is a general method of passing metadata to Content Classification along with the filesystem document. It is

advisable that you use this option when importing from filesystems, so the decision plan can see the filename/filepath and act on them. Sometimes the reason is obvious (for example, the data was already fully or partially classified in filesystem folders, or the filename can be used in a rule) and sometimes it is more subtle (for example, there are patterns in the filename/filepath that are statistically useful).

For accurate classification, try to import and use as much of the metadata as you can. Some metadata fields are now available for immediate use to Content Classification. But in other cases (such as Microsoft SharePoint files), extract them explicitly in the connector and pass them to Content Classification.

For more information about configuring the filesystem metadata, refer to the following section of the Content Collector Information Center:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/index.jsp?topic=%2Fcom.ibm.content.collector.doc%2Ficm%2Ft\\_afu\\_passing\\_metadata.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/index.jsp?topic=%2Fcom.ibm.content.collector.doc%2Ficm%2Ft_afu_passing_metadata.htm)

## Use care when defining task routes that send email to Content Classification

As currently implemented, Content Collector sends a single buffer to the Content Classification API. This buffer should contain the original email with its attachments inside. However, if simply the body of the email is sent to Content Classification, classification will be imprecise because often the content of interest is found in the attachments and not in the body.

When Content Collector extracts emails from email servers, it normally decouples the mail body from its attachments. If your task route uses Content Classification, make sure that this decoupled mail is *not* sent by Content Collector to Content Classification. You must configure the Content Collector task to extract the mail together with its attachments, as shown in Figure 6-47. Note that the **Save document without attachments** box is unchecked.

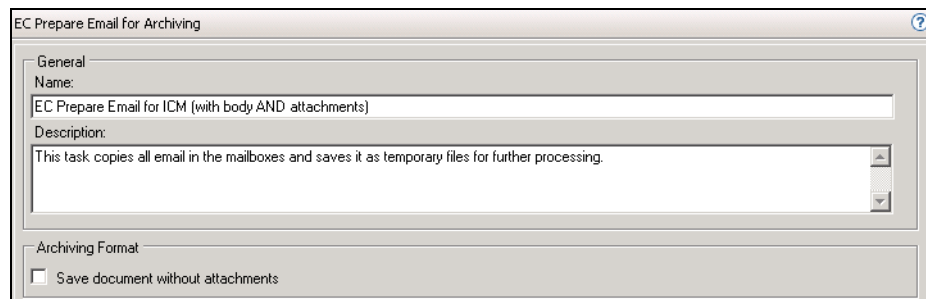


Figure 6-47 Saving body and attachments

Such errors of configuration are difficult to debug, because a decision plan that works fine in the Classification Workbench will produce erratic results when deployed in a misconfigured Content Collector task route.

Content Classification 8.8 contains debugging tools that will dump the data Content Collector sends to Content Classification. To enable debugging, run a script called GatherDebugInfo, and supply a folder name where Content Collector data and processed Content Classification data are dumped. This tool will slow down processing and eat up disk storage quickly, so use it judiciously.

For more information about enabling the debugging tool, refer to the following section in the Content Classification InfoCenter:  
[http://pic.dhe.ibm.com/infocenter/classify/v8r8/index.jsp?topic=%2Fcom.ibm.classify.trouble.doc%2Ft\\_AG\\_debug\\_folder\\_cont.htm](http://pic.dhe.ibm.com/infocenter/classify/v8r8/index.jsp?topic=%2Fcom.ibm.classify.trouble.doc%2Ft_AG_debug_folder_cont.htm)

## Inspect the input document size

When dealing with significantly large files (for example, a 200 MB video file), it is preferable not to send them to content classification. To achieve this, Content Collector should have a decision point check the file size first and sequester it when it is too large. A simple example is shown in Figure 6-48.

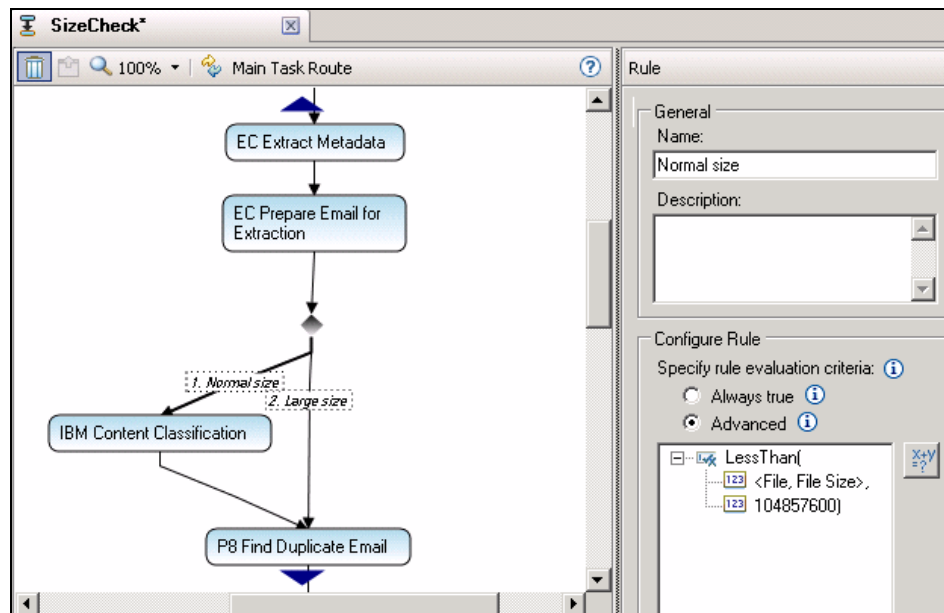


Figure 6-48 Skipping classification of large files

In a more elaborate task route, you can explicitly specify a default value for the metadata that would normally be decided in the decision plan. Example:

- ▶ In the Normal size branch, Content Classification sets a **P8Folder** field to a folder name, and the next task assigns a P8 folder based on that.
- ▶ The Large size branch assigns to a P8 folder named “LargeFiles,” so all large files are routed there automatically.

## 6.9.2 Limitation considerations

Always be aware of the limitations. Several limitations are discussed here.

### Limitations in using the Microsoft SharePoint connector

Microsoft SharePoint streams need to be converted to buffers before calling Content Classification, which might impact the task route speed.

Only one file is sent to Content Classification at a time. When there are several versions of the file, you cannot classify them jointly.

The Content Classification connector can accept Microsoft SharePoint metadata as input, in the same way it accepts filename/path metadata from the File System connector. However, unlike the File System case, Microsoft SharePoint data may contain a significantly large set of unorganized metadata columns, and it is not always practical to define them in Content Collector, so this metadata ends up being ignored.

For a known metadata column, it is easy to configure the Content Collector Microsoft SharePoint connector to pass the field to Content Classification. Note, however, that Microsoft SharePoint has a “propagated properties” feature, whereby the metadata of an MS Office file is not kept as external metadata, but is inserted by Microsoft SharePoint into the document as an internal user property. Such metadata will be ignored by Content Classification unless the Content Classification text extractor is specifically configured to extract it from the document. See the Content Classification text extractor configuration file `<ICM_SERVER_HOME>/Filters/docFilterManager.xml` and the Content Classification InfoCenter for more details.

### Limitations in using the IBM Connections connector

An IBM Connections item is extracted as one or more XML files and optional attachments. To obtain optimal results from Content Classification they should all be classified together, but as in the Microsoft SharePoint case, only one of these files can be sent to Content Classification at a time.

## 6.10 Conclusion

In this chapter we illustrated how to integrate IBM Content Classification with IBM Content Collector solutions. We explained how to use decision plans to calculate document expiration date for defensible disposal. We also described many ways of adding additional metadata (facets) to the content (yet to be archived) with Content Classification. These additional facets can be used for eDiscovery, defensible disposal, records management, and other purposes.





# Records management integration

In this chapter we describe how IBM Content Collector can be integrated with an IBM Enterprise Records solution, both with and without IBM Content Classification. As organizations establish and implement enterprise records and retention management policies, and defensible disposal solutions, integrating archived content with Enterprise Records provides a comprehensive set of capabilities needed for records management.

There are a variety of approaches to integrating an enterprise records solution. In this chapter we identify and discuss examples that illustrate several of the more common use cases.

In this chapter we discuss the following topics:

- ▶ Options for classifying and declaring records
- ▶ Record declaration requirements
- ▶ Basic record declaration from a Content Collector task route
- ▶ Use case 3: Email archiving with records declaration
- ▶ Use case 4: File system archiving with records declaration
- ▶ Considerations and guidelines

## 7.1 Options for classifying and declaring records

It is not always obvious which artifacts coming through an organization should be declared as records. In general, an enterprise records solution will typically address those business artifacts that can be identified as having clear business value and context and that can be mapped to a well-understood and established file plan. You can find more detailed information about records and records management concepts in the Redbooks publication *Understanding IBM FileNet Records Manager*, SG24-7623.

Exactly when records should be declared or why certain business documents should or should not be declared into an enterprise file plan will depend on many different factors related to the particular nature of a business or organization. It might be sufficient to use the native Content Collector expiration manager capabilities by computing a simple expiration date at the time a document is captured to effectively dispose of business content in a defensible and timely manner. However, for more mature organizations that have or require a clear set of retention policies and that understand how those policies should be applied to the various types of records in their organization, declaring records with Enterprise Records provides a more comprehensive and maintainable records management solution.

In this chapter we focus on the mechanics of how to declare records when using Content Collector to capture content for our specific use cases to illustrate some of the more useful concepts and options. As with many complex and integrated product sets, there are a variety of ways to approach a given problem or set of requirements. The examples used in this chapter are meant to illustrate the general concepts, capabilities, and options that can be applied to a wide set of use cases and solutions.

### 7.1.1 Simple retention management versus record declaration

If simple retention will meet the requirements of the solution or use case, using the Content Collector expiration manager capabilities might be sufficient and there might not be a need to declare records into Enterprise Records.

#### **When to use expiration manager**

The following conditions can indicate that simple retention with Content Collector expiration manager without declaring records might be appropriate:

- You can determine the expiration date at the time of capture and that date will never need to be changed or adjusted.



- ▶ You have another means, besides Enterprise Records hold functionality, to place documents on hold for legal or audit purposes.
- ▶ You have not yet developed an enterprise-wide retention policy and file plan.
- ▶ You have a limited set of simple retention rules that can be applied broadly.
- ▶ You have large volumes of documents that can be managed with simple retention.

## **When to declare records**

The following conditions can indicate that records should be declared:

- ▶ You have an enterprise file plan that is based on an enterprise-wide retention policy.
- ▶ You have deployed your file plan in Enterprise Records.
- ▶ You have a way to distinguish various types of records according to their retention requirements.
- ▶ You have a comprehensive set of retention rules and policies that apply to many types of records across the enterprise.
- ▶ You have the need for stricter control and monitoring of record disposition.

Deciding when or when not to declare records is an important part of designing an overall retention management solution. For example, if you are using Content Collector to capture all emails from a journal, you might set a default expiration date. You might assume that unless otherwise stated, the emails collected from the journal will be purged after a fixed amount of time such as three years, and avoid declaration at time of capture. You might then need to declare selected emails as records outside of Content Collector as part of a separate process, such as eDiscovery that would typically happen some time after using Content Collector to capture the email.

Alternatively, if you are capturing files from a file system or from Microsoft SharePoint, and you have access to metadata at the time of capture that clearly identifies the captured item as a record that should be declared, it makes sense to declare the record as part of the Content Collector task route (at the time of capture).

### **7.1.2 Determining classification**

To classify or categorize collected content as records, you need a means of determining the classification or category to assign to incoming content. Depending on the scenario you are implementing, you might or might not have ready access to associated metadata to establish a classification mapping. If sufficient metadata is not available, then Content Classification can be used to

classify unstructured content based on rules and knowledge about the nature of the unstructured content.

### **Determining classification from metadata**

The following scenarios are examples of when metadata might be sufficient to determine record classification. In these cases, there must be some common taxonomy that identifies the type of content.

- ▶ Collecting files from a file system structure or Microsoft SharePoint site where the folder path clearly identifies the type of content
- ▶ Collecting email from mailbox folders where users file email into specific folders based on the type of email
- ▶ Collecting email with direct user input
- ▶ Collecting files from a file system where an automated process includes associated metadata with each document or where the folder path can be used to derive context

### **Determining record type using Content Classification**

The following scenarios are examples of when there is not enough metadata to determine classification. In these cases, use Content Classification to determine classification based on the unstructured content.

- ▶ Collecting email from journals or inboxes where there is no indication from the metadata about what type of email is being collected
- ▶ Collecting files from a file system that has no meaningful path structure or associated metadata
- ▶ Collecting files from multiple Microsoft SharePoint sites where there is no common or established taxonomy that maps to a required classification

## **7.1.3 Use cases and examples**

We present two different use cases in this chapter:

- ▶ One use case illustrates record declaration with classification from Content Classification.
  - Use case: Email archiving with record declaration (using classification from Content Classification).
- ▶ The other use case illustrates record declaration based on classification from metadata alone.
  - Use case: File system collection with record declaration (using metadata for classification).

## 7.2 Record declaration requirements

Content Collector provides the P8 Declare Record task, which can be used in any Content Collector task route to declare a captured document as a record into Enterprise Records. In this section, we describe the general requirements (pre-conditions) in Enterprise Records for declaring a record and the options available for configuring the P8 Declare Record task in Content Collector.

One of the most important aspects of integrating content collection with an enterprise records file plan is determining how the content should be classified in the established file plan. It is often the case that collection scenarios have an element of complexity that makes this challenging. To best understand the options available for a given scenario, we start by describing the essential information required for record declaration in an IBM Enterprise Records implementation.

### 7.2.1 Prerequisites for record declaration

The following list identifies the key elements that must be in place prior to record declaration:

- ▶ Enterprise Records must be installed and configured.
- ▶ The appropriate File Plan Object Store (FPOS) and file plan must be defined and in place.
- ▶ Record classes and their properties must be defined and in place.
- ▶ Documents to be declared must be added to the P8 object store.
- ▶ The document classes you want to declare must be record-enabled.
- ▶ Determine how content should be mapped to P8 document classes in the archive object store; P8 record classes in the target FPOS; and record categories in the target file plan.

### 7.2.2 Content must be archived before declaration

If the content collected by Content Collector has not been archived to a P8 content repository, it cannot be declared as a record in Enterprise Records. One of the following Content Collector tasks must be used to archive content to P8 before you can use the P8 Declare Record task:

- ▶ P8 Create Document

This archives a single document from a Microsoft SharePoint or File System collector.

- ▶ **P8 Create Version Series**

This archives a version series from a Microsoft SharePoint or File System collector.

- ▶ **P8 Archive Email**

This archives a distinct email instance using the Content Collector email data model.

### **7.2.3 Essential information for record declaration**

There are three essential pieces of information that must be provided during record declaration:

- ▶ Target FPOS
- ▶ Target classification
- ▶ Target record class

#### **Target FPOS**

Record declaration involves the creation of a record information object (RIO) in a P8 object store known as File Plan Object Store (FPOS). A given enterprise records solution might have multiple File Plan Object Stores. So, the target P8 object store for record declaration must be identified for record declaration to happen. In Content Collector, the target FPOS is identified by selecting a P8 Connection that has been properly configured.

#### **Target classification**

Record declaration requires that a record is filed appropriately into a file plan. This involves identifying the appropriate record category or record folder in an existing file plan. Where to file a record in the file plan is typically determined by some attribute or characteristic of the record that identifies it for purposes of retention. Records are typically filed based on some meaningful attribute such as a record type, document type, record series code, or record code. Matching this attribute of collected content to the correct file plan container is a key aspect of record declaration. Content Collector provides options for determining the appropriate target record classification. Target classification can be a static value, or can be determined dynamically at run time.

It is possible to file a record into more than one container for classification purposes. However, most solutions do not file records into multiple containers; records are filed into one and only one container according to the attribute or nature of that record. A record must always be filed into at least one container at the time of declaration.

**File plan and classification scheme information:** In Enterprise Records, a file plan is also known as a *classification scheme*. Thus, target classification refers to filing a record into the file plan or classification scheme that is determined by the records and retention policies of the enterprise. See the Redbooks publication *Understanding IBM FileNet Records Manager*, SG24-7623, for more information about file plans and records management configuration.

### Target record class

Record declaration requires that a target record class is identified for each record that is to be declared in Enterprise Records. The number and variety of target record classes is dependent upon the nature of the enterprise records solution and the requirements of the file plan configuration.

Often, but not always, there will be one target record class for each P8 document class. When this is the case, there is a direct mapping between the P8 document class of the archived content and the target record class. Because the Content Collector task route must also have logic that determines which P8 document to create for the content being collected, it can use this same logic to determine the target record class for declaration. If there is no one-to-one mapping between P8 document classes for content and target record classes, then additional logic might be required to determine the target record class.

The target record class defines the metadata that will be stored with the record information object. The mapping from document class to record class will typically include properties that should be maintained as part of the record information during record declaration. Which properties to copy or set during declaration will depend on the specific requirements of the records management solution and the configuration choices of the records implementation. Content Collector provides options for determining record metadata. Record properties can be based on literal values, metadata from source files, from the source P8 document, from task status, or from regular expressions.

## 7.3 Basic record declaration from a Content Collector task route

From Content Collector, record declaration is accomplished by using the P8 Declare Record task in a Content Collector task route. Any content that is collected by Content Collector can be declared as a record after the document has been created in P8.

Figure 7-1 shows a simple task route with the P8 Declare Record task. The following steps occur in this simple task route:

1. The FSC Collector collects files from a file system.
2. The P8 Create Document task adds the files to the P8 content repository.
3. The P8 Declare Record task declares the documents as records.
4. The FSC Post Processing task completes the processing for the collected files.

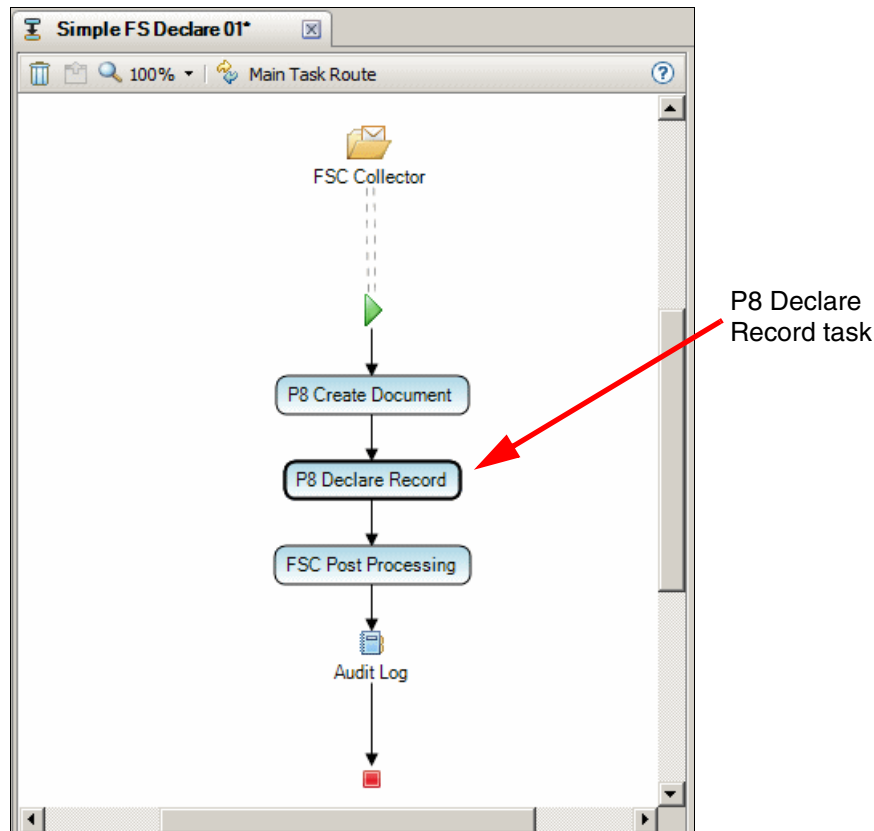


Figure 7-1 Simple task route for file system collection with record declaration

### 7.3.1 Enabling archived documents for declaration

Before you can declare an archived document as a record in Enterprise Records, the document class used for archiving must be record-enabled. This is a simple property setting on a P8 property definition in the content repository. The P8 object store where content is being archived must first be configured with the

Enterprise Records data model (that is, it must be configured as an ROS), and any document classes that are to be declared must have the CanDeclare property default value set to True.

The following configuration must be set to enable records declaration:

- ▶ For archiving email, the Content CollectorMail3 document class or its subclasses should be record-enabled.
- ▶ For archiving files, whichever document classes you are using in the content repository for archiving should be record-enabled. You would typically archive files to your own customer-defined document classes.

See the Redbooks publication *Understanding IBM FileNet Records Manager*, SG24-7623, for more information about record-enabling the content repository object store and record-enabling the document classes to be declared.

### 7.3.2 Content Collector P8 Declare Record task overview

The Content Collector task route toolbox provides a P8 Declare Record task that must be added to a task route to declare archived content as a record. This task can be added anywhere in a task route after the task that archives content to P8 and at a point where all the essential information for declaration can be derived with the configuration options for record declaration. Because a record can only be declared after a document is created in a P8 content repository, properties (metadata) about the P8 document and metadata from the original file captured is available at the time of declaration.

The P8 Declare Record task includes configuration options that determine all the essential information needed for declaration. The configuration options are separated into the following panes:

- ▶ General
- ▶ P8 Connection
- ▶ Configure Classifications
- ▶ Property Mappings
- ▶ Data Correction

Figure 7-2 shows how these panes appear in the Configuration Manager application.

Figure 7-2 P8 Declare Record - task configuration

Each of these panes has different options that you can set. The options for configuring the P8 Declare Record task are described here.

## General

Use General settings to provide the *name* and *description* for the task. This information is useful for the Content Collector task route configuration and is not used for the record declaration itself.

By default, the name is set to the generic “P8 Declare Record.” However, in more complex task routes that have multiple declaration tasks, setting this name to something that will distinguish each of the declare tasks is useful. The description field can be used to provide more details about the purpose, nature, or the specific configuration of the given declare task.



The Name field should be a short string that helps identify what the declare step does in the context of the entire task route. For example, if there is only one declare task, it would be appropriate to leave the default name “P8 Declare Record.” However, if there are multiple declare steps in one task route, it is helpful to provide more meaningful names for each, for example, “Declare Accounting Records” or “Declare Invoice Records.” If the task route is used to capture from a specific content source, you might provide a name that indicates the context, for example, “Declare Email as Records.”

## **P8 Connection**

Use P8 Connection to specify the connection to the File Plan Object Store (FPOS). The connection must already be configured in Content Collector and is selected from the pull-down list. This is one of the essential pieces of information for record declaration. Each P8 Declare Record task can only be configured to declare into one FPOS. Only P8 object stores that have been configured as FPOS will appear on this list.

The P8 Declare Record task depends on a valid Enterprise Records file plan configuration. Such a configuration involves many aspects that are too numerous to describe in detail here, including FPOS object store enablement, file plan structure, records class configuration, and others. For more information about Enterprise Records file plan configuration, see the Redbooks publication *Understanding IBM FileNet Records Manager*, SG24-7623.

## **Configure Classification**

Use Configure Classification to specify the target container in the FPOS. This is also one of the essential pieces of information required for record declaration. In most cases, the target container for each record being declared will vary. The configuration of this element will depend on the nature of your solution.

Enterprise Records supports records filed into multiple containers. Because of this, the Configure Classification settings allow you to Add, Edit, and Remove from a list of configured classifications. In a typical use case, only one classification is added to this configuration, and that classification specifies the path to a single file plan container. You must add at least one configured classification.

For each configured classification you add to the list, you can select from one of the following options to determine the target container:

- ▶ Literal
- ▶ Metadata
- ▶ Regular expression
- ▶ Calculated value
- ▶ List lookup

Figure 7-3 shows the various options for configuring the classification. This particular image shows a literal classification path, but other options are available for determining the classification.

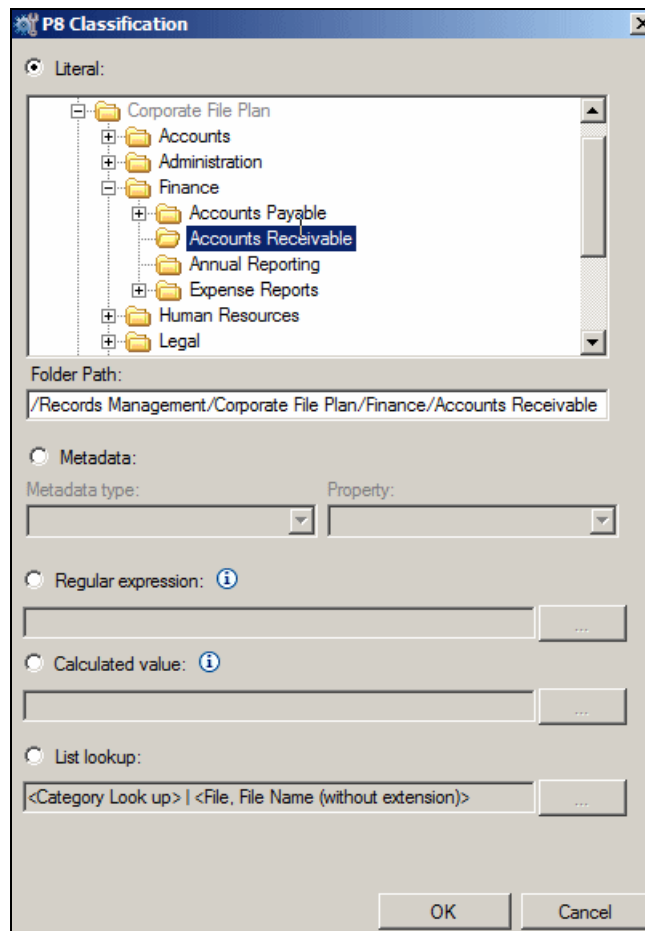


Figure 7-3 Options for record classification

Which of these options to use will depend on the data available at the time of declaration and how that data relates to the target container where the record should be filed. We describe each of these options in more detail later in this chapter.

## Property mappings

Use property mappings to specify which target record class to use for declaration and which properties belonging to that record class should be set at time of declaration. This is also one of the essential pieces of information required for

record declaration. In many cases, the target record class will depend on the document class that was used for the P8 Create Document step. However, determining the record class and which properties to map will depend on the overall nature and configuration of your Enterprise Records file plan.

The mechanics for configuring property mappings for declaration are similar to configuring property mappings for the P8 Create Document task. In fact, a record class is simply a P8 document class that is defined within the Enterprise Records data model on the FPOS. We describe the options for property mapping in more detail later in this section.

Figure 7-4 shows the lower part of the main configuration panel for the P8 Declare record task that includes the property mappings configuration.

**Configure Classifications**  
Classifications: ⓘ

**Configured Classifications**

DynamicMetadataReference[<Record Type Classification>,<File, File Name (without extension)>]
----------------------------------------------------------------------------------------------

Add... Edit... Remove

**Property Mappings**  
Record Class: ⓘ

FNTestRec Browse...

Show "Hidden Properties" ⓘ

Property	Value
Document Title	<File, File Name (without extension)>
Unique Record Identifier	
Description	
From	
To	
Cc	

Advanced... Edit... Reset value

**Data Correction**

☐ Truncate strings ⓘ

☐ Ignore choice list properties on error ⓘ

Figure 7-4 Property mappings and data correction options

## Data Correction

Use Data Correction to make adjustments to data that is used by Content Collector for record declaration. These are optional adjustments that might be useful in certain scenarios. The two options are:

- ▶ Truncate strings
- ▶ Ignore choice list properties on error

Figure 7-4 on page 233 shows these two data correction options in the lower part of the panel.

These are the same options that are available for the P8 Create Document task, and the same logic and requirements that might apply for properties being added to the target content repository would apply to setting properties during declaration.

Depending on the reason you are capturing metadata, you would typically ensure that the P8 property definitions were configured with a large enough maximum string length to accommodate the incoming property value. Or, if a string property has been configured with a choice list, the expected incoming values would match.

However, there are use cases where either of these conditions might not be met and would cause an error upon record declaration. For example, when capturing the email subject for display purposes, you might not want to store more than the first 255 characters in P8 as metadata. In the case of a choice list on the target P8 record property, sometimes the incoming property might have a null value, which would cause an error when trying to set the string that has a choice list associated with it.

Use these options as appropriate for your solution to avoid errors related to setting string properties when using the P8 Declare Record task.

### 7.3.3 Configuring record classification - options

As mentioned earlier, which option to use for configuring record classification depends on the data available at the time of declaration and how that data relates to the target container where the record should be filed. Although all five of the options shown will work, the last two (Calculated value or List lookup) are probably the most useful in terms of constructing a full path string to the target container.

## **Literal**

The Literal option allows you to browse the existing classification scheme in Enterprise Records and select a fixed target container. Optionally, you can type in the full path, but using the browser to expand the tree and select the target category or folder will ensure an accurate full path.

This option is useful if all records are expected to be filed in exactly the same target container in Enterprise Records for a given P8 Declare Record task. This might be the case if you have logic in your task route that determines the classification and you have multiple tasks that declare records, with each task configured for a different literal target container.

## **Metadata**

The Metadata option allows you to use an exact full path string from an existing metadata source in the task route. The expectation for this option is that you have access to the full path for declaration in one of the existing incoming metadata properties, such as Content Collector user-defined metadata or source file properties.

This option can be useful if you are using Content Collector to capture files from a file system or source repository where the source folder structure exactly matches the full path in the target Enterprise Records file plan, which is often unlikely in real-world situations.

## **Regular expression**

This option is useful if you can derive the full path by applying a regular expression to existing metadata. For more details about using a regular expression to derive a useful resulting string, see 4.2.1, “Configuring schedules” on page 89.

## **Calculated value**

This option allows you to calculate the entire full path by combining a series of literal or source metadata strings. This option can be useful where the file plan structure is mostly fixed and can be constructed adequately by concatenation.

## **Lookup list**

The lookup list option allows you to determine the full path for record classification by looking up existing metadata in a list. The Content Collector list effectively provides a mapping for associating a string value with the full path for classification.

This is potentially a useful and powerful option for determining the classification path because in many cases the source data will not exactly match the path

information for the target container in the Enterprise Records file plan. For example, if a string value identifying a record by record type or record code or some value that can uniquely determine the classification, this string can be used as the lookup value. It is much more likely that collected metadata will have values that are meaningful to the business context, which can be mapped to a target container in the file plan instead of having the actual file plan path value. The use case we discuss later in this chapter will illustrate how to use the lookup list to determine classification.

### 7.3.4 Configuring property mapping - options

Configuring property mapping for record declaration includes a variety of options that allows for flexibility in setting properties on each record at the time of declaration. The record class you select will determine the properties that are available to set. Each property can then be configured individually with the appropriate value. Typically, some or all of the same properties that are set on the P8 document that is created will also be set on the record during declaration. Which properties to set and how they are derived will depend on the solution requirements and the source metadata available.

Figure 7-5 highlights the main options for accessing the more detailed configuration for property mapping.

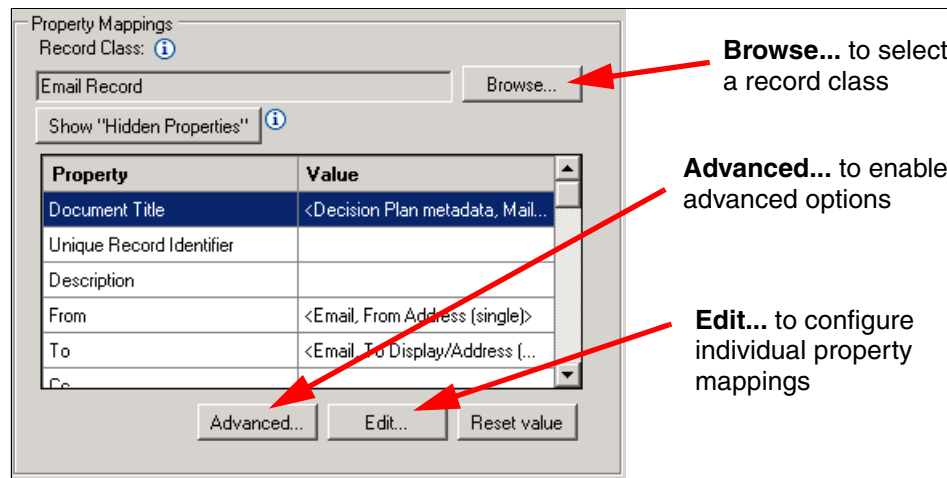


Figure 7-5 Property Mappings options

#### Select a record class

To map properties, you must first select a record class. Record classes are defined as part of the overall file plan configuration and design, and which one to select will depend on the solution requirements you have implemented for the

target FPOS. The record classes defined for the FPOS will often correspond to the document classes in use in the P8 content repository where the documents have been captured. The record class will determine which properties are available for mapping. Use the **Browse** button to navigate the available record classes. The record class can also be determined dynamically with the Advanced option for property mapping (see “Advanced options for property mapping” on page 238).

### Assign property mapping

After the record class is selected, you can individually configure each of the properties you intend to map. There might be more properties available for mapping than you need to use. Only map record properties that you intend to use or that are required for a given solution. Some of the properties exposed in this interface should not be mapped because they are internal properties used by Enterprise Records.

The following options are available for mapping each property individually when you select the property to map and click the **Edit** button.

- ▶ **Metadata** allows you to assign a value from Content Collector metadata that is available in the task route.
- ▶ **Literal** allows you to assign a literal value.
- ▶ **Advanced** allows you to use the Content Collector expression editor to construct a more complex expression for determining the value to assign to a property (see 4.2.1, “Configuring schedules” on page 89 for information about the Content Collector expression editor).

Figure 7-6 shows the options for these individual property mappings.

The screenshot shows a dialog box titled "Edit Expression" with a close button (X) in the top right corner. The dialog is divided into three sections by radio buttons: "Metadata", "Literal", and "Advanced". The "Metadata" section is selected, showing a "Source:" dropdown menu with "File" and a "Property:" dropdown menu with "File Name (without extension)". The "Literal" section is unselected and contains a large empty text area. The "Advanced" section is unselected and contains a text area with the expression "<File, File Name (without extension)>" and a small icon to its right. At the bottom right of the dialog are "OK" and "Cancel" buttons.

Figure 7-6 Options for individual property mapping

### Advanced options for property mapping

By selecting the **Advanced** button on the Property Mappings pane, you can enable dynamic selection of the record class by using an expression to determine the class. This is a useful option for task routes that might collect a variety of different types of documents for declaration that require differing record classes.

In addition, you can dynamically determine the property mappings by selecting user-defined metadata that matches the symbolic name of the target record



property. This is a useful option if the task route has been configured to populate user-defined metadata. Figure 7-7 shows the Advanced Option menu.

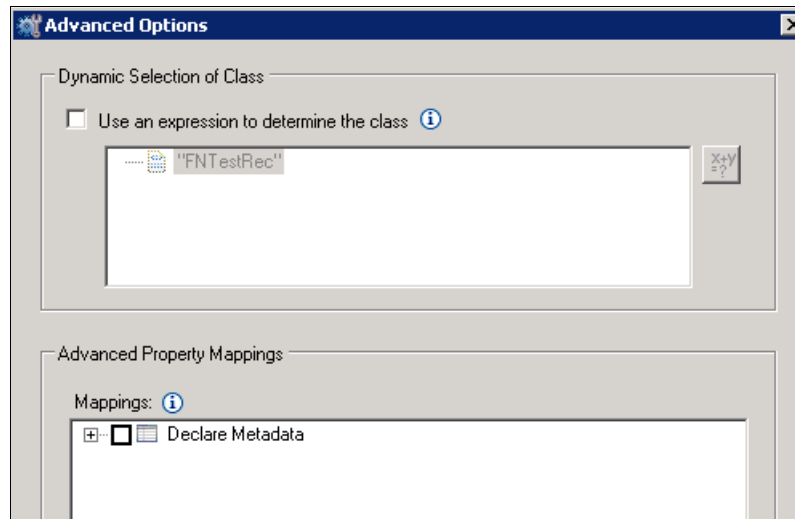


Figure 7-7 Advanced Options menu for Property Mappings

### 7.3.5 Which options to use

Not all of the options presented in this section will be useful for every use case. Depending on what metadata is available at time of capture and the nature of the documents being captured, various options will be more useful than others.

In subsequent sections in this chapter, we describe specific use cases that illustrate some of these options in a given context.

### 7.3.6 Determining record classification

Perhaps one of the most challenging aspects of designing a task route to include record declaration is determining the record classification. As described previously in this section, three essential pieces of information required to declare a record are the target repository, the target container (the record classification), and the target record class. In most cases, both the target repository and the target record class are either known or easily determined for a given collection process. The one piece of information that is not always obvious is the record classification.

At a high level, there are two distinct ways to determine record classification:

- ▶ Using associated metadata from collected content  
Email properties, file system properties, or custom metadata provided at the time of collection; this can include the folder (or the folder path) from which the content was collected
- ▶ Analysis of the content to make a determination  
Using text analytic capabilities to decide what type of document has been collected based on predefined rules or knowledge base

Depending on how incoming content has been organized, associated metadata might be sufficient for deriving the record classification. If the associated metadata does not provide enough information, Content Classification can be added to an Content Collector task route to perform the content analysis and make a determination that provides enough information for record classification.

### **Using associated metadata**

The following list includes examples of where metadata might be sufficient for record classification:

- ▶ Email collection from specific email folders
- ▶ File system or Microsoft SharePoint collection where the folder structure is well understood and maps to the Enterprise Records file plan
- ▶ File system or Microsoft SharePoint collection where custom metadata is provided at the collection point
- ▶ File system or Microsoft SharePoint collection where system properties (such as the file name) can provide enough meaning for record classification

### **Using Content Classification**

The following list includes examples of where available metadata is typically not sufficient and, therefore, Content Classification can be helpful:

- ▶ Email collection from a journal or general inboxes
- ▶ File system collection where the metadata or folder structure has no correlation with the Enterprise Records file plan
- ▶ Microsoft SharePoint collection where the content is spread across multiple sites and there is no consistent organization to the site structure
- ▶ Collection of any content where knowledge and rules about the unstructured content can provide a means of determining record classification

In the remainder of this chapter we focus on two different use cases that will illustrate these two different situations. First, we describe an email collection use

case that requires Content Classification to determine record classification. Next, we describe a file system use case where there is sufficient associated metadata for determining record classification without using Content Classification.

### 7.3.7 Task route templates for record declaration

Content Collector includes several task route templates for record declaration.

- ▶ FS to P8 Archiving (Declare as Record)
- ▶ FS to P8 Archiving (Declare as Record) with IBM Content Classification
- ▶ SP to P8 - Declare as Record

These templates can be used as a starting point for implementing your own task route. Alternatively, you can add the P8 Declare Record task to an existing task route if the requirements for record declaration are taken into account.

Figure 7-8 on page 242 shows the task route template for “FS to P8 Archiving (Declare as Record).” This template includes options that might or might not be needed, depending on the use case you are planning to implement. The essential steps include the file system collector (FSC Collector), archiving the file to P8 (P8 Create Document), declaring the record (P8 Declare Record), and FSC Post Processing.

This template includes additional features that might not be required, especially for many typical file system collection scenarios where automated business processes are used to generate documents. For example, having a decision point to separate duplicates from non-duplicates would not be needed if you are not using deduplication and are not expecting duplicates.

Also, the P8 File in Folder task might not be needed for automated record declaration scenarios because the file plan itself will be used for record classification, and an additional folder structure on the content repository might be either redundant or unnecessary.

The use case we present in 7.5, “Use case 4: File system archiving with records declaration” on page 258 does not use either of these options.

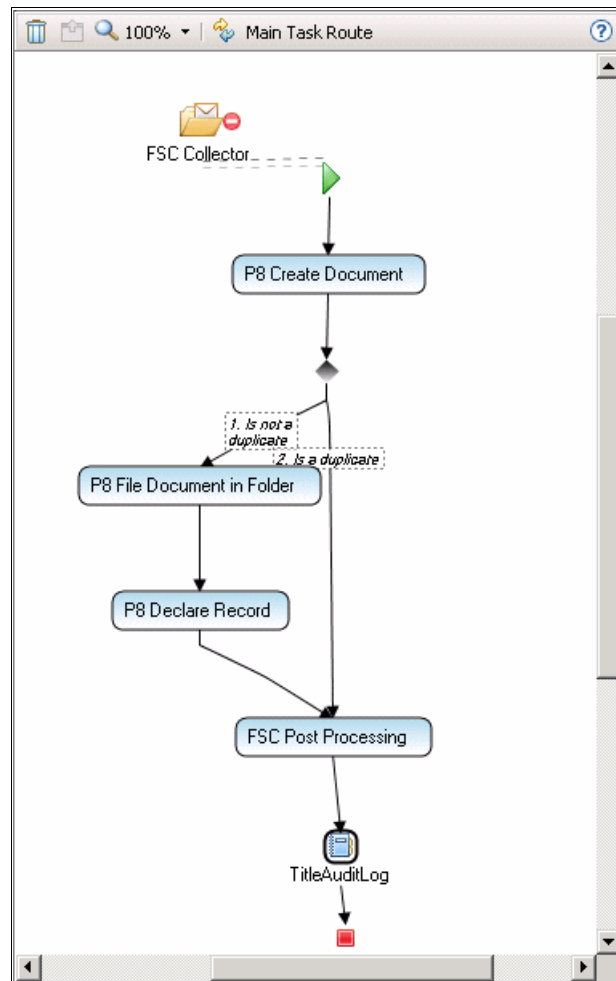


Figure 7-8 Task route template for FS to P8 Archiving with record declaration

The next two sections in this chapter describe two different use cases. We start with a use case for email archiving that builds on the use case from the previous chapter. We then present a use case for file system archiving.

## 7.4 Use case 3: Email archiving with records declaration

This use case illustrates how record declaration can be integrated with an email archiving scenario. We start with the scenario presented in 6.8, “Use case 2: Email archiving with content classification” on page 207 that uses Content Classification to determine classification results based on four example email types. We focus on what needs to be added to an existing Content Collector task route to include record declaration. In this example, we depend on the classification results from Content Classification to determine two items:

- ▶ Whether or not to declare a particular email based on the email type determined by Content Classification
- ▶ How to determine record classification based on the email type determined by Content Classification

To integrate our email archiving scenario with record declaration, the following items will be added to the task route for use case 2:

- ▶ A decision point that will determine whether to declare an email or not
- ▶ A task to archive the email without an expiration date if it is to be declared
- ▶ A task to declare the email as a record according to the result determined by Content Classification

We show the configuration for each of these items and explain the rationale for choosing our approach.

After we demonstrate a functional task route with record declaration, we enhance the task route to include added logic for handling possible error conditions.

### 7.4.1 Deciding which content to declare

As described in Chapter 6, “Document classification” on page 155, we used Content Classification to determine email classification according to the following four categories:

- ▶ Personal
- ▶ Business
- ▶ Sensitive
- ▶ Critical

For this use case, we continue to archive Personal and Business email as before, by setting the retention in Content Collector and archiving without record declaration. The rationale for this might be that personal and routine business email can continue to be managed by simple retention and avoid the overhead of record declaration, letting Content Collector continue to dispose of these email

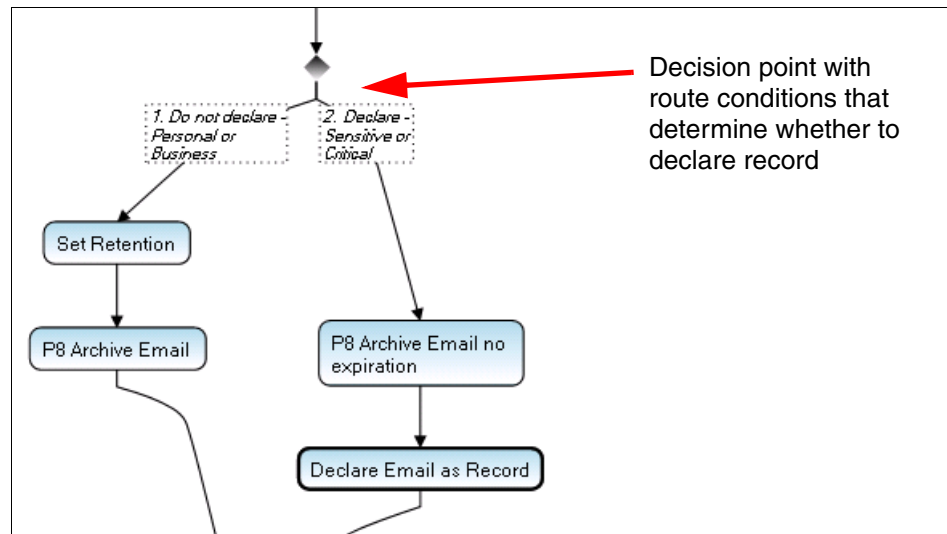
types based on the simple expiration date. However, Sensitive or Critical email will now be declared as records and the disposition will be managed by Enterprise Records. Table 7-1 on page 244 shows the logic to implement for this use case.

*Table 7-1 Logic for declaration and retention based on type of email*

Mail Type	Declare as Record	Content Collector Expiration Date
Personal	No	2 months from Date Sent
Business	No	24 months from Date Sent
Sensitive	Yes	Do not set - retention managed by Enterprise Records file plan
Critical	Yes	Do not set - retention managed by Enterprise Records file plan

Figure 7-9 shows the task route with a decision point on whether to declare email as record or not.

- ▶ Route 1 - Do not declare - Personal or Business
- ▶ Route 2 - Declare - Sensitive or Critical



*Figure 7-9 Decision point determines whether or not to declare records*

### **Route 1 - Do not declare Personal or Business**

According to the logic from Table 7-1, the condition (rule) for Route 1 is a simple OR condition that can be constructed with the expression editor to test for Mail

Type = "Personal" OR Mail Type = "Business". Figure 7-10 shows the configuration for this rule.

The screenshot shows a 'Rule' configuration window with a 'General' tab. The 'Name' field contains 'Do not declare - Personal or Business'. The 'Description' field contains 'Do not declare record if Mail Type determined by ICM is either "Personal" OR "Business"'. Below the 'General' tab is the 'Configure Rule' section. It has a 'Specify rule evaluation criteria:' label with an information icon. Two radio buttons are present: 'Always true' (unselected) and 'Advanced' (selected). The 'Advanced' section shows a tree structure: an 'Or' node containing two 'IEqual' nodes. The first 'IEqual' node compares '<Decision Plan metadata, Mail type>' with '"Personal"'. The second 'IEqual' node compares '<Decision Plan metadata, Mail type>' with '"Business"'. A small 'X+Y=?' icon is visible on the right side of the 'Configure Rule' section.

Figure 7-10 Rule detail for Personal or Business email

## Route 2 - Declare

Similarly, the rule for Route 2 is also a simple OR condition that can be constructed with the expression editor as shown in Figure 7-11.

The screenshot shows the 'Configure Rule' section of a rule configuration window. It has a 'Specify rule evaluation criteria:' label with an information icon. Two radio buttons are present: 'Always true' (unselected) and 'Advanced' (selected). The 'Advanced' section shows a tree structure: an 'Or' node containing two 'IEqual' nodes. The first 'IEqual' node compares '<Decision Plan metadata, Mail type>' with '"Sensitive"'. The second 'IEqual' node compares '<Decision Plan metadata, Mail type>' with '"Critical"'. A small 'X+Y=?' icon is visible on the right side of the 'Configure Rule' section.

Figure 7-11 Rule detail for Sensitive or Critical email

## 7.4.2 Archiving the content

Whether we declare the email as a record or not, we must still archive the content. In this section, we describe the tasks for both routes related to archiving email to P8.

Figure 7-12 shows how the tasks differ between Route 1 and Route 2. The tasks for the Do not declare route have not changed from the previous use case. Two new tasks are needed for record declaration. The email must be archived before it can be declared.

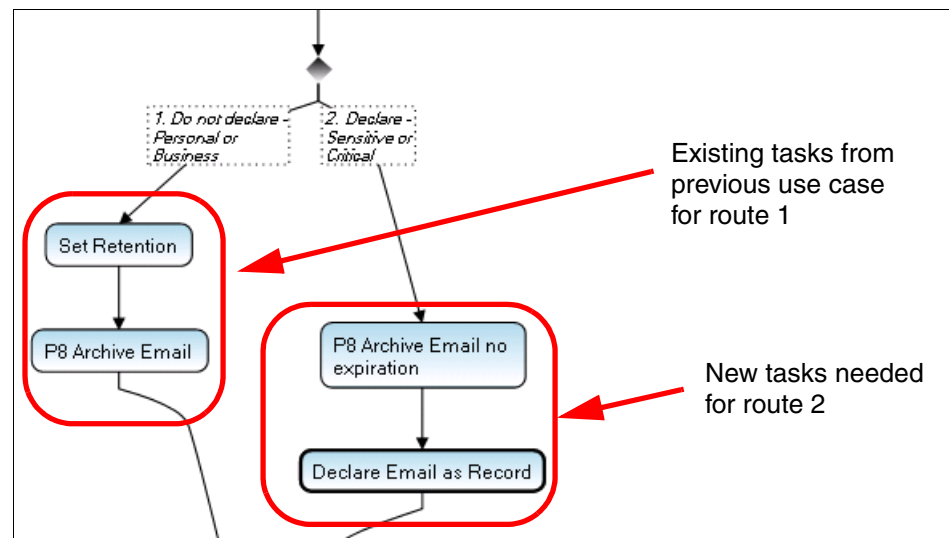


Figure 7-12 Tasks differ between Route 1 and Route 2

### Tasks for Route 1

The tasks for Route 1 remain as they did before we introduced the decision point for record declaration.

- Calculate Expiration Date (Set Retention)
- P8 Archive Email

For the email that does not get declared, we still have to set the retention and archive the email as before. The Set Retention task will still depend on the retention values specified in the Expiration Category list, because the Set Retention task uses a dynamic metadata reference to calculate the Content Collector expiration date.



Figure 7-13 shows the expression for determining Content Collector Expiration Date to be used with the Archive Email task where the number of months will be added to the email Received Date depending on the Mail Type.

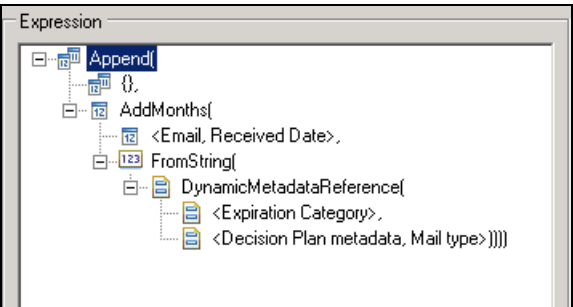


Figure 7-13 Expression for determining Content Collector Expiration Date

Figure 7-14 shows the Expiration Category list with retention values. Because of the way we configured the rule for Route 1 at the decision point, only the values for Personal and Business will be used. The other two will be ignored.

The screenshot shows a 'Value List' window with a table of retention values. The table has three columns: 'Name', 'Description', and 'Return value'. The rows are 'Personal', 'Business', 'Sensitive', and 'Critical'. The 'Personal' row is highlighted. The 'Return value' for 'Personal' is 2, for 'Business' is 24, for 'Sensitive' is 60, and for 'Critical' is 120.

Name	Description	Return value
Personal		2
Business		24
Sensitive		60
Critical		120

Figure 7-14 List of retention values for each mail type

As before, the Archive Email task must include the option to set an expiration date.

## Tasks for Route 2

Two new tasks are added for Route 2 to complete the record declaration.

- ▶ P8 Archive Email no expiration
- ▶ Declare Email as Record

Before we can declare the email as a record, we must archive the email to P8.

**Record declaration consideration:** Record declaration in Enterprise Records requires that the content has been added (archived) to the P8 repository first.

The P8 Archive Email task does *not* include the option to set the expiration date, because this would potentially conflict with the retention management defined for the record by the Enterprise Records file plan. Thus, we have labeled the task “P8 Archive Email no expiration.”

Figure 7-12 on page 246 shows the tasks for Route 2. The remainder of the configuration for the Declare Email as Record task is described in the next section.

### 7.4.3 Configuring record declaration

The P8 Declare Record task must be configured to properly determine the key elements required for declaration:

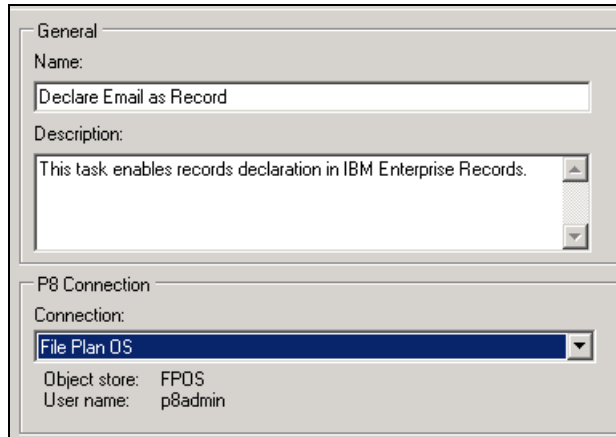
- ▶ P8 Connection to the appropriate File Plan Object Store (FPOS)
- ▶ Record classification
- ▶ Record class selection and property mappings

#### P8 Connection

For this use case, we declare emails into a single target FPOS. So the P8 Connection will be configured appropriately and will not need to vary or change.

However, if you have a situation where you need to declare records into more than one target FPOS, you would need to design your Content Collector task route to have separate P8 Declare Record tasks for each FPOS. In most situations, records are declared into a single FPOS. But as systems grow and scale up, it is possible that you might need to take multiple target FPOS connections into account, and you would need to design your task routes appropriately.

Figure 7-15 shows the P8 Connection selected. You must select the connection before configuring any of the other elements for declaration, because the details of the configuration will depend on the selected FPOS and the file plan or plans defined for that FPOS. Only valid FPOS connections will be presented in the drop-down list for selection.

The image shows a configuration window with two main sections. The top section is titled 'General' and contains a 'Name:' field with the text 'Declare Email as Record' and a 'Description:' field with the text 'This task enables records declaration in IBM Enterprise Records.' The bottom section is titled 'P8 Connection' and contains a 'Connection:' dropdown menu with 'File Plan OS' selected. Below the dropdown, it shows 'Object store: FPOS' and 'User name: p8admin'.

*Figure 7-15 The P8 Connection to an FPOS must be selected*

To complete the remainder of the configuration for record declaration, the FPOS must already be configured with an appropriate file plan and record class.

## **Record classification**

Before we can configure the record classification, we must ensure that the target FPOS has a file plan that will support our use case and requirements. We will be declaring two types of email records: Sensitive and Critical.

The whole point of identifying which emails belong in each of these categories is to be able to manage the retention and disposition for the different categories separately. Thus, we need to have a file plan that includes categories that match the different types of emails. To configure the classification in Content Collector, we need to know the full file plan path for each category into which records might be declared.

Figure 7-16 on page 250 shows the target Enterprise Records file plan with the two categories that will be used for declaring email in our use case. File plans are

usually more complex than the one shown here, but this plan illustrates how email can be separated into Sensitive and Critical categories upon declaration.

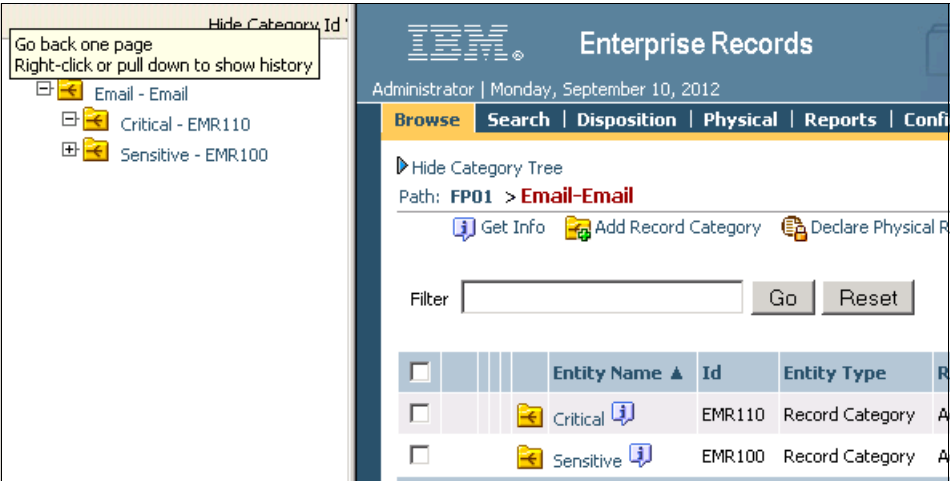


Figure 7-16 A simple file plan in Enterprise Records that has categories for different types of email

**Record categories or record folders:** In Enterprise Records, records can be declared into either *record categories* or *record folders*. In this use case, we are using an example where the records are declared directly into record categories.

For this particular task route and use case, we have a common parent file plan path for all emails, namely /Records Management/FP01/Email/. The only variation is one of two categories at level 2 of the file plan, depending on which type of email we have identified.

There are several ways to approach the configuration for record classification. For this use case, we use a calculated value where only one element in the full path is variable. This approach is suitable for the scenario we are attempting to implement with email because the main branch of the full path is fixed for all records and there is only minor variation at the end of the path.

/Records Management/FP01/Email/ + <Mail Type>

For this scenario, we add one new classification. Figure 7-17 shows the calculated value option selected, which allows the task to determine the full path to use for classification at run time.

The screenshot shows a configuration window with four radio button options: 'Metadata:', 'Regular expression:', 'Calculated value:', and 'List lookup:'. The 'Calculated value:' option is selected. Below it, a text field contains the path '/Records Management/FP01/Email/+' followed by a button with three dots. Above the text field, there are two dropdown menus labeled 'Metadata type:' and 'Property:'.

Figure 7-17 Calculated value is one way to determine classification at run time

The calculated value uses a literal string concatenated with a string value determined at run time taken from the results of the Content Classification classification. Figure 7-18 shows the configuration for the literal string concatenated with the Decision Plan metadata value for Mail Type that will be determined by Content Classification at run time.

The screenshot shows a dialog titled 'Edit Calculated Value'. It contains a table with two columns: 'Value' and 'Join Operator'. The first row has the value '/Records Management/FP01/Email/' and the join operator '+'. The second row has the value '<Decision Plan metadata, Mail type>'. Below the table are three buttons: 'Add...', 'Edit...', and 'Remove'. At the bottom are 'OK' and 'Cancel' buttons.

Figure 7-18 Details of the calculated value configuration for this use case

## Property mapping

The one remaining essential item for record declaration is the record class and property mapping configuration. Figure 7-19 shows the property mappings options as they might look for our email scenario. After a record class is determined, the properties available for that record class can be mapped. Because our scenario for this use case involves only email records, we use the Email Record class that is available by default for Enterprise Records. It is possible to use the Email Record class as shown here for real-world deployments, but it is typically advisable to design your own class hierarchy in the P8 object store to accommodate flexibility and scalability that might be needed with future growth.

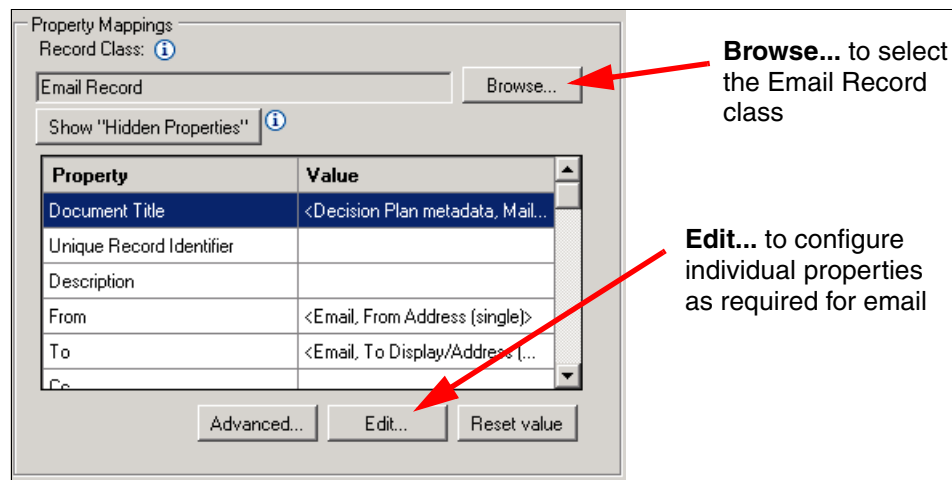


Figure 7-19 Property mappings include record class selection and editing selected individual properties

**Tip:** Always configure the Document Title property with something meaningful for the context, because Document Title is typically used as a primary display field in Enterprise Records.

Other properties that are typically useful for an email scenario are properties such as From, To, CC, Subject, Sent On, and Received On, from the available metadata associated with the captured document.

For this use case, we map the record properties in a particular way to illustrate important options. Properties can be mapped as needed for any particular requirements that are site-specific.

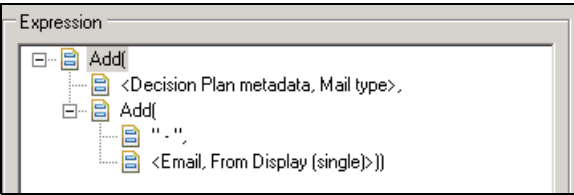
Table 7-2 on page 253 shows how we decided to map the properties for this use case.

*Table 7-2 Mapping email attributes to record properties*

Record Property	Source metadata	P8 Data Type
Document Title	<Decision Plan metadata, Mail Type> + <Email, From Display>	String
From	<Email, From Address (single)>	String
To	<Email, To Display/Address (multi)>	String (multi)
CC	<Email, CC Display/Address (multi)>	String (multi)
Subject	<Email, Subject>	String
Sent On	<Email, Sent Date>	DateTime
Received On	<Email, Received Date	DateTime

### **Document Title**

Instead of mapping a single source metadata value to Document Title, we chose to use the expression editor to concatenate the Mail Type from the Decision Plan metadata with the From Display value from the email. Figure 7-20 shows the expression that gives us this result. We do this mainly to illustrate the flexibility you have in assigning something meaningful to Document Title.



*Figure 7-20 Expression to construct Document Title from two different strings*

### **From**

Because an email is sent from only one account, we map a single value Address string from the source email into the single value string property in Enterprise Records.

### **To**

Because an email can be sent to multiple addresses, the To property in Enterprise Records has been configured as a multi-value property. So, in this case we map the string array variant from the source email into the multi-value property in Enterprise Records, as shown in Figure 7-21 on page 254.

In addition, we chose to use both Display and Address from the email instead of only the Address as we did for the From property, simply to illustrate that there are choices available for how these properties are mapped. It is also possible to map the single-value variant from the source email into a different single value record property in Enterprise Records, where the list of email recipients is stored as a single string. This might or might not be more desirable, depending on how your Enterprise Records system will be used for searching across these properties. Which of these choices you decide to use, or whether it is necessary to even map this data into Enterprise Records, will depend on your site-specific requirements.

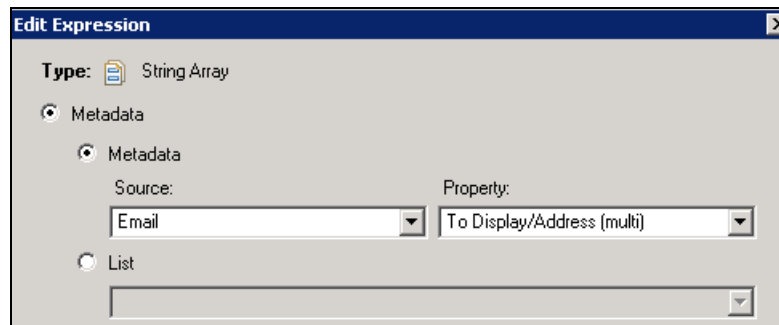


Figure 7-21 Mapping multi-value email metadata to Enterprise Records

## Cc

The same logic and choices that are used for mapping the To property can be applied to the CC list from the source email.

**Single value properties and multi-value properties:** The Content Collector data model for email includes both the To and CC lists as single value properties that can be mapped when archiving to P8. The ICCMail3 document class in P8 uses single value string properties for this information.

In contrast, the Enterprise Records data model, by default, includes *multi-value* properties for To and Cc that can be configured and used if desired on the Email Record class.

Content Collector provides recipient list data for To and CC in both single value and multi-value form to be used as needed for your solution requirements.

## Subject

For this use case, we map the email subject directly to the Subject property.



### Sent On

This property is a DateTime data type in Enterprise Records and is set to the Sent Date from the email.

### Received On

This property is also a DateTime data type in Enterprise Records and is set to the Received Date from the email.

### Data correction

For this use case, we select the Truncate strings option to ensure we do not get an error when trying to assign an unusually long subject string that might not match with the maximum string length configured in Enterprise Records.

## 7.4.4 Results in Enterprise Records

With the configuration we have described so far in this section, you would expect to see the following results in Enterprise Records after running the task route. Figure 7-22 shows that with a simple test of four emails, one for each of the four Mail Type categories for this use case, only two of those emails are declared as records and they are each declared into the appropriate category based on the Mail Type determined by Content Classification.

Find Records based on properties where :

From	=		AND
Sent On	=	Clear (MM/d/yy)	AND
Subject	=		AND
Security Folder	=	<a href="#">Select Object</a>	

AND Based on content where:

Content Contains  in Metadata

And filter by parent type: [Select](#)

[Search](#) [Clear](#) [Change](#)

<input type="checkbox"/>	Entity Name	From ▲	Sent On	Subject	Security Folder
<input type="checkbox"/>	Critical - Administrator	administrator@ecm.ibm.local	9/6/12 8:32 PM	test critical	Critical
<input type="checkbox"/>	Sensitive - Administrator	administrator@ecm.ibm.local	9/6/12 8:32 PM	test sensitive	Sensitive

Multi-Select Actions ▼

[Browse](#) | [Search](#) | [Disposition](#) | [Physical](#) | [Reports](#) | [Configure](#) [Help](#)

Figure 7-22 Email records appear in Enterprise Records after successful record declaration from Content Collector

The search results show that the Document Title (showing as Entity Name in this view) was assigned with the Mail Type and the email From Display value. The

From column shows the email Address instead of the Display Name, based on how it was configured.

### 7.4.5 Including logic to handle intermittent failure

Although not all errors or failures in task route processing can be anticipated, it is advisable to include logic, when possible, to handle specific error conditions that might arise intermittently with a given scenario.

If record declaration fails while using the email data model for archiving, the task route will not roll back any created objects. In this case, the post-processing task that marks the email as processed will not complete and the email will be collected again the next time the task route runs.

During this second collection, the email will be identified as a duplicate because it had been previously archived. However, if record declaration failed because of some intermittent problem, the declaration can be attempted again. The P8 Declare Record task will check to see if the archived email has already been declared and, if not, it will attempt to complete the declaration. As a precaution to deal with this possibility, it is advisable to include the record declaration task for both unique email and duplicates.

**Tip:** Include the record declaration task for both unique email and duplicate emails in case the duplicate email is the one that should have been declared as a record but was not due to some failure on the previous attempt.

To do this, the logic in the task route becomes somewhat more complex. Content Classification will be needed to process duplicates and unique emails if the duplicate is one that should have been declared but was not declared because of a failure.

The following key elements are included in the task route to handle this scenario:

- ▶ The test for duplicates should happen after content classification.
- ▶ The route for duplicates should include record declaration.
- ▶ A decision point is needed in case the duplicate does not need to be declared.

Figure 7-23 shows the section of the task route that includes the additional logic described here.

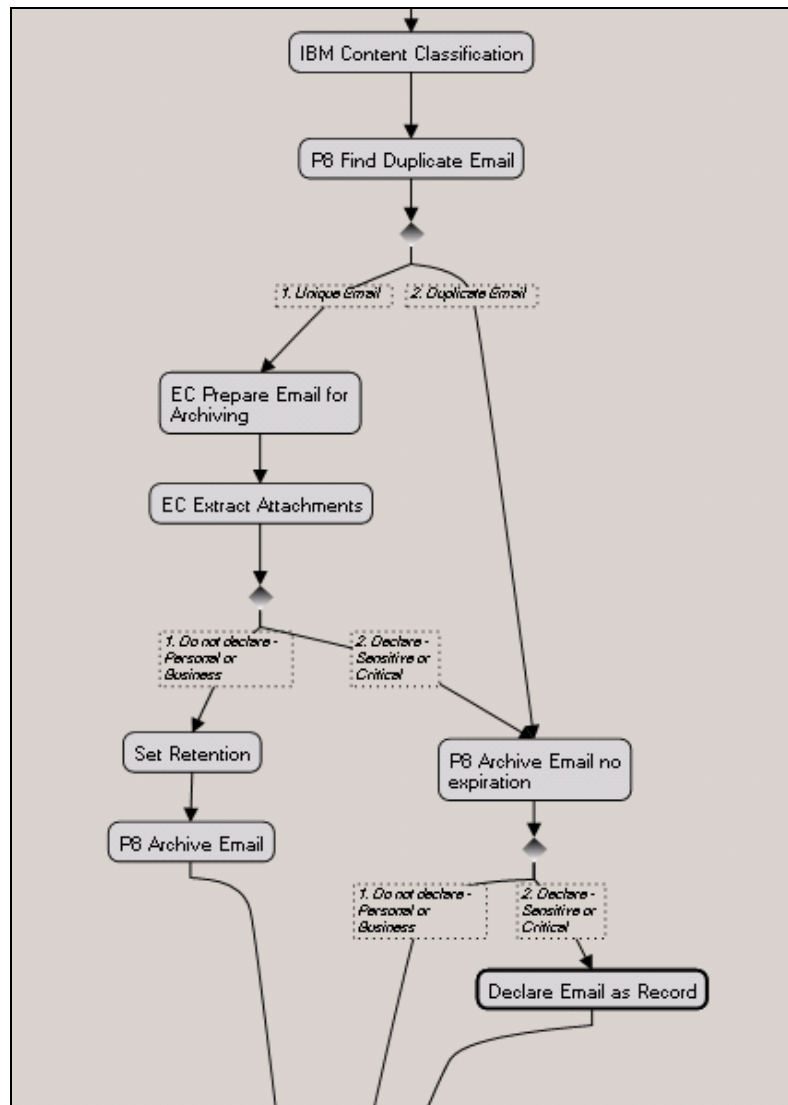


Figure 7-23 Task route logic to handle record declaration for both unique email and duplicates that might be a result of earlier failure

**Archiving files consideration:** When archiving files using the P8 Create Document or P8 Create Version series task, there is no need to process duplicates in case of failure as described in this section. If record declaration fails after file archiving with either of these tasks, the P8 document creation will be rolled back (the archived document will be deleted from the repository) so it can be collected again as a unique document. Because of this, you would *not* want to attempt record declaration on a duplicate for a file archiving scenario.

As we show with the next use case example, the complexity described here for an email archiving scenario is not required for a file archiving scenario.

## 7.5 Use case 4: File system archiving with records declaration

This use case illustrates a simple automated collection process where documents are placed on a file share by an automated business process in an organized manner, and can be ingested and declared automatically based on the file path and other attributes found on the ingested documents. As long as there is enough information from the incoming metadata to determine all the required information for declaring records with the P8 Declare Record task, there is no need for an external classification mechanism like Content Classification.

This use case can be adapted to make use of associated metadata files and many of the other options and features that Content Collector supports as described with the various examples provided in Chapter 3, “Dimensions of content archiving themes” on page 23.

### 7.5.1 Scenario and overview

The scenario for this use case involves two different types of records that need to be collected from a file share and declared. The parent folder from the file system where the documents reside before collection will determine the record type. The two examples in this use case are:

- Invoices
- Contracts

Each of these two record types must be declared in their respective categories in the Enterprise Records file plan because each of these record types will have

different retention rules and policies that apply. In addition, each of these record types will have different property mapping requirements.

The following assumptions apply to this use case:

- ▶ Files are automatically placed into appropriate subfolders on a file share by an external business process.
- ▶ Subfolders on the file share identify the type of record.
- ▶ The Enterprise Records file plan has been designed to accommodate both types of records with separate record categories and separate record classes.
- ▶ For Invoices, the File Created Date will be used to set the Invoice Date in the property mappings.
- ▶ For Contracts, the File Name will start with a strict pattern that includes the Contract Number to be used in the property mappings.
- ▶ Separate record classes will be used for each type of record, because the properties will vary.
- ▶ Documents will be deleted from the file share upon successful capture.

Table 7-3 lists the configuration requirements for this use case.

*Table 7-3 Configuration requirements for declaring records from the file system*

File System folder	Record category	Record class	Property mapping
Invoices	Invoices	Invoice Record	Set Invoice Date to the File Created Date
Contracts	Contracts	Contract Record	Set the Contract Number to the 8 digits in the file name that identify the contract

Figure 7-24 shows the task route for this use case. A decision point and rules are used to allow for variations in the configuration for the P8 Declare Record tasks for each of the two record types collected.

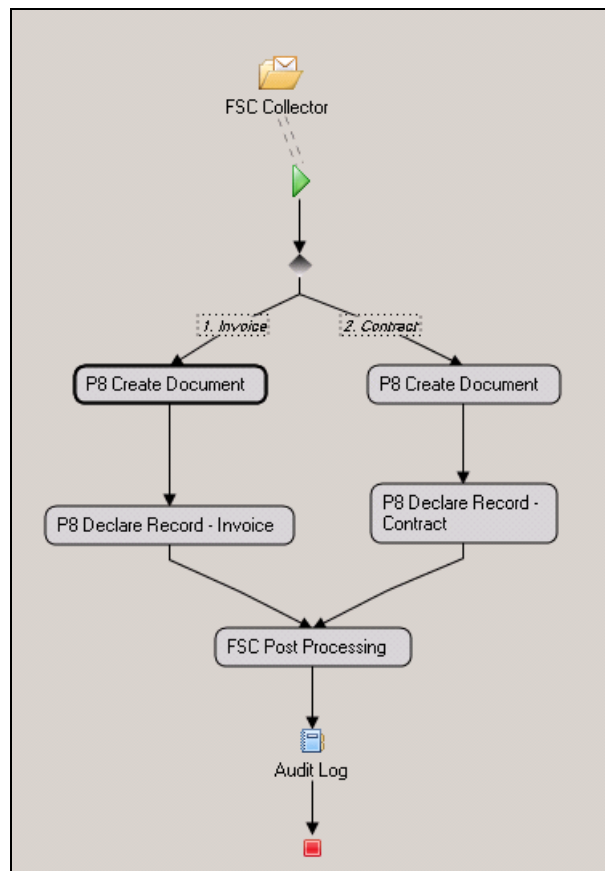


Figure 7-24 Task route for use case 4 - file system collection with record declaration

### 7.5.2 Deciding which content to declare

For this use case, we determine which content to declare by simply configuring the file system collector appropriately. We will declare all documents that are collected. Unlike the previous use case where some collected email was not declared, there is no logic needed in the task route to determine which documents to declare.

### Collector configuration

The file system collector will identify the collection source and collect all documents from the indicated collection sources. For our example, we will use the C: drive for simplicity, but the collector would typically be configured to collect from a controlled file share that was remote to the Content Collector server. The collection schedule would be configured as appropriate to coordinate with the automated business process that creates the documents on the file share. Figure 7-25 shows the collection sources for this use case, which collects from specific folders that match the logic and configuration of the task route.

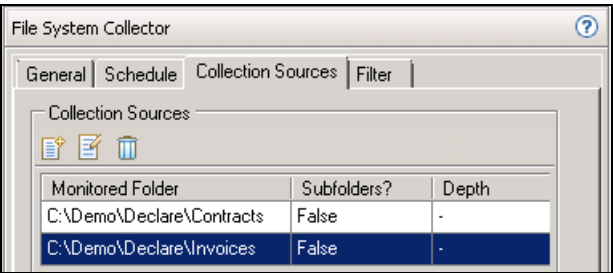


Figure 7-25 Two collection sources where all documents will be declared as records

### 7.5.3 Separating documents for declaration

In this use case, we will use a decision point with rules to separate the collected documents based on the subfolder from which they were collected. We can use a regular expression to derive the subfolder name from the folder path. Figure 7-26 shows the expression we use for Invoices to get the third folder in the path and compare it with the literal string Invoices, which we know identifies the record type. A similar rule would be used for Contracts.

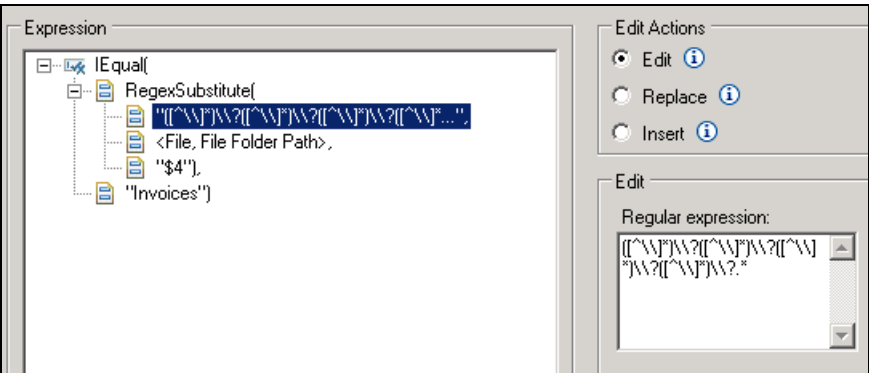


Figure 7-26 Expression used in the rule to identify Invoices based on folder name

We use the same set of tasks for each branch, but the configuration details will vary according to the P8 and Enterprise Records configuration required for each record type being declared. In each case we will use a P8 Create Document task followed by a P8 Declare Record tasks before performing Post Processing.

## 7.5.4 Creating the document in P8

Before we can declare the record, the document must be created in the P8 content repository. Because we are collecting different types of documents, we might be required to configure the P8 Create Document task differently with different target document classes and different property mappings, depending on the nature of the documents collected. The document classes used for archiving must also be record enabled.

## 7.5.5 Configuring record declaration

To configure the Declare Record tasks, we should have an understanding of how the record types map to the file plan configuration. Figure 7-27 shows the simple file plan we have constructed for this use case that illustrates the two categories we will use for each of the record types. A typical file plan would have many more record categories representing all the record types across an organization. The key to configuring record declaration is being able to map the information gathered at the point of collection to the structure and configuration of the target file plan.

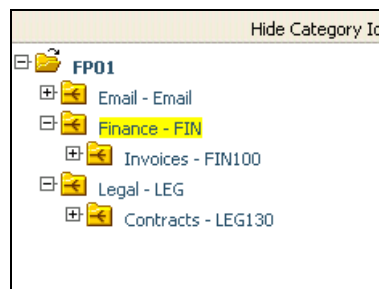


Figure 7-27 Simple file plan for Invoices and Contracts

Because our task route has a separate P8 Declare Record task for each branch based on the two record types we are collecting, the configuration details will vary slightly for each depending on the record type.



## **Name and description**

For this use case, it is useful to include the name of the record type in the task name for easy identification when viewing the task route. We can also add a specific description that gives more detail about the configuration if desired.

## **P8 Connection**

For this use case, each of the two record types will be declared into the same file plan and will therefore use the same FPOS as specified for the P8 Connection.

## **Record classification**

Because we have separate branches in our task route for each of the two record types, and because each record type will be declared into a single category, we can use a literal value in each of the P8 Declare Record tasks as appropriate for the given record type:

- ▶ For Invoices - /Records Management/FP01/Finance/Invoices
- ▶ For Contracts - /Records Management/FP01/Legal/Contracts

Because each of these categories has a different parent category, we are not able to use a common calculated value as we did with the configuration for use case 3.

Figure 7-28 on page 264 shows the literal value determined by browsing the file plan structure for the Invoices category. A different literal value would be

configured for Contracts as required. The literal value is a fixed target classification that will not vary.

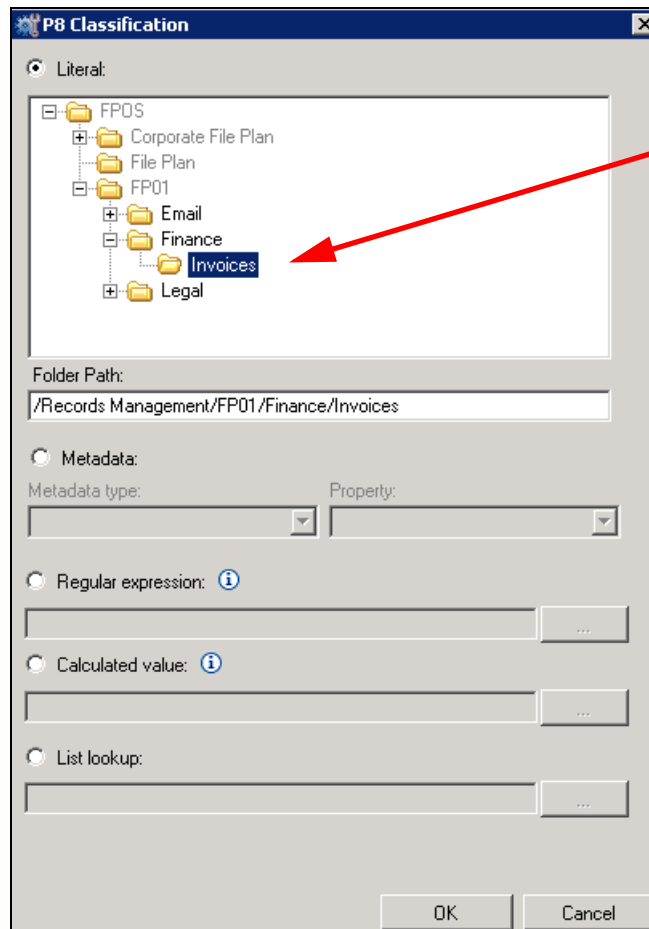


Figure 7-28 Literal value for the record classification for Invoices

## Property mapping

Each of the two record types we are declaring will require a different target record class and different property mappings.

### **Invoices**

Invoices will require the following property mapping configuration:

- ▶ Select the **Invoice Record** record class.
- ▶ Map the Document Title property to the File Name without extension.
- ▶ Map the Invoice Date property to the File Created Date.

Figure 7-29 shows the Invoice Record class has been selected. The record class will determine the available properties that can be mapped.

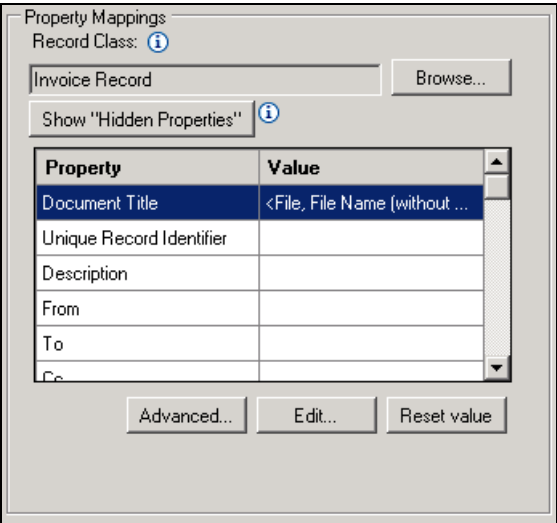


Figure 7-29 Invoice Record class is selected for property mapping

For this use case, as an example, the retention for Invoices will be based on the Created Date from the original file, based on the assumption that the automated process that generates the invoices will be creating the documents on the appropriate date.

For scenarios where the Created Date of the original file cannot be used for such a purpose, there are other strategies for indicating the appropriate date value for purposes of retention. For example, the automated process can also generate an associated metadata file with not only the Invoice Date, but other metadata for each document as well. You can also adopt a naming convention on the file name itself to indicate a relevant date, which would then have to be parsed and converted to a date value.

In this use case, Invoices are an example record type where the relevant retention trigger date can be set at time of declaration and will not need to be changed (that is, the Invoice Date is a value that will not vary and is known at time of declaration).

**Contracts**

Contracts will require the following property mapping configuration:

- ▶ Select the **Contract Record** record class.
- ▶ Map the Document Title property to the File Name without extension.

- Map the ContractNumber property to a value derived from the File Name that represents the contract number for that document.

For this use case, Contracts provide an example of a record type where the trigger date for retention is not known at the time of declaration. Retention policy for contracts is typically based on the contract expiration or contract closed date, which is usually not known when the contract record is declared. Therefore, it is important to capture some other key value to later identify the contract, to set the appropriate retention trigger date some time after declaration.

In this example, we must have at least the contract number for each document to know which contract it refers to so that we can later match up the contract number to an appropriate expiration date when the time comes. This illustrates that you can declare records without knowing their expiration or trigger date at the time of declaration if there is enough other data to make the appropriate updates later. This technique can be used when there is an automated business process that generates the file names with a strict naming pattern to ensure accuracy for deriving key metadata such as a contract number.

This use case assumes that all Contract files will have a file name that has the following pattern: the letter C, followed by a dash for a separator, followed by the 8-digit contract number, followed by any other appropriate text to identify the file and provide a file extension, as shown here:

Expected File Name pattern: C-dddddddd\*.\*

Figure 7-30 shows the regular expression used to derive the 8-digit contract number from the file name for mapping the Contract Number property during record declaration.

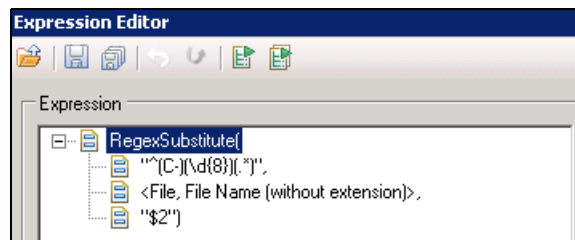


Figure 7-30 Expression to derive the 8-digit contract number from the file name

## Data correction

For this use case, we will enable the Truncate strings option as a precaution in case the Document Title is longer than the configured maximum, even though this is unlikely.

We will also enable the Ignore choice list properties on error option because the Invoice Record class has property with a choice list that will not be set.

### 7.5.6 Post processing

For this use case, we delete all documents that are successfully captured and declared.

### 7.5.7 Results in Enterprise Records

After running the task route to collect a few sample documents for this use case, Figure 7-31 shows the results of a search for records in Enterprise Records. The search results show that two Invoice records and four Contract records were collected.

Based on the way we configured the P8 Declare Record tasks, Invoices were assigned an Invoice Date based on the File Created Date and Contracts were assigned a Contract Number based on a fixed pattern from the File Name.













Entity Name	Security Folder	Invoice Date	ContractNo	Date Created
 Invoice 23456 - 2011-09-23 - PO 100763 	Invoices	9/14/12 3:49 PM		9/14/12 3:57 PM
 Invoice 12345 - 2011-10-27 - PO 654321 	Invoices	9/14/12 3:50 PM		9/14/12 3:57 PM
 C-12345678 IBM Contract for Services -Signed 	Contracts		12345678	9/14/12 3:57 PM
 C-87654321 Customer XYZ signed contract 2012-06-18 	Contracts		87654321	9/14/12 3:57 PM
 C-12345678 IBM Contract for Services 	Contracts		12345678	9/14/12 3:57 PM
 C-12345678 IBM Contract for Software 	Contracts		12345678	9/14/12 3:57 PM

Figure 7-31 Search results in Enterprise Records after collecting and declaring example Invoices and Contracts

## 7.6 Considerations and guidelines

In this section we describe guidelines, limitations, and options for designing value-based archiving solutions that include record declaration with Enterprise Records.

## 7.6.1 Preferred practices

The following list identifies general guidelines to consider when integrating records declaration with a content collection scenario:

- ▶ Start with a well-formed file plan in Enterprise Records that has been designed to meet your enterprise records retention and management needs.
- ▶ Have an understanding of how the records you collect with Content Collector will fit the file plan classification you intend to use.
- ▶ If metadata from collection is not sufficient for all the required attributes for record declaration, use Content Classification, if appropriate, to provide a means of classifying records.
- ▶ Avoid filing documents in folders on the P8 content object store (ROS) unless there is a specific business or user need.
- ▶ Avoid setting the Content Collector expiration date if a record is to be declared, because this might cause a conflict when Enterprise Records tries to dispose of the record.
- ▶ When archiving files for record declaration (from file systems or Microsoft SharePoint), it is best if the content is organized before collection with a structure or with associated metadata that can provide meaningful information for record declaration.
- ▶ When declaring email as records, consider which properties you need to map into the FPOS and store in Enterprise Records. The Content Collector data model captures and stores data in an indexable XML format that can be used for searches and eDiscovery, and you might not need to duplicate this metadata in Enterprise Records, depending on your solution requirements.
- ▶ Design the Content Collector task routes appropriately for maintainability, scalability and performance. See 4.3, “Optimizing task routes for maintainability” on page 93 for a discussion of preferred practices for optimizing task routes.

## 7.6.2 Limitations

This section describes limitations to be aware of when integrating Content Collector with record declaration in Enterprise Records.

### Target folder creation

Some declaration scenarios might require record folders to be created dynamically at run time. The Content Collector P8 Declare Record task will not automatically create target record folders or categories. If this is a requirement,

you might need to integrate the task route with an external component that creates any required record folders before the P8 Declare Record task occurs.

For example, if you decide to classify records not only by record type, but also into a record folder for each year based on the File Created Date, you might want to dynamically create the record folder for a given year as needed for the given record type.

### **Potential conflict with Content Collector expiration date**

When a record is declared in Enterprise Records, the expectation is that Enterprise Records will manage the retention of that content. If the Content Collector expiration date is also set, there can be a potential conflict with Enterprise Records only if the Enterprise Records disposition process is ready before the Content Collector expiration date. Enterprise Records will attempt to delete the content, but will fail with an error if the Content Collector expiration date has not yet passed.

For example, imagine a scenario where you arbitrarily set the Content Collector expiration date to 3 years from the date created for all documents collected, and you also declared some of these documents as records. It might be the case that after the records are declared, the retention policy dictated by Enterprise Records might call for disposition of some records before 3 years from the date created. In this case, an error would occur during the Enterprise Records destruction process if the Content Collector expiration date has not yet passed.

### **Content Collector Expiration Manager and declared records**

If the Content Collector expiration date is set on a document and the document is also declared as a record, the Content Collector Expiration Manager will ignore that document when attempting to delete expired content. This is a feature that prevents an error from occurring with Content Collector Expiration Manager on a declared record, because after a record is declared, the record is locked down by Enterprise Records and must be deleted by Enterprise Records.

## **7.6.3 Declaring a version series**

It is possible to collect multiple versions of a document by configuring the Microsoft SharePoint Collector to get either all versions or some specified number of versions, or by using the File System Collector with associated metadata that describes a version series to collect. When you have configured the task route to use the P8 Create Version Series task followed by the P8 Declare Record task, the entire version series that was collected can be declared as a single record.

The P8 Declare Record task will apply only to those items collected during a given execution of the task route. So if there were earlier versions of the same document that were collected previously, those versions would not be related to the ones being declared together during a given execution. Similarly, any subsequent versions that might be collected at a later time would not be added to the same record, but can be declared separately into a new record.



Figure 7-32 shows an example task route that includes record declaration for a Microsoft SharePoint version series. This task route declares all collected versions of each document as a single record.

Each version is archived individually into the P8 content repository as a version series, but the entire version series (one or more versions of a single document) is declared as a single record.

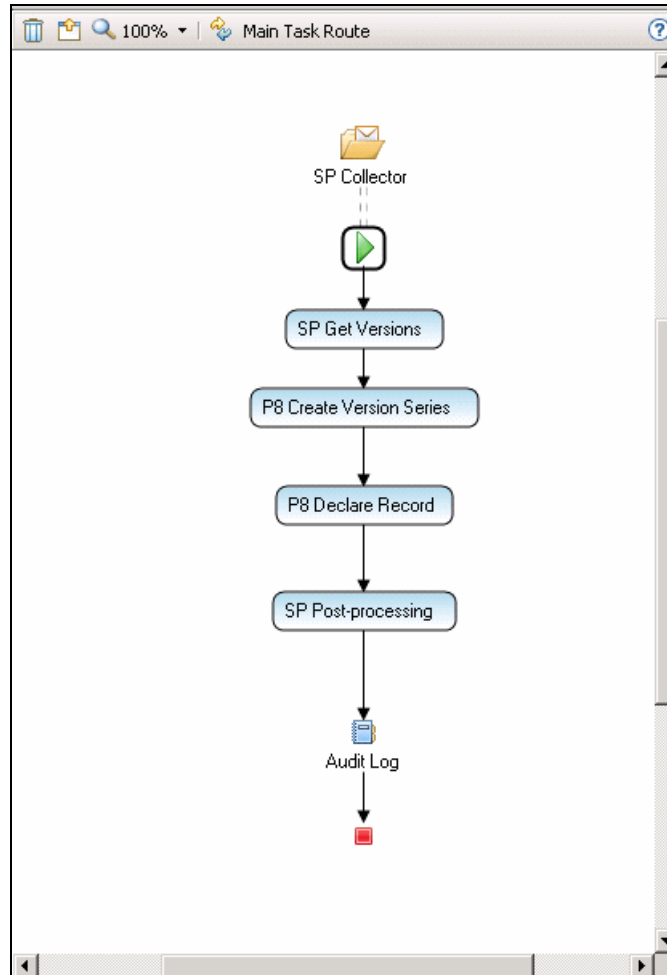


Figure 7-32 Task route showing record declaration for a Microsoft SharePoint version series

**P8 Create Version Series task:** The SP Get Versions task is intended to work in conjunction with the P8 Create Version Series task even if you have configured the SP Get Versions to collect only a single version. It is advisable to use the P8 Create Version Series task whenever you are archiving Microsoft SharePoint documents to P8.

## 7.6.4 Using deduplication for files

If you are using document deduplication when archiving files to P8, avoid executing the P8 Declare Record task when the document is a duplicate because the original has already been declared a record. In the event of an error during record declaration of a file, the system manages roll back of the original document and it will need to be collected again.

You can use a decision point to separate duplicates from those documents that are not duplicates and only include the P8 Declare Record task for the non-duplicates.

If the P8 Declare Record task is executed on a duplicate, it will attempt to declare the original, which will have already been declared a record. This will result in an unassociated record in the file plan. So it is best to avoid using the P8 Declare Record task on a route that is intended to process duplicates when collecting files.

## 7.6.5 Considerations for email

The Content Collector data model used for archiving email is more complex than the data model used for archiving files from a file system or Microsoft SharePoint. Content Collector stores a distinct email instance (DEI) document object and an email instance (EI) custom object to manage various aspects related to deduplication and mailbox retrieval in the P8 repository. In addition, an indexable XML component is stored to support content-based searching. See the IBM Information Center for more information about the Content Collector data model and how it is used for email:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/overview/c\\_afu\\_icc\\_data\\_model\\_p8.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/overview/c_afu_icc_data_model_p8.htm)

### Ensuring proper deletion

As soon as a new distinct email instance is archived, it can be declared as a record. The data model ensures that all other objects related to the distinct email instance are managed in such a way that the entire set of related objects will be deleted together when the DEI is deleted by Enterprise Records.

## **Handling intermittent failures during declaration**

Because the rollback policies for archived email are different than those used for archived files, it is a preferred practice to use the P8 Declare Record task when processing both new (unique) email instances and duplicates to handle possible failures during initial declaration, as described in 7.4.5, “Including logic to handle intermittent failure” on page 256. The P8 Declare Record task will not attempt to re-declare a duplicate email that was already successfully declared. However, if for some reason the initial declaration failed, the P8 Declare Record task will attempt to declare the distinct email instance when processing a duplicate email because it will recognize that the email has not yet been declared.

## **Changing security after declaration**

Because the email data model includes multiple components, if changes are made to security on the records in the file plan, depending on how those changes are made, the changes might not be propagated to all components of the email data model. Content Collector includes optional event handlers that manage changes to direct security on the record objects and the archived distinct email instance. However, changes to inherited security are not monitored and will only be applied to the distinct email instance through the Enterprise Records security proxy.

### **7.6.6 Considerations for large volumes**

When designing a solution for high-volume scenarios, it is generally advisable to be selective about what gets declared as a record. Although systems can be scaled to handle large volumes, there is always overhead when integrating with record declaration. Only content that has clearly been identified as having business value and context would typically be declared as a record. In other words, you know what it is and why you want to keep it as a record, and you have an established retention policy that can be applied to it.

For example, if you are archiving all versions of all documents from a Microsoft SharePoint site, you might only want to declare selected documents as records. Furthermore, you might only want to declare the most recent version as a record, and assign a Content Collector expiration date to all the older versions to be managed with simple retention.

Email archiving is another use case where you would typically want to avoid immediately declaring all emails as records, especially if you are archiving all email automatically and not relying on user selection for which emails to archive. As illustrated in our email archiving use case in this chapter, only certain emails with known business value of a certain type are declared. This avoids the added load and capacity that would be required on the system to support unnecessary record declaration. For example, it is best to avoid having large quantities of junk

or mass mailing messages in the Enterprise Records file plan, when those can be more efficiently be managed with a simple Content Collector expiration date.

Because email archiving solutions in large organizations are by nature significantly high volume, another strategy is to use the Content Collector expiration date exclusively upon archiving email, and only declare email as records during eDiscovery or some other process outside of content collection where specific emails are identified as critical or as having specific business value after they have been archived.

### **7.6.7 Declaring records after content has been archived**

There are scenarios where it might be desirable to archive the content without declaring records from the Content Collector task route and use some other means to declare records at a later time. This might be required if you do not have enough information at the time of collection and archiving to know whether a record should be declared, or if you do not have enough information to properly determine the record classification.

For example, a status change on a previously archived document, whether caused by a user update or an automated business process, might be a reason to trigger record declaration. A variety of mechanisms can be used to declare records, including a workflow subscription, a P8 CE event handler, or a bulk declaration utility that uses the API. See the Redbooks publication *Understanding IBM FileNet Records Manager*, SG24-7623, for more information about choosing the appropriate mechanism for declaring existing P8 content as records.

## **7.7 Conclusion**

Many organizations require integrating their content archiving solutions with enterprise records capability. In this chapter, we described several common approaches to accomplishing this goal by integrating IBM Enterprise Records with IBM Content Collector.



## IBM Connections integration

In this chapter we describe details about special considerations for archiving content from IBM Connections. We discuss the details of the setup for the IBM Connections system, assigning permissions for the seedlist user, and key considerations for high availability and eDiscovery Manager extension and export.

In this chapter we discuss the following topics:

- ▶ Configuring IBM Connections for IBM Content Collector
- ▶ Archiving a subset of content
- ▶ Configuring eDiscovery Manager for IBM Connections
- ▶ Adding additional index fields for specific content of IBM Connections documents

## 8.1 Configuring IBM Connections for IBM Content Collector

The IBM Connections Connector for IBM Content Collector requires a user account with sufficient rights to archive content from IBM Connections. To understand the necessary rights, we first need to look at the application programming interfaces (APIs) that are involved in archiving content from IBM Connections, as listed here:

- ▶ Seedlist SPI

The seedlist SPI provided by each application provides a stream of information about all content that resides within a application. Document addition, modification, deletions are reported back to the point where IBM Connections was installed. Creation and update events represent triggers that are picked up by IBM Content Collector for archiving a new version of a document. After no more changes are available, the seedlist will provide a time stamp to IBM Content Collector, which is saved and used to query for new changes. For more information about this topic, refer to the following site:

[http://www.lotus.com/ldd/lcwiki.nsf/dx/Seedlist\\_SPI\\_1c3](http://www.lotus.com/ldd/lcwiki.nsf/dx/Seedlist_SPI_1c3)

- ▶ Application API

For each creation or update, IBM Content Collector uses the respective application API to retrieve the primary document and all related parts from IBM Connections. The primary document might be a wiki page, whereas the related parts are attachments, comments, and version history information about the wiki page. For more information about this topic, refer to the following site:

[http://www.lotus.com/ldd/lcwiki.nsf/dx/Lotus\\_Connections\\_APIs\\_1c3](http://www.lotus.com/ldd/lcwiki.nsf/dx/Lotus_Connections_APIs_1c3)

This means IBM Content Collector needs access to the Seedlist SPI in addition to administrative access to all applications that are to be archived.

### 8.1.1 Setting up user permissions

In this section we explain how to set up the appropriate permissions for all IBM Connection applications. First, you create a separate user in IBM Connections that will be used by IBM Content Collector for archiving content. To configure permissions in IBM Connections, complete these steps:

1. Navigate to your WebSphere Integrated Solutions Console.
2. Log in.

3. Select **Applications** → **Application Types** → **WebSphere enterprise applications**.

Table 8-1 lists the IBM Connections applications.

Table 8-1 IBM Connections applications

IBM Connections application	Enterprise application name
Activities	Activities
Blogs	Blogs
Bookmarks	Dogear
Files	Files
Forums	Forums
Profiles	Profiles
Wikis	Wikis

For each of the IBM Connections applications that provide content:

- a. Select the **Application**.
- b. Click **Security role to user/group mapping** (Figure 8-1).

Figure 8-1 Activities enterprise application configuration

- c. Select the **Search-admin and the admin role** and click **Map users**.
- d. Search for the dedicated user you created for IBM Content Collector and select the user.
- e. Press **OK**.

You have now set up all the required permissions in IBM Connections to run IBM Content Collector.

## 8.1.2 Scale out and backup considerations

The seedlist passes a list of documents to the IBM Connections Connector for processing. If single entries fail to be processed, there is no way to retrieve them from the seedlist during the next collection. The collector avoids querying the same seedlist segment again, because this will lead to a high ratio of duplicate documents being created in the ECM system.

To solve this problem, the IBM Connections Connector will write a recovery file for each document that is retrieved from the seedlist. If processing of this document is successful, the recovery file is deleted. If processing fails, the recovery file is kept and used for another processing attempt during the next collection. This means that in addition to the seedlist time stamp file, the recovery files also need to be backed up to perform disaster recovery.

In a scale out environment, there needs to be a common storage for recovery files and the seedlist file. A file share is to be provided and used as a seedlist directory. Make sure the performance of the file share is adequate and does not cause a bottleneck.

## 8.2 Archiving a subset of content

It is advisable to archive the complete content of IBM Connections. Due to the nature of the seedlist, a changed filter will simply affect content that is created in IBM Connections from this moment on. This might be acceptable, depending on your scenario. If content from the past should also be archived, then special considerations are needed. For more information, see:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/collectors/r\\_afu\\_collection\\_sources\\_cx.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/collectors/r_afu_collection_sources_cx.htm)

Assume the company archives content from all members of the board of directors created within the connection system. All content that is created, updated, or commented on by these specific users is captured. Now a new member to the board of directors is appointed and consequently the list needs to



be changed to include that additional user. As a result, all content created by that user day-forward will be archived, but content that was created previously will not be captured unless a separate task route is set up to capture the past content for this user.

## 8.2.1 Content filtering for IBM Connections

The IBM Connections Connector supports filtering content based on user display names and application types (Figure 8-2).

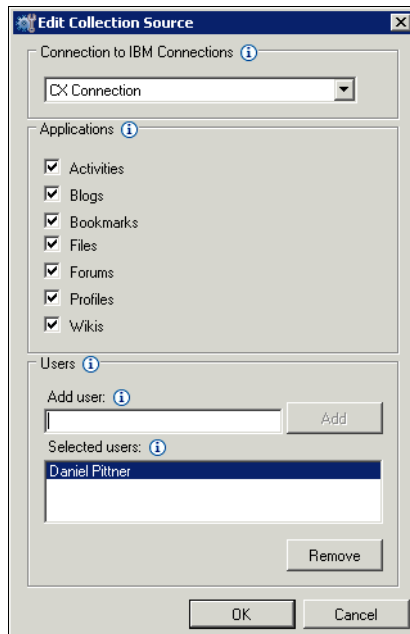


Figure 8-2 IBM Connections Collector options

Further filtering can be performed by using decision points in a task route. Task route-based filtering is based on the available metadata that is extracted by the collector or the pre-processing task.

More information about metadata extracted by the collector is available at:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/metadata\\_and\\_lists/r\\_afu\\_system\\_metadata\\_cx\\_collection.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/metadata_and_lists/r_afu_system_metadata_cx_collection.htm)

More information about metadata extracted by the pre-processing task is available at:

[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/metadata\\_and\\_lists/r\\_afu\\_system\\_metadata\\_cx\\_preprocessing.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/topic/com.ibm.content.collector.doc/metadata_and_lists/r_afu_system_metadata_cx_preprocessing.htm)

In the next section we introduce an example for task route-based filtering.

## 8.2.2 Archiving past content from a specific person

Assume that the new person who joined the board of directors is named Bob.

To archive past content that Bob created before he joined the board, first deploy the “CX to P8 - Archive” template using the New Task route option.

After the import completes, complete the following steps to customize the task route for the scenario:

1. Select the collector; the General tab will be selected.
  - a. Mark the collector as active.
2. Select the **Schedule** tab.
  - a. Set the Schedule to run at intervals, with running endlessly, starting today, running each day starting at 7:00 p.m., stopping at latest at 5:00 a.m.  
Based on the scenario, running the collection one time will be sufficient.
3. Select the **Collection Sources** tab.
4. Add the connection to your IBM Connections system as the collection source.
5. Select all applications to be archived.
6. Add only Bob to the user filter.
7. Add a new decision point before the first task.
8. Configure the rule to archive all documents that were updated before the date Bob was promoted (Figure 8-3).

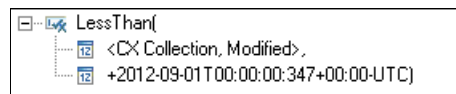


Figure 8-3 Decision to filter all documents updated before September 1

9. Configure the second rule to ignore all documents that were updated after Bob was promoted. Those documents will be captured by the task route that archived the complete board of directors.
10. Save the task route.

The task route is now set up correctly and can be promoted to production after verification (Figure 8-4).

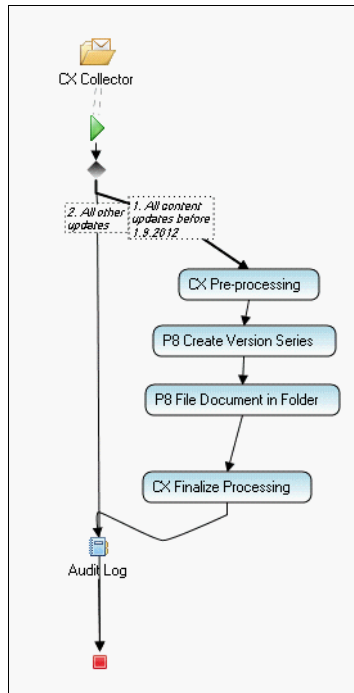


Figure 8-4 Task route for archiving all content of a person up to a specific date

The task route will be used to archive all content that Bob created in the past. After this archiving completes successfully, the task route can be deleted. Further updates will be handled by the general task route for the board of directors.

## 8.3 Configuring eDiscovery Manager for IBM Connections

IBM eDiscovery Manager supports IBM Connections content starting with release 2.2. Fix Pack 4. Starting with release 2.2.1, there is a predefined collection type for IBM Connections. When using a previous release, a user-defined collection needs to be created for IBM Connections content.

For specific details about how to set up the collection and search-mapping for IBM Connections, read the Technote that describes the necessary setup steps in eDiscovery Manager, available at the following site:

<http://www.ibm.com/support/docview.wss?uid=swg21595873>

### 8.3.1 Viewing IBM Connections documents in eDiscovery Manager

IBM Content Collector stores documents archived from IBM Connections in a ATOM-based XML format that is self-describing, but has no similar visual appearance. To provide a similar look and feel, eDiscovery Manager uses Extensible Stylesheet Language (XSL) to render from XML to HTML that has a similar look and feel as viewing the content from within IBM Connections. The rendering is built from the default theme and visual appearance of IBM Connections 3.0.1. If you customized your IBM Connections deployment, you might also want to customize the XSL used with eDiscovery Manager.

#### **Adding customization to IBM Connections viewing**

A customization example is an additional attribute in the profiles application, as described in the “Customizing IBM Connections 3.0.1” publication:

<http://www.redbooks.ibm.com/Redbooks.nsf/RedbookAbstracts/tips0854.html?Open>

By implementing this customization, IBM Content Collector will automatically archive and index the additional profiles field, but the viewing integration does not know how to display the value.

However, you can easily customize the XSL files to add rendering capabilities for customization of any kind. To add the “contractAgency” field to the profiles rendering, follow these steps:

1. Navigate to your eDiscovery Manager config directory and locate the folder named “Connections.”
2. Locate the file `profiles.xsl` within this folder.
3. Create a backup copy of the file in case you want to revert later.
4. Open `profiles.xsl` with a editor.
5. Locate the section that renders the contact information section (starting at line 700).

- Duplicate the section containing display information for one attribute (lines 719 - 726, as shown in Figure 8-5).

```
718 </td>
719 </tr><tr>
720 <th scope="row"> Role:
721 </th>
722 <td>
723 <p> <xsl:value-of select="high:highlight($highlighter,atom:entry/atom:co
724 </p>
725 </td>
726 </tr>
727 <tr>
```

Figure 8-5 Section for rendering a contact information attribute

- Change the field label to Contract Agency.
- Change the select statement to:  
`high:highlight($highlighter,atom:entry/atom:content/xhtml:div/xhtml:span/xhtml:div[@class='contractAgency'])`
- Restart your eDiscovery Manager instance.
- Search and view a profiles document to verify that the output looks as you intended.

### 8.3.2 Extending searching capabilities

By default, eDiscovery Manager provides you with the ability to search for all content of IBM Connections documents and attachments. However, if you have specific search requirements that cannot be fulfilled with the default search configuration, you can customize the configuration to provide search fields for your specific scenario.

Assume the company uses IBM Connections activities to track different tasks in the HR department. For this purpose a specific template is used that has a custom field labelled **Responsible person** (Figure 8-6).

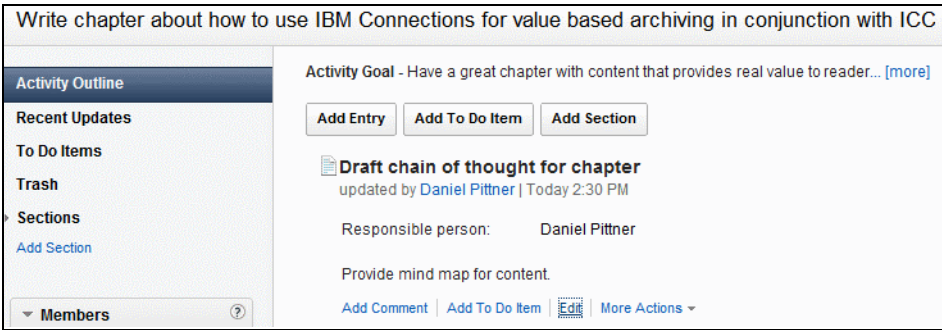


Figure 8-6 IBM Connections activity with responsible person field

In this example, company lawyers want to search for all content that a certain person was responsible for according to the field value. Searching using the default settings will retrieve these documents, because all content is indexed. However, documents in which this person was mentioned in other fields will also be found, leading to a huge overhead. Looking at the ATOM/XML that is archived, the field is easy to identify (Figure 8-7). Thus, the company decides to add a specific search field to eDiscovery Manager that allows for searching on the **Responsible person** field.

```
<snx:field name="Responsible person" fid="FFFG5749781881c240a9a1eb967a0f586399" position="1000" type="text" >
  <summary type="text">Daniel Pittner</summary>
</snx:field>
```

Figure 8-7 Adding search field for Responsible person

### Extending the IBM FileNet P8 Content Search Services index

If your content is stored in IBM FileNet P8 and indexed using IBM Content Collector Content Search Services Support, several simple steps are required to add a specific search field for the Responsible person IBM Connections field. Because the complete XML-based content is indexed, no configuration change to the indexing engine is necessary. Changing the eDiscovery collection and search template is sufficient; simply complete these steps:

1. Open eDiscovery Manager administration.
2. Select the collection you defined for IBM Connections.
3. Add a new collection field:
  - a. Assign the name PERSON\_RESPONSIBLE.
  - b. Leave the Content Server Property area blank.

- c. Set the Type to String.
- d. Text Index is the xpath to the element in question.
- e. Enter the xpath `//field[@name="Person responsible"]` as value.
- f. Save the collection (Figure 8-8).

Collection Field	Content Server Property	Type	Text Index
MODIFIER		String	//modifier/name
NOT_COMMENT_CONTENT		String	//icc_part[@mimetype!="application/icc-comment-atom+xml"]
PERSON_RESPONSIBLE		String	//field[@name="Person responsible"]

Figure 8-8 Responsible person collection field

4. Select the search mapping you defined for IBM Connections.
5. Check the Person responsible search field to be visible.
6. Save the search mapping.
7. Log in and out of eDiscovery Manager to see the new search field, as shown in Figure 8-9.

Figure 8-9 Search mask with Person responsible search field added

### Extending the IBM Content Manager index

If IBM Content Manager in conjunction with the IBM Content Collector Text Search component is used, you need to perform multiple changes to add a specific search field for Person responsible. For details about the general process of customizing, configuring, and troubleshooting IBM Content Collector Text Search component, see the indexing guide:

<http://www.ibm.com/e-business/linkweb/publications/servlet/pbi.wss?CTY=US&FNC=SRX&PBL=SH12-6979-00>

Similar steps can also be performed during the creation of the index.

### Extending an existing index

1. Add the attribute to the Net Search Extender (NSE) model file.

**Changing the attribute description:** The attribute description can only be changed by recreating the NSE full text index.

2. Create a copy of the currently used NSE model file and name the copy `afu_conn_extended.xml`.
3. Add an NSE field for Person responsible (also see “Adding attributes to the model file” in the indexing guide):

```
<XMLFieldDefinition name="icc_personResponsible" exclude="NO"
locator="/icc_document/icc_connections/icc_content/icc_personResponsible" />
```
4. Save the model file.
5. Add the attributes to the item type configuration file:
  - a. Locate the configuration file for the item type (also see “Adding attributes to the configuration file of the item type” in the indexing guide).

**Mapping information:** For each mime-type there is a section in the `ini` file that describes how to map the XML content to the intermediate XML that is produced for NSE. NSE will map the intermediate XML to index fields as configured in the model file.

- b. Locate the `[MIME-TYPE:application/icc-activity-atom+xml]` section.
- c. Add an entry for the person responsible field:

```
//atom:feed/atom:entry/snx:field[@name="Person responsible"] ->
/icc_document/icc_connections/icc_content/icc_personResponsible
```

**Xpath statement:** The Xpath statement on the left side of the assignment operator is evaluated against the content of the IBM Connections document, and the result is mapped to the Xpath of the intermediate document given on the right side of the assignment operator. Namespaces must be defined in the global `[NAMESPACES]` section.

- d. Save the configuration file.
6. Use the **afuRecreateIndex** tool and specify the model file you modified as new model for the new NSE full text index.
  - a. Specify the command line argument `-modelfile afu_conn_extended.xml` to specify the new model file.
  - b. Re-index the content of the item type by running **afuIndexer**.



**Querying consideration:** Recreating the index for an item type might take a considerable amount of time, depending on the amount of data. While the index is being recreated, your ability to query for content of the item type for eDiscovery is limited to the documents that have already been re-indexed.

Perform the following customization for IBM eDiscovery Manager to consume the newly defined index field:

1. Open eDiscovery Manager administration.
2. Select the collection you defined for IBM Connections.
3. Add a new collection field:
  - a. Assign the name `PERSON_RESPONSIBLE`.
  - b. Leave the Content Server property blank.
  - c. Set the type to `String`.
  - d. Set the text index field to `icc_personResponsible`.
  - e. Save the collection.
4. Select the search mapping you defined for IBM Connections.
5. Check the **Person responsible** search field to be visible.
6. Save the search mapping.
7. Log out and log back in to eDiscovery Manager to view the new search field.

### 8.3.3 Considerations for eDiscovery Manager document export

IBM eDiscovery Manager can export IBM Connections documents either in native ATOM/XML format or in HTML format. Because interlinked content is not exported implicitly, you might want to export related content also. An example for such a situation is the export of wiki pages. Assume you are exporting wiki pages pertaining to a certain legal case. Because within IBM Connections wiki page authors are cross-linked with their profiles, you might want to include the profiles in the search and export of documents.

As with the current release, pdf export is not supported. Instead, a third-party tool can be used to convert the exported HTML files into pdf format, if needed.

### **8.3.4 Conclusion**

In this chapter we explained the details of how to set up the archiving of IBM Connections content for compliance purposes. We outlined key considerations for backup and ensuring good performance. We also demonstrated how to extend eDiscovery Manager search and viewing capabilities for IBM Connections to match your specific requirements or IBM Connections customization.

# Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

## IBM Redbooks

The following IBM Redbooks publications provide additional information about the topic in this document. Note that some publications referenced in this list might be available in softcopy only.

- ▶ *IBM Classification Module: Make It Work for You*, SG24-7707  
(IBM Content Classification was formerly known as IBM Classification Module)
- ▶ *Understanding IBM FileNet Records Manager*, SG24-7623
- ▶ *IBM Content Collector Integration with IBM Classification Module*, REDP-4705

<http://www.redbooks.ibm.com/redpapers/pdfs/redp4705.pdf>

You can search for, view, download or order these documents and other Redbooks, Redpapers, Web Docs, draft and additional materials, at the following website:

[ibm.com/redbooks](http://ibm.com/redbooks)

## Online resources

These websites are also relevant as further information sources:

- ▶ IBM Content Collector Information Center  
<http://pic.dhe.ibm.com/infocenter/email/v3r0m0>
- ▶ IBM Content Classification Information Center  
<http://pic.dhe.ibm.com/infocenter/classify/v8r8>
- ▶ Defensible Disposal Library  
<http://www.ibm.com/software/ecm/disposal-governance/library.html>

- ▶ White paper “Decision Plan best practices”  
<http://www.ibm.com/support/docview.wss?uid=swg27023248>
- ▶ White paper “Achieve compliance and control costs with automatic categorization of email for records management”  
<http://public.dhe.ibm.com/common/ssi/ecm/en/zzw03051usen/ZZW03051USEN.PDF>
- ▶ Integrating Content Classification into Content Collector  
[http://pic.dhe.ibm.com/infocenter/email/v3r0m0/index.jsp?topic=%2Fcom.ibm.content.collector.doc%2Ficm%2Ft\\_afu\\_enabling\\_rc\\_icm\\_integration.htm](http://pic.dhe.ibm.com/infocenter/email/v3r0m0/index.jsp?topic=%2Fcom.ibm.content.collector.doc%2Ficm%2Ft_afu_enabling_rc_icm_integration.htm)
- ▶ Technote: Apply IBM Classification Module to email archiving  
<http://www.ibm.com/developerworks/data/library/techarticle/dm-0812chitivel/index.html>
- ▶ Detailed information about Content Collector for SAP Applications is beyond the scope of this book, but more information is available online at:  
<http://www.ibm.com/software/data/content-management/content-collector-sap>

## Help from IBM

IBM Support and downloads

[ibm.com/support](http://ibm.com/support)

IBM Global Services

[ibm.com/services](http://ibm.com/services)



## Creating Value-Based Archiving Solutions with IBM Content Collector

(0.5" spine)  
0.475" <-> 0.875"  
250 <-> 459 pages







# Creating Value-Based Archiving Solutions with IBM Content Collector

**Content archiving and retention management with use cases**

**Integration with IBM Content Classification**

**Integration with IBM Enterprise Records**

This IBM Redbooks publication describes how the IBM Content Collector family of products can help companies to create value-based archiving solutions. IBM Content Collector provides enterprise-wide content archiving and retention management capabilities. It also provides IT administrators with a high level of control over the archiving environment. From a common interface, organizations can implement policies that define what gets archived from which source system, make decisions about how content gets archived based on the content or metadata of the information, and determine the retention and governance rules associated with that type of content. Content Collector enables IT staff to implement granular archiving policies to collect and archive specific pieces of information.

IBM Content Collector helps with the following tasks:

- ▶ Eliminating point solutions and lowering costs with a unified collection, management, and governance approach that works effectively across a broad range of source systems and information types
- ▶ Appraising, improving understanding of, culling, and properly selecting the information to archive
- ▶ Retaining, holding, and disposing of archived content efficiently and defensibly
- ▶ Eliminating the costs and risks inherent with over-retention

This book covers the basic concepts of the IBM Content Collector product family. It presents an overview explaining how it provides value-based archiving and a defensible disposal capability in the archiving solutions. With the integration of IBM Content Classification and IBM Enterprise Records, the book also showcases how these products can be used to add more flexibility, power, and capabilities to archiving solutions.

The book is intended for IT architects and solution designers who need to understand and use IBM Content Collector for archiving solution implementations. Use cases are included to provide specific, step-by-step details about implementing common solutions that fulfill some of the general business requirements.

**INTERNATIONAL  
TECHNICAL  
SUPPORT  
ORGANIZATION**

**BUILDING TECHNICAL  
INFORMATION BASED ON  
PRACTICAL EXPERIENCE**

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

**For more information:**  
[ibm.com/redbooks](http://ibm.com/redbooks)