

Advanced Case Management with IBM Case Manager

Understanding case management
and IBM Case Manager

Developing IBM Case Manager
solution with use case example

Integrating with external
products and components



Wei-Dong Zhu
Michael Kirchner
Tom Ko
Michael Oland
Balunaini Prasad
Mike Prentice
Monique Ruggiero

Redbooks



International Technical Support Organization

**Advanced Case Management
with IBM Case Manager**

May 2013

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

Third Edition (May 2013)

This edition applies to Version 5.1.1, IBM Case Manager (product number 5725-A15).

© Copyright International Business Machines Corporation 2011, 2012, 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

Notices	xiii
Trademarks	xiv
Preface	xv
The team who wrote this book	xv
Now you can become a published author, too!	xvii
Comments welcome	xviii
Stay connected to IBM Redbooks	xviii
Summary of changes	xxi
May 2013, Third Edition	xxi
May 2012, Second Edition	xxiii
Part 1. Concept and overview	1
Chapter 1. Case management concept	3
1.1 Case management overview	4
1.2 What makes case management unique	6
1.2.1 Case Management Solutions versus BPM Solutions	6
1.2.2 Cases are goal driven and unpredictable	9
1.2.3 Cases are knowledge intensive	10
1.2.4 Modeling	11
1.2.5 Tasks	11
1.2.6 Routine work and knowledge work	12
1.3 The need for Enterprise Content Management	13
1.3.1 Information complexity	13
1.3.2 Security	14
1.3.3 Retention	14
1.4 The need for collaboration	15
1.5 Case management solutions	16
1.5.1 Case data	17
1.5.2 Case folder	17
1.5.3 Tasks	19
Chapter 2. Typical case management applications	21
2.1 Typical applications	22
2.1.1 Solution patterns	22
2.1.2 Applications by industry	23
2.2 Complaints management example use case	25

2.2.1 Complaint scenario	26
2.2.2 Resolving the case	27
2.2.3 Content related to the case	28
2.2.4 Integration with other systems	28
Chapter 3. IBM Case Manager overview	31
3.1 IBM Case Manager capabilities	32
3.1.1 Case design and deployment	32
3.1.2 Case management	32
3.1.3 IBM Case Manager API	32
3.1.4 Tools	32
3.1.5 Solution templates	33
3.1.6 Content management	33
3.1.7 Business process management	33
3.1.8 Collaboration and social software tools	33
3.1.9 Business rules	34
3.1.10 Case analytics	34
3.1.11 Reporting	34
3.1.12 Content Analytics	34
3.1.13 Forms integration	35
3.2 IBM Case Manager environments	35
3.2.1 Environments and case management phases	35
3.2.2 Solutions and deployment in development environment	37
3.3 IBM Case Manager components	38
3.3.1 IBM Case Manager core components	40
3.3.2 IBM Case Manager optional components	47
3.3.3 IBM Case Manager extensions	49
3.4 IBM Case Manager architecture	51
3.4.1 Case Manager Builder	52
3.4.2 Case Manager Client	52
3.4.3 Case Manager API	52
3.4.4 Content Engine	53
3.4.5 Process Engine	53
3.5 IBM Case Manager configurations	53
3.5.1 Development environment configuration	54
3.5.2 Production environment with high availability	57
3.5.3 Production environment with partitioning	61
Chapter 4. Inner workings of IBM Case Manager	65
4.1 IBM Case Manager object model	67
4.1.1 Solution	68
4.1.2 Case types	71
4.1.3 Roles	75

4.1.4 Roles in-basket	75
4.1.5 Personal in-baskets	76
4.1.6 Configuring manual work assignment	78
4.1.7 Document types	85
4.1.8 Task type	86
4.1.9 Tasks	92
4.2 Case object model implementation	97
4.2.1 Design object store	97
4.2.2 Target environment	99
4.2.3 Project areas for supporting parallel project development	101
4.2.4 Solution deployment process	103
4.3 IBM Case Manager spaces and pages	108
4.3.1 Default pages in Solution space	109
4.3.2 Default pages in Case page space	109
4.3.3 Default pages in Task page space	110
4.3.4 IBM Case Manager widgets	111
4.4 Tasks and associated workflow processes	111
4.4.1 Workflow association	111
4.4.2 Workflow data fields	112
4.4.3 Workflow attachments	116
4.4.4 Step parameters and post assignments	116
4.4.5 Starting task workflows	116
4.4.6 Workflow processing and task state	117
4.5 Object model for IBM Content Manager integration	118
Part 2. Solution development	121
Chapter 5. Designing case management solutions	123
5.1 Business goals of a case management solution	124
5.2 Designing the solution	124
5.2.1 Agile and iterative methodologies	125
5.2.2 Case design	126
5.2.3 Task Design	128
5.2.4 Case documentation	129
5.2.5 Business rules	130
5.2.6 Modeling how cases are to be created	130
5.3 Implementing the solution	131
5.4 Level 1: Defining your solution	132
5.4.1 Defining case properties	133
5.4.2 Defining case types	133
5.4.3 Defining document types	134
5.4.4 Defining roles	134
5.4.5 Defining in-baskets	134

5.4.6 Defining tasks	135
5.5 Level 2: Refining your solution with advanced configuration.	141
5.5.1 Advanced in-basket configuration	143
5.5.2 Changing the object model in the solution	144
5.5.3 Using forms to enhance user interfaces and data collection.	145
5.5.4 Configuring the security definitions	147
5.5.5 Configuring pages for advanced user interfaces.	148
5.5.6 Configuring decision services with iLog JRules	149
5.5.7 Configuring case analytics and reporting	150
5.5.8 Configuring the Cognos Real-Time Monitor	152
5.6 Level 3: Customizing and integrating your solution.	153
5.7 Documenting your solution	156
Chapter 6. Building a simple solution: Part 1	159
6.1 Building the sample solution overview	160
6.2 Artifacts to be implemented in the sample solution.	160
6.2.1 Case types	161
6.2.2 Tasks.	161
6.2.3 Roles	162
6.2.4 In-baskets	163
6.2.5 Documents	164
6.2.6 Properties	164
6.2.7 Property choice lists	166
6.3 Creating the Customer Complaint solution	166
6.4 Setting up and configure artifacts for the solution.	178
6.4.1 Setting up properties	179
6.4.2 Setting up roles	183
6.4.3 Setting up In-baskets	184
6.4.4 Setting up document types	186
6.4.5 Setting up case type	187
6.5 Setting up tasks for the solution	195
6.5.1 Creating the Upsell Opportunity container task.	196
6.5.2 Editing the Upgrade Product task	198
6.5.3 Editing the Upgrade Plan task.	202
6.5.4 Editing the Call Customer task	202
6.5.5 Editing the Verify Billing task.	203
6.5.6 Editing the Verify Complaint task	204
6.5.7 Editing the Review Product Complaint task	205
6.5.8 Editing the Review Non-Product Complaint task	206
6.5.9 Editing the Investigate Product Safety task.	206
6.5.10 Editing the Send Corresponding Letter task	207
6.5.11 Editing the Close Complaint task	209
6.5.12 Editing the Investigate Employee task	209

6.5.13 Editing the Request Assistance task	210
6.5.14 Editing task summary	210
Chapter 7. Building a simple solution: Part 2	213
7.1 Configuring workflow diagrams by using Step Editor	214
7.1.1 Creating the Upgrade Product task diagram.....	214
7.1.2 Creating the Upgrade Plan task diagram	222
7.1.3 Creating the Call Customer task diagram	223
7.1.4 Creating the Verify Billing task diagram	224
7.1.5 Creating the Verify Complaint task diagram	227
7.1.6 Creating the Review Product Complaint task diagram	228
7.1.7 Creating the Review Non-Product Complaint task diagram	229
7.1.8 Creating the Investigate Product Safety task diagram	230
7.1.9 Creating the Send Corresponding Letter task diagram.....	232
7.1.10 Creating the Close Complaint task diagram	237
7.1.11 Creating the Investigate Employee task diagram	238
7.1.12 Creating the Request Assistance task diagram	241
7.2 Saving and validating the solution.....	243
7.3 Deploying the Customer Complaints solution	245
7.4 Testing the Customer Complaints solution	246
7.5 Improving the solution	257
7.5.1 Solution icon	257
7.5.2 Property choice list	257
7.5.3 Solution properties	257
7.5.4 Document properties	258
7.5.5 In-basket properties	258
Chapter 8. Solution deployment	259
8.1 Deploying a solution	260
8.1.1 Tools for migration	261
8.1.2 Effects of redeployment	262
8.2 Prerequisite steps	263
8.2.1 FileNet Deployment Manager	263
8.2.2 Case Manager Administration Client (CMAC).....	266
8.2.3 Content Engine	267
8.3 Identification	269
8.4 Migrating external assets and processes	270
8.5 Migrating non-Case-Manager FileNet-based assets	270
8.5.1 Half-maps	271
8.5.2 Exporting the assets	272
8.5.3 Building the target half map	277
8.5.4 Creating the source-destination mappings	279
8.5.5 Importing the assets	281

8.5.6 Analyze the dataset.....	281
8.5.7 Disconnected deployment.....	284
8.6 Migrating a Case Manager solution.....	285
8.6.1 Exporting solution package.....	285
8.6.2 Importing the Case Manager solution.....	286
8.6.3 Deploying the solution to the target object store.....	286
8.6.4 Post deployment changes.....	287
Chapter 9. Solution development	289
9.1 Solution development	290
9.1.1 Roles	290
9.1.2 Team collaboration	291
9.1.3 Tracking and packaging solution assets	294
9.2 Security considerations	299
9.2.1 Security concepts	300
9.2.2 Security example.....	308
9.3 Solution change and configuration management	314
9.3.1 Solution documentation.....	315
9.3.2 Source code and release management	315
9.4 Sandbox options	317
9.4.1 Copying a solution.....	317
9.4.2 Copying a solution by using Case Manager Builder	318
9.4.3 Copying a solution by using CMAC.....	320
9.4.4 Extra options for an IBM Case Manager administrator	324
9.4.5 Creating and using solution templates	326
9.4.6 Creating solution templates.....	326
9.4.7 Using solution templates.....	329
9.4.8 Available solution templates	333
Chapter 10. User interface	335
10.1 Using default widgets and their payloads	336
10.1.1 Standard widgets provided by IBM Case Manager.....	336
10.1.2 Widgets payload	337
10.1.3 Wiring a widget to another widget.....	338
10.1.4 Payload types of Case Manager Client widget	341
10.1.5 In-basket widget event and corresponding payload	342
10.1.6 Case Search widget event and corresponding payload	345
10.2 Creating and deploying a custom iWidget.....	346
10.3 Manipulating pages in business space	352
10.3.1 Security	352
10.3.2 Default pages	353
10.3.3 Custom pages	354
10.4 Example of customizing the Investigator Step	357

10.5	Changing the runtime appearance	370
10.5.1	Changing space user interface style	370
10.5.2	Changing page user interface style	371
10.5.3	Changing widget user interface style	371
10.6	User interface navigation and case accessing	372
10.6.1	Navigation	372
10.6.2	Accessing cases with case link	373
10.7	Managing documents in context of a case	375
10.7.1	Integration of the IBM Content Navigator viewer	375
10.7.2	Manually adding documents	383
10.7.3	Adding case documents to a task	387
10.8	Using the Case Form widget	389
10.8.1	View and edit modes	390
10.8.2	Overview of data merging rules	392
10.8.3	Configuring Case Form widget-related pages	393
10.8.4	Configuring Case Form widget	396
10.8.5	Using Case Form with FileNet eForm	401
10.9	Using IBM Forms with IBM Case Manager	402
10.9.1	IBM Forms overview	402
10.9.2	Installing the IBM Forms Server	405
10.9.3	Configuring IBM Forms with IBM Case Manager	406
10.9.4	Installing IBM Forms Designer	407
10.9.5	Integrating IBM Forms with IBM Case Manager	407
Chapter 11.	Development topics	413
11.1	Using project areas when developing solutions	414
11.1.1	Creating the default project area	416
11.1.2	Creating a project area	417
11.1.3	Assigning solutions to a project area	419
11.1.4	Removing solutions from a project area	420
11.1.5	Assigning users or groups to a project area	421
11.1.6	Removing users or groups from a project area	423
11.1.7	Removing a project area	424
11.1.8	Determining the project area that you are in	425
11.1.9	Summary	426
11.2	Automated handling of ingested documents	427
11.2.1	Generalized approach for handling ingested documents	427
11.2.2	Implementation methods	428
11.2.3	Sample Javascript-based implementation procedure	428
11.3	Splitting a case	433
11.3.1	Configuring the case toolbar to show the Split Case button	434
11.3.2	Splitting a case into two cases	439
11.3.3	Registering the Split Case page for upgraded systems	441

11.4 Reuse of an existing FileNet workflow	441
11.4.1 Reusing an existing FileNet process as a case task.....	443
11.4.2 Native tasks and tasks by using reused processes comparison..	454
11.5 Multilingual support	455
11.5.1 Use cases for Case Builder designer	456
11.5.2 Use cases for Case Client user.....	456
11.5.3 Business Space localization	456
11.5.4 Process Engine element localization	457
11.5.5 Content Engine element localization.....	459
11.5.6 IBM Content Manager element localization	460
11.5.7 FileNet eForms element localization.....	461
11.5.8 IBM Forms element localization	463
11.5.9 Toolbar, Case toolbar, and Work Item toolbar localization	464
11.5.10 Multilingual limitations.....	466
Chapter 12. Round-tripping workflow editing.....	467
12.1 Introducing round-tripping	468
12.2 Adding CE_Operations in a round-tripping edit.....	468
12.2.1 Modeling in Case Manager Builder.....	469
12.2.2 Round-tripping to Process Designer	476
12.2.3 Round-tripping back to Case Manager Builder.....	485
12.3 Creating more in-baskets	486
12.4 Exposing error handling in the Case Manager Client	495
Part 3. Integration	509
Chapter 13. Integration points	511
13.1 REST APIs	512
13.1.1 Case Manager REST API.....	512
13.1.2 Process Engine REST API	513
13.1.3 Content Management Interoperability Services API	514
13.2 Using Java APIs	514
13.2.1 Integrating by using Content Engine Java API	515
13.2.2 Integrating by using Process Engine Java API	516
13.2.3 Integrating by using the Case Java API	517
13.3 Using Case Manager Client side integration.....	518
13.3.1 Web Site widget	518
13.3.2 Script Adapter widget	518
13.3.3 Custom widget	519
13.4 Using Process Engine workflow integration	519
13.4.1 Custom components	520
13.4.2 Web services.	520
13.4.3 Database interaction	521
13.5 External data service.....	521

Chapter 14. Integration with WebSphere ILOG JRules	523
14.1 Benefits of using rules engine software	524
14.2 Incorporating a business rule in the case solution	525
14.2.1 Getting started: Creating a JRules project	526
14.2.2 Design: Defining the business model for your rules (XOM)	527
14.2.3 Orchestrate: Creating a rule package and rule flow	533
14.2.4 Author: Creating the business rules	534
14.2.5 Deploy: Creating the rule application project	537
14.2.6 Integrate: Configuring web services to start the rule	541
Chapter 15. Integration with IBM Content Manager	549
15.1 Benefits of Content Manager integration	550
15.2 Architecture of the integration	550
15.2.1 Overview of the architecture	551
15.2.2 Interaction between IBM Case Manager and Content Manager	552
15.3 Integration configuration steps	553
15.3.1 Prerequisites for integration	554
15.3.2 Overview of the configuration steps	554
15.3.3 Detailed configuration steps on Content Manager server	555
15.3.4 Detailed configuration steps on IBM Case Manager server	563
15.3.5 Starting event monitor and case handler on Content Manager	568
15.4 Using Content Manager documents in case solutions	569
15.4.1 Using project area configured with Content Manager	569
15.4.2 Using document type from the Content Manager item types	570
15.4.3 Adding Content Manager documents in cases	570
15.4.4 Security implications	571
15.4.5 Resetting the test environment	571
15.4.6 Troubleshooting	571
Chapter 16. External Data Service Framework	573
16.1 Implementing an External Data Service	574
16.1.1 Sequence for case creation using external data service	576
16.1.2 Implementing the POST method for external data service	578
16.1.3 Request parameters in the POST payload	579
16.1.4 Case data persistence	580
16.1.5 Response payload and HTTP response codes	581
16.2 Registering the External Data Service	583
Chapter 17. Integration with IBM Business Process Manager	587
17.1 Introduction to IBM Case Manager and IBM BPM	588
17.1.1 IBM BPM components	588
17.1.2 IBM Case Manager integration for IBM BPM	590
17.2 Implementing a task as IBM BPM process application	590
17.2.1 Configuring IBM Case Manager for IBM BPM integration	591

17.2.2 Reusing an IBM BPM process definition.....	592
17.2.3 Mapping a task to business process definition	596
17.3 Implementing a task as an IBM BPM automated process.....	599
17.4 Configuring an Integrated Inbox	600
17.4.1 Setting up business space for IBM Case Manager.....	601
17.4.2 Creating a saved search query in IBM BPM Process Portal.....	603
17.4.3 Configuring In-baskets in the Customer Complaints solution	605
17.4.4 Adding Integrated Inbox to the Customer Complaints solution	607
17.4.5 Configuring the Inbox window for the integrated space	608
17.4.6 Configuring the Work on tasks window for the integrated space ..	612
Chapter 18. Integration with IBM Content Analytics	617
18.1 IBM Content Analytics overview	618
18.2 Installing Content Analytics Version 2.2	619
18.2.1 Installing Content Analytics	620
18.2.2 Installing Content Analytics V2.2 FixPack 2	624
18.3 Installing IBM FileNet Content Engine Client	626
18.4 Configuring script to resolve Content Engine Client paths	628
18.5 Configuring crawlers to access case solutions	629
18.5.1 Starting the administration console and creating collections	629
18.5.2 Configuring crawler and parser for the case solution	632
18.5.3 Gathering and parsing the case artifacts	644
18.6 Rapid insights detection with the Text Mining application	645
18.6.1 Starting the Text Mining application	646
18.6.2 Default facets of case solutions	647
18.6.3 Representation of case properties identified at the crawl time ..	652
18.6.4 Insights that are related to customer service	655
Related publications	663
IBM Redbooks	663
Online resources	664
Help from IBM	664

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 about the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

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

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

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.

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.

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:

AIX®
Cognos®
DB2®
Domino®
FileNet®
IBM®

ILOG®
Jazz™
Lotus®
Lotus Notes®
Notes®
Quickr®

Rational®
Redbooks®
Redbooks (logo)  ®
Sametime®
WebSphere®

The following terms are trademarks of other companies:

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

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

Organizations face case management challenges that require insight, responsiveness, and collaboration. IBM® Case Manager, Version 5.1.1, is an advanced case management product that unites information, process, and people to provide the 360-degree view of case information and achieve optimized outcomes. With IBM Case Manager, knowledge workers can extract critical case information through integrated business rules, collaboration, and analytics. This easy access to information enhances decision making ability and leads to more successful case outcomes. IBM Case Manager also helps capture industry best practices in frameworks and templates to empower business users and accelerate return on investment.

This IBM Redbooks® publication introduces the case management concept. It includes the reason for and benefits of case management, and why it is different from the traditional business process management or content management. In addition, this book addresses how you can design and build a case management solution with IBM Case Manager, and integrate that solution with external products and components.

This book is intended to provide IT architects and IT specialists with the high-level concepts of case management and the capabilities of IBM Case Manager. In addition, it serves as a practical guide for IT professionals who are responsible for designing, building, and deploying IBM Case Manager solutions.

The team who wrote this book

This book was produced by a team of specialists from around the world working at IBM Software Development Lab in Costa Mesa, California.

The Redbooks publication team consists of the following members:

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 the Southern California.

Michael Kirchner is a Senior Solutions Consultant for IBM Enterprise Content Management Technical Sales in Germany. He has 15 years of experience in the ECM and Case Management marketplace. Michael holds a Master of Electrical Engineering from the Ruhr-University in Bochum (Germany). He joined IBM in 2007 with nine years of FileNet® experience, where he worked in the Professional Services organization as Team Lead, Project Manager, and Senior Systems Consultant. During this time, he was responsible for designing and implementing ECM and Case Management solutions that were based on IBM FileNet products. Michael has a strong expertise in IBM FileNet P8 architecture, and he regularly advises clients regarding their ECM strategy.

Tom Ko is a Systems Support Engineer for IBM ECM products in California in the United States. He has 25 years of experience in computer hardware and software fields. He has worked at IBM and previously FileNet for 15 years. His areas of expertise include IBM Case Manager, IBM FileNet Business Process Manager, IBM FileNet eForms, and various relational database technologies.

Michael Oland is an IT Specialist with IBM Industry Solutions Software Services in Washington, D.C. He has 13 years of experience implementing enterprise content management solutions with various IBM and FileNet products. For the past year, Michael has worked as a Case Manager implementation designer and developer. Before that he was an architecture and transition specialist assisting customers as they changed from various older and third-party repositories. Michael has a degree in Broadcasting from the University of Tennessee, Knoxville.

Balunaini Prasad is a developer in IBM Software Group at India Software Labs. He develops IBM Case Manager Builder. Before this product, he worked in IBM eDiscovery Analyzer, IBM Content Analyzer, DB2® and AIX® development teams. He joined IBM in 2000. He holds a Master's degree in Electrical Engineering from the Indian Institute of Technology, Kanpur.

Mike Prentice is a business process management (BPM) and Enterprise Content Management (ECM) Solution Architect with IBM Software Group in the United States. He has 20 years of experience in architecture, designing, and building BPM and ECM solutions as a consultant and supporting IBM Business Partners. He has worked at IBM for six years. His areas of expertise include Business Process Manager, Case Management, Enterprise Content Manager, solution architecture, and solution design with object-oriented principals. Mike has written extensively on FileNet and non-FileNet best practices around solution implementation for BPM and ECM.

Monique Ruggiero is an ECM Solution Consultant with the IBM Software Group. She has more than 15 years of experience with enterprise content management technologies. Her roles have included implementation, consultant, and technical sales in the United Kingdom, Switzerland, Germany, and Singapore. She joined

IBM in 2008. She has a Bachelor of Science degree from Imperial College London in Computer Science.

Thanks to the IBM Case Manager development management and architect team for their support of the project:

Martin Shramo
Huzefa Hakim
Sean Yang
Mike Marin
Lauren Mayes

Thanks to the following people for helping us during and after the residency:

Barry Beach
Thuy Do
Laurent Dubois
David Hanson
Steven Hsieh
Kraig Newkirk
Darik Siegfried
Dave Perman
Yvonne Santiago
Patricia Sort De Sanz
IBM Software Group

Thanks to the authors of the previous editions of this book.

- ▶ Authors of the second edition, Advanced Case Management with IBM Case Manager, published in May 2012, were:
Wei-Dong Zhu (Jackie), Thuy Do, Anand Iyer, Srinivas Jandhyala (Srini), Mike Marin, Sathis Marudanayagam, Armando Mastracci, Lauren Mayes, Alan Morton, Eugene Rozhdestvensky, Monique Ruggiero, and Shao Hong Xu.
- ▶ Authors of the first edition, Advanced Case Management with IBM Case Manager, published in February 2011, were:
Wei-Dong Zhu (Jackie), Boris Becker, Julie Boudreaux, Scott Braman, Thuy Do, David Gomez Jr., Mike Marin, and Andrew Vaughan.

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

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
- ▶ Send your comments in an email to:
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>

Summary of changes

This section describes the technical changes made in this edition of the book and in previous editions. This edition might also include minor corrections and editorial changes that are not identified.

Summary of Changes
for SG24-7929-02
for Advanced Case Management with IBM Case Manager
as created or updated on May 9, 2013.

May 2013, Third Edition

This revision reflects the addition, deletion, or modification of new and changed information described below.

New information

- ▶ Chapter 8 “Solution deployment” is new.
- ▶ Section 9.2 “Security consideration” is new.
- ▶ Section 10.4 “Example of customizing the Investigator Step” is new. It shows an example of customizing the standard step page by using the Content List widget.
- ▶ Section 10.7, “Managing documents in context of a case” on page 375 is new. It includes descriptions of integration of the IBM Content Navigator viewer, manually adding a document, and adding a case document to a task.
- ▶ Section 11.5, “Multilingual support” on page 455 is new.
- ▶ Section 13.2.3, “Integrating by using the Case Java API” on page 517 is new.

Changed information

- ▶ All existing chapters in part 1 have been updated.
- ▶ All existing chapters in part 2 have been updated.
- ▶ Chapter 13 in part 3 has been updated.

This edition includes new features and enhancements description added in IBM Case Manager, Version 5.1.1. All the chapters in part 1 and 2 have been revised

and updated. Chapters in part 3 do not require changes except the first chapter in the part.

Chapter 1 “Case management concept” and Chapter 5 “Designing case management solutions” have been revised to better explain the concepts of case management. These concepts include its differentiation from business process management solutions, and the designing of the case management solutions.

Chapter 2 “Typical case management solutions” has a revised use case description. Chapters 6 “Building a simple solution: Part 1” and 7 “Building a simple solution: Part 2” provide updated, step-by-step instructions on building this use case solution. The revised use case solution incorporates new and enhanced features, including container task, case properties update based precondition, precondition group and roles, and update in-basket table.

Chapter 3 “IBM Case Manager overview” and Chapter 4 “Inner workings of Case Manager” have descriptions for new features and enhancement description. They include task enhancements, work visibility and distribution, case document operations, work item toolbar configuration, document viewer, and security.

The old Chapter 8 has been split into two chapters: Chapter 8 “Solution deployment” and Chapter 9 “Solution development”. The original content has been rearranged and revised with a few sections added to them, one of which covers security considerations.

Chapter 10 “User interface” now includes descriptions of the ContentList widget and its usage, solution page customization, and document viewing by using Content Navigator viewer. It also describes how to manually add a document, and add a case document to a task.

Chapter 11 “Development topics” now includes a multi-lingual support description.

In Chapter 12 “Development topic: Round-tripping workflow editing”, the chapter was revised based on the simplified way of doing round-tripping workflow editing.

Chapter 13 “Integration points” now includes a description of using Case Java API.

This book covers IBM Case Manager, Version 5.1.1, whereas the previous edition covers IBM Case Manager, Version 5.1.

May 2012, Second Edition

This revision reflects the addition, deletion, or modification of new and changed information described below.

New information

- ▶ All chapters in part 2 except chapter 5 are new.
- ▶ All chapters in part 3 are new.

Changed information

- ▶ All chapters in part 1 have been updated.
- ▶ Chapter 5 in part 2 has undergone major revision.

This is a major revision of the first edition of the book. The focus is no longer on installing and setting up the IBM Case Manager environment. Rather, the focus is on a more in-depth explanation of the case management concept and the IBM Case Manager product. It also covers how to use Case Manager to develop case solution, and how to integrate other products or components with IBM Case Manager to provide more capabilities.

This book covers IBM Case Manager, Version 5.1, whereas the previous edition covers IBM Case Manager, Version 5.0.



Part 1

Concept and overview

This part introduces the concept of case management and IBM Case Manager. It provides case scenarios for case management systems, and addresses the design approach for those solutions.

This part includes the following chapters:

- ▶ Case management concept
- ▶ Typical case management applications
- ▶ IBM Case Manager overview
- ▶ Inner workings of IBM Case Manager



Case management concept

This chapter introduces the basic concepts of case management, including those aspects of case management that make it unique compared with other types of business applications. The relationship between case management and enterprise content management, collaboration, and business process management are also explored. Finally, the concept of case management solutions is introduced.

This chapter includes the following sections:

- ▶ Case management overview
- ▶ What makes case management unique
- ▶ The need for Enterprise Content Management
- ▶ The need for collaboration
- ▶ Case management solutions

1.1 Case management overview

Case management was developed because certain business applications performed by knowledge workers require a great deal of flexibility, adaptability, control, and collaboration to achieve successful outcomes. Traditional Enterprise Content Management (ECM) systems and the structured control of business process management (BPM) are insufficient to meet the requirements of these applications. In certain industries, such as health care, insurance, and the legal profession, case management is fairly well understood. The case management approach, however, can be successfully applied to a broad set of business applications. Doing so gives knowledge workers and businesses the capabilities they need to achieve their business objectives.

Case management is built around the concept of processing a *case*, a collection of information and coordinated activities, by knowledge workers or case workers. A case typically has a “subject”, similar to the subject of a sentence or a narrative. The subject can be one of several types:

- ▶ A single person such as a patient, customer, employee, or taxpayer
- ▶ A legal entity such as a business, church, or government
- ▶ Two or more people/entities such as in legal cases (Jones v. Smith)
- ▶ An event such as a fraud occurrence, a security violation, or a system outage

A *case folder* is a mechanism that allows knowledge workers to store and retrieve information that pertains to the case. It also tracks the tasks that are required to process the case.

A *case management solution* is the application of case management technology to a particular business problem, typically in a particular domain. You can use case management solutions in scenarios such as these:

- ▶ Customer complaint management where the subject is a customer
- ▶ Benefit enrollment where the subject is an employee
- ▶ Legal cases where the subject is a defendant
- ▶ Allow processing where the subject is a citizen
- ▶ Health care claim reimbursement where the subject is a patient
- ▶ Credit card dispute management where the subject is a customer

For any case management solution, there can be many active cases. For example, the customer complaint management solution creates a case for each customer complaint. The term *case instance* is sometimes used to refer to these individual cases.

Case management solutions are knowledge intensive. These solutions require case workers to coordinate data, tasks, processes, and services to achieve a positive outcome. The case workers need a large degree of flexibility and

adaptability to process the case. The requirement for flexibility makes these solutions hard to implement in traditional BPM systems because BPM systems require predictable and repeatable processes. Cases are less predictable than traditional processes because the case worker judgment and experience influence how the case is handled, and therefore the outcome. However, case management solutions are also repetitive, and therefore require process support that makes it hard to implement them using an ECM system.

Case workers are not the only individuals who interact with a case. A case *participant* might be a user who helps process and close a case. A participant might also be a user who performs management operations such as assessments, audit, and outcome analysis. Management functions can be performed on a single case instance, or across many case instances. Case management solutions need to provide participants with views into the case that allow them to efficiently complete their assignments. However, not all the participants in a case need the same level of flexibility or access to the case folder. In most situations, one or a few knowledge workers control the case and other participants are restricted to performing well-defined activities. The participants who interact with a case can be organized by roles. For example, in the credit card dispute case solution, roles might include customer service representative, dispute agent, dispute supervisor, data clerk, and fraud investigator.

Implementing a robust and effective case management solution requires a software platform that consists of a range of functions. These functions include content management, process management, business rules, collaboration, and analytics. The solution must integrate seamlessly into the work environment of the case participants. The user interface must also provide flexibility and allow a high degree of collaboration among the participants. For example, the interface must allow a user to dynamically add more tasks to a case already in process.

A case management solution must provide case workers with the full context of the case they are working on. This context is called a *360 view* of the case. In practice, case workers with enough privileges must have access to all the information pertinent to a case. This information includes history, documents of various media types, and content added by other case workers. Have the 360 view include all the process information for the case. By having all the relevant information available, case workers can make better informed decisions.

1.2 What makes case management unique

Case management empowers the knowledge worker to solve problems. It does so with a flexible solution that bundles the case information, documents, rules and all the tasks that might be required to solve the business problem.

The main focus around a case management solution is the goal-oriented nature of the problem being solved. All the detailed aspects of the solution work to achieve that goal. Whether it is a rule, a document, a task, or simply a group of data elements, these characteristics all work to help bring the case to its end state.

This focus is where the uniqueness of a case management solution originates. The goal-oriented approach when compared to BPM-focused solutions are around process almost exclusively. The focus around a case and its interactions with data, rules, content, processes is the complexity that differentiates them.

1.2.1 Case Management Solutions versus BPM Solutions

To compare these solutions, it is important to first realize the intent of the solution that is being developed. Case management solutions have a primary focus on the data around the case and what is needed to complete it. In a BPM solution, the focus is on the process and what is needed to complete it. The confusion at times comes when the case uses a defined process. The use of these defined processes (which can use BPM) is needed within the case solution, but is not the focus of the solution. The processes are short lived and achieve a specific purpose, such as obtaining specific information, processing a rule, gaining approval, asking for input, or updating other systems with decision. All of these processes are included in a single BPM process.

Case management and business process management solutions have these key differences:

- ▶ Cases tend to be more unpredictable, and rely more on a knowledge worker's judgment than on system control or business-rule-based control flow.
- ▶ The emphasis of case management is on designing a flexible process to drive a business goal. The emphasis of business process management is on designing a repeatable step by step procedure to solve a business problem.
- ▶ In case management, the business problem is solved by a knowledge worker by using tools in the form of tasks that are not necessarily modeled. In business process management, the business problems are always solved by following a modeled process.

- ▶ Case management solutions have a focus on the case and its data and activities, and how to complete the case. BPM solutions have a focus exclusively on the process and how to best accomplish that specific process.
- ▶ Cases have an indeterministic lifecycle and can exist for a long time. BPM processes have a deterministic lifecycle and focus on processing within a shorter timeline.
- ▶ Cases always involve human participants, whereas a business process management solution might not involve any human participants.
- ▶ Cases always involve content because content is what knowledge workers use to make decisions, whereas processes might not include content at all.
- ▶ The knowledge workers who process a case decide which tasks are required to complete the case. In a business process management system, the software requires activities that are specified in process model and by business rules.
- ▶ When a case is closed, the case folder remains accessible. Case closure (or completion) is a relative term in that a case is always accessible and can be reopened for more processing as required.

Case management solutions that are intended to solve a business goal are considered partially repeatable. That means that much of the processing is repeatable, but not entirely. The business goal itself always happens, but the path to accomplish it is non-deterministic. The activities might or might not be performed, which is different from traditional BPM. From a case management perspective, recognize that many solutions cannot be fully described in a BPM system because of one or more of these factors:

- ▶ The focus is on the case data, not the process itself
- ▶ Not all the activities are known beforehand
- ▶ The order of the activities is unknown

There are several technologies that provide process functionality. These technologies go from formal, structured processes at one end of a spectrum to informal processes at the other. Formal processes were called *production workflow* in the 1980s, and today are implemented by BPM systems. These solutions are also referred to as process-centric

A formal process encodes the business goals the process is designed to accomplish, giving the BPM system full control over the business goal. Formal processes might not have any human intervention. If they do, the participants on the process do not need to know the business goal. If the participants run their assigned activities, the BPM system ensures that the business goals are achieved. The BPM system is in complete control, which allows vendors to provide a full suite of functions to model, track, monitor, and manage the

processes. The process can contain business rules that change the execution path of the process, but the path that is taken is modeled.

Informal processes, at the other extreme of the spectrum, were called *ad hoc workflows* in the 1990s. They evolved into collaborating technologies, including email and instant messaging. The informal process is not modeled and cannot be easily tracked by the system. The participants in the process must agree on and understand the business goal, and have enough information about how to achieve it. There is no process to follow, and the system does not know when the goal is achieved or when the process is started. These types of processes and technologies are useful for non-repetitive and one-of-a-kind assignments that do not need to be tracked or audited.

Case management tries to achieve a balance between formal and informal processes by formalizing parts of the process. These parts are called *process fragments tasks*. In addition, the system is aware of when the case starts and when it finishes so it can track and monitor the cases. In the case management system, the case workers are in control on how cases are processed. However, they do not need to know all the details about how to achieve the goal, and the system can detect when it is achieved. The process fragments tasks are available as tools for the case worker to use.

Figure 1-1 shows aspects of the process spectrum and positions case management against formal and informal processes. It does not cover all aspects of case management, but it includes some of the aspects that are common with business process management.

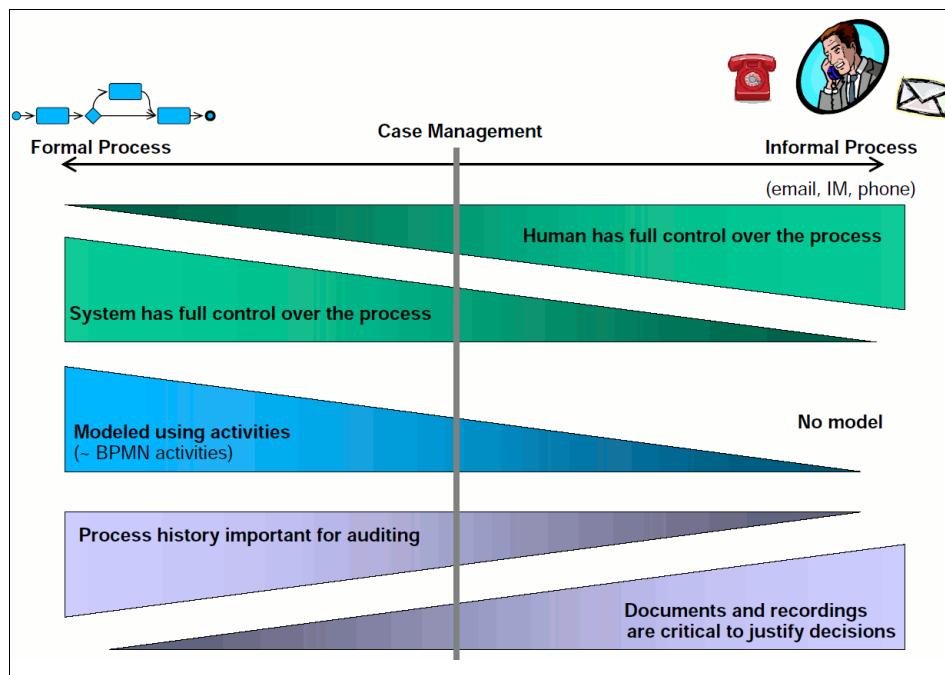


Figure 1-1 Process spectrum

One aspect that is not illustrated in this figure is management and control. Within a case management system, this is an over-arching aspect of a solution. For example, as illustrated in Figure 1-1, the line is drawn between “Process History important for auditing” and “Documents and recordings are critical to justify decisions” separating the nature of formal and informal processes. In contrast, case management has an overarching view on both sides of the spectrum concerning the case.

1.2.2 Cases are goal driven and unpredictable

The concept of a goal driven case solution is to bring focus on what is needed to be done to a case. A case is to be processed, completed, paid, or resolved. The data, rules, and activities that are identified exist to support this end goal. Although the case constructs are defined, each individual case instance has enough variability that a knowledge worker judgment is required to manage the case to its end state.

For example, in a credit card dispute case, the business goal is to resolve the dispute. Although each case instance has the same goal, the outcome of each case can be different. In the credit card dispute example, some disputes end with the merchant returning money to the customer. Other disputes end with the customer maintaining the charge in the credit card, the bank fixing an accounting mistake, and so on. In this example, each case is completed with a different outcome, but all of them met the same business goal of solving the credit card dispute.

A case management solution is designed to provide the tools to solve a repeatable situation. In the credit card dispute resolution, the situation is customers who are disputing charges to their credit cards. Although all the processing patterns are similar, each dispute is different enough to present a certain level of unpredictability in the process.

The overall case design might look similar, but the case worker must react to new information in ways that cannot be predetermined. New information can change the outcome of a case in ways that can be determined only by the case worker.

However, not all aspects of a case are unpredictable, and so BPM technology can be used to model the predictable aspects of case solutions. These predictable aspects of the case processing can be encapsulated into process fragments that can be used as part of the toolset that case workers use.

Case management recognizes that a business goal can be achieved in multiple ways, not all of which can be predetermined. The path that follows from the start of a case to the completion of that case can be different for each case instance. Not all activities or steps that are required to complete a case can be known when you design the case solution. And for the known activities, the order of execution might be unknown.

1.2.3 Cases are knowledge intensive

The reason BPM and ECM technologies are insufficient for implementing case management is that cases are knowledge intensive, requiring the judgment of case workers. Therefore, the outcome of the case depends more on human judgment than on the underlying technology. The technology, however, must support the case worker by providing the tools that are needed to advance the case to its resolution. It cannot replace the judgment of the case worker, and should not unnecessarily constrain the actions of the case worker.

Although the outcome of the case depends on case worker judgment, case workers cannot arbitrarily decide the fate of a case. Case workers must be able to justify their decisions and collaborate, as necessary, to reach those decisions. The case management solution must support this knowledge-intensive activity by

providing the tools and facilities to help the knowledge workers accomplish their work. At the same time, it must provide the persistence, history, tracing, and monitoring needed to justify and audit case worker actions and decisions.

1.2.4 Modeling

Case and process modeling are separate. Modeling is the definition of the solution. It uses a tool to describe the solution that must be run to the system. The tool is normally called a modeling or design tool, and the model is the human readable output or printout that is produced by the tool. The BPM process modeling is well understood and formalized by standards like the Business Process Modeling Notation (BPMN). Case management has not been studied as long or as deeply. Therefore, case modeling is not as well understood. There is an effort underway at the Object Management Group (OMG) to define a modeling notation called Case Management Modeling and Notation (CMMN). This notation will help clarify and formalize case management modeling.

From a modeling perspective, case modeling focuses on knowledge worker needs. It allows the person modeling the case to define the tools that are needed to complete a case. Those tools can be modeled as tasks, which can be process fragments. The modeling of cases focuses on what must be done to complete a case instead of how it is done. Tasks in a case then describe what must be done. Not defining the how gives the case workers more flexibility to decide the best tasks for a particular case instance.

BPM models focus on the activities that are required to achieve the business goal and the efficient ordering of those activities. Therefore, BPM models describe how the processes are done, in addition to what must be done.

1.2.5 Tasks

In order for the case model to achieve a balance between formal and informal processes, the formal processes must be broken into process fragments. This concept is the case *task*, which corresponds to a process fragment, but can also be implemented with other non-BPM technologies.

Tasks break the model of cases into two levels of abstractions. A task represents a higher level of abstraction than process, and describes “what” must be done. A task can also describe “why” it must be done. For example, a task to review a customer’s application must be done if a new customer application is received. This concept allows the person modeling the case to model at a higher level of abstraction and avoid describing the details of “how” a task must be done. Therefore, when you model a case, you are trying to answer the “what” and “why”

questions by using tasks. In contrast, when you model a BPM process, the designer answers the “what”, “who”, “when”, and “how” questions.

The implementation of the task details corresponds to the lower level of abstraction. Implementation might involve modeling a process fragment where you answer the what, who, when, and how questions. However, it can also be implemented by other technologies or applications.

Figure 1-2 shows the IBM Case Manager case task page with a model of a case that contains seven tasks. Tasks are not connected with lines as they are in traditional BPM systems because there is no execution order between them. The tasks are designed as tools for the case worker.



Figure 1-2 Example of tasks modeling in Case Manager Builder

1.2.6 Routine work and knowledge work

Most case solutions require both knowledge workers and routine workers. Routine workers are sometime called heads-down workers, and are assigned repetitive work that requires little judgment. Routine workers are suited for data

entry applications such as scanning correspondence, and certain types of BPM applications.

In processing cases, organizations must manage the use of routine workers and knowledge workers assignments. Most organizations manage the two pools of workers separately, and separate routine workers roles and knowledge workers roles. The concept of role allows organizations to fulfill those assignments by moving workers between the roles, or having workers who are assigned to multiple roles. Roles allow managers and supervisors to quickly and easily move resources where they are needed.

1.3 The need for Enterprise Content Management

During the processing of a case, multiple documents can be produced and used by the various participants who interact to solve the case. Documents are a key part of any case management solution because knowledge workers depend on documents to complete their tasks. Case workers are no different, and so, while resolving a case, multiple documents can be created and used. These documents can be emails, text documents, spreadsheets, voice recordings, images, video clips, and presentations. All this information must be stored and organized as part of the case. This process requires cases to have a flexible containment mechanism, similar to a folder structure in a file system.

However, a typical file system is not flexible enough to store case information. For example, a document might be used simultaneously in multiple cases, and therefore must be in each folder. A change to the document must be reflected in all the cases. It is also important that metadata, version, and security are maintained so the complete history of the case is kept current and traceable.

These requirements are achieved in case management solutions by using an ECM system to implement a case folder to collect the information that is related to the case. A case folder (and all its contents) is maintained after the case is complete for legal and compliance reasons. The retained case can also be used in future cases or for analysis across cases. A well-designed case management solution must include the design of the content management aspects like folder organization and document types, including metadata.

1.3.1 Information complexity

The amount of information or documents that is required to complete a case can be different for separate instances of the same case solution. Certain cases require more documentation or investigation than others, with means that cases can have different amounts of information. This complexity translates into case

information complexity, which requires high levels of organization. Case documents and information are normally organized in a hierarchical containment structure similar to a traditional folder structure called a case folder. For example, a customer dispute case might have a folder structure as shown in Figure 1-3 on page 15. The folder structure is populated with the documents and files that are required to solve the case.

A case can contain multiple documents of various types. To manage these documents, you must associate metadata with them. You also need content and metadata search capabilities. Finding information in a case is different from finding information in a database because that information might be inside documents that are stored in the case file. Case workers must easily find the information they need to work on the case, even if it is buried inside documents. ECM systems make it easier to work with this complexity of information.

The need to find case information is not just important for case workers who handle a single case instance. It is also important for participants such as managers or analysts who deal with aggregations of case instances. These participants must be able to extract patterns from case information. For example, they might want to determine the frequency that a particular pattern occurs in a collection of cases. In the credit card complaint solution, a manager might want to find the percentage of complaints that are filed for a particular product or a particular merchant. This information might be present only in the documents or emails that are contained in the cases.

1.3.2 Security

Using an ECM system to store case management content, including information and documents, allows you to maintain a fine-grain access control over the information. It is common for cases to deal with sensitive information, and so these documents must be protected from unauthorized access. In certain scenarios, even those working on the case should have limited access to the case information.

1.3.3 Retention

Case information is long lived and can be subject to the lifecycle of content objects in an ECM system. In particular, when the processing of a case is completed, all the case information remains and is still available for users with the correct security privileges. Cases therefore can be reopened for processing or auditing if required.

Case information, including history and documents, can have regulatory compliance retention requirements. Cases are normally implemented as a case

folder (see Figure 1-3) to collect all information and documents that are related to a particular case. The case folder is retained after case completion. Retaining case folders allows future cases to use the existing case information, and facilitates compliance with legal and regulatory requirements.

Even when regulatory compliance is not required, the retention of case information is important for auditing purposes. For example, certain documents might need to be retained for five years, whereas others must be retained for 10 years. The case management system must be able to enforce these regulatory compliance policies and satisfy audit requirements.



Figure 1-3 Case for customer 123 showing case folder structure

1.4 The need for collaboration

Case workers not only need information about the case to make decisions, but also must collaborate with other participants in the case. Most of this collaboration is through the documents and information that is contained in the case folder. These items include case notes, extra documents added to the case,

and decisions made. Other collaboration can occur through case worker communication. Case workers must communicate with others who are working the same case through case notes, instant messages, email, and phone conversations. Technology supporting a case management solution must provide these types of collaboration technologies. It also needs the capability to maintain the collaborations as part of the case folder for case resolution and audit purposes.

Case management recognizes that not all the tasks, steps, or activities of a solution can be predefined. Therefore, case worker judgment and collaboration are used to resolve the case. In certain case solutions, negotiation and collaboration between case workers can be more important than imposing a fixed set of activities in a particular order. However, too much negotiation and collaboration can slow down the resolution of a case. Therefore, a balance must be achieved. In many situations, experienced case workers can achieve the correct balance. In other situations, business rules and analytics in the form of decision support can be used to help strike the correct balance.

1.5 Case management solutions

A case management solution provides the software environment for case workers to collaborate on the completion of cases. The solution can be modeled and deployed by using a case management system. As described earlier, the solution is implemented by using multiple technologies that include ECM, collaboration, and BPM. A case management system is used to bring all those technologies into a coherent framework.

Although the case is the focus of a case management solution, the solution might be composed of multiple types of cases. For example, a customer complaint solution might have multiple types of complaints, depending on the product or service the customer is complaining about. Each type of complaint can be slightly different and can be managed by different case workers. Therefore, a solution is normally composed of one or more types of cases.

Case management software solutions are appropriate for situations in which the work that must be accomplished is goal-oriented, knowledge intensive, and highly collaborative. These types of solutions are based on documents such as emails, faxes, pictures, video, and voice. These documents are required for the knowledge workers to make decisions that are based on their judgment. Case management provides the tools and collaboration environment for the knowledge workers to complete their cases.

Case management solutions are also appropriate for works that involve complex decision making by knowledge workers that is based on the information and

documents that are associated with case instances. Typical applications include exception handling, complaint or dispute management, contract management, lending applications, benefits enrollment, invoice processing, change request, and incident reaction. These types of solutions require the integration of capabilities from multiple technologies that include content management, BPM, collaboration tools, social software, business rules, and analytics. The case paradigm is applicable in multiple industries and environments that include insurance, banking, health care, government, and utilities.

1.5.1 Case data

A case consists of a collection of information that is used to achieve a business goal. The first step to designing a case management solution is the identification of this information. This information is in addition to the folders, documents, and tasks. The case data defines what is needed for the case to be processed. This data can come from a number of sources, forms, documents, external systems, and user input. After it is collected, this data is maintained throughout the lifecycle of the case to allow the knowledge worker to process the case through completion. In addition, it includes data that is relevant for reporting and processing (tasks).

Understanding the data and their source is needed to design the solution as a starting point. Then, understanding of when and where to obtain the data from, and how and where it needs updating are the next step. The processing of data might be presented as a separate business activity that is needed or a separate data window for the knowledge worker to complete.

1.5.2 Case folder

The natural way to organize case information is to use a case folder in an ECM system. The case folder can contain a folder structure with subfolders that contain documents, history, and other information that is used to process the case. In general, the case folder contains all the information that is used to process and manage the case.

The case folder provides the context for the case workers to do their work. The information that is contained in the case folder can be categorized into these types:

- ▶ Properties: Not all case information is contained in documents. There are discrete properties that identify the case and hold important status information about the case. These properties, such as the customer name, case priority, and account number, must be associated with the case folder.

- ▶ Documents: Case management is content-centric, and case workers as knowledge workers base their work on documents, including emails, text documents, pictures, spreadsheets, and others. The case folder provides the container to collect all the documents that are used in solving the case.
- ▶ Tasks: The case folder contains all the tasks that can be used in the case, and each task contains state information. Some tasks might be waiting to be executed, executing, or completed. This state information and history must be preserved in the case folder.
- ▶ History: Everything that happens to the case and its content must be recorded as history. The case folder provides the container to keep that history. The history includes when the case was created, when documents and information were added, case comments, when tasks were initiated and completed, and so on.

This collection of diverse and rich information is important to maintain in a central place for case consistency and integrity. That central container is the case folder. Having the case folder in an ECM system provides the governance, lifecycle management, searching capability, archival integrity, and security required. Case information is long lived, so after the case is completed, the case folder and its information must be maintained in the ECM system.

Cases must react to what is happening in the case folder so new tasks can be executed. New tasks might be needed when documents are added to the case, case properties are modified, or based on case actions. Case workers in the correct role and with enough security privileges can view the state of the case tasks. These case workers can also disable, execute, and add new tasks to the case.

Figure 1-4 shows the basic functions of the case folder.

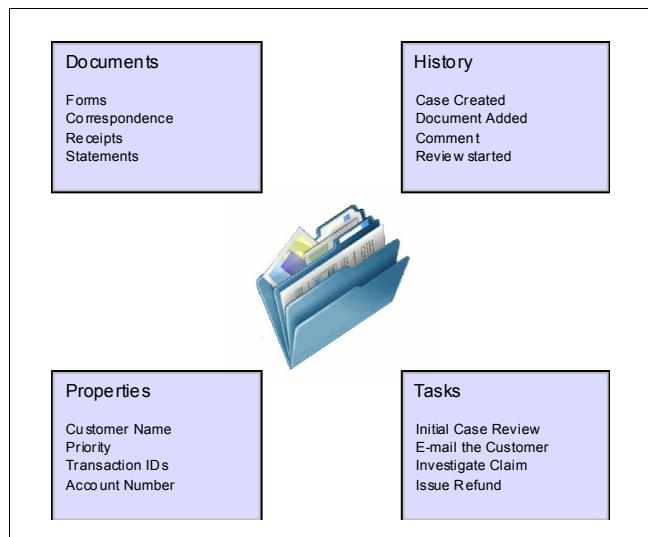


Figure 1-4 Case folder concept

1.5.3 Tasks

In a case management solution, tasks are tools the case workers use to process a case. The case worker or workers in charge of a case must decide which tasks must be executed to complete the case. Depending on the modeling of the case solution, tasks might be initiated by the system, or might be classified as required or optional. Case worker with enough privileges can start, disable, or add tasks, depending on the requirements of the current case instance. Figure 1-5 on page 20 shows the interface that a case worker can use to interact with the tasks in a case solution implemented by using IBM Case Manager.

Tasks that are understood can be defined as part of the modeling of the case solution. Figure 1-2 on page 12 shows an example of such a model. Not all of the tasks are understood at modeling time. The case workers can add tasks when they are working a case. In case modeling, tasks introduce a higher level of abstraction than the process fragments. During the processing of an actual case, tasks introduce a higher level of control. Case workers with the correct privileges can see and control the tasks by using an interface similar to the one shown in Figure 1-5 on page 20. Other case participants simply see work to be done in their role or personal in-baskets, and might not have access to the high-level tasks. Therefore, tasks are not assigned to people, but personnel assignments can be done in the process fragments implementing the tasks.

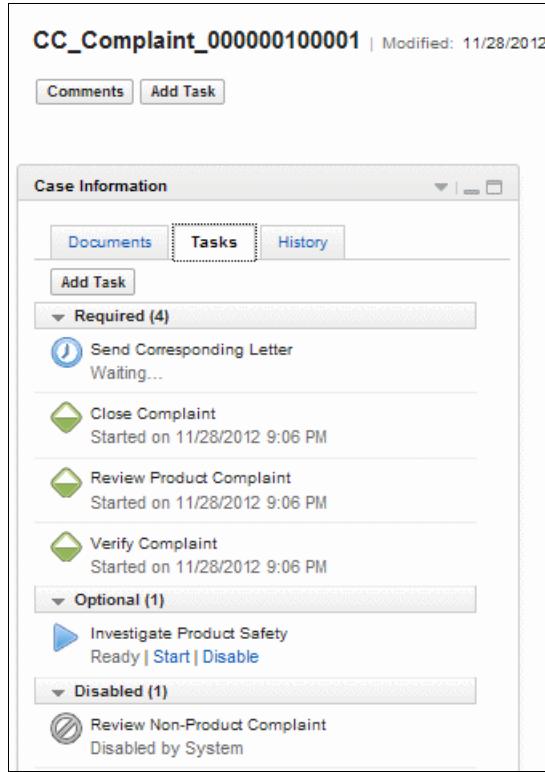


Figure 1-5 Example of tasks available to the case worker

Neither ECM systems or BPM systems use tasks as implemented by case management systems. Therefore, flexible case solutions that use the task concept can be implemented only by using a case management system.



Typical case management applications

This chapter addresses solution patterns in industries that are suitable for case management applications. It also explores a specific example of a case management application, a complaint management application, in greater detail.

This chapter contains the following sections:

- ▶ Typical applications
- ▶ Complaints management example use case

2.1 Typical applications

As described in 1.1, “Case management overview” on page 4, some solutions are better suited for case management than for other technologies. The need for case management emerges from the need to provide flexible solutions for knowledge workers. These workers need to solve complex problems that involve documents, collaboration, and processes. These solutions are traditionally implemented by using business process management or enterprise content management technologies.

Competition and a more complex business and regulatory environment is driving companies in various industries to optimize their operations. Knowledge workers are faced with complex business situations. These knowledge workers require flexible software technologies to help solve these situations. Case management solutions equip knowledge workers with the tools they need to achieve their goals in a complex business environment.

The case management paradigm of having a case folder to contain all pertinent information that is required to solve a problem is used in multiple industries. In certain situations, the pattern is industry-specific, such as mortgage servicing or court case management. A few solutions are applicable to every industry such as customer dispute, customer complaints, and fraud investigation. Fraud investigation is a solution pattern that is applicable across most industries, which include banking, insurance, healthcare, government, and utilities. For example, a bank customer might question activity in an account, resulting in a fraud investigation. Another example is a government agency that is looking into the possibility of improper benefits being paid to a citizen.

2.1.1 Solution patterns

Providing knowledge workers with the tools to solve structured, repetitive, goal-oriented work with a great deal of control and collaboration introduces a few solution patterns. Case management solutions are based on these patterns:

- ▶ The use of a folder as the central focus of the solution with all interactions and supporting artifacts in that central container.
- ▶ The solution is designed for knowledge workers or experts to solve a repetitive, goal-oriented business situation. The experience and judgment of the worker or expert affect how the resolution is achieved.
- ▶ The need to provide collaboration technologies to support the knowledge workers and experts who are solving the business situation.

- ▶ The use of non-predetermined activities as a major element of the solution, allowing knowledge workers and experts to add activities to the solution as needed.
- ▶ The use of rich content as part of the solution. Most complex business situations require multiple type of documents, including text documents, spreadsheets, presentations, video, and pictures.
- ▶ The use of event-driven activities. For example, a bank customer filing out a dispute form might automatically start a credit card fraud investigation.
- ▶ The use of analytics for seeking patterns among real-time or historical data.

2.1.2 Applications by industry

Industry faces many challenges. One of them is how to improve the productivity of their knowledge workers and improve customer satisfaction. Case management technology helps by providing a flexible technology that can be used to create knowledge worker applications that help with customer interactions. Although the goal is similar, each industry has specific applications that are suited for case management.

Healthcare

With ever increasing regulations and a growing need for healthcare services, the demands on the healthcare industry have forced it to become more efficient. The complete healthcare workforce is composed of highly skilled knowledge workers, and so traditional highly structured and rigid applications imposes unnecessary constraints. This environment requires the flexibility and collaboration that is provided by case management solutions. The following healthcare industry challenges are ideal candidates for case-based solutions:

- ▶ Patient discharge
- ▶ Care coordination
- ▶ Healthcare fraud and abuse
- ▶ Clinical trials coordination
- ▶ Claims adjudication
- ▶ Member management
- ▶ Provider management

Government

The demands of increasingly aware citizens have forced government agencies to respond accordingly. Government-citizen interactions by their nature lend themselves to a case management solution. Case management solutions address the following typical challenges:

- ▶ Benefit enrollment administration
- ▶ Grant request and administration
- ▶ Citizen management
- ▶ Taxpayer management
- ▶ Court case management

Insurance

In the insurance industry, the need for efficiently acquiring customers and retaining them is a central theme in all case management solutions. The following insurance industry challenges are ideal candidates for case-based solutions:

- ▶ Policy issuance
- ▶ Policy underwriting
- ▶ Claim handling
- ▶ Annuity management

Banking

The banking industry recently has seen many changes because of regulation and the economic cycle. The efficient management of customer interactions or inquiries is as important as asset management. The following banking industry challenges are ideal candidates for case-based solutions:

- ▶ Loan request
- ▶ Dispute resolution
- ▶ Customer inquiry
- ▶ Mortgage servicing
- ▶ Account opening and maintenance
- ▶ Credit card inquiry
- ▶ Personal credit line request
- ▶ Investment and wealth management

Utilities

The utilities industry has worked with the concept of cases for many years. However, utilities have worked in this context either manually or using technologies that are not designed for this purpose. Government case management solutions in the utilities industry address the following challenges:

- ▶ Rate case justification
- ▶ Claim management

- ▶ Permit request management
- ▶ Land rights management
- ▶ Property management

2.2 Complaints management example use case

Throughout this book, an example that is based on complaints management is used to illustrate the key aspects of the IBM Case Manager. Complaints management is used because it is a good fit for the Case Management paradigm. This section outlines a set of business requirements for this scenario.

There are many types of complaints management systems, including handling service complaints, product complaints, and government-related complaints. The example that is used here is a simplified and relatively generic version so that it can be applicable to various different industries. Although intended to be realistic, it is fictitious and is intended for illustrative purposes only. It does not describe any exact processes or industry best practices.

In the example scenario, a company sells products and provides a number of services. It has a centralized complaints management system that handles any issues from customers. A customer calls in to lodge a complaint and a representative completes a complaint form, which in turn creates the case that is used to track this complaint.

The complaint case is first reviewed for validity and then routed to a specialist, depending on the category of the complaint. The process of resolving the case does not necessarily take the same path every time. However, it almost always involves collecting documents, collaboration between individuals and teams, and tasks such as reviews and investigations. Different tasks must be carried out depending on the nature of each complaint. The outcome of the case relies on the experience and judgment of the person who is working on the complaint case. That person is guided by the company rules and regulations.

In this example, if the complaint is from a customer with a high rating, more tasks must be performed. The customer rating is based on business rules. This rating can be dependent on a number of different factors such as transactions revenue, number of previous complaints, and time of membership.

In the real world, almost every case management system must be integrated to one or more external systems. The example shows how this configuration is supported through different integration points.

Many companies have a team of analysts who are responsible for analyzing the historical patterns of the complaints and their outcomes. This process allows the

company to gain a better insight of customer-related issues. By learning from previous complaints cases, the company improves not only its case processing, but also improves its products and services. The optional Content Analytics component can be used to analyze the complaint data, comments from the case workers, and case documents across many complaint cases. For more information, see Chapter 18, “Integration with IBM Content Analytics” on page 617.

2.2.1 Complaint scenario

In the example, a company provides commercial products and services to many customers. The company realizes that customer complaints can be a gold mine of important information. They consider a complaint as valuable as any positive feedback, maybe even more so. They want to be able to process these cases efficiently, and collect all the feedback and related information centrally to support their continuous quality improvement processes.

There are a number of ways the company can receive complaints:

- ▶ The customer calls the support line or a particular account representative to lodge a complaint.
- ▶ The customer emails the complaint to the company.
- ▶ The customer fills out an online form that is provided on the company website.
- ▶ The customer sends in a written complaint by mail or fax.

The company has a Customer Relationship Management (CRM) system that manages all the details about their customers. All complaints must be matched to a record in the CRM system, and the relevant information added to the case so that it is available to the reviewer.

The company uses a multi-dimensional system for rating its customers to help give a balanced view of the customer and support the customer communication processes. If a complaint is received from a high value customer, the account manager for the customer is notified. The account manager then contacts the customer to assure them that the complaint is receiving attention.

In some situations, the client manager or customer relationship staff might also suggest a different product to the customer as an alternative. From the case management perspective, this up-sell activity means that a separate set of tasks must be processed and tracked in the context of the case.

The handling of the complaints can span many departments in the organization. The process typically starts at the support contact center, and, depending on the nature of the issue, it is categorized into an appropriate type: Product, Service,

Billing, or Others. There are separate guidelines for the processing of each of these types of complaints.

Based on the type of the complaint, specialists might need to review the case in detail. If the complaint is related to a safety issue, it must be routed to a special department that carries out the necessary investigations.

2.2.2 Resolving the case

The complaint review process starts upon receipt of the complaint and a review case is opened. The process ends, and the case is closed, when a final determination is made by the principal assigned reviewer and communicated back to the customer.

Company policy requires a response to all complaints. The reviewer is responsible for finding a resolution in a timely manner. The first step is to screen the complaint. For spurious complaints, no further action might be needed, and the case is closed. All non-spurious complaints are reviewed by trained specialists who have a deep understanding of regulatory requirements, and company policies and procedures.

Reviewing product-related complaints requires a good understanding of the products and their components. In these cases, a specialist in the product department is assigned to review the case. If it is a non-product-related complaint, a specialist from the complaints department takes the case.

The specialist collects all necessary documentation related to the case. This process might involve requesting more information from the customer, or getting information from other systems inside the company. In complex cases, a small team of specialists might work on the case cooperatively. If a product safety issue is suspected at any point, a safety investigator is required to participate in the review, and provide an assessment.

After the final determination is made, the outcome is communicated back to the customer. Depending on the severity and complexity of the case, a formal report might be filed in the case folder.

The specialist teams are led by their team managers. The managers are responsible for the monitoring of service levels, the distribution of work, and ensuring correct operations by the teams. The managers have the authority to assign tasks, and also to reassign work from one member of the team to another.

2.2.3 Content related to the case

Like in most case management situations, handling complaints involves the collection of documentation that is related to the case. The documentation includes all forms of communication with the customer related to the complaint, internal review and investigative reports, and any documentation required for compliance purposes. The case reviewer is responsible for ensuring the completeness and consistency of the documentation. The information must be able to be used as evidence to support the decision making process. Case information related to complaints is used by the Quality Review Board in their regular sessions. It is also used to support the analysis of trends, and to help identify new issues and systemic discrepancies.

The following types of documents that might be collected as part of a complaint case:

- ▶ Complaint form: Standard form that is completed by the customer, and sent in through mail, email, or fax that describes details of the issue.
- ▶ Correspondence and supporting documents: Any written communication with the customer, including extra documents received such as receipts, invoices, emails, and pictures.
- ▶ Reports: Internal reports from reviewers or investigators.
- ▶ Phone recording: Any phone calls recorded during the case.
- ▶ Contracts: Any updated or new contracts that involve product up-sell or service agreement changes.

There are associated tasks that run automatically whenever certain types of documents are added to the case. For example, a review process when a new contract arrives.

2.2.4 Integration with other systems

There are several other systems that are involved in this example:

- ▶ The company Customer Relationship Management (CRM) system. Details about the customer's account must be retrieved from the CRM system and added to the case information.
- ▶ The Customer Rating system. This rules system is used to calculate the customer rating. The rating is requested at the start of the case, and added to the case information.

- ▶ The account managers portal. For high rating customers, the appropriate account manager on the portal is sent a task to reach out to the customer.
- ▶ Analysis system. Case information, including all the documentation and comments, must be made available for analysis.



IBM Case Manager overview

IBM Case Manager enables you to manage structured and unstructured processes and content in dynamic, highly collaborative, and flexible ways. IBM Case Manager supports the agile methodology by providing an environment in which you can design, develop, validate, and test case solutions iteratively. IBM Case Manager also provides a rich set of tools for administration, template management, and importing, exporting, and transferring of solutions to and among various environments.

This chapter includes the following sections:

- ▶ IBM Case Manager capabilities
- ▶ IBM Case Manager environments
- ▶ IBM Case Manager components
- ▶ IBM Case Manager architecture
- ▶ IBM Case Manager configurations

3.1 IBM Case Manager capabilities

IBM Case Manager provides capabilities for integrating content, processes, and people into a platform for building case management applications. This section addresses the capabilities of IBM Case Manager including content management, business process management, collaboration and social software tools, business rules, and analytics.

For more information about licensing of IBM Case Manager components, see the following website and search for the term “Case Manager”:

<http://www.ibm.com/software/sla/sladb.nsf>

3.1.1 Case design and deployment

IBM Case Manager allows you to design and deploy case solutions that model real world business solutions. Business analysts can use the Case Manager Builder interface to create case solutions such as Claim Processing, Loan Processing, or Complaint Management in insurance and banking sectors. The solutions can then be easily deployed into the production environment.

3.1.2 Case management

IBM Case Manager enables case workers to easily create new cases, monitor their in-baskets for outstanding work, process case work, and browse and search cases. The user interface can be configured based on user roles and customized depending on specific needs of managing cases of a particular type.

3.1.3 IBM Case Manager API

The IBM Case Manager API serves as the integration tier between IBM Case Manager components. For example, at design time, Case Manager Builder uses the API to fulfill the requests related to deployment of a solution. At run time, Case Manager Client uses the API to create, run, and manipulate cases and tasks. Custom applications can use the IBM Case Manager API within custom widgets for the Case Manager Client or a new user interface.

3.1.4 Tools

IBM Case Manager provides various tools such as IBM Case Manager administration client, Document Generator (DocGen), and precondition checker.

Administrators and Business Analysts use these tools to configure and manage their case management system for optimal use.

3.1.5 Solution templates

Solution Templates help organizations to jump-start the process of developing case management solutions. The IBM DeveloperWorks website contains a set of templates for several lines of business applications. Business analysts can build new solutions from these templates. The IBM Case Manager administration client tool provides support for creating solutions from templates or templates from solutions. It can also be used to copy solutions. Solution templates contain a predefined set of artifacts. These artifacts include case metadata, roles, in-baskets, document types, case types, task definitions, and user interface designs. They can be deployed into an IBM Case Manager environment. Solution templates typically capture best practices for a case management use case, and can be modified to fit the individual needs of an organization.

3.1.6 Content management

IBM Case Manager provides a complete content management environment. These capabilities include check-in, check-out, major and minor versioning, referential containment (hierarchical folder structures), browsing, searching, compound documents, and auditing. This capability is part of the IBM FileNet Content Manager component. IBM Case Manager can also use an existing IBM Content Manager installation as document repository as described in Chapter 15, “Integration with IBM Content Manager” on page 549.

3.1.7 Business process management

Business process management is another capability of IBM Case Manager. This capability enables a solution to have manual, automatic, and user-creatable case-oriented tasks using business process management workflows. This capability is part of the IBM FileNet Business Process Manager component. IBM Case Manager can also use an existing IBM Business Process Manager installation for implementing tasks workflows as described in Chapter 17, “Integration with IBM Business Process Manager” on page 587.

3.1.8 Collaboration and social software tools

IBM Case Manager brings the power of collaboration to case management. This capability allows case workers to ask other knowledge workers or experts for advice in real time without leaving their case management environment. IBM

Lotus® Sametime® Entry is seamlessly integrated into the IBM Case Manager Client, and enables case worker to use instant messaging.

3.1.9 Business rules

IBM Case Manager uses the power of a business rules management system (BRMS) to help manage cases intelligently. The integration allows you to access and evaluate a centrally defined business rule in a step of a task. This process enables business analysts to manage those rules centrally and use them in individual task workflows as opposed to defining them individually at the level of each workflow. Business rules management is made available by the IBM WebSphere® ILOG® JRules component.

3.1.10 Case analytics

IBM Case Manager allows you to analyze, monitor, and report information at the case level. It enables case workers, line of business managers and business analysts to identify dynamic trends, and track the case and task processing. It also allows them to monitor key performance indicators (KPIs) or service level agreements (SLAs) in real time. This capability is part of the Case Analyzer component, which processes information from IBM FileNet Content Manager and IBM FileNet Business Process Manager.

3.1.11 Reporting

IBM Case Manager supports powerful reporting that is based on IBM Cognos® BI reports and Real Time Monitoring dashboards. You can customize reports to view historical and work-in-progress data that are generated by Case Analyzer by using online analytical processing (OLAP) cubes.

3.1.12 Content Analytics

IBM Case Manager uses IBM Content Analytics for text and content analysis. This capability enables knowledge workers to look into the unstructured data within cases and identify statistically significant correlations. Content Analytics uses case documents, case comments, and case metadata for the analysis.

3.1.13 Forms integration

IBM Case Manager integrates with two form management components and products: IBM FileNet eForms and IBM Forms. This integration allows you to use customized, easy-to-use forms in your case solutions.

3.2 IBM Case Manager environments

IBM Case Manager is built on top of the IBM FileNet P8 Platform. This configuration takes advantage of existing business process and content management to simplify creating, deploying, testing, and managing case solutions. IBM Case Manager uses IBM FileNet Content Engine for creating and accessing case solution objects. IBM FileNet Process Engine provides workflow processing and task execution within cases. IBM Case Manager seamlessly mixes the use of underlying Content Engine and Process Engine objects to provide a comprehensive case management solution.

3.2.1 Environments and case management phases

IBM Case Manager supports three phases to accomplish end-to-end case management: Design, test, and manage. To realize the three phases, IBM Case Manager supports three environments: Development, test, and production. The case design tool (Case Manager Builder) is used in the development environment where the solution design phase takes place. After the design is finished, case solutions are deployed on the test system. The test environment is similar to a production environment, but it is used for user-acceptance testing or pre-production testing. This process is called the testing phase. Finally, solutions are deployed and run on a production environment, where the manage phase takes place.

Development environment

The development environment consists of the IBM Case Manager design object store, target object store, and the associated Process Engine isolated region. Use the development environment to design, deploy, test, and refine case solutions before you move them into the test, pre-production, or production environment.

In the design phase, you create a case solution for the business problem. Lay out all the necessary solution artifacts such as properties, roles, case types, tasks, and process workflows by using Case Manager Builder tool. The design object store contains all the case solution artifacts that were created during design

phase by using Case Manager Builder. Target object store contains the deployed solution objects for testing.

The development environment allows for distinct project areas, each with their own target object store and Process Engine region. The development environment supports multiple project areas to allow different business analysts to create their own solutions and reset their environments separately if required.

Test environment

The test environment is similar to the production environment, but is used for user-acceptance, performance, and other pre-production testing. The test environment consists of design object store, target object store, and the associated Process Engine isolated region. However, it does not have Case Manager Builder because solutions are designed in the development environment only. Small test environments are sometimes configured to share resources with a development environment, for example by using the same database or LDAP server.

For the test phase, the solution artifacts in the development environment are exported or transferred to the test environment. Do so by using the IBM Case Manager administration client and FileNet Deployment Manager tools. After you are satisfied with the performance of the solution in the test environment, it can be moved from the development environment to the production environment.

For more information about these tools, see 3.4.1, “Case Manager Builder” on page 52 and 3.4.2, “Case Manager Client” on page 52.

Production environment

The production environment contains the FileNet Content Engine and Process Engine. It is where the case solutions are deployed, run, and managed. The production environment consists of design object store, target object store, and the associated Process Engine isolated region. Similar to the test environment It does not have Case Manager Builder tool. Typically, this environment separate from other environments so the production system is isolated from any testing problems.

For the manage phase, the solution package and related artifacts are exported or transferred from the development to the production environment. You can move them using IBM Case Manager administration client and FileNet Deployment Manager tools. They are then deployed to the target object store and Process Engine region for execution.

Figure 3-1 shows the IBM Case Manager phases and environments. CMAC in the figure stands for IBM Case Manager administration client. The solid arrow indicates that you deploy the solution to the test environment first. After successful testing, you deploy the solution to the production system as shown by the dashed arrow.

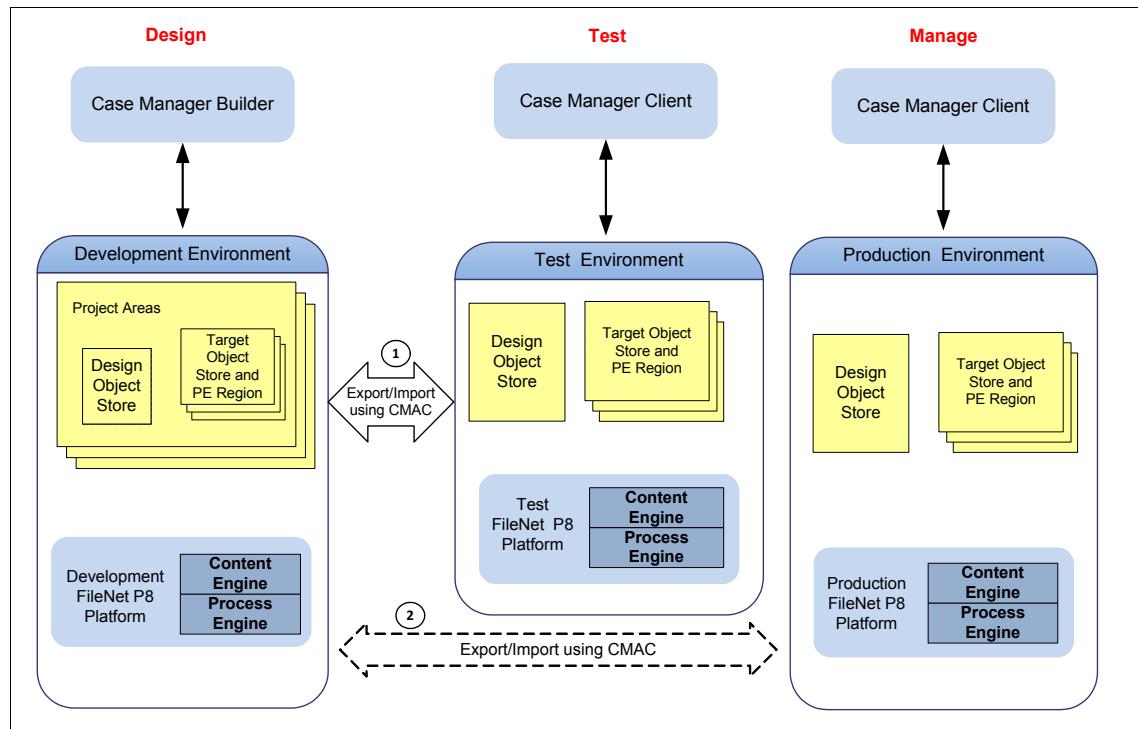


Figure 3-1 IBM Case Manager phases and environments

Explanation: Figure 3-1 explicitly shows only three environments because they differ in the way they are used. Case Manager supports having more than three environments. The extra environments are typically more test environments that focus on dedicated aspects like performance or integration testing.

3.2.2 Solutions and deployment in development environment

Use the development environment to design your solutions. Solutions that are being designed are in the IBM Case Manager design object store. Case Manager Builder is used to build solutions, initiate development deployments, and reset the target object store and region.

The development environment provides project areas with their own target object stores and Process Engine regions. Project areas help you test your solutions before you move them to the test, pre-production, or production environment. You can modify, refine, and test solutions within the development environment iteratively.

After the design cycle is finished and the solution is ready for testing, use IBM Case Manager administration client to transfer the solution package from the development environment. You can transfer the solution to any test or production environment.

3.3 IBM Case Manager components

This section describes all the components that comprise IBM Case Manager. The core platform for IBM Case Manager consists of the IBM FileNet Content Engine and IBM FileNet Process Engine. The IBM FileNet P8 Platform has been extended to include the IBM Case Manager native case object model. Except for the Business Space database, IBM Case Manager itself does not contain any process or content data. Rather, IBM Case Manager is an integration layer that is supported by a core FileNet P8 services layer.

All data that are related to cases managed in IBM Case Manager solutions are stored in the Process Engine and Content Engine repositories. The Business Space configuration data are stored in a separate database. This database can also be collocated with the Content Engine and Process Engine database.

Figure 3-2 on page 39 shows the building blocks of IBM Case Manager. These abbreviations are used in the figure:

- ▶ CMAC: IBM Case Manager administration client
- ▶ FDM: FileNet Deployment Manager
- ▶ FEM: FileNet Enterprise Manager
- ▶ CE: Content Engine
- ▶ PE: Process Engine
- ▶ BSpace: Business Space

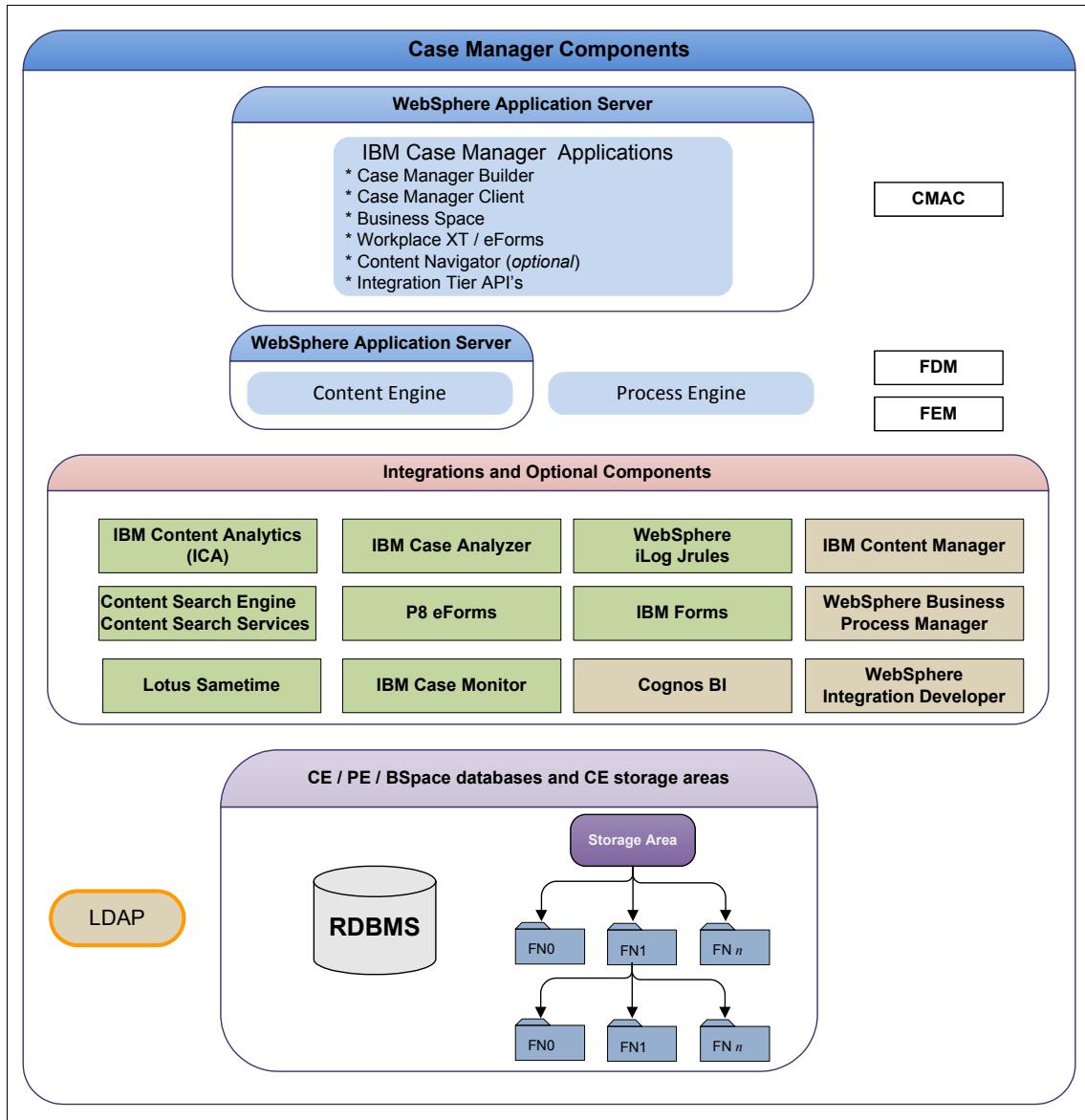


Figure 3-2 IBM Case Manager components

IBM Case Manager components can be divided into these categories:

- ▶ Core components
- ▶ Optional components
- ▶ IBM Case Manager extensions

The optional components, which are shown in green in Figure 3-2 on page 39, are bundled with the IBM Case Manager license. The brown color indicates integrated components that must be licensed separately.

3.3.1 IBM Case Manager core components

The following IBM Case Manager components provide core functionality for designing and processing case solutions:

- ▶ Case Manager Builder
- ▶ Case Manager Client
- ▶ Case Manager API
- ▶ Case Manager administration client
- ▶ Content Engine
- ▶ Process Engine
- ▶ FileNet Workplace XT

This section introduces each of these components.

Case Manager Builder

IBM Case Manager provides a comprehensive design tool, Case Manager Builder, to model case solutions to solve real world business problems. Case Manager Builder is an intuitive, easy-to-use, web-based application. It enables you to create all case solution artifacts such as tasks, document types, and properties. You can also associate workflows and tasks. The tool can also deploy these artifacts to a pre-configured project area that consists of the target object store and Process Engine region within the development environment.

Case Manager Builder creates a package of XML files known as the solution package. It uses the Case Manager REST API to deploy the artifacts that are defined in the solution package into the target object store and Process Engine region.

Case Manager Builder is used in the development environment to design and develop the solution. When the solution is ready for system-level testing, it is transferred and deployed into the test environment by using the IBM Case Manager administration client tool. This process can be done for performance testing and user-acceptance testing.

Figure 3-3 shows the Case Manager Builder home page that you see after you log in to Case Manager Builder. This page lists all available solutions in the design object store and their deployment status inside the development environment.

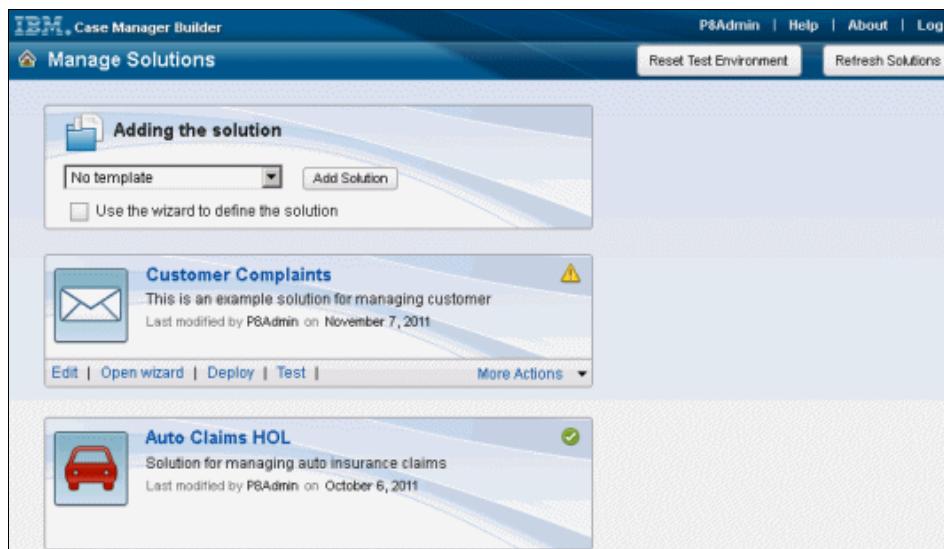


Figure 3-3 Case Manager Builder home page

Case Manager Client

Case Manager Client is an application that case workers use to create and process cases, and collaborate with others. It also allows business analysts or IT professionals to create customized page layouts for cases and work items to provide case workers with all information they need to make an informed decision as quickly as possible.

Case Manager Client is based on IBM iWidget technology, and is built on top of the IBM Business Space Mashup container. It uses the Dojo JavaScript toolkit to deliver a set of ready-to-use widgets that can be used for your case management user interface. Case Manager Client integrates with the IBM Lotus Sametime LiveName application programming interface (API) for Sametime awareness and web chat capability to support case processing collaboration.

Case Manager Client uses Case Manager Representational State Transfer (REST), Process Engine REST, and Content Management Interoperability Services (CMIS) REST APIs to communicate with the underlying FileNet P8 environment.

Figure 3-4 shows a Case List page in the Case Manager Client for a Claim Management solution.

The screenshot shows the 'Customer Complaints' application interface. At the top, there's a navigation bar with 'Work' and 'Cases' tabs, and a search bar labeled 'Search:'. Below the search bar is a 'Complaint Category' dropdown set to 'Billing'. There are two buttons: 'Search' and 'Advanced Search'. The main area is titled 'Case List' and displays two case entries. Each entry includes a case number, case source, status, category, and a 'More' link. The first case is 'CC_Complaint_000000110001' with case number 00002, source Fax, status Open, category Billing, and a 'More' link. The second case is 'CC_Complaint_000000110003' with case number 04, source Fax, status Closed, category Billing, and a 'More' link. Both cases also have a 'Complaint Category: Billing' link below them. Navigation links 'Items 1 - 2', 'Previous | Next', and 'Sort by: Date Modified' are at the top of the list, and 'Sort: Ascending' is at the bottom right.

Figure 3-4 Case Manager Client for Customer Complaint Management solution

Case Manager Client comes with a default set of spaces and pages. When you deploy a solution, it is created with default spaces and pages to process the cases in IBM Business Space. You can then customize the default pages, and create and register new custom pages in Case Manager Client.

For more information about spaces and pages, see 4.1, “IBM Case Manager object model” on page 67 and 4.2, “Case object model implementation” on page 97.

Case Manager API

Since IBM Case Manager release 5.1.1, two different API are available the Case REST API and the Case Java API. The Case REST API was published with the initial release of IBM Case Manager, but the Case Java API was not exposed in former releases.

Case REST API

The Case REST API consists of three REST-based APIs: CMIS, Process Engine REST, and Case Manager REST. The IBM Case Manager API provides an integration tier between the IBM Case Manager application components and the underlying FileNet P8 engines.

The IBM Case Manager API has a full set of APIs for users to build custom applications on top of IBM Case Manager. For details about the CMIS and Process Engine REST APIs, see the FileNet P8 documentation.

The Case Manager REST API supports the REST architecture style. The Case Manager REST API uses the Process Engine Java API and the Content Engine

Java API. These applications handle case solution requests from the IBM Case Manager components, notably those from Case Manager Builder and Case Manager Client.

Case Java API

The Case Java API is provided as a single JAR file (`acmapi.jar`) and the Case Manager Information Center provides both a Java Docs reference and a development guide. From a functional point of view, the Case Java API roughly corresponds to the capabilities of the Case REST API. From an architectural point of view, the Case Java API existed in earlier versions and provided the foundation for the Case REST API as shown in Figure 3-5.

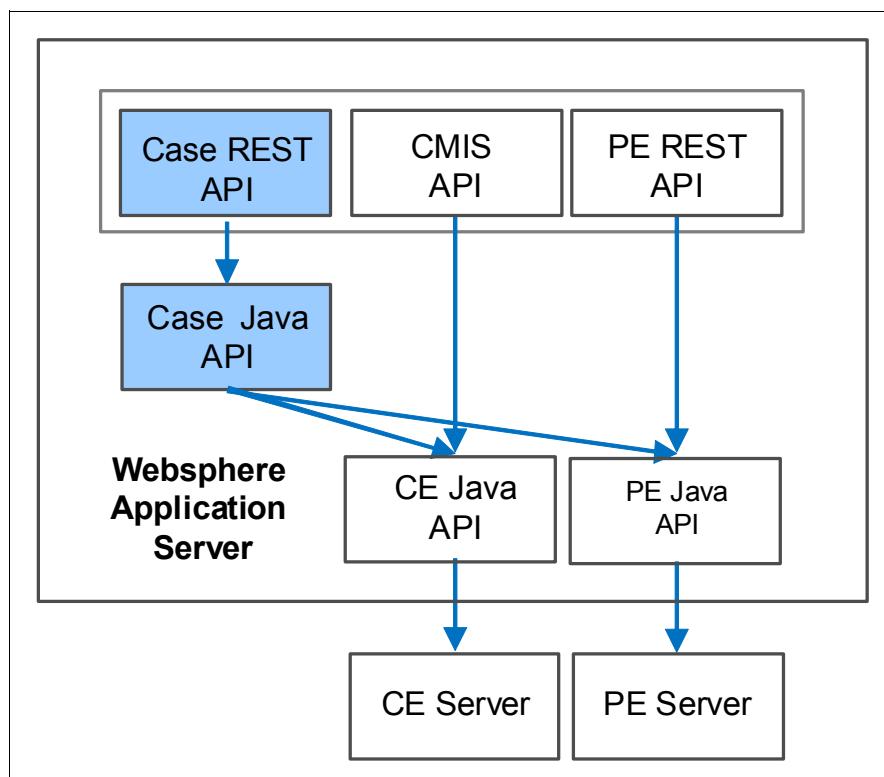


Figure 3-5 IBM Case Manager API architecture

The following is a partial list of the capabilities both IBM Case Manager API support:

- ▶ Discovering information about deployed solutions that includes configuration parameters like document types or Process Engine Connections Point used
- ▶ Creating case instances

- ▶ Updating case properties
- ▶ Retrieving case tasks
- ▶ Splitting or relating cases
- ▶ Adding or retrieving case comments
- ▶ Retrieving case history
- ▶ Start manual tasks
- ▶ Enable or disable certain tasks
- ▶ Create discretionary tasks

For a full listing of the IBM Case Manager API capabilities, see the IBM Case Manager documentation. Open the IBM Case Manager Information Center and select **Developing case management applications** → **Creating and managing case objects by using the IBM Case Manager REST protocol** or **Developing case management applications** → **Java application development** → **IBM Case Manager Java API**.

Case Manager administration client

The Case Manager administration client is a tool for configuring the IBM Case Manager environment. IBM Case Manager administration client organizes configuration tasks into two profiles:

- ▶ Development environment profile

This profile contains a set of tasks for setting up the development environment. It is used to deploy and configure Case Manager Builder, Business Space, Case Manager Client, and IBM Case Manager API web applications to a WebSphere Application Server profile. You can prepare, design, and target object stores with all the requirements needed to develop and deploy solutions in a test environment.

This profile also contains tasks for configuring bootstrap settings, default project area, and object stores, and for deploying the forms application and integrating IBM Content Navigator. Finally, there are tasks to configure the Business Space Environment to be used to register External Data Services and to generate resources for globalization.

- ▶ Production environment profile

This profile contains tasks that are related to setting up of production environments. It does not contain tasks to configure Case Manager Builder because that tool applies only to the development environment. It also does not provide the task to set up the default project area.

In addition, this profile contains tasks to create connection definitions for deploying solutions to different regions and target object stores for the

production environments, and a task to deploy solutions to these production environments.

Figure 3-6 shows the IBM Case Manager administration client user interface for the Deploy Case Manager Builder Application task in the development environment profile.

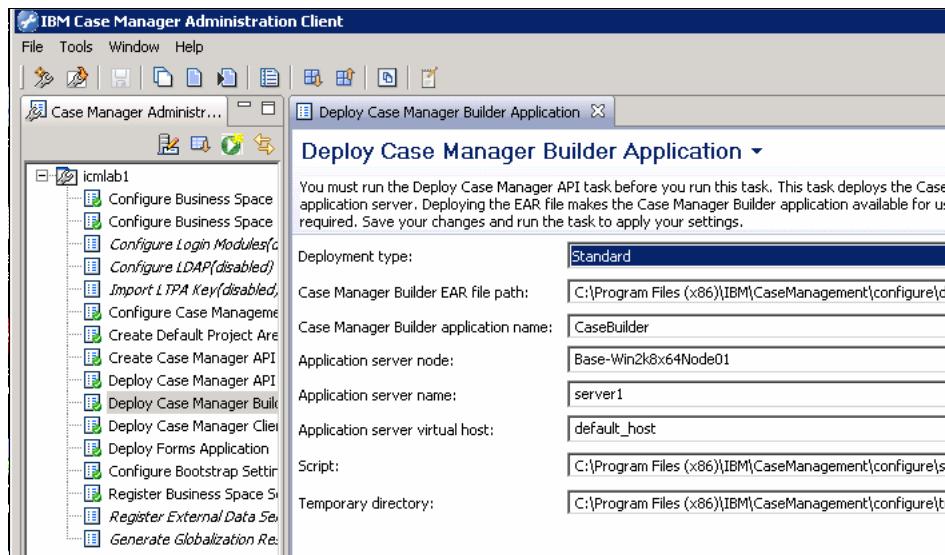


Figure 3-6 IBM Case Manager administration client

In addition to system configuration, you can also use IBM Case Manager administration client to complete these tasks:

- ▶ Create templates from solutions, or solutions from templates
- ▶ Copy solutions
- ▶ Manage project areas
- ▶ Import and export solution package from development environment to production environment, or to another development environment

For more information about using IBM Case Manager administration client and configuring IBM Case Manager, see the IBM Case Manager Information Center at:

<http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/topic/com.ibm.casemgt.install.doc/acmin002.htm>

Content Engine

IBM Case Manager uses Content Engine to maintain the design and target object stores. These object stores contain the solution-related files. The design object store maintains the solution package files that are created through Case Manager Builder. Each solution has its own folder to keep its package of files. The target object store maintains the deployed objects of the solution. Case instances, task objects, and case documents are managed in the target object store. Case Manager client interacts with the target store to create and process cases in the solution.

In addition to storing the solution-related files, Content Engine has been upgraded with the following features to support IBM Case Manager:

- ▶ **Task**
The task class is a Content Engine base class. An instance of the Task class represents a piece of work in a case.
- ▶ **CaseType**
Content Engine implements the case type as a subclass of the Case Folder class (CmAcmCaseFolder).
- ▶ **Content activity monitoring**
Content activity monitoring is an extension of the Content Engine auditing framework that allows you to monitor and analyze property changes.
- ▶ **Case object model**
The Content Engine case object model provides add-ons for the design and target object store metadata, code modules, and subscriptions for handling events from cases.

In addition to the preceding features, Content Engine also provides support for the documents, folders, annotations, and pages that are needed to process a case solution.

Process Engine

Process Engine supports IBM Case Manager mainly in two areas:

- ▶ Designing the solution in Case Manager Builder:
 - Support for creating roles, in-baskets, tasks, and workflows that are associated with case management solutions.
 - Support for workflow authoring in Case Manager Builder.
 - Support for sharing existing Process Engine workflows in solution packages using Process Designer and associating them with Case Type tasks.

- Support for offline validation of workflows in Case Manager Builder.
- Support for augmented solution authoring in Process Designer, which is also known as round-tripping authoring.
- ▶ Execution of cases in Case Manager Client:
IBM Case Manager uses Process Engine to process work items for tasks that are associated with a case instance. This process includes provisioning roles and corresponding in-baskets that contain the workflow steps to be completed by case workers. Process Engine also provides the capability to retrieve, manipulate, and update case instance properties during the execution of work items.

FileNet Workplace XT

FileNet Workplace XT provides a user interface for the IBM FileNet Content Manager and Process Engine applets.

IBM Case Manager release 5.1.1 has a dependency on FileNet Workplace XT for the following functions:

- ▶ To host Process Designer to support the workflow round-tripping between Process Engine and Case Manager Builder
- ▶ To render and display eForms used in the Form Widget.

Additionally, FileNet Workplace XT can be used to make the following capabilities available to IBM Case Manager

- ▶ To provide the document viewer that is used by the Viewer Widget if IBM Content Navigator is not used for that purpose
- ▶ To allow users to add documents by using entry templates.

IBM Case Manager implements its own dialogs for adding and accessing documents and metadata. If case workers must use entry templates with IBM Case Manager for adding documents to a case, WorkplaceXT dialogs must be configured for that purpose.

3.3.2 IBM Case Manager optional components

The following components are bundled and integrated with IBM Case Manager as optional components:

- ▶ Case Analyzer
- ▶ IBM Cognos Real-time Monitoring
- ▶ IBM Content Analytics
- ▶ IBM Content Navigator
- ▶ WebSphere ILOG JRules

- ▶ IBM FileNet eForms
- ▶ IBM Forms
- ▶ Lotus Sametime

Although these components are integrated with IBM Case Manager, they are not essential for solution authoring or execution. However, they can add great value and power to IBM Case Manager solutions.

Tip: Some of the optional components that are mentioned are licensed on a limited use basis. For more information, see the licensing terms in the documentation.

Although Cognos BI is not included, Case Analytics reports can be loaded and viewed in the Cognos BI report studio.

Case Analyzer

Case Analyzer integrates with Cognos Real Time Monitor and Cognos Business Intelligence to generate real-time and historical data reports for case events. These reports are based on Content Engine and Process Engine events. Case Analyzer is an enhancement of IBM FileNet Process Analyzer to support IBM Case Manager. As of IBM FileNet P8 Version 5.0 release, Case Analyzer replaced Process Analyzer as part of the IBM FileNet Business Process Manager package.

Cognos Real-time Monitoring

Cognos Real-time Monitoring is used to monitor and run trend analysis for case data. It allows you to define dashboards for different roles that can monitor critical parameters like execution cycle times or in-basket queue depth. You can define watermarks and actions to be taken when a watermark is exceeded. Cognos Real-time Monitoring is also used for IBM FileNet Case Monitor, and has been enhanced to support IBM Case Manager. As of IBM FileNet P8 5.0 release, Case Monitor replaced Process Monitor as part of the IBM FileNet Business Process Manager package.

IBM Content Analytics

IBM Content Analytics supports crawling of structured and unstructured data of cases. Content Analytics contains a crawler for mining case data for analysis and for finding document content, case comments, and case data that are related to individual cases. You can use IBM Content Analytics to explore correlations between unstructured and structured case content with case data. For example, you can search for patterns which repeat for cases that show an unusual long processing time.

IBM Content Navigator

IBM Content Navigator provides enhanced capabilities for viewing documents as described in 10.7.1, “Integration of the IBM Content Navigator viewer” on page 375. It includes rendering and annotation support for MS Office formats and PDF and side-by-side viewing of multiple documents.

WebSphere ILOG JRules

IBM Case Manager integrates with WebSphere ILOG JRules to manage complex business rules for processing cases. This integration allows you to separate the definition and evaluation of the business rules from the definition of task workflows. By using ILOG JRules, you can define and manage business rules centrally in a language that is close to business terms, and use these rules in task workflows.

IBM FileNet eForms

IBM FileNet eForms integration provides electronic form function in IBM Case Manager. It allows you to use a customized form as the user interface for displaying data in a task workflow step and for viewing case properties. eForms facilitates sophisticated data entry capabilities, graphical layout options, and intelligence like calculations and parameter validation.

IBM Forms

IBM Case Manager also provides integration with IBM Forms allowing the user to use rich customized forms in case solutions. It provides a powerful ability to use a form as the user interface for displaying data in a task workflow step and for viewing case properties. It allows the transfer of the data that are entered in form fields to case properties through mapping.

Lotus Sametime

Using Lotus Sametime, case workers can collaborate while processing cases, including initiating instant messaging sessions with others who are working on the case.

3.3.3 IBM Case Manager extensions

IBM Case Manager supports integration with following products as extensions:

- ▶ IBM Business Process Manager (IBM BPM)
- ▶ IBM Content Manager (CM8)

These extensions are not bundled with IBM Case Manager product and are separately licensed. For more information about product licensing, see:

<http://www.ibm.com/software/sla/sladb.nsf>

IBM Business Process Manager (IBM BPM)

IBM Business Process Manager (IBM BPM) integrates with IBM Case Manager. It provides business analysts with an easy way to associate BPM processes to case solutions in Case Manager Builder. IBM Case Manager administration client provides a task to configure IBM BPM Server with IBM Case Manager.

While you are designing a solution, Case Manager Builder looks for a process application on IBM BPM Server with the same name as the solution name. It associates this application to the case solution by using default snapshot. Also, Case Manager Builder provides an enhanced user interface for task editing. This interface allows you to associate process definitions of the process application to Case Type tasks.

Solution deployment checks and validates the process application, snapshot, and property mappings against the settings in IBM BPM Server.

A new Content Engine event handler handles the launching of snapshots on IBM BPM Server and propagates property information based on the defined mapping. At run time in Case Manager Client, a process instance starts when corresponding task is initiated. When the process instance ends, it updates the corresponding output parameters to mark the task complete.

For more information about configuring IBM BPM with IBM Case Manager, see Chapter 17, “Integration with IBM Business Process Manager” on page 587.

IBM Content Manager

The integration of IBM Case Manager with IBM Content Manager allows you to use content that is stored in IBM Content Manager repository for access in case solutions. The integration allows any existing IBM Content Manager content repositories to work with IBM Case Manager seamlessly. IBM Content Manager content remains in its repository, and does not need to be moved or federated to FileNet P8. New case documents are created in IBM Content Manager.

Configuring IBM Content Manager with Case Manager is easy using the provision of connection definition in the project area. Use Case Manager Builder to define IBM Content Manager item type document dependency and preconditions on Case Types and tasks. Event handlers are available in IBM Content Manager and IBM Case Manager to exchange events between repositories.

For more information about configuring IBM Content Manager with IBM Case Manager, see Chapter 15, “Integration with IBM Content Manager” on page 549.

3.4 IBM Case Manager architecture

This section addresses the high-level architecture of IBM Case Manager and describes the interaction between various components as shown in Figure 3-7.

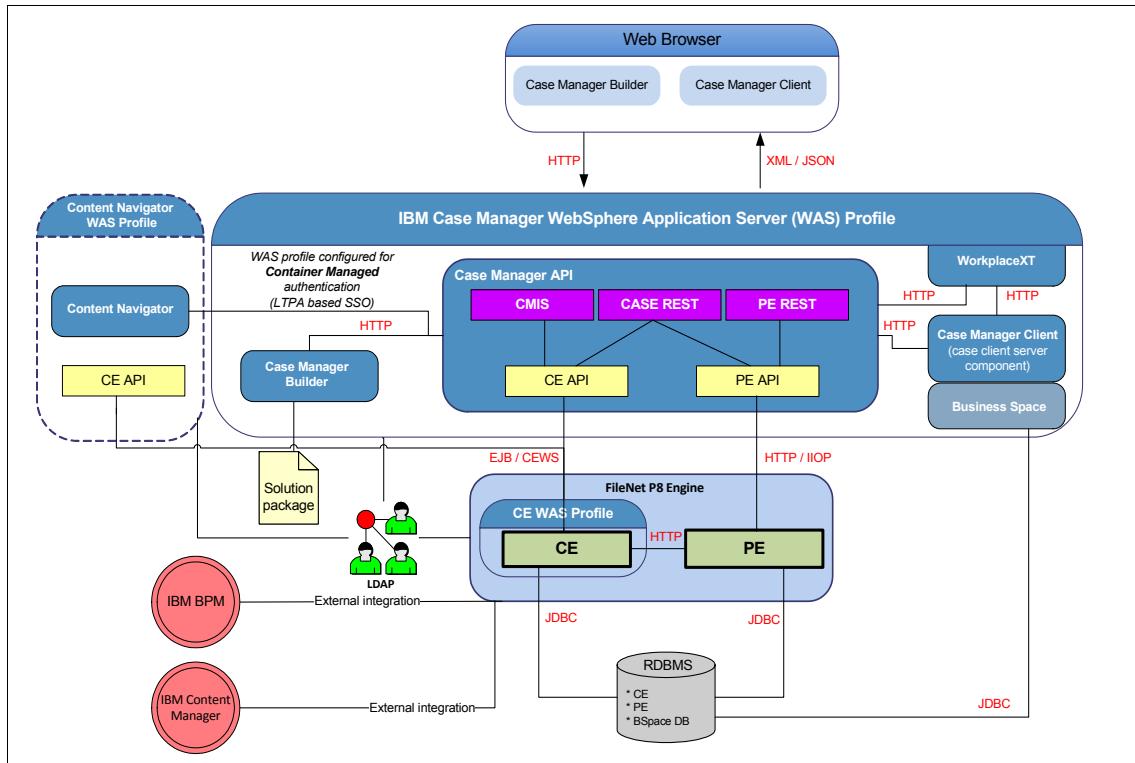


Figure 3-7 IBM Case Manager high-level architecture

Specifically, this section addresses the protocols, communication data, and artifacts for the following applications:

- ▶ Case Manager Builder
- ▶ Case Manager Client
- ▶ IBM Case Manager API
- ▶ Content Engine
- ▶ Process Engine

3.4.1 Case Manager Builder

The Case Manager Builder client runs in your browser, and communicates with the server application using HTTP protocol with JavaScript Object Notation (JSON) payload. For certain actions like deployment, it directly starts the IBM Case Manager REST API by using HTTP. Case Manager Builder creates solution files in the design object store by using the Content Engine Java API.

3.4.2 Case Manager Client

Case Manager Client is built on top of Business Space using the iWidget specification. It starts the IBM Case Manager REST API and Process Engine REST API by using HTTP protocol with JSON and XML payloads.

3.4.3 Case Manager API

Case Manager provides the Case REST API and Case Java API.

Case REST API

The IBM Case Manager REST API has three REST APIs: CMIS, Process Engine REST, and Case Manager REST. These APIs are developed by using the REST architecture style, and the Content Engine and Process Engine Java APIs.

CMIS API accepts HTTP requests from the Case Manager Client and other custom applications. CMIS uses the Content Engine Java APIs to run operations against the Content Engine.

Process Engine REST API accepts HTTP requests from the Case Manager Client and other custom applications. Process Engine REST uses the Process Engine APIs to run operations against the Process Engine.

IBM Case Manager REST API accepts HTTP requests from both Case Manager Builder and Case Manager Client. IBM Case Manager REST API responds to Case Manager Builder and Case Manager Client requests by using HTTP responses with JSON payloads. Depending on the requests, IBM Case Manager REST API connects to Content Engine, Process Engine, or both to run the required operations.

IBM Case Manager REST API creates and maintains numerous artifacts in the design object store and target object store. It also creates Process Engine metadata. The metadata is created by transferring Process Engine configuration and Process Engine workflow definition files to a Process Engine region when Case Manager REST API deploys a solution.

Case Java API

The Case Java API is a separate layer that is called by the Case REST API to access Content and Process Engine by using their respective Java APIs as shown in Figure 3-5 on page 43. With Case Manager release 5.1.1, this API has been officially documented and exposed for public use. The Case Java API also contains private classes that are not publicly exposed, for example for managing artifacts at design time.

You can use the Case Java API in situations where using the Case REST API would introduce significant processor usage. A typical use case is the need to access IBM Case Manager functions in a Java based environment. This situation might occur when you implement a servlet or use the FileNet Process Engine Component Manager.

3.4.4 Content Engine

Content Engine accepts both HTTP and Internet Inter-ORB Protocol (IIOP) requests from its clients. IBM Case Manager uses Content Engine Java API to store, retrieve, and update solution artifacts, solution metadata, and case instances.

3.4.5 Process Engine

Process Engine processes HTTP and IIOP requests. Process Engine retrieves and updates case data while it processes work objects associated with a case task. IBM Case Manager uses Process Engine REST API and Process Engine Java API to communicate with Process Engine and to manage process metadata and work items.

3.5 IBM Case Manager configurations

This section addresses configurations for the three types of IBM Case Manager environments:

- ▶ Development environment configuration
- ▶ Production environment with high availability
- ▶ Production environment with high availability and solution partitioning

3.5.1 Development environment configuration

IBM Case Manager development environments normally consist of single instances of each component. No hardware-based load balancing, server level clustering, or WebSphere ND clustering is required. For development environments, installing IBM HTTP Server (IHS) to front the IBM Case Manager WebSphere Application Server profile is optional.

Figure 3-8 shows the IBM Case Manager and FileNet P8 components that make up the development environment.

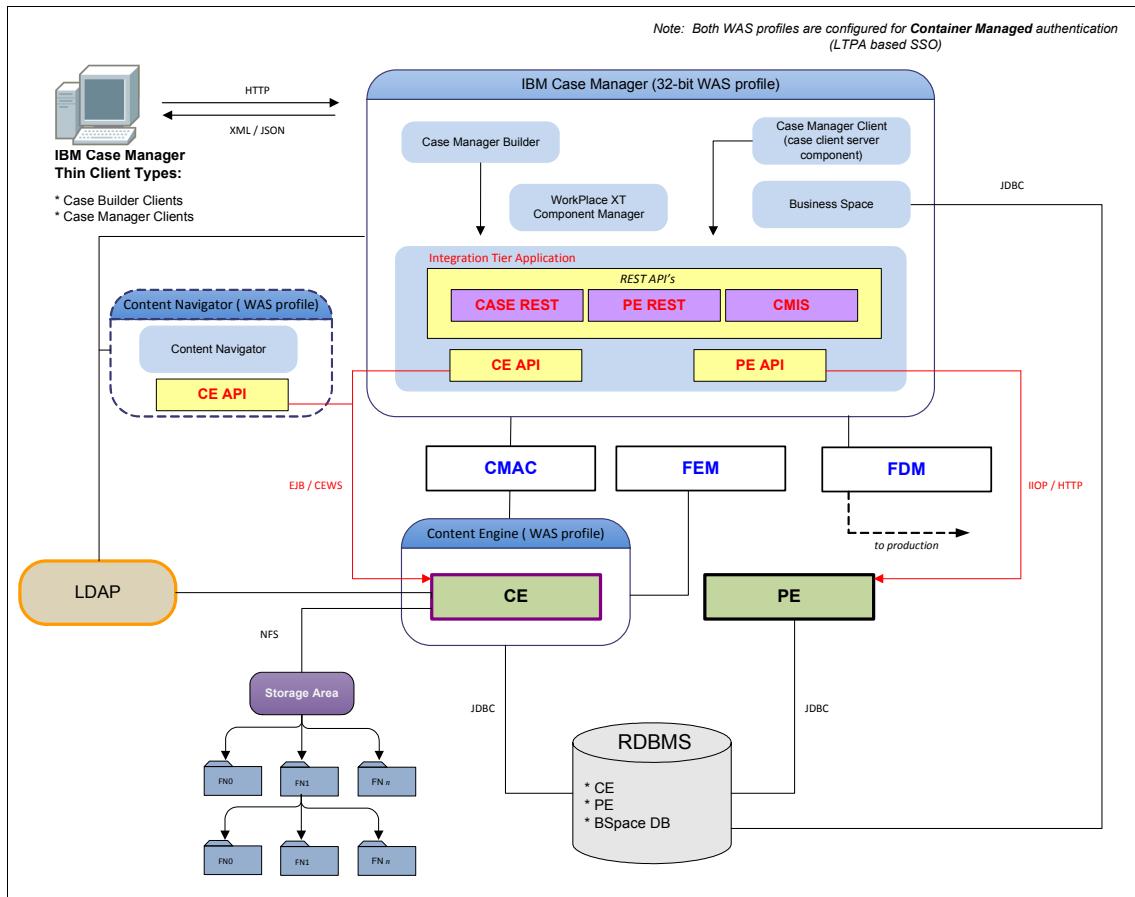


Figure 3-8 Logical view of IBM Case Manager development environment

This environment consists of the following objects:

- ▶ IBM Case Manager WebSphere Application Server profile
- ▶ IBM FileNet Content Engine WebSphere Application Server profile
- ▶ IBM FileNet Process Engine
- ▶ Other administration and configuration applications:
 - IBM Case Manager administration client
 - FileNet Enterprise Manager
 - FileNet Deployment Manager
- ▶ External infrastructure components:
 - Relational database management system (RDBMS)
 - Content storage subsystem
 - Lightweight Directory Access Protocol (LDAP)

Hardware and software requirements for IBM Case Manager and the underlying IBM FileNet P8 Platform are documented in the “IBM FileNet P8 Hardware and Software requirements” guide at:

<http://www.ibm.com/support/docview.wss?rs=3278&uid=swg27013654>

Important: The IBM Case Manager applications must all be in the same WebSphere Application Server profile. The IBM Case Manager administration client does not run under WebSphere Application Server, but must be installed on the same server as the IBM Case Manager applications.

The IBM Case Manager WebSphere Application Server profile and FileNet P8 components (Content Engine and Process Engine) can all be collocated and run by a single operating system. Alternately, they can be separated and run on multiple servers. The servers can be physical servers or guest hosts that run on supported virtualized platforms. The optional IBM Content Navigator component can be collocated with IBM Case Manager or the FileNet P8 components, or it can be installed on a separate server.

Because IBM Case Manager solutions depend on a tight integration between the Content Engine and Process Engine, fidelity between the Content Engine and Process Engine databases must be maintained. To simplify database administrative tasks and support this fidelity, IBM Case Manager requires that the Content Engine object store and associated Process Engine table spaces are created in the same database within the same database schema.

Restriction: The IBM Case Manager Content Engine target object store can be associated with one, and only one, Process Engine database schema.

If multiple IBM Case Manager solutions are deployed into the same target environment, they must all share a single Content Engine object store and Process Engine database. If database isolation between solutions is required, multiple target environments must be defined. These target environments, which consist of separate target object stores and associated Process Engine isolated regions, can exist in the same FileNet P8 domain. They can also be in separate domains.

The WebSphere Deployment Manager is not shown in Figure 3-8 on page 54. However, both profiles are contained in separate WebSphere Application Server “cells,” and both are controlled by the same WebSphere Application Server Deployment Manager. You can deploy the Content Engine into the same WebSphere Application Server profile that is used by the IBM Case Manager applications.

The IBM Case Manager WebSphere Application Server profile that is illustrated in Figure 3-8 on page 54 contains the following applications:

- ▶ **Case Manager Builder Application:**
Solution builder tool that is only deployed in the development environment.
- ▶ **IBM Case Manager run time (also known as the Case Manager Client):**
Runtime environment that is used for starting, processing, and interacting with cases. This run time is deployed in both the development and production environments.
- ▶ **Business Space:**
Used to design user interface layouts and custom pages, and hosts the Case Manager Client iWidgets.
- ▶ **FileNet Workplace XT and Component Manager:**
Case Manager Client uses FileNet Workplace XT for viewing documents (if Content Navigator is not configured) and eForms. A solution can be extended with more functions such as incorporating FileNet Workplace XT entry templates. The component manager runs background operations in task workflows.
- ▶ **Integration Tier Application (also known as IBM Case Manager API):**
Communication layer between components of the IBM Case Manager applications. Also provides you with the ability to customize standard capabilities.
- ▶ **Content Navigator:**
This optional component can be configured to provide extended document viewing capabilities like rendition and annotation support for office documents

and PDF. It can also provide side by side viewing of multiple documents in the same viewer.

3.5.2 Production environment with high availability

The IBM Case Manager production environment is where the Case Manager Clients interact with the IBM Case Manager Runtime environment. The example configuration describes the topology of a sample production IBM Case Manager environment that is configured for high availability. It is focused only on configurations that are specific to IBM Case Manager. Detailed information about high availability solutions for the FileNet P8 Platform can be found at:

<http://www.redbooks.ibm.com/cgi-bin/searchsite.cgi?query=filenet>

IBM Case Manager production environments differ from development systems in the following ways:

- ▶ The Case Manager Builder application is not installed or used in a production environment. Solutions can be designed in the development system only. The solution package must be moved from the development FileNet P8 domain to the production FileNet P8 domain by using IBM Case Manager administration client.
- ▶ In a development environment, Case Manager Builder is used to deploy solutions. In a production environment, IBM Case Manager administration client is used.
- ▶ Solutions cannot be modified in the production environment. Changes to solutions can be made only in a development environment.
- ▶ Production IBM Case Manager environments can contain DMZs, hardware load balancers, HTTP servers (IHS), and WebSphere Application Server ND clustering. Connection strings that map to multiple instances of the application are required for the following URLs:
 - Content Engine EJB URL
 - IBM Case Manager API URLs
 - Case Manager Client URL
- ▶ IBM Case Manager WebSphere Application Server clusters require a network share directory (NFS mount) that is accessible to all the nodes in the cluster. This share directory contains files that are needed by IBM Case Manager API for globalization resources. These files are needed when the IBM Case Manager deployment profile is created.
- ▶ Production IBM Case Manager environments can take advantage of FileNet P8 distributed deployment and domain partitioning configuration options.

- ▶ The LDAP security configuration for production solutions is more restricted than the development environment. Production environments can use a different LDAP than development. The LDAP is tied to the FileNet P8 domain that hosts the IBM Case Manager production domain.
- ▶ Production environments can contain a secure DMZ-fronting IBM Case Manager.
- ▶ A single FileNet P8 domain is used for the IBM Case Manager development and target environment. Production environments can consist of either single or multiple FileNet P8 domains, although each FileNet P8 domain requires its own design object store. Each production domain can have one or more target environments.

Figure 3-9 shows a production IBM Case Manager environment with built-in redundancy for each component to provide higher availability. This diagram does not include the optional Case Manager components.

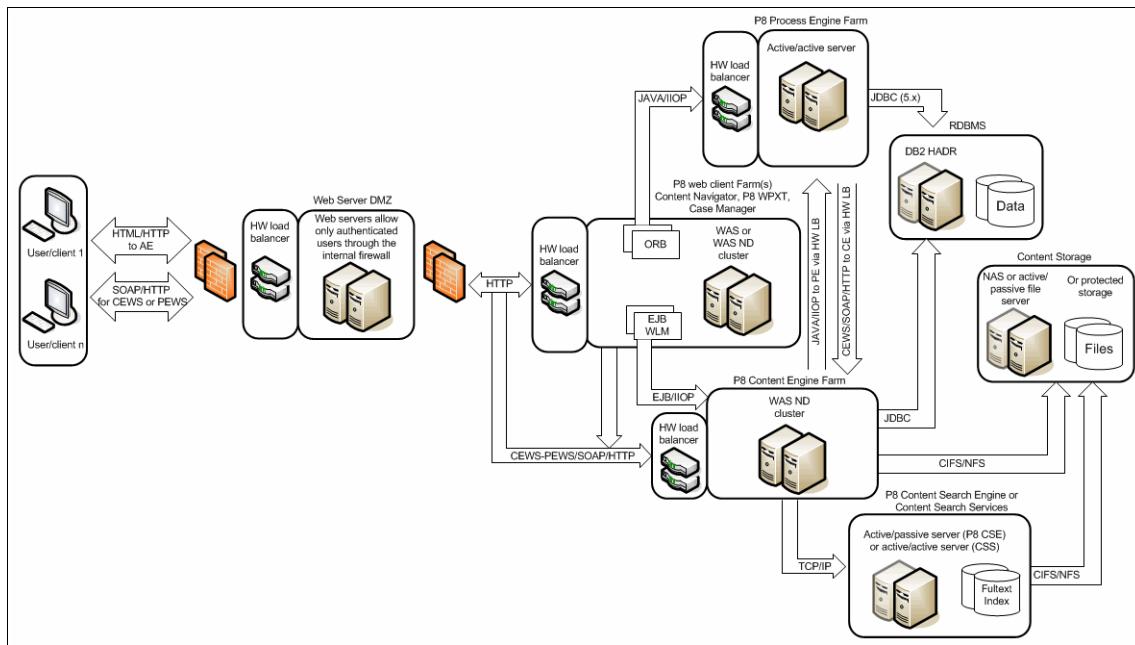


Figure 3-9 Overview of a highly available Case Manager environment

Case worker clients connect to the Case Manager Client URL to log in and interact with cases. In a highly available IBM Case Manager WebSphere Application Server profile, there are two or more instances of Business Space (of Case Manager Client) application running. Load balancing and failover for the Business Space application instances are achieved by using WebSphere

Application Server Network Deployment for this application cluster, fronted by a load balancer.

Typically this solution is either IBM HTTP Server (IHS) or a hardware-based device. Case Worker client connections require session persistence. This means that the load balancing mechanism must support session affinity or “sticky sessions”. This session persistence ensures that clients are always directed to the same Business Space instance resource initially chosen to receive the traffic.

Other web client components like Content Navigator or FileNet Workplace XT can be deployed into either WebSphere Application Server Network Deployment or WebSphere Application Server Standard Edition clusters.

Figure 3-10 shows a Business Space Virtual IP address that is mapped to two Business Space servers that are running in a WebSphere cluster. The Business Space Servers are shown running on two separate IBM LPARs. This configuration provides both load balancing and fault tolerance.

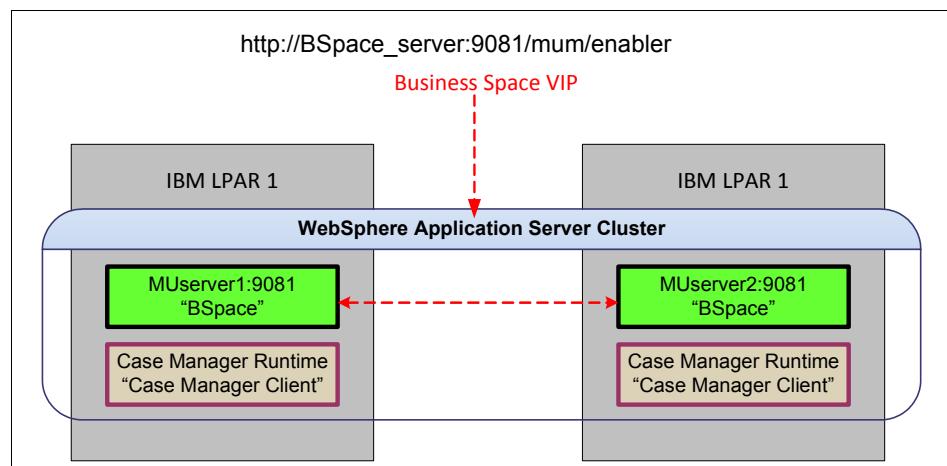


Figure 3-10 Business Space servers configured for high availability and fault tolerance

In an IBM Case Manager environment, both the Enterprise JavaBeans (EJB) and web services interface are used to interact with the Content Engine. More information about high availability and scalability for the Application Engine, Content Engine, and Process Engine can be found at:

<http://www.redbooks.ibm.com/abstracts/sg247700.html?Open>

<http://www.ibm.com/support/docview.wss?uid=swg27010422>

<http://www.redbooks.ibm.com/cgi-bin/searchsite.cgi?query=filenet>

Each target environment can have only one Process Engine isolated region. Therefore, the Process Engine farm in an IBM Case Manager environment can

connect only to a single isolated region in the database. If the production environment contains more than one target environment, more Process Engine farms are required.

Figure 3-11 shows two highly available Process Engine farms, each connecting to separate target environment isolated regions.

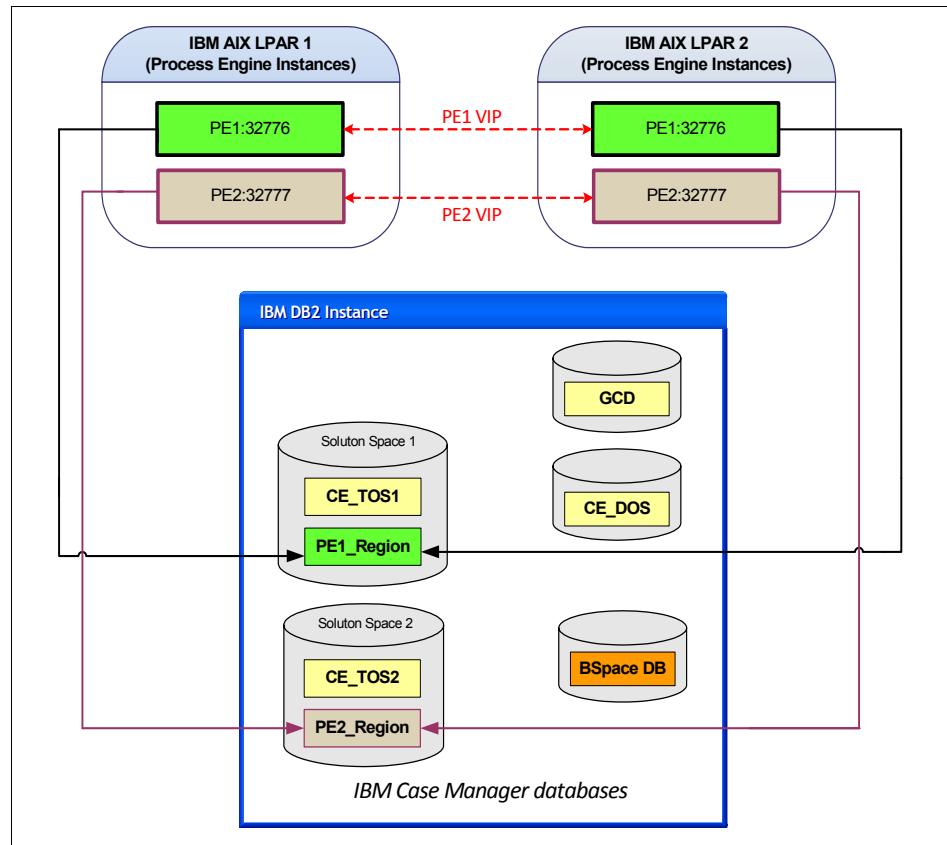


Figure 3-11 Example of Process Engine farming and IBM Case Manager environment

In this example, each Process Engine isolated region is paired up with its associated Content Engine target object stored in the same database. The two pairs create two target environments. This configuration satisfies the IBM Case Manager requirement that the Content Engine and Process Engine table spaces for each target object store be collocated in the same database. Also, because both solutions are contained in the same FileNet P8 domain, there are only one design object store and one Business Space database.

In Figure 3-11 on page 60, a virtual IP address (VIP) is used to address each Process Engine farm.

3.5.3 Production environment with partitioning

Some IBM Case Manager implementations require that individual solutions be segregated within a single production FileNet P8 domain or be isolated into separate FileNet P8 domains. This section covers some sample configurations for IBM Case Manager illustrating separation options for individual solutions.

When you are planning IBM Case Manager solutions, take into account the following considerations:

- ▶ Estimated number of solutions and case workers
- ▶ Anticipated number of concurrent solution designers
- ▶ Do LOBs require dedicated environments, or can shared target environments be used?
- ▶ In terms of software maintenance, is it acceptable to have multiple environments that need patching and updating?
- ▶ In terms of operational considerations, is being able to stop, start, and back up IBM Case Manager solutions independently a requirement?

The simplest configuration is to host all IBM Case Manager solutions in a single target environment. In this type of configuration, all of the solutions are deployed to a single target object store and share a single Process Engine isolated region.

Figure 3-12 shows a single FileNet P8 domain with three solutions being deployed from the CE_DOS design object store to the CE_TOS1 target object store. The Business Space database for the Business Space application WebSphere Application Server ND cluster and Content Engine global configuration database (GCD) are also shown.

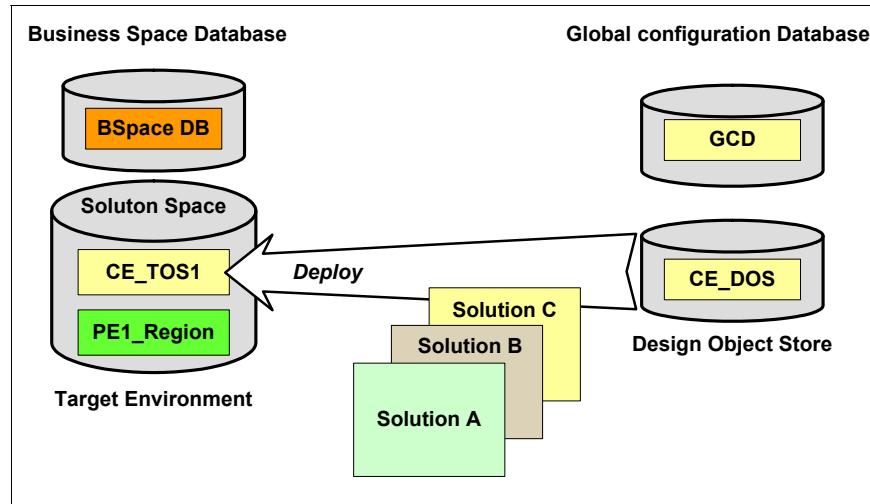


Figure 3-12 One target object store hosting three IBM Case Manager solutions

Instead of having all the IBM Case Manager solutions be in a single shared target environment, each solution can be deployed into a separate target environment. In this configuration, each solution is provided with a dedicated Content Engine target object store and Process Engine isolated region. This configuration provides data separation between the solutions, thus providing more flexibility for database operations and maintenance.

Figure 3-13 shows this type of configuration.

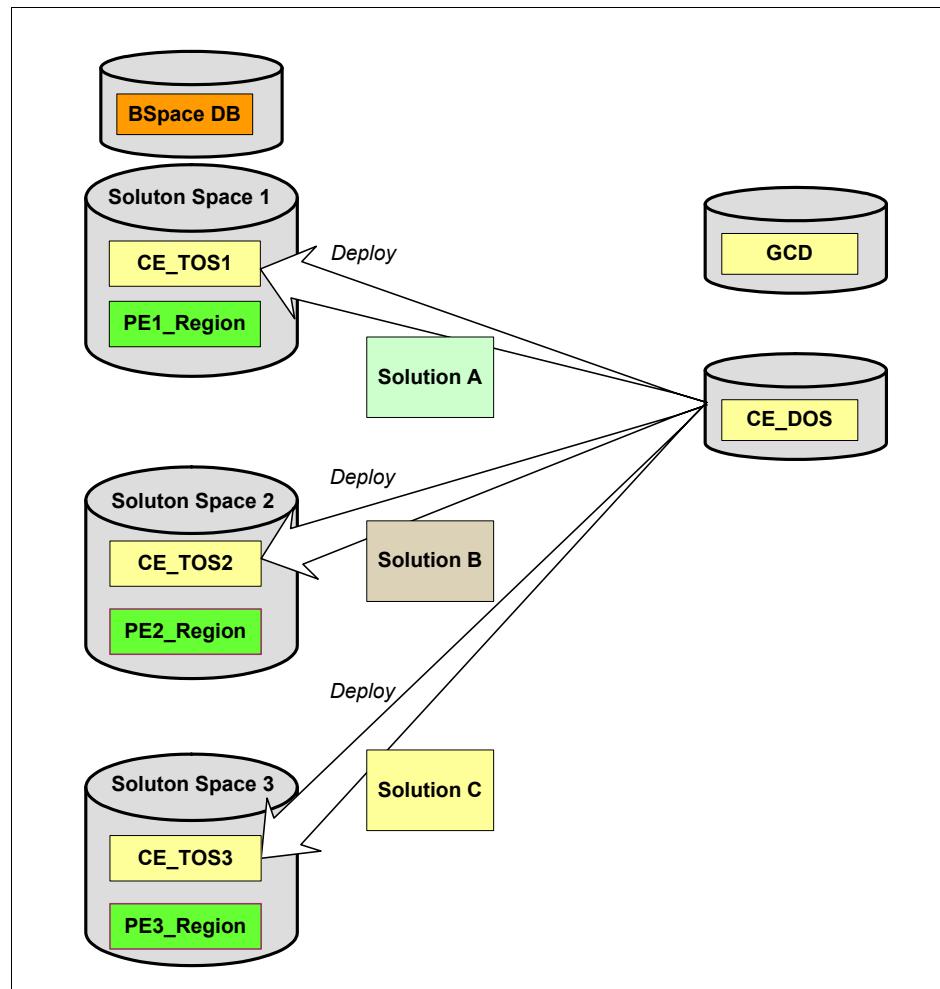


Figure 3-13 Multiple target object stores with one solution in each

If you want total solution separation, each solution can be deployed into its own FileNet P8 domain. Figure 3-14 illustrates this type of configuration, showing that the databases for all three domains are hosted by the same DB2 instance.

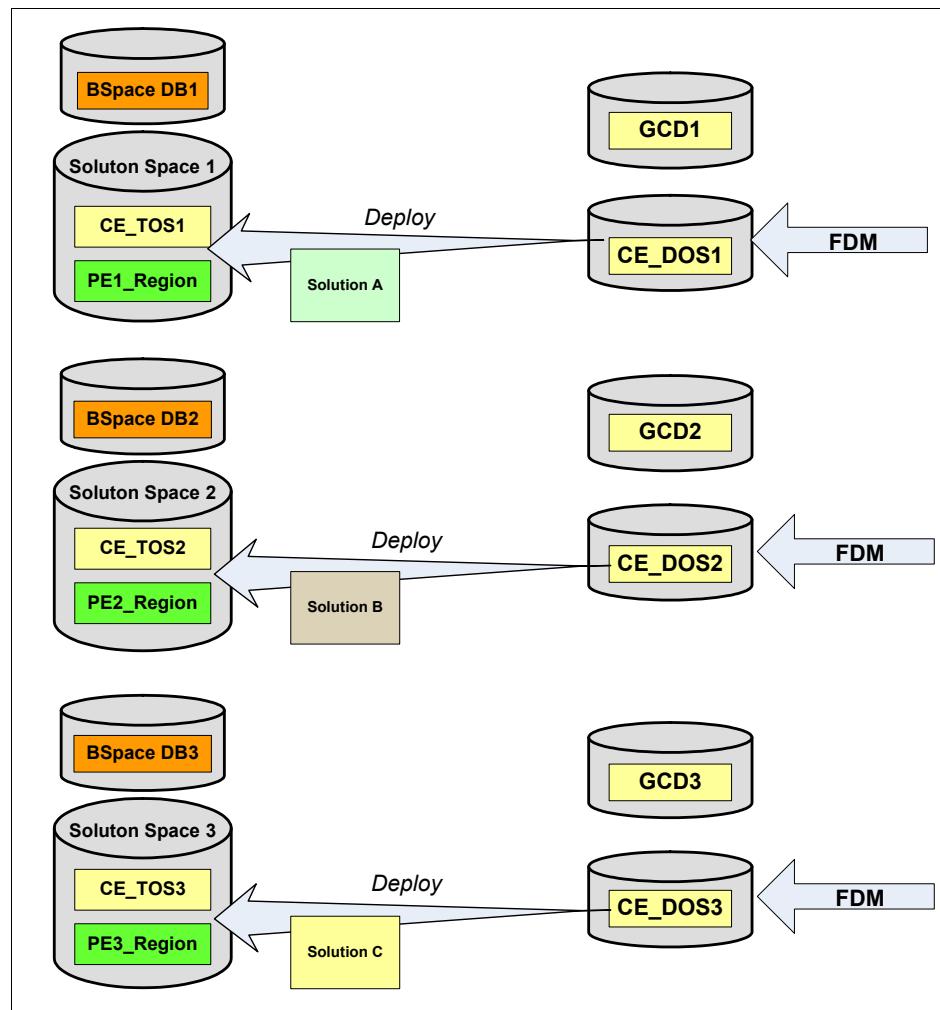


Figure 3-14 Dedicated FileNet P8 domain for each IBM Case Manager solution

It is also possible to host the databases for each domain in separate instances or database servers. This type of deployment provides the most flexibility for production environments. In Figure 3-14, the term *FDM* denotes the *FileNet Deployment Manager*, which you use to transfer the solution packages to the design object stores in the target environments.



Inner workings of IBM Case Manager

If you plan to develop customized IBM Case Manager solutions, a good understanding of the inner workings of IBM Case Manager is vital. In addition, this understanding is also useful when troubleshooting IBM Case Manager solutions.

This chapter describes the inner workings of the IBM Case Manager. It addresses the IBM Case Manager object model, and the way it uses the internal services from the IBM FileNet Process Engine and IBM FileNet Content Engine. This chapter also describes the integration with the IBM Content Manager from the data model perspective.

This chapter includes the following sections:

- ▶ IBM Case Manager object model
- ▶ Case object model implementation
- ▶ IBM Case Manager spaces and pages
- ▶ Tasks and associated workflow processes
- ▶ Object model for IBM Content Manager integration

Important: This chapter assumes that you are familiar with IBM FileNet P8 Platform. If not, see the following Redbooks and product information center:

- ▶ *IBM FileNet P8 Platform and Architecture*, SG24-7667
- ▶ *IBM FileNet Content Manager Implementation Best Practices and Recommendations*, SG24-7547
- ▶ *Introducing IBM FileNet Business Process Manager*, SG24-7509
- ▶ IBM FileNet P8 Platform main information page at:
<http://www.ibm.com/software/data/content-management/filenet-p8-platform>

4.1 IBM Case Manager object model

IBM Case Manager object model consists of the solution, case type, task, external process, step, document type, and role. Figure 4-1 shows the IBM Case Manager object model.

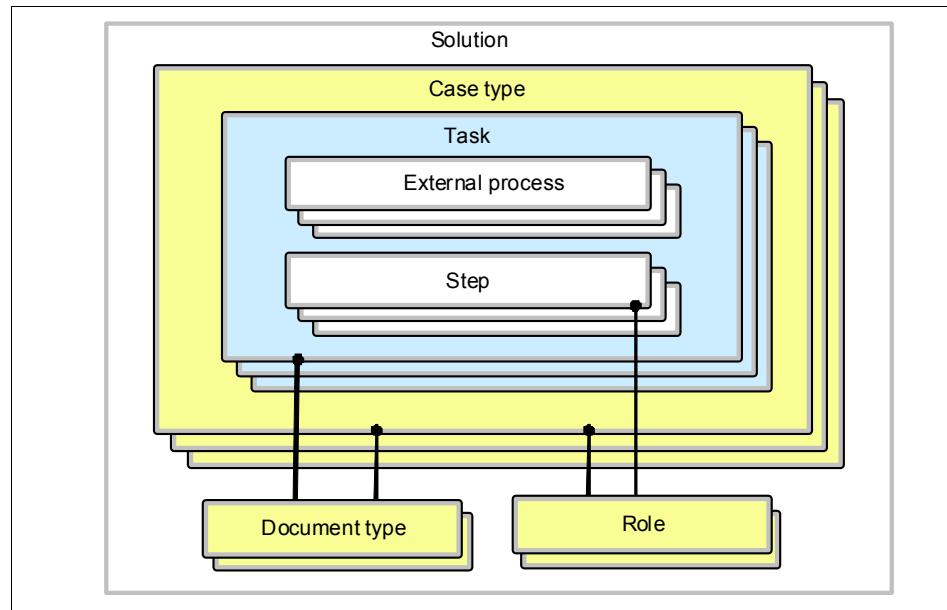


Figure 4-1 IBM Case Manager object model

There can be many *solutions* in an IBM Case Manager system. For example, you can have one for Complaints Management, one for Human Resource Case Management, and one for the Legal Department.

Each solution can handle a number of different *case types*. For example, the Complaints Management system must handle customer complaints, internal employee complaints, and customer feedbacks. The solution might have a case type for each of these inputs.

Each solution shares a number of metadata definitions (properties), document types, and roles.

Each case type can have a number of *tasks*. A task has one or more steps that must be completed. A task can also start external processes. For example, a task in the Complaints Management use case might be to review a product-related complaint.

There are different kinds of tasks:

- ▶ Repeatable tasks
- ▶ Inclusive or exclusive sets of tasks
- ▶ Optional tasks
- ▶ Discretionary tasks
- ▶ Event driven tasks

A *role* represents a specific business function. For example, in the Complaints Management system, a role might be a Reviewer or Investigator. You assign users or groups of users to a role. People with different roles can access cases in different ways, and roles are assigned to run a particular step in a task.

Document types help you to organize and classify the documents that belong to a case. You can provide more information about the documents by assigning properties to the document type. For example, a document type in the example might be a complaint letter. Different types of cases can access different types of documents. Documents are typically used by the case workers while they process task steps.

The following sections provide more detail about each entity within the IBM Case Manager object model.

4.1.1 Solution

A *solution* consists of a user interface together with document types, property types, case types, and tasks. For example, a solution in a Complaints Management system might include these case types:

- ▶ Customer complaints
- ▶ Internal employee complaints
- ▶ Customer feedbacks

The different types of cases within this solution are related, but have different tasks that must be completed to complete the case. Roles are defined at the solution level so case workers with roles in this solution can handle different case types within it. The case types all use similar document types such as complaint letter, and the properties such as the date when complaint was received.

Figure 4-2 shows a diagram of the solution components.

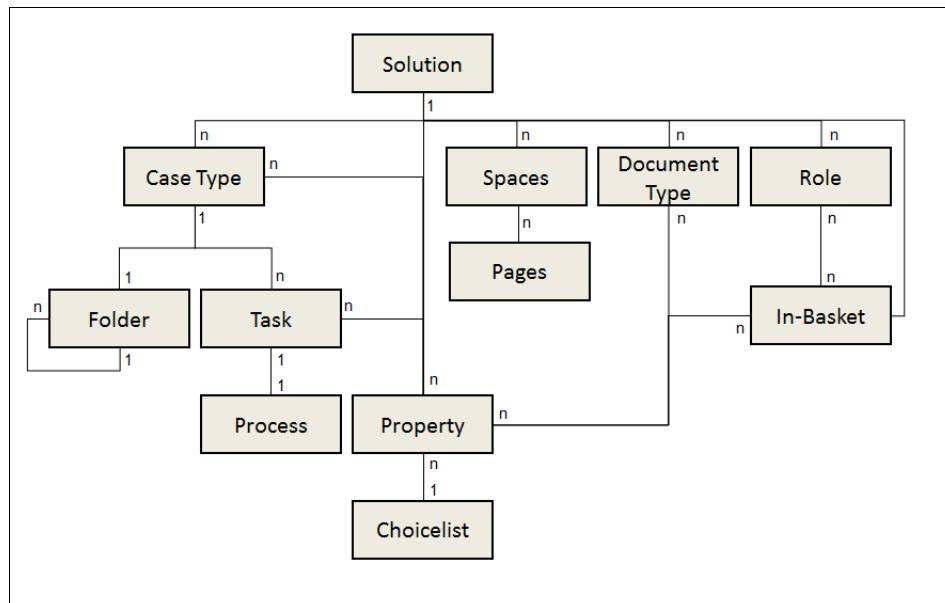


Figure 4-2 Solution object model

Each IBM Case Manager solution has a solution prefix to provide unique identification in a Content Engine object store and Process Engine isolated region.

Case Manager Client represents a solution as a *solution space* into which users log to manage and process their cases. Solution spaces contain the pages for case and step data. These pages are known as *Case pages* and *Step pages*. The pages display user interface widgets that allow users to view in-baskets of available work, process work, and search for cases.

A solution is represented in a Content Engine target object store as a folder of type Deployed Solution (CmAcmDeployedSolution). The deployed solution folder contains definitions for case types and their instances. The name of the deployed solution folder is the solution name.

For details about the CmAcmDeployedSolution class, see the IBM Case Manager Information Center. Click **Developing case management applications** → **Content Engine add-on extensions for Case Manager Builder** → **IBM Case Manager target object store extensions** → **Custom subclasses of the Folder class** → **Deployed Solution class**.

A solution is represented as an *application space* in a Process Engine isolated region. An application space contains the roles and their in-baskets. A roster is defined in a Process Engine isolated region for each solution. The application space name is the solution name. The roster name is a normalized version of the solution name to conform with the naming convention for a roster in Process Engine.

Solution package

A *solution package* is a set of XML files and folders that are the artifacts of a solution. Figure 4-3 shows the solution package content.

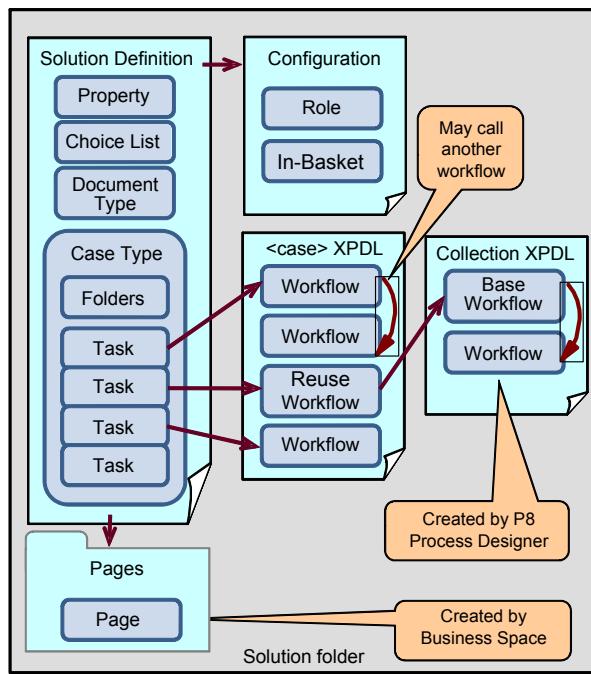


Figure 4-3 Solution package

A solution package contains the following files and folders:

- ▶ **Solution Definition File (SDF)**
The SDF contains definitions of property types, document types, case types, and task types that are used in the solution.
- ▶ **Process Engine Configuration File**
The Process Engine Configuration File defines the Process Engine configuration for a solution. The file contains definitions for the application

space, roster, event logs, queues, in-baskets, roles, and step processors. A step processor is the step page in Case Manager.

- ▶ **Case Type Workflow Definitions (Case XPDL)**

The XML Process Definition Language (XPDL) is a collection of workflows that are associated with each case type. Case Manager Builder creates a separate XPDL file for each case type. Each task in a case type corresponds to a workflow in the XPDL. The XPDL file can also contain zero or more workflows that are not associated with a task. These workflows can be used by any task workflow in the XPDL file.

For the task with reused workflow, Case Manager workflow inherits the original workflow that was imported in the solution workflow collection.

- ▶ **Solution Workflow Collection (Collection XPDL)**

The XPDL contains the reusable FileNet BPM workflow definitions. These definitions are created by using FileNet Process Designer. They are available to be used as a “Reuse workflow” in the solution if their base class is changed to CaseWorkObject.

- ▶ **Pages folder**

The pages folder contain page objects that represent the Case Manager Client user interface layouts (Business Space and pages). The page objects provide a mechanism for Case Manager Builder to display a list of available page layouts for task steps and for role-based case detail views. When a solution package is created for deploying into another environment, space and page definition files are uploaded from the Business Space database. These files are stored as content on the page objects.

In addition to these artifacts, there might be more artifacts included in a solution package, such as forms and document templates.

4.1.2 Case types

A case type defines the tasks, the document types needed to support them, and their steps to solve a particular business problem. The definition of a case type also includes the case folder structure, and allocation of solution properties that are displayed to case workers in the Case Manager Client. A case in an instance of a case type.

A case type has the following items:

- ▶ A collection of properties that become case folder properties.
- ▶ A collection of property views that specify which properties Case Manager Client displays in the case widgets.

- ▶ A folder structure for storing documents.
- ▶ A collection of tasks to specify business processes and process data.
- ▶ A page layout that specifies which Client widgets to use to create cases, view cases, and split cases at run time.
- ▶ Optionally an initiating document type. In an IBM Case Manager system, users can create a case instance directly from the Case Manager Client user interface or by adding a document of the initiating document type.

Consideration: When a document of the initiating document type enters the system, it triggers the creation of a case. You can also create a case directly without an initiating document.

Content Engine implements the case type as a subclass of the Case Folder class (CmAcmCaseFolder). The Case Folder class is a subclass of the Base Case class (CmAcmBaseCase).

For more information about the CmAcmCaseFolder class, see the IBM Case Manager Information Center. Click **Developing case management applications** → **Content Engine add-on extensions for Case Manager Builder** → **IBM Case Manager target object store extensions** → **Custom subclasses of the Folder class** → **Case Folder class**.

When you design a solution, Case Builder defines a roster for each solution and an event log for each Case Type in that solution by default. The roster is used by all task workflow definitions for that solution (across all case types). The event log is used for all task workflows that are defined for the case type. You can create an extra event log in Process Designer and assign it to any workflow in the case type XPDL.

The event log name is *<solution prefix>_<normalized case type name>*. Ensure uniqueness within each Process Engine isolated region because the *solution prefix* provides identification for a solution in a Process Engine isolated region. This is important because multiple solutions can have the same case type name.

If the case type properties are used in the steps of the workflow, they become Process Engine data fields for the workflows that are associated with the case. For more information about Process Engine and Content Engine interaction for a case regarding case properties, see 4.4, “Tasks and associated workflow processes” on page 111.

Cases

A *case* is an instance of a case type. It is also called a *case instance*. A case is persisted by using a case folder structure.

Figure 4-4 shows a case instance. As illustrated in the diagram, a case or case instance is created from case type. A case instance comes with a case folder, case properties, optionally tasks that are associated with the case, and optionally case comments.

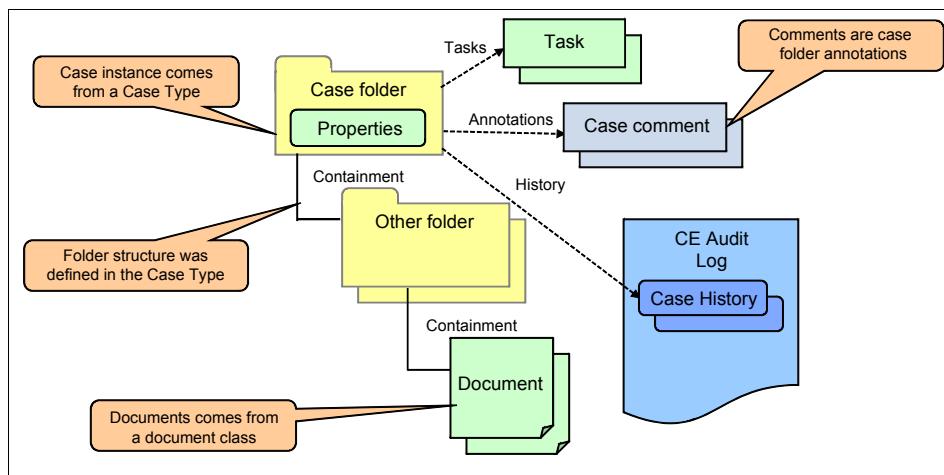


Figure 4-4 Case instance

A case contains a case folder and case history that is associated with it. The case history comes from the CE audit log and Process Engine event log (not shown in the figure). The case folder can have subfolders to structure the document content. The comments from case workers throughout the case lifecycle are also associated with the case.

Users can create cases by completing one of the following tasks:

- ▶ Creating a case folder of subclass CmAcmDeployedCaseType. Case Manager Client uses this method to create a case when users start the add case operation. For information about CmAcmDeployedCaseType and a solution root folder, see “Creating the solution root folder” on page 105.
- ▶ Creating a document of the initiating document type. This method works only for a case type that has an initiating document property defined.

In an IBM Case Manager environment, case workers can update the case properties in one of the several ways:

- ▶ Update the case properties from Case Manager Client.
- ▶ Update the case properties from the data that are associated with a task for a case. The task can also include data from a Process Engine workflow process or an external process such as IBM WebSphere Process Server.
- ▶ Update the case properties by using custom applications.

Case state

At any moment, a case has one of the following states, which the Content Engine event handler manages:

- ▶ NEW
A case is set to this state when Content Engine creates it.
- ▶ INITIALIZING
A case is set to this state while Content Engine sets the case properties.
- ▶ WORKING
A case is set to this state when there are incomplete tasks.
- ▶ COMPLETE
A case is set to this state when all its required tasks are completed.
- ▶ FAILED
This state is not currently used by the default IBM Case Manager. However, you can use this state in your custom applications to indicate a case that is in an exception or abnormal state.

Figure 4-5 shows the state diagram for a case.

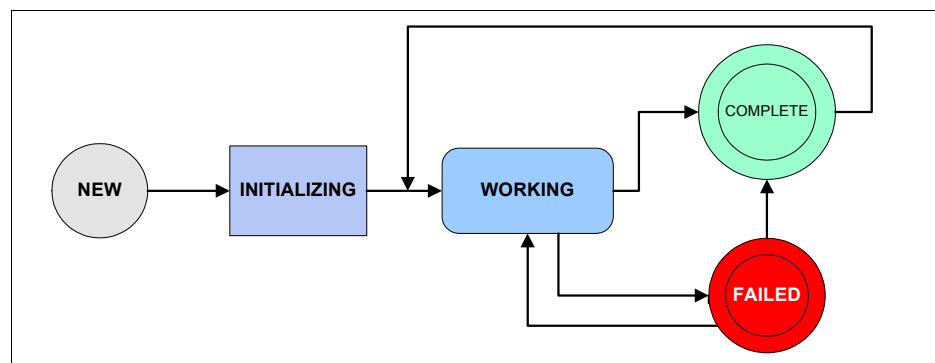


Figure 4-5 Case state transition

Remember: Capitalized letters and underscores are used for the Case and Task states in this book. The actual states that are represented in the Content Engine differ because they use spaces and are not capitalized. For example, the actual state is “Disabled by user” instead of “DISABLED_BY_USER”.

Content Engine Case Folder Update event handler enforces case state transition rules as follows (see Figure 4-5 on page 74):

- ▶ The user cannot reset the state to NEW.
- ▶ The user cannot reset the state to INITIALIZING.
- ▶ If a case is in the WORKING state, the case state can be set to either COMPLETED or FAILED.
- ▶ If a case is in the COMPLETED state, the case state can be set to WORKING.
- ▶ If a case is in the FAILED state, the case state can be set to WORKING or COMPLETE.

A case is set to COMPLETED state if all the started tasks and required tasks are completed. The required tasks here are the tasks that are marked as required at design time. After a task is started, the Case Client also displays the started task as a required task.

4.1.3 Roles

A role represents a specific business function. For example, in a Complaints Management system, a role might be a Reviewer or Investigator. You assign users or groups to roles. Roles can access only tasks that are assigned to those roles.

Roles are defined at the solution level, so roles are shared across case types. During the task design process, steps in a task are associated with roles. At run time, case administrators are responsible for assigning actual users to the roles.

Each role has at least one in-basket that is associated with it. The role determines the spaces and pages to which the users or groups of the role have access. When you define a case type, you can select different Case page layouts for different roles.

4.1.4 Roles in-basket

The *roles in-basket* is a bucket that contains the work items that the users or groups that belong to the role can access. The role in-basket is created automatically by IBM Case Manager Builder when you create a role. For more information about configuring the details of the role in-basket like the columns, sorting, and in-basket filters, see 6.4.2, “Setting up roles” on page 183. A role can access more than one role in-basket as shown in 12.3, “Creating more in-baskets” on page 486.

For each role and associated in-basket defined in a solution, Case Manager Builder defines a Process Engine role, associated in-basket, and a work queue in the Process Engine configuration file. You can also use Process Designer to create a role without creating a work queue. In addition, you can use Process Designer to delete the work queue that is created for a role that is defined in Case Manager Builder.

The properties that are specified for a role in-basket become the displayed data fields for the corresponding Process Engine work queue.

Table 4-1 specifies the mapping between roles in a solution and Process Engine configuration elements.

Table 4-1 Roles in solution to Process Engine element mapping

Roles in solution	Process Engine roles	Process Engine queues
Role name example: Customer Service Representatives	Same as role name in solution	<solution prefix>_<normalized role name> Example: CC_CustomerServiceRepresentatives

The normalized role name is the role name that uses only characters that are valid for a queue name in Process Engine.

4.1.5 Personal in-baskets

The *personal in-basket* displays the workflow step items that are assigned to a particular case worker.

A step item is assigned to the personal in-basket as a result of any of the following actions:

- ▶ Case workers move a step item to their own personal in-basket.
- ▶ Case workers reassign an item to a different case worker.
- ▶ An automated processing step assigns the item to a specific workgroup.

Each user typically has a personal in-basket. You can, however, configure a role without a personal in-basket. Therefore, having an in-basket for a user is optional.

IBM Case Manager implements personal in-basket as an in-basket of the Process Engine “Inbox” queue.

IBM Case Manager release 5.1.1 supports configuring how the personal in-basket is displayed at the level of each role. When you define a role, you can select between the options as shown in Figure 4-6.

The screenshot shows a configuration dialog for a role named 'Investigator'. The 'Role' field is filled with 'Investigator'. The 'Description' field contains 'Investigating Specialist'. Under 'In-baskets currently associated with this role:', 'My Work, Investigator' is listed. The 'Select the type of personal in-basket to display for this role:' section contains three radio button options: 'Personal (Common): Show the common view' (unchecked), 'Personal (Role): Show a custom view for this role' (checked), and 'Do not show common or role personal in-baskets' (unchecked). The 'Work assignment options to display for this role:' section contains two checked checkboxes: 'Role members can move work into their personal in-basket' and 'Role members can reassign work to others'. The 'Assignment in-basket:' section contains one unchecked checkbox: 'Show the in-basket that displays all assigned work'.

Figure 4-6 Configuring the personal in-basket for a role

► **Personal (Common)**

All roles that are configured for this setting share a personal in-basket definition. As a result, users see the same columns, sorting and filtering when they access their personal in-basket, independent which role they currently use. This is the behavior that earlier releases of IBM Case Manager used.

► **Personal (Role)**

This new setting enables you to have custom in-basket definitions for certain roles. You can use different columns, sort orders, and filters for each role to represent the work shown in the personal in-basket for that particular role.

► **Do not show personal in-basket**

This setting hides the personal in-basket from being shown for a role. It is helpful if the role is not intended to use a personal in-basket at all. When this option (as shown in Figure 4-6) is set, Case Manager Client hides the personal in-basket at run time.

Remember: Users cannot access personal work that is assigned to them if they are member of a single role that hides the personal in-basket. In this case, you must make sure that you do not directly assign work to users in this role.

At the level of the Process Engine, all solutions inside a Project Area share an Inbox queue for the personal in-basket. IBM Case Manager uses a default workflow data field named “SolutionIdentifier” to filter work items into personal in-baskets based on solution name.

4.1.6 Configuring manual work assignment

IBM Case Manager uses in-baskets to present users the work, which is either assigned to them personally or to the roles that they are a member of. IBM Case Manager release 5.1.1 introduces a number of new configuration options to provide better control of how users can assign work to themselves or to other case workers.

Moving work to the personal in-basket

When you configure a role, you can define whether role members are allowed to transfer work from the role in-basket to their personal in-basket as shown in Figure 4-7.

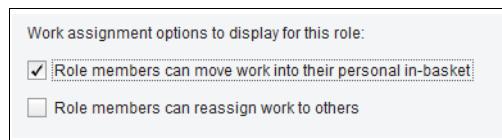


Figure 4-7 Configuring a role to allow moving work to the personal in-basket

When this option is enabled, case workers are allowed to transfer work from the role in-basket to their personal in-basket. They do so by selecting one or more items in the role in-basket and using the menu option **Move Item to personal In-basket**. Likewise, they can move work back from their personal in-basket to the role in-basket. They can do so by selecting the corresponding entries from their personal in-basket and using the menu option **Return Item**. You can use this configuration for roles that might have to spend a longer time to complete their work. You might also need to pend work and you want to make sure that the person that initially started to work on an item can continue this work later on.

You might want to disable this configuration for roles that perform heads down processing, or usually can completely process each work item that they open. If work will not be assigned to users in this role in any other way (either from users

in other roles or by automated rule-based assignments in the process itself), you can hide the personal in-basket completely for the role. For more information, see 4.1.5, “Personal in-baskets” on page 76.

At the technical level, this configuration is stored as a boolean attribute for the role called ECM_canMoveWorkToPersonalInBasket.

Assigning work to other users

Another configuration setting defines whether a role is allowed to assign work to other users as highlighted in Figure 4-8.

The screenshot shows a configuration dialog with the following text:
Work assignment options to display for this role:
 Role members can move work into their personal in-basket
 Role members can reassign work to others

Figure 4-8 Configuring a role to allow reassigning work to other users

When this option is enabled, users can select one or more entries from the role in-basket or their personal in-basket and assign them to another user. When reassigning work from your personal in-basket, you see an extra option in the dialog as shown in Figure 4-9.

The screenshot shows a configuration dialog with the following text:
 Have the currently assigned participant approve the work item before it progresses to the next step in the workflow

Figure 4-9 Optional approval check when reassigning work

Selecting the check box means that you want to approve the work item after the person you assigned it to has completed it. As a result, this work item will reappear in your in-basket in *the same workflow step* after the other person completed it. Otherwise, the work item will be placed automatically into the next workflow step.

At the technical level, this configuration is stored as a boolean attribute for the role called ECM_canAssignWork.

Important: IBM Case Manager allows a user to assign work to other users only if all of the following conditions are true:

1. The user is member of at least one role with ECM_canAssignWork = true.
2. The work item is not locked by another user.
3. The current workflow step has the reassign option enabled.

As a default, the reassign option is enabled for each process step you create in either IBM Case Manager Builder Step Editor or in the Process Designer.

However, you might encounter situations where you want to ensure that the same person performs two sequential workflow steps, or you want to prevent users from reassigning a workflow step for other reasons. In that case, you can disable the reassign option for the step. This setting can be modified in the Step Editor or by opening the **General** tab for the step in Process Designer.

Viewing all assigned work

IBM Case Manager release 5.1.1 introduces an in-basket that allows users to view all work items assigned to them throughout the complete solution. This in-basket is helpful in situations where a role member must quickly determine which work is assigned to a particular user. For example, a supervisor who gets notified that a team member is sick must find all work that is assigned to that person to decide which work items must be shifted to other case workers.

To display this in-basket for a role, enable the setting that is shown in Figure 4-10.

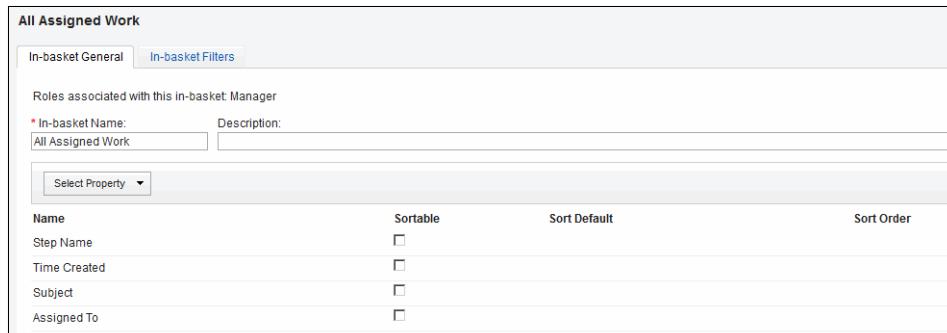


Assignment in-basket:

Show the in-basket that displays all assigned work

Figure 4-10 Configuring a role to show the in-basket for all assigned work

As a result, a new in-basket named **All assigned work** is available in IBM Case Manager Builder as illustrated in Figure 4-11.



All Assigned Work

In-basket General In-basket Filters

Roles associated with this in-basket Manager

* In-basket Name: Description:

Select Property ▾

Name	Sortable	Sort Default	Sort Order
Step Name	<input type="checkbox"/>		
Time Created	<input type="checkbox"/>		
Subject	<input type="checkbox"/>		
Assigned To	<input type="checkbox"/>		

Figure 4-11 Configuring the in-basket that shows all assigned work

The default columns that are configured for this in-basket are Step Name, Time Created, Subject, and Assigned To. You can change these columns and add filters in the same way as you would for a regular role in-basket. Generally, do not remove the Assigned To column, which shows the users Process Engine user ID alongside with their short name as configured in the LDAP directory.

Figure 4-12 shows how the “All Assigned work in-basket” is presented to the user in the IBM Case Manager Client. The sorting for the work item in this basket is by user ID and then by the column you select.

In-baskets			
My Work	Manager	All Assigned Work(4)	
Review Corresponding Letter	11/9/2012 4:24 PM	Send Corresponding Letter	50(p8admin)
Verify complaint	11/15/2012 2:46 AM	Verify Complaint	57(sue)
Review and Close Case	10/31/2012 11:32 AM	Close Case	73(jan)
Process Request	10/31/2012 12:50 PM	Request Assistance	74(isabelle)

Figure 4-12 In-basket showing all work assigned for a solution

For this use case, configure the supervisor role to show the “All assigned work” in-basket and to enable this role to reassign work to others. The supervisor role can then easily find all work assigned to a particular user, and can reassign some or all of it to other case workers.

You can configure the access to the “All assigned work” in-basket for several roles in a solution. At the technical level, this configuration is stored as a boolean attribute for the role called ECM_viewAssignedWorkInBasket.

Default settings

When you create a role in IBM Case Manager Builder, the configuration parameters default to the values listed in Table 4-2.

Table 4-2 Default parameters for new roles

Role custom attribute	Default value
ECM_canMoveWorkToPersonallnBasket	true
ECM_canAssignWork	true
ECM_viewAssignedWorkInBasket	false

As a result, a role member can move work from a role in-basket to their personal in-basket and to assign work to other users. You can change the role custom attributes either in IBM Case Manager Builder or by opening the Process Designer for a solution. In the Process Designer, open **View** → **Roles** and then click the **Custom Attributes** tab to display the window that is shown in Figure 4-13.

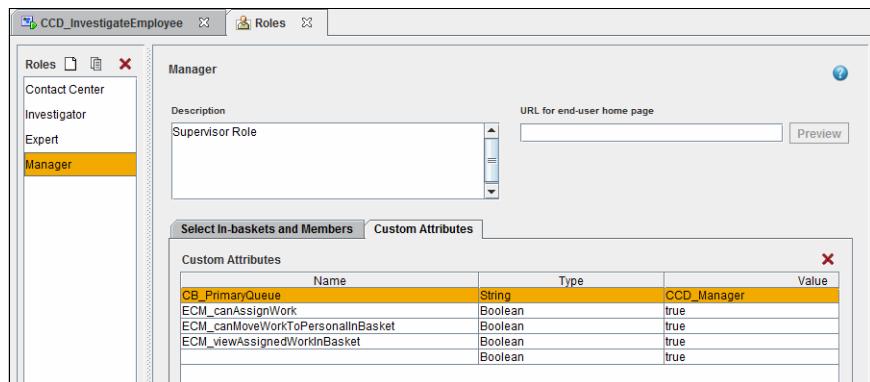


Figure 4-13 Modifying role custom attributes by using Process Designer

For more information about how to access the Process Designer from a solution in IBM Case Manager Builder, see 12.2.2, “Round-tripping to Process Designer” on page 476.

If you changed custom attributes for a role, you must redeploy your solution for the new configuration to be picked up.

Tip: For test purposes, you can directly modify the role attributes in the runtime environment by using the Process Configuration Console. However, any changes you apply here will be overwritten with the information in the solution package after you redeploy the solution.

Role-based reassignment window

IBM Case Manager release 5.1.1 introduces a new window that simplifies the selection of users when you reassign work. You can select the person whom you want to assign the work to based on their role. Additionally, it is still possible to select the users by searching for names in the LDAP directory.

Configure the work reassignment window in Case Manager Space Settings in IBM Business Space. You must log in to the IBM Case Manager Client as a user who has **Owner** permissions for the **Solution Space** in IBM Business Space. Click **Actions** → **Case Manager Space Settings** to display the configuration

window. The reassignment window is controlled by the setting that is highlighted in Figure 4-14.

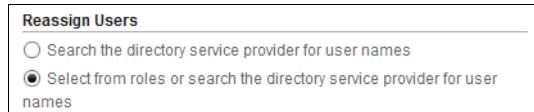


Figure 4-14 Configuring the role-based reassignment window

The default setting for this configuration is “Search for the directory service provider for user names”, which brings up the same dialog that was used in previous releases of IBM Case Manager. After you change the setting and save the configuration, users will see the new role-based assignment window shown in Figure 4-15.

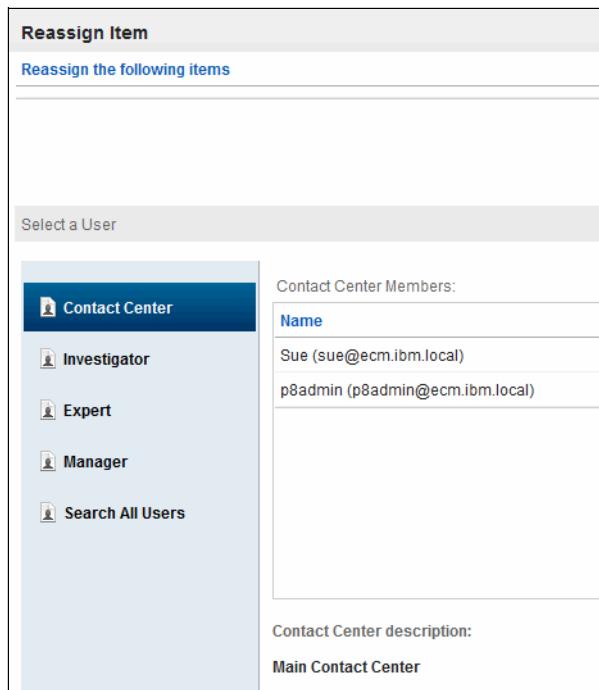


Figure 4-15 Role based user selection for reassigning work

When you select a role from the list, the users that are assigned to that role are populated in the right side of the window. You can then select the user whom you want. The number of roles that are displayed depends on the security settings, which are described in the next section.

Case workers can still search the directory services for the user name by selecting the option **Search All Users** at the bottom of the role list.

Work reassignment and security

In addition to the configuration settings, the security that is configured for Process Engine also impacts whether users can move or reassign work, and which information is displayed in the role-based assignment dialog.

By default, all users are not restricted for their use of rosters, event logs, and queues that are defined in a solution. Therefore, set the security on these objects as required to satisfy your security requirements. When you change security for Process Engine objects by using the Process Designer in the design phase (or the Process Configuration Console after the design phase), adhere to the rules listed in Table 4-3.

Table 4-3 Access right requirements for reassigning work

Action	Access right
Move work to personal in-basket from role in- basket	<ol style="list-style-type: none">1. Query and Process right on the current work queue2. Query right on Inbox queue3. ECM_moveToPersonalInbasket=true for role that uses in-basket
Reassign work from role in-basket	<ol style="list-style-type: none">1. Query and Process right on the current work queue2. Has ECM_canAssignWork=true
Return work from personal in-basket	Query on destination work queue
Reassign work from personal in-basket	Has ECM_canAssignWork=true
Return work from “All assigned work” in-basket	Query on destination work queue
Reassign work from “All assigned work” in-basket	<ol style="list-style-type: none">1. Query and Process on Inbox queue2. Has ECM_canAssignWork=true

The roles listed in the role-based reassignment window depend on the access rights the user has on the Application Space of the solution (Table 4-4).

Table 4-4 Access rights for role-based user selection window

Roles displayed	Access right
Only roles that the user is member of	No write access to Application Space
All roles that are defined for the solution	Write access to Application Space

Without more configuration, only the members of the **Process Engine Administrator Group** and **Process Engine Configuration Group** (configured when you install the Process Engine) have write access to the Application Space. These groups therefore can see all roles for the solution in the role-based work reassignment window. For example, add write access rights to the supervisor group to ensure that a supervisor can see all roles when reassigning work even if the supervisor is not member of these groups.

4.1.7 Document types

Document types help you to organize and classify the documents that belong to a case. You can provide more information about the documents by assigning properties to the document type. For example, a document type in the example might be a complaint letter.

The document type maps to a document class in the FileNet Content Engine. It maps to an item type in IBM Content Manager, if IBM Content Manager integration is being used.

Documents that are associated with cases are categorized into types. A document type can trigger a case or task creation. When you design the case types, you can assign a document type to be the initiating document type (also called starting document type) for a case type. When users create a document of the initiating document type for a case type, IBM Case Manager creates a case instance of that case type. A document type can be the initiating document type for one or more case types.

When you design task types, you can specify a document type as a filing precondition for a task creation. This precondition dictates when IBM Case Manager creates a task for a case instance. If a user files a document in a case instance folder, IBM Case Manager creates a task for the case instance. This process occurs if the task has the filing precondition and the document class is the same as the document type specified for the precondition. The task with filing precondition can be repeatable.

Tip: The display name for a document type is the document type name. The symbolic name for a document type has the format of *<solution prefix>_<normalized document type name>*.

4.1.8 Task type

A case type contains tasks. A task has one or more steps that must be completed. For example, a task might be to review a product-related complaint. A task is not completed until all the steps in the main work item are completed. A case is not completed until all required tasks are completed or manually disabled.

Consideration: Required tasks are all the tasks currently running (the started tasks) and the tasks that have the required attributes.

There are three types of starting tasks:

- ▶ Container task
- ▶ Task with reused workflow
- ▶ Case Manager normal task

Tasks can be started by these preconditions:

- ▶ Document filing
- ▶ Property update

Only tasks with the filing and property update preconditions can be repeated. The filing and property update preconditions can have property expression as extra constraints.

As an example, you can define a repeatable task with filing (of document class A) and property condition ($P1 > 100$ and $P2 = \text{true}$). For another example, you can define a repeatable task with property $P1$ update and property expression condition ($P2 = \text{true}$). In the first example, Case Manager promotes the task if a document of type A is filed in the case folder and ($P1 > 100$ and $P2 = \text{true}$). In the second example, Case Manager promotes the task if $P1$ is updated (regardless of the value) and ($P2 = \text{true}$).

Document filing and property update preconditions are always evaluated first. The property expression is evaluated after them.

Task can also be designated as optional or required. A required task must be completed before the case is set to complete. Even tasks that are marked as

optional must be completed for the case to complete after the task is in the working state.

Tasks can be included in sets. A set is either inclusive or exclusive. Placing a task in an inclusive set means that all tasks in that set must be completed for the case to complete. Placing a task in an exclusive set means that only one task in the set must be completed.

Hidden task

IBM Case Manager release 5.1.1 allows you to define hidden tasks. A hidden task works the same way as a task of the same type, but is not shown in the **Tasks** tab in IBM Case Manager Client. Hidden tasks are helpful for background processing tasks like initialization or cleaning up that are started based on certain conditions, but do not need to be visible to the case worker. Typically, you configure hidden task as being started automatically based on a condition or you would start them based on API calls. Even though hidden tasks are not shown in the **Tasks** tab, their status changes and any comments you programmatically add in a hidden task are displayed in the **History** view.

Container task and subtask

IBM Case Manager release 5.1.1 introduces new concepts that are called *container task* and *subtask*. A container task is similar to a task in that it can have preconditions. It can start automatically, manually or discretionary, and it can be defined as hidden or repeatable. Also, all required container tasks must be completed or manually disabled for a case to be closed. However, there are important features that distinguish a container task:

- ▶ It does not have a workflow that is associated to it
- ▶ It contains one or more subtasks

A container task contains one or more subtasks. Subtasks can be either container tasks themselves or tasks. Subtasks can be only automatic or manual tasks, not discretionary tasks. Subtasks can have preconditions, can be required, and can be hidden. When you organize subtasks in sets, all subtasks in a set must belong to the same container task. Task sets cannot cross container task boundaries. You can move existing top-level tasks into newly created container tasks.

Container tasks are useful to group a number of tasks together. You can also make them available to the case worker only when certain conditions of phases in the case management process are reached. This way, container tasks help to reduce the number of tasks that are presented in the IBM Case Manager Client. This is especially useful for complex case types. For example, a comprehensive task type can have large number of single tasks defined. However, depending on the progression of the case, many of these tasks never need to be started

because their corresponding preconditions are not met. Therefore, they are of no value for the case worker in that case. Without container tasks, all these tasks are visible in the “Task” tab of the Case Information widget. However, most of them are in the waiting state and might never get started. By using high-level container tasks, the worker sees only a few container tasks waiting as opposed to all the subtasks that are part of the container tasks.

Container tasks also reduce the resources that are spent for a case. Each task instance corresponds to Task Object in the Content Engine even if the task is in waiting state. Another advantage of the container task is the performance for a case creation. If you have a case with 200 independent tasks, Case Manager creates 200 task objects. However, if the 200 tasks are grouped into five containers and 35 stand-alone tasks, then Case Manager creates only 40 task objects.

Discretionary container tasks are powerful tools that allow case workers to bring in a predefined set of tasks into a case. This is useful when you cannot describe all preconditions that automatically or manually start the tasks. It provides support for highly dynamic and adaptive case management by enabling case workers to judge and decide which tasks must be used on a case-by-case basis.

Because container tasks do not have a workflow associated with them, you cannot convert an existing task into a container task. In that case, create a container task instead and move your existing task into that container task.

A container task is completed only when the following conditions are true:

- ▶ All required subtasks are completed or manually disabled (for manual tasks only). All required tasks that are in the FAILED state must also be terminated.
- ▶ Any working subtasks are complete or stopped/aborted.

A case is completed when the following situations are true:

- ▶ All required tasks (containers included) are completed or aborted.
- ▶ All running tasks are completed or aborted.

An optional container task in WAITING or READY state is not evaluated for the case completion computation, even if the container task includes a required subtask. This is because the subtasks are not started until the container task is in the RUNNING state. If a container task contains a required subtask that is intended to impact the case completion computation even if the container task is not started, configure the container task as a required container task.

Task Classes

A *Task type* is a definition of a task class. IBM Case Manager release 5.1.1 uses one single task type, which is *CmAcmCaseTask*. Previous IBM Case Manager

releases used a separate task type *CmAcmCaseTaskWithInitiatingDocument* for tasks that are associated with a document filing precondition. This class is deprecated in release 5.1.1. When you redeploy a solution, the existing tasks that use the deprecated class are automatically converted to use the *CmAcmCaseTask* class. The document that started the task (the initiating document) is stored in the *CmAcmTriggerDocument* property with the task.

For more information about the *CmAcmCaseTask* class, see the IBM Case Manager Information Center. Click **Developing case management applications** → **Content Engine add-on extensions for Case Manager Builder** → **IBM Case Manager target object store extensions** → **Custom subclasses of the CmTask class** → **Case Task class**.

Case Task class properties

The following are the Case Task class (*CmAcmCaseTask*) properties:

- ▶ Task Name

If a task is a discretionary task, case workers specify the task name when they create the task. Otherwise, the task has the name of the task class, which is the name that you provided for the task in IBM Case Manager Builder.

- ▶ Disabled State

The value of this property indicates whether a task is disabled. If it is disabled, the value also indicates how it was disabled. This integer property has a choice list associated with it as listed below. The actual integer value is shown in brackets.

- ENABLED(0): The task is not disabled.
- DISABLED_BY_USER(1): A task has been disabled by a case worker.
- DISABLED_BY_EXCLUSIVE(2): A task is disabled because it belongs to an exclusive group and one of the tasks in the group has been promoted to the WORKING state.
- DISABLED_ABORTED(3): A task is disabled because its associated workflow has been aborted.

- ▶ Required State

The value of this property indicates whether the task completion is required to complete a case. A case is completed only when all the required tasks are completed. This integer property has a choice list that has these values:

- OPTIONAL(0): The task is not required for a case. This is the default value.

- REQUIRED_BY_USER(1): This value indicates that the task is marked as required when you define a case type for a solution.
- REQUIRED_BY_INCLUSIVE(2): This value indicates that the task belongs to an inclusive group and one of the members of the group is in the WORKING state.

► **Launch Mode**

The value of this property specifies the state of a task when IBM Case Manager creates the task, and whether the task is repeatable. This integer property has a choice list with the following values:

- SYSTEM_MANUAL(0): This value indicates that the task is defined as a manual task. When IBM Case Manager creates a task by using the Content Engine event handler, IBM Case Manager promotes the task from WAITING to READY. This process occurs unless the task has a precondition that prevents it from being promoted. The task remains in the READY state until a case worker starts the task, which then promotes it to the WORKING state. For more information, see “Case object model implementation” on page 97.
- SYSTEM_AUTOMATIC(1): This value indicates that this task is an auto-launch task. When IBM Case Manager creates a task by using Content Engine event handler, if the task has a precondition that is met, Case Manager promotes the task from WAITING to WORKING. Case Manager then starts its associated workflow if the task has no precondition that prevents it from being promoted. For more information, see 4.2, “Case object model implementation” on page 97.
- USER_AUTOMATIC(2): The property indicates that the task is marked as a discretionary task. IBM Case Manager creates a task only when a case worker requests one. After it is created, IBM Case Manager promotes the task from WAITING to WORKING state and starts its associated workflow. For more information, see “Case object model implementation” on page 97.
- SYSTEM_MANUAL_REPEATABLE(3): This value specifies the task is manual and repeatable. This value has the same attribute as “System Manual”.
- SYSTEM_AUTOMATIC_REPEATABLE(4): This value specifies that the task is auto-launch and repeatable. This value has the same attribute as “System Automatic”.

Remember: For a repeatable task, multiple instances of the task can be created for any case. For a non-repeatable task, only one instance of the task can be created. Only tasks with filing or property update preconditions can be marked as repeatable.

► Group Mode

This property specifies whether a task is stand-alone, inclusive, or exclusive. This property is a choice list property with the following values:

- Not Grouped (0): The task does not belong to any group. This value is the default value.
- Exclusive (1): The task belongs to an exclusive group. When IBM Case Manager promotes a member of an exclusive group from READY to the WORKING state, the rest of the group is set to DISABLED.
- Inclusive (2): The task belongs to an inclusive group. When IBM Case Manager promotes a member of an inclusive group from READY to the WORKING state, the rest of the group is promoted to WORKING.

Restriction: A repeatable task cannot belong to a group. A non-repeatable task can belong to no more than one group.

► Is Container

This property specifies whether a task is a container task, and can have the values TRUE or FALSE.

► Parent Task

This object valued property contains the parent container task for Subclass.

► Filing Document

This object valued property contains the document, that if filed, triggers the start of the task, if the task has the document filing precondition.

► Process Instance ID

This string valued property contains the ID of the process started on the Process Engine for the task.

► Last Failure Reason

This string property provides information why the task was set to the FAILED state. It is automatically populated when Process Engine moves a task into this state. For example, CmAcmError Launch Failed indicated that IBM Case Manager was not able to launch the corresponding task workflow.

Both Content Engine and Process Engine can update the task state and this field.

► Last Restart Date

This date and time property is updated by IBM Case Manager when a task is restarted.

► Restart Count

This integer property is used by IBM Case Manager to track the number of restart attempts for a task. You can use this information, for example, to abandon restarting a task after it reaches a number of unsuccessful restarts.

4.1.9 Tasks

Tasks are the instances of task types that are created by IBM Case Manager when a case is created.

Task states

A task can be in one of the following states:

- ▶ WAITING

A task is in this state when IBM Case Manager first creates it. A task can enter this state when it is disabled (demoted) from the READY state.

- ▶ READY

A task is in this state when the precondition for the task is met and the task has the manual launch mode.

- ▶ WORKING

IBM Case Manager starts a workflow that is associated with a task when it is in this state. A task can enter this state when one of the following occurs:

- A case worker creates a discretionary task.
- A case worker starts a manual task currently in the READY state.
- The task was in COMPLETED state and a **Restart** was issued successfully through the API.
- The exception for the task in the FAILED state is rectified. The Disabled State property for the task is not set to DISABLED_ABORTED, and a **Restart** was issued successfully through the API.
- The task was in FAILED state with Disabled State equal to DISABLED_ABORTED, and a **Restart** was issued successfully through the API.

Case workers cannot disable a task in this state.

- ▶ COMPLETE

The task is completed. Any workflow that is associated with the task is also completed. Case workers cannot disable a task in this state.

- ▶ FAILED

A task can enter this state under the following conditions

- The associated workflow is in the FAILED or MALFUNCTION state. If the failure condition is rectified, the Process Engine sets the state back to WORKING and the work item can continue to be processed.
- All the work items for a workflow that is associated with the task are deleted abnormally. Also, in this case the state of the task can be set back to WORKING by issuing a **Stop** request to gracefully clean up the existing work objects. Then, issue a **Restart** request.
- The task was in WORKING state and a **Stop** request was issued successfully through the API. This action sets the Disabled State property to DISABLED_ABORTED.

Case workers cannot manually disable a task in FAILED state. However, you can provide the capability to stop and restart the task by issuing the corresponding requests through the Case Manager API.

Stopping and Restarting Tasks

With IBM Case Manager 5.1.1, you can issue **Stop** and **Restart** requests for tasks at API level.

You can use a Stop request to gracefully end a task in WORKING state. As a result, any existing workflows for this task are ended. Stopping tasks is useful in these situations, among others:

- ▶ A task no longer must be completed
- ▶ A task process has failed and cannot be recovered
- ▶ A case must be ended, for example when a customer decides not to pursue a complaint anymore

Additionally, a Stop request cleans up a task in the FAILED state.

In both cases, the task ends in the FAILED state with the Disabled State property set to DISABLED_ABORTED. If wanted, you can restart the task by issuing a Restart request.

A Restart request can be used to demote a successfully finished task from COMPLETED back to WORKING state. As a result, a new instance of the corresponding workflow is started. You can also issue a Restart request to demote a task in FAILED state back to WORKING state if one of the following conditions is met:

- ▶ The tasks Disabled State property is equal to DISABLED_ABORTED
- ▶ The tasks Last Failure Reason property has the value “CmAcmError Launch Failed”

Figure 4-16 shows the state diagram for a task.

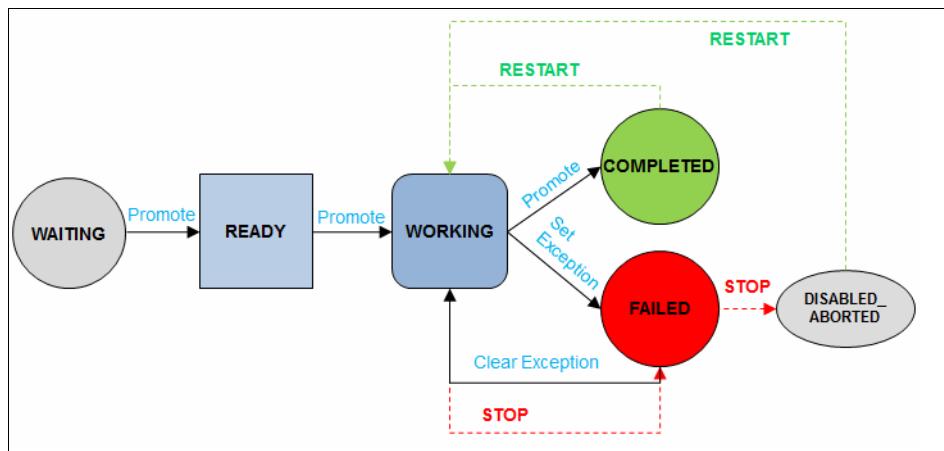


Figure 4-16 Task state transition

IBM Case Manager only supports Stop and Restart invocation against a Process Engine-based regular task. It is not supported to stop or restart IBM BPM based task and container tasks. Additionally, discretionary tasks can only be stopped.

A stop request deletes all of the active work items for a single workflow process instance. This process updates the corresponding task's state to "FAILED" and the Disabled State property to DISABLED_ABORTED. Any further workflows that are launched by this main workflow are not ended by the request itself. Furthermore, ending a workflow does not compensate any changes that are applied to external systems like values changed by using *DBExecute* or Web Services calls.

Restarting a task always starts the latest workflow that was deployed for that task. If the solution has been redeployed since the initial start of the task, the workflow that is launched at restart can differ from the initial one. The restart does not affect the *Date Started* property of the task, which remains the time of the original first start. However, it does change the properties *Task State*, *Process Instance ID*, *Disabled State*, *Last Date Restarted*, *Restart Count*, and *Last Failure Reason*.

Case Analytics stops capturing task state data after a task reaches the COMPLETE or DISABLED_ABORTED state. When the task is restarted, Case Analytics does not restart to capture task state data.

The user who initiates a Stop request must have query rights on the roster, and query and process rights on all queues of the solution. For a Restart request to succeed, the requesting user must have create rights for the solution roster.

Existing and running task instances of earlier IBM Case Manager versions can be stopped, but they cannot be restarted after the system is upgraded.

Task preconditions

There are four possible preconditions to start a task:

- ▶ Filing

To create a task, case workers must file into a case instance a document of a document class that is specified in the filing precondition for the task. IBM Case Manager sets the state of a task based on its launch mode. You can select which document types are evaluated for a filing precondition. Multi-selection of document types is supported. A filing precondition can be combined with property expression preconditions.

- ▶ Property update

Creates a task if one or more case properties are changed. If multiple properties are selected for a property update precondition, the properties are logically combined in an OR condition. An example that is Complaint Category OR Complaint Description is changed. More case property expressions can be added to the property update precondition with an AND operator. The case property update precondition can be combined by AND or OR. For example, (Complaint Category OR Complaint Description is changed) AND (Complaint Status = Processing OR Complaint Status = Pending).

- ▶ Property expression

When an expression that contains case properties satisfies the condition that is specified in the property precondition, IBM Case Manager sets the state for a task. This state is based on its launch mode.

- ▶ No precondition

IBM Case Manager promotes the state of a task to WORKING immediately after creation.

As mentioned earlier, a task precondition can be a filing or property update precondition, with added property expression for extra constraint.

Task precondition evaluation

When creating a case, IBM Case Manager takes the following actions:

- ▶ Top-level non-discretionary tasks are created and put into their initial state, which is WAITING.
- ▶ Tasks with no precondition or with property preconditions that are met are promoted to READY state.

- If a task is configured as automatic, it is moved to WORKING and the associated workflow launches.
- Tasks with a property update or document filing precondition wait until the property condition is met or the document is filed.

Consideration: When you create a case and a value is entered for a property, this action does not count as a property update precondition.

For repeatable tasks with a property update precondition when the update event triggers, the systems checks whether a task of that type is in the WAITING state. If not, a new task is created.

When the container task is moved into the WORKING state, the following processing is started:

- The corresponding top-level subtasks are created and put in WAITING state
- Subtasks with no precondition or property preconditions that are met are promoted to READY state
 - Subtasks that are configured as automatic are promoted to WORKING state and the corresponding workflow starts
- Subtasks with property update or document filing conditions wait for the appropriate event to happen before they are evaluated

With IBM Case Manager release 5.1.1, you can change the preconditions for tasks in a deployed solution by redeploying the solution. This process can change the precondition itself and the precondition type. For example, you might decide that a verification task be started based on a property update instead of a property condition. You can also change a property-based precondition to a document filing precondition, or vice versa.

Important: Generally, do not change the task from Automatic or Manual to Discretionary for production systems because starting a discretionary task is different from a system triggered start of a task. Discretionary tasks use a different user interface to start, and often require more information to be passed when they are started. This process can cause incompatibilities with existing task instances.

If you change the precondition for task and redeploy the solution on a production system, run the Precondition Checker utility to update running cases. The Precondition Checker checks for any existing tasks that are currently in a Waiting state. If the tasks satisfy the changed precondition, they are promoted to a

READY state. It is mandatory to run the Precondition Checker in the following situations:

- ▶ A property precondition has been changed
- ▶ A property update or document filing precondition has been changed to a property precondition
- ▶ An existing precondition of any kind has been changed to “No precondition”

You do not need to run the Precondition Checker if a property update precondition or document filing precondition is used or changed. These precondition types are automatically evaluated the next time that a document is filed or the property is updated.

Important: For documents that are already in a case, a new or changed document filing precondition is not evaluated.

4.2 Case object model implementation

This section addresses how IBM Case Manager services organize and manage the objects that are addressed in 4.1, “IBM Case Manager object model” on page 67 as data.

IBM Case Manager organizes its objects into these main categories:

- ▶ Design object store (solution packages or artifacts).
- ▶ Target environment (solution instances or deployed solutions). The target environment consists of one target object store and its associated Process Engine region.
- ▶ Spaces and pages

4.2.1 Design object store

IBM Case Manager uses a design object store to organize and manage the solution packages and artifacts. A design object store is a Content Engine object store with IBM Case Manager add-ons. There are two kinds of add-ons: Design object store add-ons and target object store add-ons.

In a development environment, the object store contains the solution package (such as SDF, Process Engine configuration, case type XPDL) is called the design object store.

IBM Case Manager administration client installs IBM Case Manager design object store add-ons while it configures an IBM Case Manager environment. IBM Case Manager administration client also creates a folder structure for IBM Case Manager to organize IBM Case Manager solution packages and artifacts under.

Figure 4-17 displays the root folder structure for IBM Case Manager in a design object store.

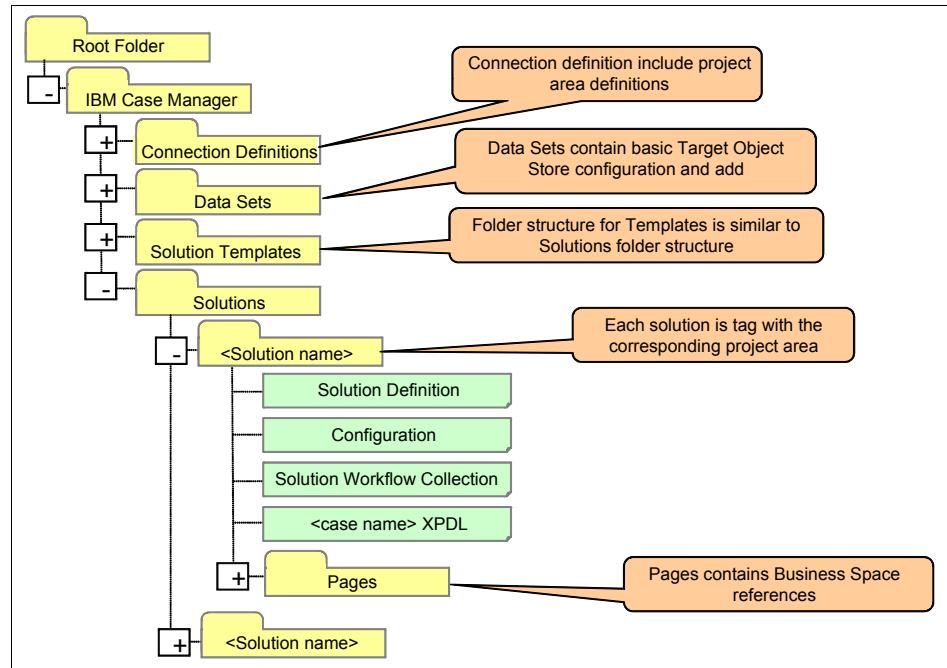


Figure 4-17 Design object store folder structure

A design object store contains following artifacts:

- ▶ Solution definition

A solution definition is a collection of solution definition files (SDF), Process Engine configuration file, Process Engine workflow definition files (XPDL), and pages. A solution package is stored under the Solutions folder in a design object store. For definition of SDF, Process Engine configuration file, XPDL, and pages, see “Solution package” on page 70.

- ▶ Solution template

A solution template is a collection of IBM Case Manager assets that can be customized and extended to build a complete solution. A solution template is stored in the Solution Templates folder.

- ▶ Connection definition

A connection definition specifies the target environment to which an IBM Case Manager solution is deployed. It contains the information about Process Engine connection point, and a logical to physical page mapping. Using Process Engine connection point, IBM Case Manager can discover the target object store and Process Engine region number.

Remember: A solution definition refers to the solution as defined in the Design Object Store. A solution package refers to the archive file of the solution that is created when you export the solution.

4.2.2 Target environment

The target environment consists of one target object store and its associated Process Engine region. A target object store is a Content Engine object store with IBM Case Manager add-ons required for a target object store. IBM Case Manager administration client installs IBM Case Manager target object store add-ons as part of configuring an IBM Case Manager environment. IBM Case Manager administration client also creates a folder structure for IBM Case Manager to organize deployed solutions.

Figure 4-18 displays the root folder structure for IBM Case Manager in a target object store.

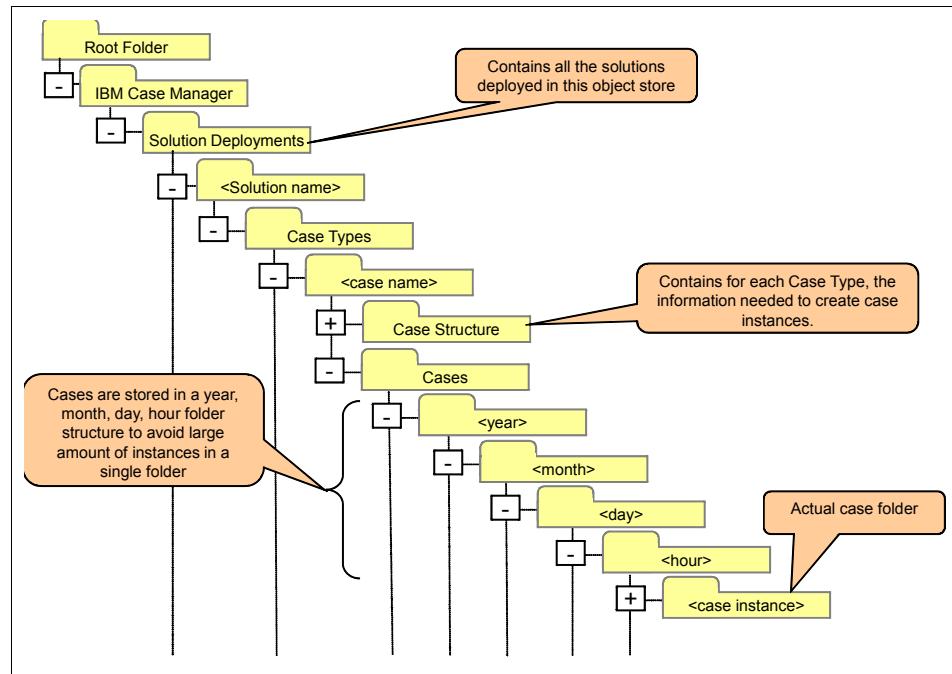


Figure 4-18 Target object store structure

A target object store contains the following artifacts for a deployed solution:

- ▶ Deployed solution folder

A deployed solution folder is a folder structure for a deployed solution in the target object store. Case instances are created under this structure.

- ▶ Case class

A case class is a representation for a case type in a solution.

- ▶ Document class

A document class is a definition of a document type.

- ▶ Property template

A property template is a definition of a property type.

- ▶ Task class

A task class is a definition of a task type.

- ▶ Subscription

A subscription relates a case or task event condition to corresponding Content Engine event actions.

Each target object store is associated with a Process Engine region. A Process Engine region stores process data for the solutions. When users deploy the solutions, IBM Case Manager API transfers the Process Engine configuration information and workflow collections for the solution to the Process Engine region. Process Engine metadata includes application space, roster, event logs, queues, roles, in-baskets, and workflow classes.

Figure 4-19 shows the Process Engine region of a target object store.

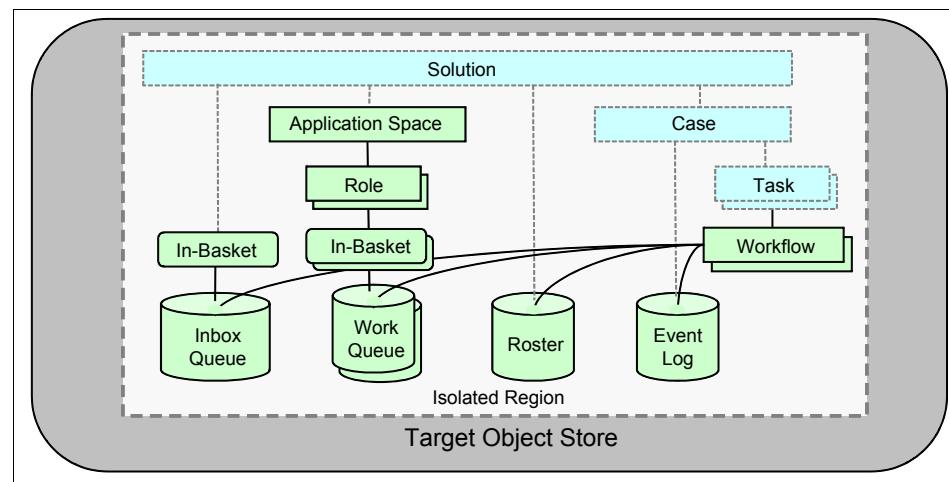


Figure 4-19 Target object store Process Engine region

4.2.3 Project areas for supporting parallel project development

Project areas are used when the development environment for IBM Case Manager must be shared between different solution development teams. Project areas allow solution teams to develop their solutions iteratively without affecting other development teams that share the environment. In particular, developers can isolate the effects of resetting the test environment to a single project area.

Project areas are used only in development environments. A single design object store can hold multiple project areas, and each one maps to one development environment. A solution must be unique at the design object store level. That is, two project areas cannot have the same solution.

Project areas are used to determine the access to the solutions. Each has its own target object store and Process Engine region. Each project area can have multiple solutions and be shared by multiple developers. However, each developer uses only one project area.

Figure 4-20 illustrates how project areas in the design object store can be mapped to the target object stores.

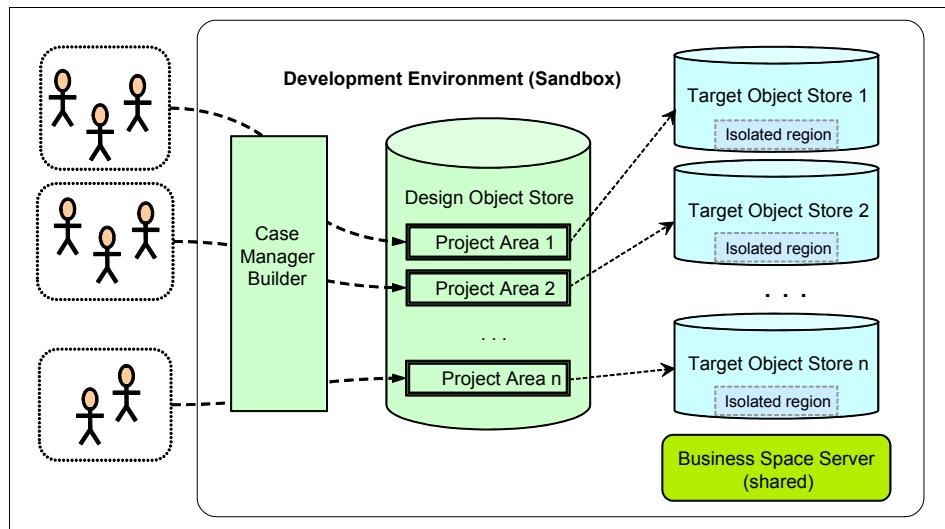


Figure 4-20 Project areas

Project areas are created and managed by IT administrators. Typically, an IT administrator runs a task to create and configure a default project area. Other project areas can be set up by using IBM Case Manager administration client.

IBM Case Manager administration client handles the privileges that are required for a user to access the design object store of the project area. However, it does not add the privileges for the user in the target object store and Process Engine region. To avoid reconfiguration later, plan the necessary user access to a project area before you create the target object store and Process Engine.

Keep in mind these restrictions when you are planning project areas:

- ▶ A test environment reset requires that the target object store users have system administration privileges. This action clears out the data in the target test environment to provide a clean environment for further development.
- ▶ Users who work in project areas must be assigned to the Process Engine SysConfigG or SysAdminG privilege groups.

Remember: For IBM Content Manager integration, each target object store in FileNet P8 maps to an IBM Content Manager Server (Library Service). Therefore, you need multiple IBM Content Manager servers for multiple project areas.

For more information about setting up Project Areas by using IBM Case Manager administration client, see the IBM Case Manager Information Center. Click **Designing and testing your case management solution** → **Preparing the design environment** → **Setting up a development environment with project areas**.

4.2.4 Solution deployment process

The IBM Case Manager API handles solution deployment requests from Case Manager Builder or custom applications by using the IBM Case Manager REST API. This section describes the operations the IBM Case Manager API runs during the deployment process.

Figure 4-21 shows a summary of the deployment flow for a solution.

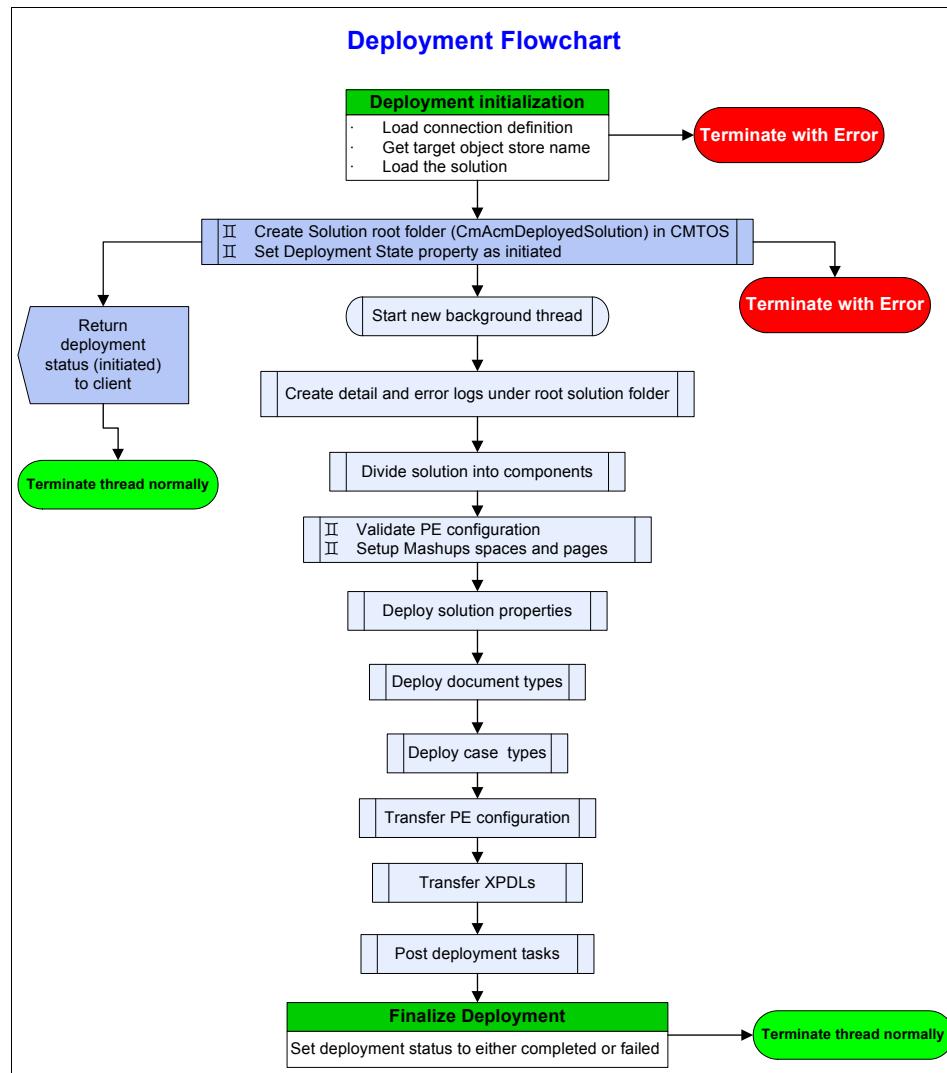


Figure 4-21 Solution deployment flowchart

In general, the IBM Case Manager API runs all the operations that are required for deploying a solution. IBM Case Manager ends the process only for these kinds of unrecoverable errors:

- ▶ Failure to load a connection definition
- ▶ Failure to load a solution definition file (SDF) for a solution.

- ▶ Failure to retrieve target object store name that is associated with a Process Engine region specified in the Process Engine connection point. The Process Engine connection point is in a connection definition.
- ▶ Failure to create a solution root folder in a target object store.

For all other errors, the IBM Case Manager API continues with the deployment process after it logs the errors in the solution error deployment log. If there are no errors in the deployment, the error deployment log is empty. The IBM Case Manager API also records all operations and messages to make troubleshooting solution deployment issues easier.

Most of the operations in Figure 4-21 on page 104 are self-explanatory. However, some need more explanation. For more information about deploying the different assets of a case management solution, see Chapter 8, “Solution deployment” on page 259.

Creating the solution root folder

When a solution is deployed, the following Content Engine and Process Engine metadata are created:

- ▶ Property templates
- ▶ Document classes
- ▶ Process queues
- ▶ Workflow definitions

The IBM Case Manager API also creates a deployed solution folder structure in the target object store as shown in Figure 4-18 on page 100.

A solution root folder is of type *CmAcmDeployedSolution*, a subclass of the Folder class. The CmAcmDeployedSolution class is part of the set of target object store add-ons. IBM Case Manager administration client installs the add-ons when users configure the environment for IBM Case Manager.

Under the solution root folder, IBM Case Manager API creates other folders:

- ▶ Case type name

The name of the folder is *<solution prefix>_<normalized version of case type name>*. The normalized version of the case type name contains only characters that are valid as a folder name in Content Engine. The folder class is *CmAcmDeployedCaseType*, a subclass of Folder.

For more information, see the IBM Case Manager Information Center. Click **Developing case management applications** → **Content Engine add-on extensions for Case Manager Builder** → **IBM Case Manager target object store extensions** → **Custom subclasses of the Folder class**.

Tip: If a user creates a folder of <solution prefix>_<normalized version of case type name> type, IBM Case Manager creates a case of type <case type name>.

- ▶ CaseStructure

When the business analysts design a case type, they can design a folder structure for that case type. When IBM Case Manager creates an instance, it creates the same folder structure for that case instance under the CaseStructure folder of a deployed solution folder.

- ▶ Cases

The Cases folder contains the case instances. IBM Case Manager organizes the case instances under *YEAR / MONTH / DAY / HOUR* subfolders.

Setting up business spaces and pages

IBM Case Manager creates the Business Mashups spaces and pages during solution deployment. For more information, see 10.3, “Manipulating pages in business space” on page 352.

Deploying case types

Each type of case has its set of tasks, views, and case folder structure. For each case type, IBM Case Manager creates a class that is derived from Case Folder class (CmAcmCaseFolder) that is used for the implementation of the case folder. The symbolic name of a case class has the naming convention of <solution prefix>_<normalized case type name>.

The case class contains all case properties that are defined for a case type. A case class inherits the Initiating Document (CmAcmlInitiatingDocument) property. An initiating document (also called a starting document) in Case Manager Builder specifies a document type that triggers the creation of a case class.

As part of deploying a case type, IBM Case Manager must deploy all the task types belong to a case type. For more information, see “Deploying task types” on page 107.

Deploying task types

For each task type in a case type, IBM Case Manager creates the task class and its precondition. IBM Case Manager runs the following validation and configuration steps as part of the process:

- ▶ Validate the task's preconditions

IBM Case Manager makes sure the preconditions such as the following specified for a task are valid:

- The property expression is a valid expression.
- The document type that is used in the filing precondition exists.
- The property that is used in the property update precondition is valid.
- The task, if marked repeatable, can indeed be repeated.

- ▶ Validate the task groups

In this operation, IBM Case Manager ensures the case type definition that is defined in the SDF follows the restriction for a group task:

- A repeatable or discretionary task cannot belong to a group
- A task belongs to only one group.
- ▶ Validate the discretionary tasks.
- ▶ Ensure that there is a workflow for each task: If a task does not have an associated workflow, IBM Case Manager logs a warning message in the solution error log. However, IBM Case Manager does not consider this error as unrecoverable.
- ▶ Create a task class.
- ▶ Configure the task properties.
- ▶ Configure workflow subscriptions. The subscription indicates which workflow IBM Case Manager using Content Engine event handler starts when a task is in the WORKING state. For more information, see “Case object model implementation” on page 97.

4.3 IBM Case Manager spaces and pages

A space is a collection of mashup pages. An IBM Case Manager application consists of a set of spaces and pages as shown in Figure 4-22.

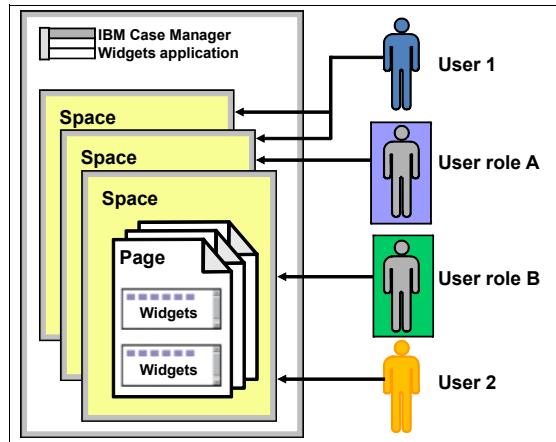


Figure 4-22 Spaces and pages

Case Manager Client stores spaces and pages information in the business space database. Case Manager Client supplies the following default spaces:

- ▶ Solution space: The Solution space contains *Work page* and *Case page* layouts.
- ▶ Case page space: The Case page space contains pages for case layouts such as *Add Case* and *Case Details*.
- ▶ Step page space: The Step page space contains pages for step layouts such as *Add Tasks* and *Work details*.

To customize applications, you can modify the default spaces, create more solution spaces for different user roles, and create new page layouts for steps and cases. For more information about spaces, see Chapter 10, “User interface” on page 335.

4.3.1 Default pages in Solution space

Default pages in the solution space include a Work page and a Case page.

Work page

The Work page contains in-basket widgets that display the work items in the in-basket of a role and a personal in-basket. By default, the Work page contains the **Add Cases** and **Manage Roles** buttons. **Add Cases** allows case workers to create new cases. **Manage Roles** allows the case administrators to assign role membership.

For more information, see 4.1.4, “Roles in-basket” on page 75

Case page

The Case page provides case search capability. The page also displays search results in a Case List widget.

4.3.2 Default pages in Case page space

In the Case page space, the default pages include Add Case, Add Case Form, Case Detail, Case Detail Form, and Split Case.

Add Case page

The Add Case page is a default Case page for manually creating a case instance.

Add Case Form page

The Add Case Form page is similar to the Add Case page. However, it uses a Form widget instead of a Case Data widget for capturing case data when it creates a case. Using the Add Case Form page as a form can provide extra capabilities such as dynamic input regions, lookups, or verifications. These capabilities can be used to guide the case worker through the case creation process.

Case Details page

The Case Details page is a default Case page for displaying case data. This page contains widgets for various case views such as the Case Data view, the Case History view, and the Case Task view.

Case Details Form page

The Case Details Form page is similar to the Case Details page, but it uses the Form widget instead of the Case Data widget. This page contains widgets for

various case views such as the Case Data view, the Case History view, and the Case Task view.

Split Case page

The Split Case page is used to create a case that is based on an existing one. It shows the data and documents for two cases next to each other in preparation of the split.

Remember: New form and split pages are not automatically available when you upgrade solutions that were created from IBM Case Manager Version 5.0 to Version 5.1.1. You must manually add them to your solutions. For more information, see Chapter 10, “User interface” on page 335.

4.3.3 Default pages in Task page space

The Task page spaces contain the following default pages: Add Task page, Add Task Form page, Work Details page, Work Details Form page, and Form Attachment Work Details page.

Add Task page

Case Manager Client displays this page when a case worker adds a discretionary task for a case.

Add Task Form page

The Add Task Form page is similar to the Add Task page, but it uses the Form widget instead of the Case Data widget.

Work Details page

Case Manager Client displays this page when a case worker opens work items from an in-basket.

Work Details Form page

The Work Details Form page is similar to the Work Details page, but it uses the Form widget instead of the Case Data widget.

Form Attachment Work Details page

This page allows the user to enter data to a form, which is then saved as a form document attached to the work item.

4.3.4 IBM Case Manager widgets

A page contains a set of widgets. For a detailed list of standard widgets that are available in IBM Case Manager, see 10.1.1, “Standard widgets provided by IBM Case Manager” on page 336.

For more information about the standard widget, see the IBM Case Manager Information Center. Click **Designing case management applications** → **Modifying the default case management client application** → **Widget Reference**.

For more information about enhancing your IBM Case Manager solution by developing and adding your own widgets, see 10.2, “Creating and deploying a custom iWidget” on page 346.

4.4 Tasks and associated workflow processes

In IBM Case Manager, each non-container task or a task that does not have subtasks has a workflow that is associated with it. A workflow in IBM Case Manager environment derives from the *CaseWorkObject* class that was introduced in FileNet Process Engine, Version 5.0.

When a non-container task is promoted to WORKING state, IBM Case Manager starts a workflow that is associated with it. A task enters the Working state when one of the following conditions is satisfied:

- ▶ A user creates a discretionary task
- ▶ A user starts a manual task currently in the READY mode
- ▶ A precondition is satisfied for an auto-launch task

For more information, see 4.2, “Case object model implementation” on page 97.

This section addresses the Content Engine and Process Engine enhancements that are used by IBM Case Manager.

4.4.1 Workflow association

Task types are associated with workflow definitions. There are different ways how workflow definitions can be defined connected to a task:

- ▶ New FileNet Process Engine workflows

Workflow definitions can be created by using the Step Editor in IBM Case Manager Builder. They can be enhanced by using the Process Designer. In addition, you can edit workflow definitions directly from the task page.

- ▶ Reuse FileNet Process Engine workflow

Existing FileNet BPM workflow definitions that were previously created by using Process Designer can be used when you create a task in IBM Case Manager Builder.
- ▶ Created by WebSphere Integration Designer

Existing workflow definitions that were defined in WebSphere Integration Designer that run in WebSphere Process Server can be used when you create a task in IBM Case Manager Builder.
- ▶ Created by WebSphere BPM Process Designer

Workflow definitions that were created by using WebSphere BPM Process Designer and available to run in WebSphere Process Server can be used when you create a task in IBM Case Manager Builder.

For more information about using the workflow in WebSphere Process Server, see Chapter 17, “Integration with IBM Business Process Manager” on page 587.

IBM Case Manager uses the *CmAcmTaskWorkflowLaunchEventAction* action type to wire a task type to a workflow subscription. IBM Case Manager starts the workflow that is associated with a task when such an event action is triggered.

4.4.2 Workflow data fields

To provide a tighter integration between Content Engine and Process Engine, case properties can be accessed directly from a workflow instance. Case properties can be used in expressions of step parameters and route conditions, and to assign system instructions.

Functions

In the IBM Case Manager Builder, case properties are used to define in-basket columns, and are displayed as step parameters. The work item data that are displayed in the in-baskets are workflow field data, not case properties. Because of this restriction, IBM Case Manager Builder adds “shadow” data fields into the workflow definition. These fields use the same symbolic name and type as the corresponding case properties. A shadow field is added to a workflow definition for each case property. This field is used in one of the workflow steps (in either edit or read mode). This process is done automatically by IBM Case Manager Builder when you edit a task workflow in the Step Editor.

When IBM Case Manager runs a workflow (that is, creates a workflow instance), IBM Case Manager initializes the shadow data fields from the current case property values. When the business analysts define steps in the Step editor, IBM Case Manager Builder creates post assignments for any writable step parameter

that references a case property. These post assignments update the shadow data fields with their corresponding case property values. This process keeps the shadow data field values synchronized with their corresponding case properties.

IBM Case Manager Builder also adds these extra data fields that are required by IBM Case Manager components:

- ▶ **SolutionIdentifier**

The SolutionIdentifier is a string-valued field that contains the value *<solution prefix>_<solution name>*. Case Manager Client uses the SolutionIdentifier to identify which solution a work item comes from when displaying the personal in-basket.

- ▶ **F_CaseFolder**

F_CaseFolder identifies the case class and its instance. Process Engine uses the F_CaseFolder data field to retrieve and update the values for the case properties.

- ▶ **F_CaseTask**

F_CaseTask identifies the task class and its instance. Process Engine uses the F_CaseTask data field to update the state for a task.

When you modify a process definition by using the Process Designer, you can also directly access case properties without using the corresponding workflow definition shadow field. This is helpful if you must read or update a case property that you did not use in any workflow step in the workflow definition, and thus has no shadow field. In that case, you can access the case property by using the syntax F_CaseFolder.propertySymbolicName. In the example environment, the solution prefix is CC. Therefore, the case property Customer Name can be accessed in a workflow step by using F_CaseFolder.CC_CustomerName.

You can directly assign a new value to the case property or use it in a workflow expression the same way as any other workflow data field.

IBM Case Manager 5.1.1 provides enhanced support to access the case and task object properties in a workflow using **Business Objects** in the expression builder as shown in Figure 4-23.

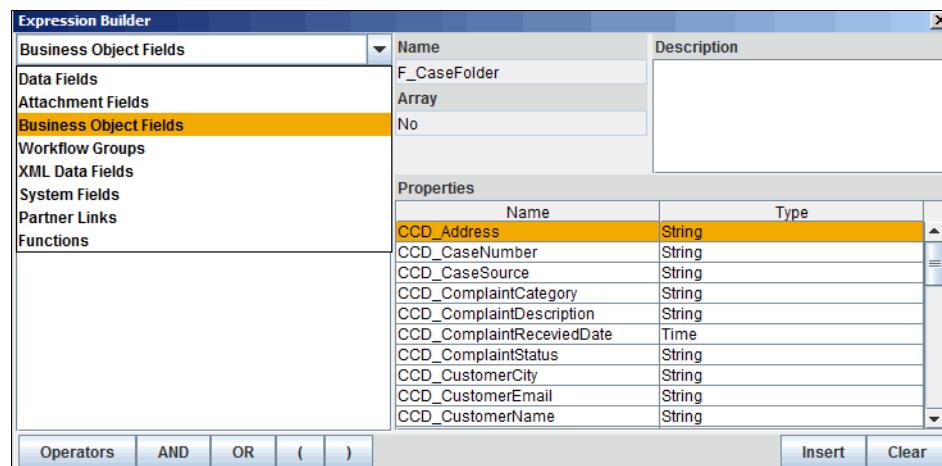


Figure 4-23 Accessing case and task properties in the Expression Builder

With this process, you can easily access case or task properties from a workflow definition. It also helps avoid errors due to misspelling. To access the business objects data fields, open the expression builder and select **Business Objects** from the drop-down list on the left side. Then, select **F_CaseFolder** or **F_CaseTask** as the business object, and double-click one of the associated data fields from the **Properties** list.

For more information about using the Expression Builder, see the FileNet P8 Information Center. Click **Integrating workflow into document management** → **Process Designer** → **Define workflow properties** → **Expression Builder**.

You can also define step parameters directly from Process Designer and initialize them with `F_CaseFolder.<properties>`.

Saving data in a task step

IBM Case Manager uses work object shadow data fields to track each case property that is used in one of the workflow steps. The information that is displayed in either the Properties widget or the Form widget when a task is opened from an in-basket reflects the *case* data. However, the information that is displayed in the in-basket widget reflects the *work object* data. Case data and work object shadow fields are synchronized whenever the work object is dispatched in a step by the post assignments. IBM Case Manager Builder automatically adds these assignments for case properties that are marked as

read/write for this workflow step. For more information, see 4.4.4, “Step parameters and post assignments” on page 116. Process Designer does the same when you define the step parameter for the step. Process Designer creates the post assignments as well.

When data is saved in a workflow step by clicking **Save** without dispatching the work item, any data change is only applied to the *case* object. The change will be applied to the work object only at the time when it is completed. It will be dispatched to the next workflow step. As a result, case data and work object data are out of sync until the user completes the step.

This is usually not a problem. Case workers typically do not save data changes without also processing the step. However, you can use IBM Case Manager to suppress the **Save** option from the Work Item Toolbar widget to avoid this problem. To do so, click to clear **Allow saving changes without completing a work item** as shown in Figure 4-24.

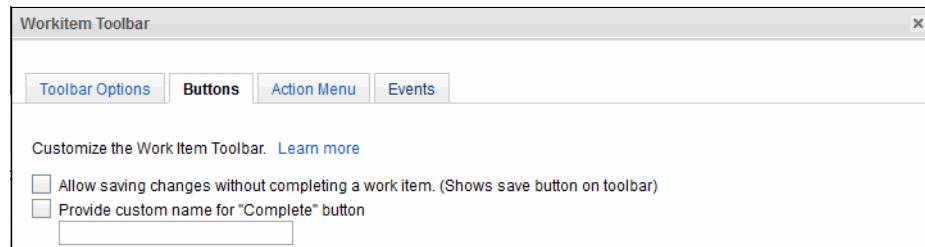


Figure 4-24 Work Item Toolbar widget configuration options

If you need to allow case workers to save changes without dispatching the work item in a certain step, add a response that loops back to the same step. You can call it, for example “Save Data”. This process allows users to change the data, which is applied to the case object. By running the response “Save Data”, the work item is dispatched and updated with the changed data from the case object. The *F_EnqueueDate* time stamp is updated in this case to reflect the time when the “Save Data” response is run. You can use a different workflow data field to track the initial enqueue time stamp if required.

If a case has two running tasks. The work items for two tasks are in the same queue. The work item for task 1 is opened, updated, and saved. Using the approach mentioned in the previous paragraph, the work item in the basket is synchronized with the case data. However, the work item in task 2, which is also in the same queue and same in-basket, still has the old data. The synchronization does not update the values of all work objects in the same queue.

In the same window, you can also define a different name for the “Complete” option, which is displayed for each step that does not have explicit responses defined.

4.4.3 Workflow attachments

From the Step Editor of IBM Case Manager Builder, you define the attachments for a workflow and assign them to a step as the step parameters. If a task has a filing precondition, one of the attachments is selected as the workflow that initiates attachment. The initiating attachment is added as a read/write parameter for the launch step of a workflow. If you do not want to update the attachment, you can add it as read-only parameter.

When IBM Case Manager starts the workflow by using the Content Engine event handler, Content Engine assigns the filing document to the attachment parameter for a task.

4.4.4 Step parameters and post assignments

Process Engine uses step parameter expressions to retrieve and update values for case properties. Process Engine uses step post assignments to update the Process Engine “shadow” fields. Process Engine retrieves the values of referenced case properties and assigns the values to the corresponding step parameters when the step is opened. Process Engine updates the values of case properties with the values of the corresponding step parameters when case workers or external services save or complete a work item. Process Engine updates only the case properties with the values of the corresponding step parameters if the step parameters have read/write or write permission.

4.4.5 Starting task workflows

The method IBM Case Manager uses to run a workflow that is associated with a task depends on a launch mode of the task. The following alternatives exist:

- ▶ Auto launch

This mode is for a task that starts automatically. The Content Engine event handler is responsible for creating and starting a task when it creates a case instance. Content Engine starts the task if it has no precondition or the precondition is satisfied. When it starts the workflow, the Content Engine event handler initializes the workflow data fields with the values from the case properties. If the launch fails, IBM Case Manager sets the task properties *State* to FAILED, *Disabled State* to DISABLED_ABORTED and *Last Failure Reason* to “CmAcmError Launch Failed”.

- ▶ Manual launch
- This mode is for tasks that start manually. The tasks must be in the READY state before users can start the task. When users start the task, IBM Case Manager, using Content Engine event handler, starts the workflow after it starts the workflow data fields with the values from the case properties. If starting the workflow fails, the same task properties are set as described above.

- ▶ User

This mode is for a discretionary task. When users click **Add Task** in IBM Case Manager Client, IBM Case Manager Client presents the users with the Add Task window so they can edit the workflow data fields. The Add Task window displays only the workflow data fields that are part of the step parameters of the launch step for the corresponding task workflow. For a workflow associated with a discretionary task, you can edit the launch step from the Step Editor of Case Manager Builder or by using the Process Designer.

IBM Case Manager Client updates the corresponding case properties with the parameter values and starts the workflow by using the IBM Case Manager API.

For more information about launch mode, see 4.1.8, “Task type” on page 86.

4.4.6 Workflow processing and task state

Process Engine updates the state of a task that is associated with a workflow during workflow processing to one of the following task states:

- ▶ FAILED

If an exception occurs, Process Engine sets the state of the task to FAILED. When the error is rectified, Process Engine clears the exception, resetting the task state back to WORKING.

If a user stops the workflow abnormally, Process Engine sets the task state to FAILED and the task property Disabled State to DISABLED_ABORTED. A workflow can be stopped abnormally if the user deletes all the work items for a workflow. For information about gracefully stopping and restarting tasks using the IBM Case Manager API, see “Stopping and Restarting Tasks” on page 93.

- ▶ COMPLETED

If a workflow is completed normally, the Process Engine sets the state of the task to COMPLETED.

► WORKING

This is the initial state of a task when a workflow is started. A workflow is in progress when a task in it has the WORKING state.

4.5 Object model for IBM Content Manager integration

The goal of the integration with IBM Content Manager is to enable IBM Case Manager to store and use documents in IBM Content Manager. In other words, this integration allows IBM Content Manager to be the document repository for all the documents that are attached to cases and tasks.

You can interact with IBM Content Manager documents attached to cases and tasks in the same way as you would with documents stored in FileNet P8. You can attach the IBM Content Manager documents, update them, and delete them from the case. In addition, document events can trigger tasks.

Restriction: A solution can use either FileNet P8 or IBM Content Manager as document repository, but not both.

The integration depends on the tight linking of documents in IBM Content Manager, and folders in IBM Case Manager. This integration is done by using proxy objects in each of the respective repositories.

Each document in IBM Content Manager that is attached to a case is represented by a proxy document in FileNet P8. This proxy document is then filed in the case folder.

Each case folder in FileNet P8 is represented as a proxy folder in IBM Content Manager. These objects are monitored by the IBM Content Manager adapter, and the IBM Case Manager adapter. When events occur on these objects, they are mapped with the appropriate action.

Figure 4-25 shows the object model that is used for the IBM Content Manager integration. CM8 in the figure stands for IBM Content Manager. ICM in the figure stands for IBM Case Manager.

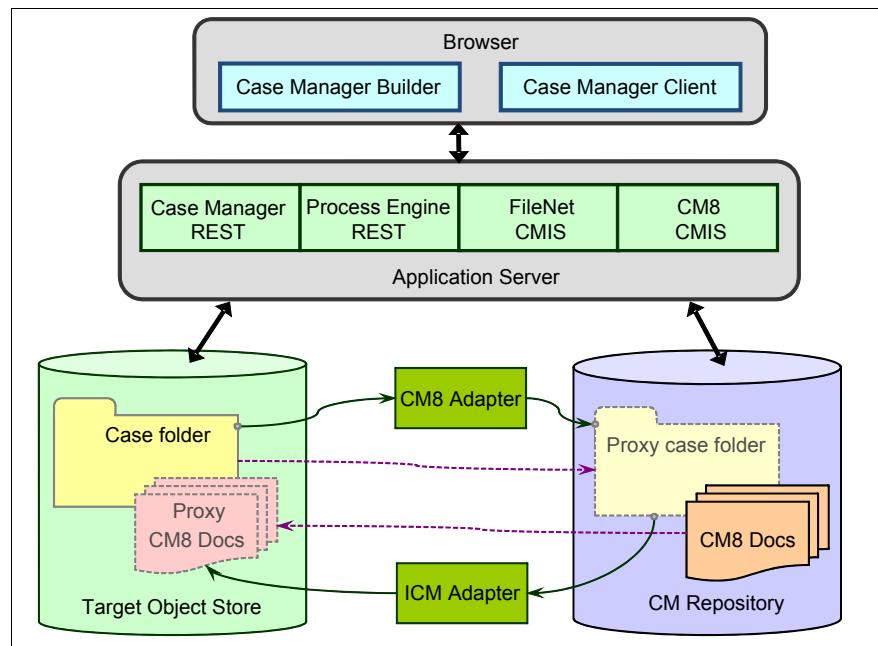


Figure 4-25 Object model for IBM Content Manager integration

Restriction: With IBM Content Manager integration, the case folder does not support the use of subfolders. All documents are held in one level. However, you can use the CE_Operations in the workflow to move documents to subfolders.

For a list of limitations with the IBM Content Manager integration, see the IBM Case Manager Information Center. Click **Installing and configuring IBM Case Manager** → **Extending your case management system** → **Integrating with IBM Content Manager** → **Known limitations and differences for IBM Content Manager**.

Event subscriptions are supported as with the FileNet P8 repository. For example, you can configure a solution so a new case is created when a case worker creates a document in an IBM Content Manager item type. Adapters are used to monitor the events and ensure that the appropriate actions are carried out.

The associations of event subscriptions and IBM Case Manager item types are configured when you set up the integration.

For each document associated with a case in IBM Content Manager, there is a proxy document in IBM Case Manager (FileNet P8), of class CmAcmCM8ProxyDocument. This proxy object holds no user properties.

For more information about the object class, see the IBM Case Manager Information Center. Click **Developing case management applications** → **Content Engine add-on extensions for Case Manager Builder** → **IBM Case Manager target object store extensions** → **Custom subclasses and properties of the Document class** → **Proxy Document Class**.

For each case folder in IBM Case Manager (FileNet P8), there is a proxy object in IBM Content Manager. This proxy object does not include any user properties. Instead, it contains the following properties that identify the case folder in IBM Case Manager:

- ▶ Case ID
- ▶ Case folder GUID
- ▶ Initiating document ID
- ▶ Object store symbolic name
- ▶ Solution name
- ▶ Case type
- ▶ Case folder name

The information about the integration connection with IBM Content Manager is held in a special custom object of the IBM Content Manager Integration Data class, CmAcmCM8IntegrationData. The information is managed by IBM Case Manager administration client.

For more information about the class, see the IBM Case Manager Information Center. Click **Developing case management applications** → **Content Engine add-on extensions for Case Manager Builder** → **IBM Case Manager target object store extensions** → **Custom subclasses of the Custom Object class**.

For more information about IBM Content Manager Integration, see Chapter 15, “Integration with IBM Content Manager” on page 549.



Part 2

Solution development

This part guides you as you get started with IBM Case Manager solution development.

This part includes the following chapters:

- ▶ Designing case management solutions
- ▶ Building a simple solution: Part 1
- ▶ Building a simple solution: Part 2
- ▶ Solution deployment
- ▶ Solution development
- ▶ User interface
- ▶ Development topics
- ▶ Round-tripping workflow editing



Designing case management solutions

This chapter addresses how to design solutions with IBM Case Manager. It covers the various approaches, alternatives available, and key considerations. The Complaint Management example is used to illustrate the design process. This chapter is intended to help the solution designer to plan and design a solution. It addresses design principles and gives an overview of the range of tools and features that can be involved in building an IBM Case Manager solution.

This chapter includes the following sections:

- ▶ Business goals of a case management solution
- ▶ Designing the solution
- ▶ Implementing the solution
- ▶ Level 1: Defining your solution
- ▶ Level 2: Refining your solution with advanced configuration
- ▶ Level 3: Customizing and integrating your solution
- ▶ Documenting your solution

5.1 Business goals of a case management solution

The goals of the solution must be well understood before you design it. The many different types of case management solutions across many industries have their own unique set of design goals. However, there are some common goals that are associated with solutions built on the IBM Case Manager:

- ▶ Support the knowledge worker to enhance the quality of the decision making for each case
- ▶ Provide faster, more accurate, and fair case resolutions
- ▶ Collect all information related to a case such as documents and comments
- ▶ Improve productivity by supporting a collaborative environment where information can be securely and easily shared
- ▶ Automatically initiate activities that are mandatory for the case
- ▶ Allow the case worker to initiate other optional tasks, or add new tasks, when needed
- ▶ Automatically initiate tasks according to predefined conditions or events
- ▶ Provide a framework in which all the tasks related to a case can be initiated and tracked together in the context of the case
- ▶ Provide auditing information about how case outcomes were determined
- ▶ Allow the case worker to search for cases based on categories, data ranges, document content, and so on
- ▶ Allow the case worker to search for ongoing and historic cases
- ▶ Support the use of analytic tools that can help to gain insights from past cases, and mitigate risks and reduce fraud
- ▶ Improve agility by using business rules for decision support

IBM Case Manager provides many capabilities and tools that can be used to help meet these goals when you design a solution.

5.2 Designing the solution

This section addresses various topics that are related to designing of case management solutions.

Before you design a solution with IBM Case Manager, get at least a basic understanding of the IBM Case Manager components and data model. For an overview of the components, see 3.3, “IBM Case Manager components” on

page 38. For an overview of the data model, see 4.1, “IBM Case Manager object model” on page 67.

Consider reusing an existing solution template, and consider using your new design as a template for future solutions. For more information about solution templates, see:

- ▶ 9.4.6, “Creating solution templates” on page 326
- ▶ 9.4.7, “Using solution templates” on page 329
- ▶ 9.4.8, “Available solution templates” on page 333

5.2.1 Agile and iterative methodologies

Businesses face increasing challenges:

- ▶ Meeting customer service demands
- ▶ Compliance and regulatory requirements
- ▶ Reducing risks that are associated with business activities
- ▶ Managing the growth and complexity of both internal and external information sources

As these challenges increase, knowledge workers must handle more varied tasks. To stay competitive, businesses are automating and outsourcing routine processes and seeking more productivity from their knowledge workers.

To manage this growing complexity, the traditional waterfall application development methodology is inappropriate. Time-consuming, waterfall-based methodologies often create solutions that are obsolete before they are delivered. Requirements are often collected outside the context of capabilities that can be delivered rapidly and with low cost.

Successful use of technology in this environment requires businesses to act and react with agility, flexibility, and assurances that they can handle the changing demands.

IBM Case Manager supports the agile methodology by providing a rich set of tools that supports prototype creation, template usage, and case management solutions refinement.

The toolset offers a default user interface that allows initial requirements gathering to be a process in which the tool itself is used. Case Data, Document Tasks, and an outline of tasks can be identified and walked through with the business users. This time-boxed requirements analysis can foster a collaborative environment between the project team and the business users of the system. It can show a live system that they can interact with, without spending time

developing a user interface. Depending on the requirements and detail technical aspects of the project, this interaction can continue as the user interfaces and tasks are updated within the system.

Iterative methodology helps ensure that the requirements are more inline with what the business requires. It also provides early feedback in what the system will look like and how it behaves. Agile and iterative methodology allows the creation of flexible systems that handle adversity and can be delivered in multiple phases. This is a general practice statement because individual projects vary in how they must be implemented to best serve the solution.

5.2.2 Case design

This is the first part of all case management projects, and the most important. A clear and precise definition of what the case is defines all other aspects of the solution. The tasks, document types, rules, and reports are all created and used to support the case. Thus, the identification and definition of this object is crucial.

Sometimes a case, such as a claim, can be easily identified. Sometimes, it might not be so easy to identify what is a case. In general, a case can be anything that has a purpose, meaning, and life to it. It needs a beginning, middle, and end. Use those attributes to identify your case.

After the object itself is identified, the next part is to obtain the data around it. This data can come from a number of sources, as shown in Figure 5-1.

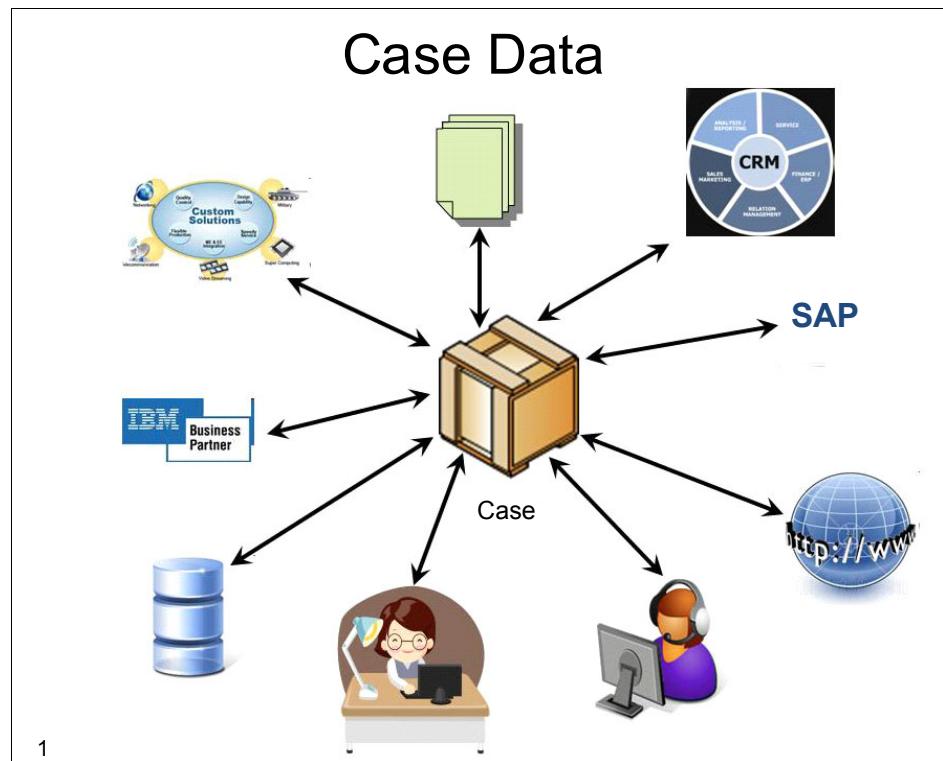


Figure 5-1 Multiple sources for case data

As shown, the case can derive its data from a number of sources, and can interact with that data at any point within the case's lifecycle. The identification of the interaction points and the timing of the data are part of the design of the case itself. The case data addresses these questions

- ▶ What data is needed to complete the case?
- ▶ What data or systems need to be aware of the case's progress?
- ▶ What systems does the case need data from?
- ▶ What KPIs and other reporting metrics are needed against the case?
- ▶ What synchronization is needed with the case management solution and the system of record (assuming the case management solution is not the system of record)?

After these questions are answered, the case is defined.

5.2.3 Task Design

The concept of case management design is to understand what the various pieces of work in the overall business process are. You can then encapsulate them as separate tasks. The tasks become part of the case worker tool set. The way to identify these tasks is to look to discover how a case gets completed. Look for verbs in requirements to represent discrete activities that can be mapped to an implementable task.

After the activities are defined, work with users to understand:

- ▶ How a user interacts with the tasks.
- ▶ When and why the tasks are needed or defined.
- ▶ How and when should these tasks be started?
- ▶ Who should work on the tasks?
- ▶ Are there dependencies among defined tasks?

Predictable tasks can be configured to initiate automatically. The less predictable tasks are available to the case worker, who can decide if and when to initiate them.

A task in IBM Case Manager is implemented as a workflow process. A case can have many tasks that are associated with it. Each task involves part of the work necessary to complete the case. Each task can have one or many steps and one or more participants. The tasks can be run in sequence, in parallel, independent of each other, or in relation to each other. IBM Case Manager tracks all the tasks in the context of each case.

When you design a case management solution, some of the main design decisions are related to the definition of tasks. How do you decide how to break down the necessary work into tasks? Is it better to have bigger tasks that handle larger segment of work, or many smaller tasks? In general, the best design is based on a logical model of the business process. However, some situations can especially benefit when modeled as a case task.

The following are some of the common situations that tend to work well modeled as an IBM Case Manager task:

- ▶ When you need to initiate work when an event happens, and that event can occur at any point in the life of the case. Map this to a separate Case Manager Builder task rather than define the event trigger multiple times in all the relevant processes. This mapping makes the process clearer, and reduces the risk of unexpected issues arising from overlapping responses.
- ▶ When a set of work needs to be completed, but there is no particular defined sequence in which they must be carried out.

- This situation is commonly modeled in BPM tools as a “star” or “spider” shaped process diagram. This work set can also be modeled as a many-to-many flow or as a split/join.
 - Using IBM Case Manager, this set can be modeled by using an all-inclusive set. In this kind of set, each task must be completed for the case to be completed. However, the tasks can be completed independently of each other. In addition, each task can be set up to start manually or automatically when preconditions are met. For more information, see “Using all-inclusive sets” on page 141.
- Decisions and workflow paths are too numerous or complex, and it is simpler to allow knowledge workers to decide whether the work is needed.
- Defining a long complete sequence of work like in a BPM process that identifies every decision, event, and exception is too involved. Instead, define a set of tasks, and allow the knowledge-worker to decide which tasks must be carried out and when.
- Workflow paths that depend on a number of factors that are not known during workflow design time.
- Often is it better to enable knowledge-workers to decide on the activities and their order during process execution. This is especially important when they use a different sequence in different situations and even among themselves.
 - With IBM Case Manager, you can define all the necessary tasks that are associated with the case type as per business guidelines. You then rely on the knowledge-worker to initiate the tasks when appropriate.

5.2.4 Case documentation

A case almost always includes documentation. This documentation can directly and indirectly support the case. Documents might arrive throughout the lifecycle, some of which help instantiate a case and others that are there for support. When you design the document types for a case solution, all of these types must be accounted for.

Forms also play a role in what is considered content within the system. A form can be a set of data or help to visualize a set of data. The differences here can have some impact on design. Forms that are used for a user interface construct must exist at design time to ensure that it is part of the solution package and to tie it to the digits and widgets within the user interface.

Outside of these design considerations, the rest of the design follows a typical ECM focus. Document Types are displayed within the case. They have properties and security that must be implemented

5.2.5 Business rules

Within the scope of the case management solution, business rules can be implemented. These business rules are implemented in areas that allow for a more dynamic and flexible solution. Although it is common in the first implementation to not include these, look into them to determine where the solution can grow and expand and incorporate them.

Externalizing the determination of decisions is one of the key reasons to use a rules engine. The incorporation of rules can occur in tasks, user interface, and data elements of the solution. Rules within a task are the most common approach to rule execution and use. They can help remove coding efforts within the task and externalize them to allow for changes that do not affect how a task is used going forward. The same approach can be used within the digit events and forms. After rules are defined, they can be used within any aspect of the system to help ensure consistency and accuracy when data is involved.

The design of rules is also the same as it is in any other solution. There is no need to adjust the project or process when rules are used. The main is to consider is when and where to use them to the fullest extent possible.

5.2.6 Modeling how cases are to be created

When you design a case management solution, consider how the cases are created. This process depends on the specific application. Generally speaking, there are four ways in which cases are created using the features of IBM Case Manager:

- ▶ Manually using a form

Case workers use the default add case form or a custom form to create cases manually. In addition, other users can access a form that can be used to create the case. For example, an online complaint form might be used to create a case.

- ▶ Created when a specified document is created or added

A particular document type can be defined to be the initiator of a case. When a document of this type is created, the system automatically creates a case. It then passes the document metadata to the case properties and files the document into the case folder. For example, an insurance application document might automatically create a case.

- ▶ In a workflow process

Case can be created from a workflow processes. For example, a loan application process can start a case to investigate a potential fraud situation. Using the CE_Operations component queue, a workflow process can use the

createFolder operation to create a case. The workflow process can be any workflow process that runs on the system. This process can be either in a case task workflow or other workflow process started by an event subscription.

- ▶ Programmatically

Other systems can create cases in various programmatic ways:

- Using the Case Representational State Transfer (REST) application programming interface (API).
- Using the Content Management Interoperability Services (CMIS) REST API interface. External data services are not started when you use the CMIS REST interface.
- Using the Java/.NET/Web Services Content Engine API interface. Like the CMIS REST API, external data services are not started.

Calling a published Process Engine Web Services Description Language (WSDL) might then call the *createFolder* operation in CE_Operations. The Process Engine WSDL is a workflow process that publishes a WSDL from a Receive Web Service step in the process

5.3 Implementing the solution

Implementing a case management solution in the real world is typically not trivial. However, you can get to a useful level quickly by using the Case Manager Builder tool.

Implementing a case management solution is an involved process. It ranges from simple configuration, to advance configuration to coding and integration. This section addresses the implementation of your solution and provides an overview of the work involved. After you have a list of requirements, use this overview to plan what range of work is needed to meet those requirements. It is not intended as an exhaustive list of the capabilities of IBM Case Manager.

For description purpose, the solution implementation can be categorized into the following levels of complexity:

- ▶ Level 1: Defining the solution with Case Manager Builder

Use Case Manager Builder to rapidly develop the main structure of the solution through configuration. This process creates an executable stand-alone solution, complete with user interface, case metadata, and workflows. This solution not only forms the foundation for further enhancing the solution, but can be used to meet simpler requirements. It is commonly used as a rapid prototype to validate business requirements.

- ▶ Level 2: Refining the solution using more advanced configuration
IBM Case Manager provides rich set of enhancements and tools. These enhancements can be used to further refine the solution and meet more advanced requirements without the need for coding.
- ▶ Level 3: Customizing the solution using custom code
Most real world implementations of case management solutions involve all three levels. For example, you might use advance configurations to meet special security requirements, or need custom code to integrate with other system in the environment.

Although each level focuses onto increasingly fewer and more specific areas of the solution, it also increases in complexity, time, and effort. Figure 5-2 illustrates the three levels.

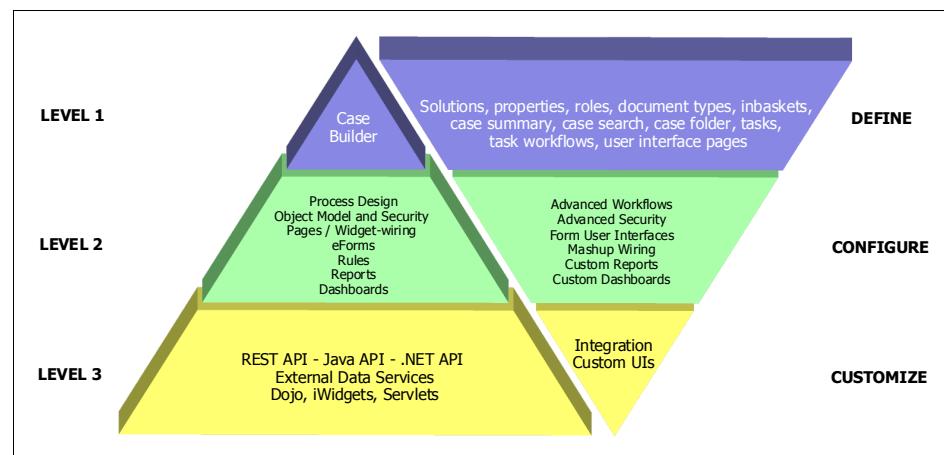


Figure 5-2 Definition, configuration, and customization levels

5.4 Level 1: Defining your solution

Using the Case Manager Builder tool, Level 1 solution definition and creation activities have the following characteristics:

- ▶ High-level business abstractions
- ▶ Performed by *business analysts*
- ▶ Rapid
- ▶ Core case management functions
- ▶ No programming or scripting needed

The goal of Level 1 definition activities is to rapidly create a solution. It also defines the fundamental structure of the solution, including its data model and tasks.

For an overview of the IBM Case Manager data model concept, see 4.1, “IBM Case Manager object model” on page 67.

Configuration at Level 1 normally involves direct participation by line of business experts. These experts have in-depth information about the needs and requirements of the users of the case management solution. This direct involvement enables better alignment with the needs and goals of the users and promotes productivity improvements / usability enhancements to be realized.

5.4.1 Defining case properties

The case properties are used to describe and identify each case for search, processing, reporting, and auditing purposes. Default case properties are available in the Content Engine object store. Using existing properties enables the solution developer to maintain a uniform document taxonomy for all solutions.

For each property, the following information must be provided:

- ▶ Data type (string, Boolean, date-time, float, or integer)
- ▶ Single value / multiple value

Optionally, the following information can be provided for a property:

- ▶ Maximum length (string) or minimum/maximum value (integer)
- ▶ Default value
- ▶ Choice list (integer and string)

5.4.2 Defining case types

Each solution in IBM Case Manager can have one or more case types. A case type is made up of properties, views, folders, and tasks. It is a specific instance type of the overall case that has been defined. The Case type is a level of encapsulation of the overall case into a specific business case. Each case type that is defined in a solution shares with other case types:

- ▶ The pool of properties available in the solution
- ▶ The document types in the solution
- ▶ The roles and in-baskets defined in the solution

5.4.3 Defining document types

Each solution has zero or more document types defined. Document types are defined within Case Manager Builder, and can have properties associated with them from the list of available properties in the solution. Default document types are available in the Content Engine object store. Reusing document types allows the solution developer to maintain a uniform document taxonomy for all solutions.

Document types also include documents that are stored as structured form data objects.

Remember: Not all documents that are stored in cases by the users must be identified as a solution document type. Any document in the system can be stored in a case folder. However, those document types that trigger case creation, start tasks, or have specific meaning and purpose within the solution must be identified in Case Manager Builder.

The Complaint Management solution example has two document types:

- ▶ *Correspondence*: Triggers a task for document approval before it is mailed to the complainant
- ▶ *Supporting Document*: Used for any other document that is filed into a case such as an email or scanned in letter

5.4.4 Defining roles

Roles are representative of the functions that individuals or groups of individuals perform on the system. Create a role for any functional group that acts as a participant in a workflow process. In addition, roles can be defined for non-active workflow participants who can search and retrieve case information only.

5.4.5 Defining in-baskets

In-basket definition at Level 1 creates a base in-basket configuration. Each role has a shared work queue, and a single in-basket for that queue. When you define the in-basket, you set up the properties that display, the ability to sort, the default sort order, and one or more filter options.

All work that is assigned to this role within the task map created in Step Editor goes to these shared work queues. The work is presented in the in-baskets that are defined in Case Manager Builder.

5.4.6 Defining tasks

Tasks are associated with individual case types. The name of each task often starts with an action word such as “Review”, “Perform”, “Assess”, and “Update”.

Each case type can have many tasks associated with it:

- ▶ Standard case management workflow processes
- ▶ Workflow definitions that are reused from existing processes that run on the FileNet Process Engine
- ▶ Processes that are started and run in the IBM BPM platform

Required automatic

Required-automatic tasks generally function as the main workflow processes that run when the case is created. They run in parallel immediately after the case is created.

In the Complaint Management solution example, the “Review Complaint” task is defined as a Required-automatic because it must be completed for every complaint that is received.

Required automatic with property preconditions

Required automatic with property preconditions tasks are workflow processes that must complete, but do not start until a precondition is met. These tasks must be completed at some point in the lifecycle of the case. Typically these tasks are completed when the case reaches a certain stage, or when an event is triggered.

Tip: Required tasks with preconditions forces the case to remain in the Working state until the precondition is satisfied and task is completed.

In the Complaint Management solution, the “Respond to Invalid Complaint” task has a precondition that is set when a call or email is received against a complaint marked invalid. For an invalid complaint, the system waits for five days before it closes the complaint to see whether the complainant responds about the complaint they issued. This approach enables the business to track how many complaints are marked invalid that actually should be valid.

Optional automatic with property preconditions

Optional automatic tasks with property preconditions allow work to be performed, but are not required for the case to be completed. These tasks can be triggered by the preconditions at any point in the lifecycle of the case. The tasks can be triggered even after the case is completed. Optional automatic tasks with

preconditions change the case state to “Working” when they are triggered, allowing a case to be reopened using this method.

This task is commonly used with cases that are initiated by request forms and where the form allows the user to make multiple selections. Each selection can map to a particular task workflow process that pertains to the selection made.

As an example, a service request/application form might have an area where the form user can check which services they want to request/apply for. Each check box requires a set of activities to be performed that are specific to that service. By mapping each check box to a property and having tasks started when the service check box is checked, multiple parallel task workflow processes are started within the case. Each runs independently, but they are aggregated together as part of the case. This configuration allows any user who opens the case to easily see which services were requested and how close each is to completion.

Another common use case is when tasks work in concert with the iLog JRules business rule engine. When rules are used to determine customer rating or scoring, for example, tasks can be triggered when the rating/score is above or below a certain threshold.

In the Complaint Management solution, the “Review Product Complaint” is an example of an optional automatic with precondition task. It runs only when the Complaint Category is “Product” and when Valid = true.

An example in the complaints scenario is a “Provide Immediate Feedback” task, which is started only when a complaint is received from a highly rated customer. For this task, the rules engine determines whether the customer is highly rated based on a set of evaluation rules. This task is a high-priority work item for the contact center to call and assure the customer that the complaint is being investigated.

Required manual

A required manual task is a task that must be completed or disabled in order for the case to be completed. It is initiated/disabled at the discretion of the user. Manual tasks are typically used when the time the work is initiated is not known in advance, or when the conditions needed to initiate the task are complex.

Users can start this task (and must manually launch/disable it) at any point in the case lifecycle. Often, users review the information that is stored in the case, in particular the comments and unstructured information, to decide whether the task must be started. For example, upon reading the comments and relevant documents in the case, a user might determine that the case can be closed. The user then starts a manual task to initiate a case closing workflow process.

In the Complaint Management solution, the “Close Complaint” can be initiated manually only after the worker reviews the details of the complaint. The case worker must also determine that there is sufficient information to be able to close it.

Required manual with property preconditions

Required manual with preconditions tasks allow users to initiate task workflows only when a condition is met. The case cannot be set as completed until this work is completed. Users can disable the task at anytime. This task waits within the case until the preconditions are met before it can be started. For example, a user might decide to start the task upon review the information in the case *and* the preconditions are true. This is a more precise implementation of the required manual task.

Optional manual

An optional manual task is a task that can be initiated/disabled at the discretion of the user. Manual tasks are typically used in situations when you do not know when the work will be initiated, or the conditions needed to initiate the task are complex. Users are able to start/disable this task at any point in the case lifecycle. Often, users review the information that is stored in the case, in particular the comments and unstructured information, to decide whether the task can be started. For example, a user might decide that a specific piece of information is needed to make other decisions in the case.

In the Complaint Management system example, the “Perform Customer Outreach” task is an optional manual task started by a user. It is used when the user feels the complaint requires a more formalized process to contact the customer through an existing outreach process.

Optional manual with property preconditions

Similar to optional tasks, optional manual with preconditions tasks are used to allow users to initiate task workflows only when a condition is met. Users can disable the task at anytime. This task waits within the case until the preconditions are met before allowing a user to initiate it. For example, a user might decide to initiate this task after reviewing or noticing specific information in the case *and* the preconditions are true. This is a more precise implementation of the optional manual task.

Document is added to the case precondition

Any tasks with preconditions (required/optional - automatic/manual) can be configured to initiate when a document is received. They can be configured to run when a document of any type is received or when a document of a certain type is received.

This precondition type has a special option that enables it to be started repetitively. The case indicates that one or more of these tasks are triggered in this situation.

This special type of precondition is ideal in situations where work is to be performed when a document is filed into the case. Often the filing of a document in a case means that additional work such as the review of the document must be initiated.

Advanced topic: Task workflows that use a WaitForCondition system step for trailing documents can be set up. They can be in addition to tasks where the document is added to the case preconditions. In the former, tasks cannot be completed until the specified document arrives. In the latter, tasks cannot be initiated until the specified document arrives.

This precondition is often used in situations where documents are received during the lifecycle of the case. For example, it can be used when emails or scanned mail are added to the case. This precondition is also used in situations where case workers create content and file it into the case, often to trigger a review and approval process.

In the Complaint Management system example, a Review Complaint Correspondence task is started every time an official complaint response letter of type *Correspondence* is generated. The task workflow process follows a review and approval process, including a legal review. This process continues until the letter is printed and mailed to the complainant.

Discretionary or user-created tasks

Discretionary / user-created tasks enable the business analyst to define tasks that users can start as required. These tasks are not displayed on the tasks tab until they are added by a user. User-created tasks have the following traits:

- ▶ They can be added (created, started) after the case is started. They can be created multiple times and each instance has its own task name, making it uniquely identifiable among the other tasks in the case.
- ▶ They are ideally suited to situations where new work emerges while the case is in progress. Users can create new tasks at their discretion.
- ▶ They map to a workflow process, and launch parameters can be set by the user when the task is created. In the Launch step of the discretionary task, the user configures the fields/parameters that govern the process instance. The user also assigns users to workflow groups if this is configured in the task
- ▶ They differ from manual tasks in that they are repeatable and only discretionary tasks display the launch step. The name of the task is set by the

user at task creation. Setting the task name allows it to be recognized uniquely, unlike manual tasks that have predefined names.

- ▶ They are often used to create work for other case participants as required by a user. For example, when more information is needed, a discretionary task can be added for a contact center to call the customer and ask for that information.
- ▶ Discretionary tasks do not exist until they are created at run time by users. Many discretionary tasks can be defined for a case type. These tasks can match the various workflow processes needed to process work involved in the case.

Consideration: User-created tasks are a key component of IBM Case Manager systems. They allow workflow processes to be added by knowledge workers as required after the case is created. Using discretionary tasks, the case worker is more in control of the information and collection process because the activity is tracked in the case.

The Complaint Management system example has two user-created tasks:

- ▶ “Request Assistance” is a task that a user can initiate to request help from the contact center. This task is mostly used to communicate with the complainant. It might also be used to request any additional help from the contact center.
- ▶ “Investigate Employee” was modeled as a discretionary task because it might be started multiple times for each employee who is mentioned in the complaint. Because it is a repeatable, manual task, it was set up as a discretionary task.

Task comparison chart

Table 5-1 compares the task types, their characteristics, and the use cases.

Table 5-1 Comparison of task types

Task type	Characteristics	Use cases
Required Automatic	Started by: System Repeatable: Yes	A workflow that runs when the case is created. It must be completed.
Required Automatic with property preconditions		A workflow that runs whenever a property condition is met either upon creation of the case or after a property value changes. It must be completed. A user can start the workflow by updating the case properties to meet the preconditions.
Optional Automatic with property preconditions		Same as required automatic, except this task does not have to be completed to complete the case.
Required Manual	Started by: User Repeatable: Yes	A workflow that runs when a user starts. It must be completed.
Required Manual with property preconditions		A workflow that runs when a user starts it. It must be completed. A user can start the workflow only if the precondition is met.
Optional Manual		A workflow that runs when a user starts it. This task does not have to be completed.
Optional Manual with property preconditions		When the precondition is satisfied, the user can start the workflow. This task is the same as optional manual, but has property preconditions.
When a document of a certain type is added precondition	Started by: System or user Repeatable: Yes	A required, optional automatic, or manual workflow process that runs when a document of a certain type is filed in the case. If the task is flagged as repeatable, every document of that type that is received starts the workflow.
When a document of any type is added precondition		A required, optional automatic, or manual workflow process that runs when a document of any type is filed in the case. If the task is flagged as repeatable, every document that is received starts the workflow.

Task type	Characteristics	Use cases
User created task (discretionary)	Started by: User Repeatable: Yes	A predefined workflow started by a user where the user can name the task and supply the launch parameters. These parameters include case properties, user/workgroup assignment, and documents to attach to the workflow.

Using all-inclusive sets

An all-inclusive set is a group of tasks. If any of the tasks in the set is started, all of them must be completed. An all-inclusive set is a way to designate a group of tasks as having to be completed for the case to be completed.

All-inclusive sets form a simple abstraction of parallel and required work activity. In practice, all-inclusive sets are often used with a group of optional manual tasks. When a user initiates or disables one of the tasks, the other tasks in the group must be started or disabled as well.

Using mutually exclusive sets

A mutually exclusive set of tasks is a group of tasks where only one task can be started. After one task is started, the other tasks are disabled.

The Complaints Management application uses a mutually exclusive set to determine whether a Review Product Complaint or Review Non-product Complaint task is to be performed. Both tasks cannot be performed. The task to be performed is determined by the Complaint Category property. This property allows for easy determination and comparison of complaints that are product-related or non-product related.

5.5 Level 2: Refining your solution with advanced configuration

This section addresses the features, benefits, and considerations of the advanced configuration capabilities available with IBM Case Manager.

As in Level 1, the capabilities characterized as Level 2 here require no coding. However, you need a good understanding of the IBM Case Manager architecture, object model, and FileNet BPM. Many capabilities are available from the advanced tools, including IBM FileNet Process Designer, and in some cases the FileNet Enterprise Manager administration tool.

Using both Case Manager Builder and FileNet Process Designer to configure solutions is known as “round-tripping”. The solution designer goes back and forth between Case Manager Builder and IBM FileNet Process Designer to take advantage of the most advanced functions. This configuration is an important aspect of building solutions with IBM Case Manager. For more information, see Chapter 12, “Round-tripping workflow editing” on page 467.

For details about capabilities available within the IBM FileNet Process Designer, see *Introducing IBM FileNet Business Process Manager*, SG24-7509.

Table 5-2 provides a list of common requirements that can be met by using more advanced configuration, and a brief description of what is involved. This list assumes that you are familiar with the capabilities of FileNet Process Designer.

Table 5-2 Common uses in application design

Usage	Description
Advanced routing	Case Manager Builder allows the business analyst to assign responses to routes in the process. Using Process Designer, routes can have broader and more complex branching calculations.
Starting tasks by setting preconditions	A task with preconditions can be started by modifying the matching properties of the case folder object. It can also be started by using the createDocument operation from the CE_Operations library to create a document in the case. This document in turn starts a task with a filing condition.
Updating case/document properties	Folder properties can be updated in task workflow processes by using simple assignment. Other cases, documents, and any objects can have their properties updated by using the operations available in CE_Operations.
Updating in-basket data without moving to next step	To update in-basket queue data while staying on the same step, a response can be set up that branches and returns to the step. This configuration enables any changes to case properties to propagate to the in-basket data fields.
Sending Automated Emails	Individual email notifications can be set up under user preference settings in Workplace or WorkplaceXT. In addition, send mail operations available in CE_Operations can be used to send emails during a task workflow process.
Assigning work to a user defined in a case property	To assign a work step to a designated user at run time when the user is defined in a case property, use Field Assignment on the Before Execution tab. This process assigns the user value to the correct workgroup/participant.

Usage	Description
Automated filing of documents into cases	An event action can be used to automate the filing of a document. These documents can include scanned documents, emails from IBM Content Collector and Quickr® services, FIMO documents, and WorkplaceXT entry templates. For more information, see 11.2, “Automated handling of ingested documents” on page 427.
Managing trailing documents	When task workflows are waiting until a required document is received, the workflow can use a WaitForCondition system step. This step starts a workflow process when the document is received. This type of task is useful in situations where tasks are used to track the collection of documents in the case.
Using rule-based decision services	Workflow processes can start rule services from iLog JRules by using a Web Services partner link.
Gathering data from other systems	Case properties can be populated with data from other systems. Workflow processes can obtain this data by using dbexecute system steps, other system steps, Web Services steps, or custom component integration operations.
Sending case data to other systems	When data from the case management solution is required by other systems, tasks workflow processes can send case data by using the built-in Web Services capabilities.

5.5.1 Advanced in-basket configuration

You can use the Process Configuration Console through Process Designer to configure more in-baskets in IBM Case Manager. Each extra in-basket that is configured for a role is displayed as an extra tab on the Work page.

Important: When you use Process Configuration Console to modify settings, you must also make the changes manually in the target system.

IBM Case Manager supports both shared work queues and personal in-baskets that are defined for the roles in the solution. In-baskets are configured to provide specific presentation and capability of the work queues/inbox:

- ▶ Definition of properties to be displayed on the worklist
- ▶ Configuration of predefined filters on a work queue or personal inbox

Configuring a predefined filter allows you to present users with only those work items that match the criteria of the filter. For example, you can show only work items where “valid insurance” equals true.

- ▶ Definition of properties that can be filtered by the user or through widget wiring
- In addition to allowing users to filter work items for search purposes, filter definitions can be wired with other widgets to enable a *runtime* filtering capability. This runtime filter allows the solution designer to create in-baskets that present the worklist for users filter for their specific context within the application. As an example, a user might select their role and geographical region, and see only in-basket work items that pertain to that geographical region.

Table 5-3 provides common uses of in-baskets in application design.

Table 5-3 Common uses in application design

Usage	Description
Multiple in-baskets per role	More in-baskets can be created and assigned to roles for other work queues. They can be customized with modifications and filters.
Changing the order of in-baskets	The order in which in-baskets are displayed in the Case Manager Client can be configured.
Adding system fields as columns	Fields that were not defined in the Case Manager Builder, but are available in the system, and can be added to the work queue and in-basket as columns.
Using pre-filtered, sorted in-baskets	Different views of work queues can be displayed to a user/role by configuring more in-baskets that contain pre-filtered, sorted work items.
Using advanced in-basket filters	Creating an in-basket runtime-based filter can be configured through wiring of the in-basket iWidget to another iWidget. This filter enables in-baskets to be filtered at run time based on contextual information.

5.5.2 Changing the object model in the solution

You can create a more flexible object model and enable more advanced subscriptions. Managing the object model hierarchy that is defined for your solution is done by using the FileNet Enterprise Manager.

Table 5-4 shows the common uses of object models in application design.

Table 5-4 Common uses in application design

Usage	Description
Creating solution document types in a hierarchy	When the solution involves many document types, it is often a good practice to define a parent document class, followed by subclasses for each document type. One advantage of this design is that you can assign the parent class to start a task in Case Manager Builder. The task then starts automatically for all the document subclasses.
Using advanced form data types for structured data case objects	To store structured information in the case, extend the Form data object class with a new subclass for your structured data object. Add the properties that you need to this class as required.
Creating relationships between cases: Simple method	A simple method to enable relationships between cases in IBM Case Manager is to configure a multi-value property as part of the case types. This property maintains the IDs of related cases. The Case Manager Client can be configured to search and display related cases using widget wiring.
Creating relationships between cases and other objects: Enhanced method	A more comprehensive method to create and manage relationships among objects in IBM Case Manager is to extend object classes to store association properties.

5.5.3 Using forms to enhance user interfaces and data collection

Forms in IBM Case Manager are used to extend the data collection capabilities available. Form templates can be designed to enrich the user experience in interacting with information to allow improved user productivity and provide stricter data validation.

IBM Case Manager supports both FileNet eForms and IBM Forms. Form templates take advantage of the available forms design, templates, validation, and runtime scripting capabilities. This section outlines these capabilities. For more information about using these forms products with IBM Case Manager, see 10.8, “Using the Case Form widget” on page 389 and 10.9, “Using IBM Forms with IBM Case Manager” on page 402.

Remember: Each form product has its own unique user interface functions and features. For example, IBM Forms supports dynamic forms. These forms allow the visibility of form fields, controls, and sections to be controlled based on pre-filled data and field value entries such as property information.

Forms in IBM Case Manager provide these functions:

- ▶ Forms used as a user interface for data entry in Case Manager Client
- ▶ Form policy-based operations to create/edit form data objects in cases
- ▶ Form policy-based operations to create cases
- ▶ Forms as step processors in custom workflow processes

Forms used for data entry in Case Manager Client

In the IBM Case Manager Case Manager Client interface, forms are configured with the Case Form widget. Forms can be used for the following purposes:

- ▶ Entering case properties when manually adding a case
 - Using a form template when you add a case can greatly simplify and improve the data entry and user interface.
- ▶ Modifying case properties in a case details page
- ▶ Entering/modifying step properties as part of a workflow step
 - Forms that are used in this manner can be configured to store the generated form data in the case. Storing data is useful in situations where you need to keep a history of property information changes.
- ▶ Entering launch properties for a discretionary, user-created task

Form policy-based operations to create or edit form data objects in cases

Form data objects that are filed in a case folder open with their associated form template, allowing users to review and update the form data. Form data that are stored in form data objects are stored as XML objects in the target object store.

Form data objects can be extended to provide more business object functionality by extending the form data object class. This extended object class can be configured to use case properties, such as enabling normalized data collection in the case. The extended form data object class can be added to the solution to allow you to define its uses in the solution. This method is useful for collecting structured data objects to be used for analysis and reporting purposes.

Form policy-based operations to create cases

There are essentially two techniques that can be used to create cases by using a form policy:

- ▶ Using a form document policy, a form can be used to create a form data object. This object is designed as the “Starting document type” for a case type in the solution.

- ▶ Using a form workflow policy, a form can be used to start a workflow process that runs any number of process operations. These operations can include the creation of the case folder by, for example, having a step in the process that calls the `createFolder` operation in `CE_Operations`.

Tip: These techniques do not require users to be logged in to IBM Case Manager to create cases. For example, links to these form policies can be published to a portal where users are able to complete a form to start the case.

Forms as step processors in process workflows

For process definitions that use *Form HTML (FileNet)* as the step processor, the Case Manager Client automatically opens the form defined in the form workflow policy for that step. This method can benefit from the features enabled by form workflow policies that might not be available in the Case Manager Client step processor pages. These features include the ability to quickly configure the presentation of process milestones and the ability to manage the presentation of form tabs.

5.5.4 Configuring the security definitions

The configuration of case security uses security modeling that is part of the FileNet P8 Platform.

There are three aspects of security definitions to consider when you set up your production environment.

- ▶ Security on the case management objects that are controlled by Content Engine
- ▶ Security for workflows on Process Engine queues, rosters, and application spaces
- ▶ Access permission to changing spaces and pages in Business Space

For more information, see the IBM Case Manager Information Center at:

<http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/index.jsp?topic=%2Fcom.ibm.casemgmt.design.doc%2Facmdc001.htm>

Also, see 10.3, “Manipulating pages in business space” on page 352.

Tip: In general, use class security and inheritance to manage the security of the cases.

Setting the appropriate security on the case type class, case folder class, and task classes controls who has permissions to those objects. These permissions include creating cases, accessing case information, creating tasks, and other actions.

Table 5-5 provides a few requirements that are related to security and how they can be met.

Table 5-5 Common uses in application design

Usage	Description
Changing the default security on a case type	Each case type is defined by a corresponding case folder class in the target object store. This class can be modified to deny users and groups permission to the case instances. Modify the default instance security of the case folder class with the FileNet Enterprise Manager tool.
Task authorization	Similar to the previous requirement, each task definition is defined by a corresponding task object class in the target object store. Modify this class to deny users and groups permission to the task instances. Modify the default instance security on the task object class with the FileNet Enterprise Manager tool.
Creating a restricted page	Pages that are defined in Business Space can be configured with permissions to allow only select users/groups to access them.

If users must control security, marking sets can be used. Each value in a marking set is called a marking. Each marking is assigned a security template that defines who can set and change that marking, and who is exempt from the restriction of the marking. For more information about using marking sets, see the IBM Information Center for FileNet P8 at:

<http://publib.boulder.ibm.com/infocenter/p8docs/v5r1m0/topic/com.ibm.p8.security.doc/p8psa057.htm>

Tip: Using marking sets limits migration. If you migrate the solution across a domain, you must redo the marking sets for the target environment.

5.5.5 Configuring pages for advanced user interfaces

Both business analysts and solution developers are able to create and modify pages by adding and wiring new iWidgets to pages. The solution developer uses the client interface to directly configure pages. Using the client interface allows the developer to quickly and easily change and test their changes. Include the business analyst during this process to provide immediate feedback.

For more information about customizing the user interface, see Chapter 10, “User interface” on page 335.

Table 5-6 provides the common uses of pages in application design.

Table 5-6 Common uses in application design

Usage	Description
Displaying related/linked cases	To display related or linked cases to the currently open/selected case, the Case List widget can be added to the page. The Case List must be wired with a widget that provides the object payload to run the search. This wiring capability is a simple method to display related cases to the current case. This method is useful when there are other cases that have a defined relationship with the current case.
Scripted URL addressing	In many situations, case property values can be used to construct URLs in GET and POST operations to other websites. This method is a simple way to present information from other websites to the user interactively on the Business Space page.

5.5.6 Configuring decision services with iLog JRules

Using the Process Designer, the solution developer can configure the execution of decision services in the iLog JRules rules engine. This execution is configured by using the partner link, web services function available in Process Designer. In addition, IBM Forms can also call JRules rules services for validation and for dynamic forms capabilities. For more information, see Chapter 14, “Integration with WebSphere ILOG JRules” on page 523.

Table 5-7 provides the common uses of the decision services with iLog JRules in application design.

Table 5-7 Common uses of decision services in application design

Usage	Description
Case scoring	During the execution of a workflow process that is associated with a task, case property information can be used to call a decision service. This service then “scores” the case or related information in the case. For example, based on information in the case, the urgency of the case can be scored and used to prioritize work in the in-baskets.

Usage	Description
Initiate a task	While running a workflow process associated with a task, case property information can be used to call a decision service to populate other case properties. The property values returned by the decision service can be mapped to one or more tasks that are initiated with property-based preconditions.
Complete work	In some process, user decisions are based on an evaluation of case property structured information. Consider using a decision service rules invocation to automate the completion of that work.
In an IBM Form	IBM Forms used in IBM Case Manager can be configured to use decision rule services from iLog JRules for validation and other dynamic forms capabilities. These capabilities include expanding the form to request more information.

5.5.7 Configuring case analytics and reporting

Case analyzer manages the data that are used in providing analytics and reports for IBM Case Manager solutions. Case, task, and workflow information is collected from logs and added to an analytics database. Reporting tools can then use this database to generate both in-progress and historical information. Case analyzer processes this information into cubes and fact tables.

Cubes

Cubes are multi-dimensional tables of information. The types of cubes that are managed by Case Analyzer are listed in Table 5-8.

Table 5-8 Types of cubes that are supported by Case Analyzer

Cube	Description
Case load	Stores historical information about the number of cases in process during a specified time period.
Cases in progress	Stores nearly up-to-the-moment information about all cases currently in progress in the system. The information includes the number of cases, processing times, and status.
Task load	Stores historical information about the number of tasks in process during a specified time period.
Tasks in progress	Stores nearly up-to-the-moment information about all tasks currently in progress in the system. The information includes the number of tasks, processing times, and status.

Cube	Description
Queue load	Stores historical information about the amount of work that is done in each queue during a specified time period.
Work in progress	Stores nearly up-to-the-moment information about all work currently in the system. The information includes the number of work items, processing times, and status. The information can be broken down by workflow or work item, and by user.
Work item processing time	Stores historical information about processing time for work items during a specified time period. The information can be broken down by user.
Routing	Stores historical information about the number of work items going from a source step to a destination step. The information can be broken down by user, time, and exposed user-defined fields.
Work load	Stores historical information about the amount of work that is done in the system during a specified time period.
Workflow in progress	Stores nearly up-to-the-moment information about all work currently in the system. The information includes the number of work items, processing times, and status. The information can be broken down by workflow or work item.

Reporting using cubes enables reports to be “sliced and diced” and allowing users to drill down to obtain more detail. Cubes are generally used to supply summarized information about cases, tasks, and workflow information.

IBM Case Manager includes reports that are based on Microsoft Excel. These standard reports provide a basis for configuring new reports as required.

Fact tables

Fact tables are single-dimensional tables of information. Using fact tables, you can generate reports that contain detailed case information. Reports list all cases, tasks, or workflows, and can group, total, and provide averages using tabular calculations in the report itself. The fact tables do not summarize the data directly from multiple cases/tasks/workflows.

Exposed field

An exposed field in the reporting database is either a case property or a workflow data field. Each cube (fact) table can have zero or more exposed fields. Exposing fields allows case properties to be made available for reporting purposes.

Configuring a solution for reporting involves the creation of exposed fields and the setup of reports. This process can use Cognos BI, Microsoft Excel, or other tools that can generate reports from a reporting database.

In the Complaint Management solution example, the following case properties might be exposed for reporting: Complaint Type, Received Date, and Customer number. Cognos BI can be used to generate graphical reports that show the frequency of complaints submitted by Complaint Type.

Table 5-9 provides the common uses of reports in application design.

Table 5-9 Common uses of reports in application design

Uses	Description
Generating detailed reports on cases in progress	Case properties to be used in the reports are set up as exposed fields. These exposed fields are then added to the Case In Progress fact tables. Reports are generated from the fact tables.
Generating detailed reports on completed cases	Case properties to be used in the reports are set up as exposed fields. These exposed fields are then added to the Case Load fact tables. Reports are generated from the fact tables.
Generating detailed reports on tasks or workflows in progress	Properties and data fields to be used in the reports are set up as exposed fields. These exposed fields are then added to the task/workflow in progress fact tables. Reports are generated from the fact tables.
Generating graphical, historical management reports	Properties and data fields to be used in the reports are set up as exposed fields. These exposed fields are then added to the cubes that are used for the management reports. Reports are generated from the cubes.
Storing reports in cases	Reports that are in object format, such Microsoft Excel/Word or a PDF, can be stored in the target object store as documents and opened by users. These documents can be generated and viewed, for example, in situations where decision-making on specific cases depends on status/management reports on other cases.

5.5.8 Configuring the Cognos Real-Time Monitor

Real-time monitoring allows you to present key performance indicators that are used by line-of-business managers. Dashboard objects can display summary and detailed performance information based on business matrixes. For example,

dashboards objects can be configured to show the number of cases that are created over a specific period.

Because real-time monitoring uses the same cubes and fact tables that are used in reporting, similar procedures are used to extend the reporting database to display case, task, and workflow information. Dashboard objects can then be configured to display more information as required.

Table 5-10 lists common uses of real-time monitor in application design.

Table 5-10 Common uses of real-time monitor in application design

Uses	Description
Modifying dashboard objects with case properties	As in reporting, case properties that are used in the dashboard objects must be set up as exposed fields. These exposed fields are then added to the Case In Progress cube. Dashboard objects are modified or created by using the cube (summary dashboard objects) or fact tables (detailed dashboard objects).
Create watch point email notifications	Watch points enable line-of-business managers to receive emails when a threshold is reached, such as when more than the expected number of cases are generated. Watch points can be configured directly on the real-time monitor dashboard objects.

5.6 Level 3: Customizing and integrating your solution

IBM Case Manager provides flexible configuration options at both Level 1 and Level 2 to accommodate a broad range of solution requirements. However, often you must build extra functions that are beyond the scope of what is available with the product. For customized application building, IBM Case Manager includes an extensive set of application programming interfaces (APIs) that expose all the functions through various levels. Programmers can select the most appropriate one to meet their needs.

Use the published APIs with IBM Case Manager to ensure that the code is compatible with future updates and upgrades. Using these APIs allows the software to be kept up-to-date without unnecessary costs and risks.

For more information about the APIs, see Chapter 13, “Integration points” on page 511.

Table 5-11 highlights the various APIs and interfaces available within IBM Case Manager.

Table 5-11 APIs and interfaces available in application design

API / interface	Description
Case Manager REST API	<ul style="list-style-type: none">▶ A Representational State Transfer interface typically used to develop advanced Web 2.0 user interfaces▶ Can be used for services within a service-oriented architecture
Process Engine REST API	<ul style="list-style-type: none">▶ A Representational State Transfer interface typically used to develop advanced Web 2.0 user interfaces▶ Can be used for services within a service-oriented architecture
CMIS REST API	<ul style="list-style-type: none">▶ A Representational State Transfer interface to the open Content Management Interoperability Services (CMIS) standard▶ Typically used to develop advanced Web 2.0 user interfaces▶ Can be used for services within a service-oriented architecture
Process Java API	<ul style="list-style-type: none">▶ Java-based API▶ Typically used to develop integrations with other systems, custom component queue operations, and custom step processors
Content Java API	<ul style="list-style-type: none">▶ Java-based API▶ Typically used to develop custom user interfaces, integrations with other systems for content ingestion, searching and extraction, custom event actions, and content batch processes
Content .NET API	<ul style="list-style-type: none">▶ API based on .NET▶ Typically used to develop custom user interfaces; integrations with other systems for content ingestion, searching, and extraction; and content batch processes
Process Engine Web Services API	<ul style="list-style-type: none">▶ Web Services-based API▶ Typically used to develop integrations with other systems where Java or .NET development environments are not available

API / interface	Description
Web Services in Process Workflows	<ul style="list-style-type: none"> ▶ Web Services-based interface ▶ Process workflows that implement <i>Receive Web Service</i> steps to publish an associated WSDL. ▶ Process steps can be started by other systems (both launch and in-progress steps) ▶ Can be used as services within a service-oriented architecture
Content Engine Web Services API	<ul style="list-style-type: none"> ▶ Java-based API ▶ Typically used to develop integrations with other systems where Java or .NET development environments are not available
Content Engine SQL	<ul style="list-style-type: none"> ▶ SQL-based language interface ▶ Enables access to the content metadata by using SQL-like language ▶ Typically used in programs to locate or enumerate information in an object store for search and reporting purposes
FileNet eForms JavaScript	<ul style="list-style-type: none"> ▶ JavaScript-based API ▶ Typically used to develop enhanced forms that are used for data entry ▶ User interface applications
IBM Sametime Software Development Kit	<ul style="list-style-type: none"> ▶ C/C++ software development kit ▶ Enables the extension of capabilities of the Sametime client used in IBM Case Manager
IBM Forms Server API	<ul style="list-style-type: none"> ▶ Provides low-level access to XFDL forms in Java, C, and COM ▶ Develop applications that analyze, validate, and create XFDL forms ▶ Functional Call Interface for runtime custom XFDL capabilities
External Data Services	<ul style="list-style-type: none"> ▶ Interface between cases and other information sources ▶ Loads case data at query time ▶ Enables real-time presentation and storage of case data

API / interface	Description
Business Space powered by WebSphere	<ul style="list-style-type: none"> ▶ Business space environment ▶ iWidgets and applications with web-based interfaces can be wired together to enable advanced user interactions ▶ Integration is transient and data is interchanged only when needed by the iWidgets in real time
Component Integrator	<ul style="list-style-type: none"> ▶ Automated work processing that is performed by Java object or Java Message Service (JMS) queue ▶ Runs functions from within a workflow ▶ Extends process functionality without full application deployment

5.7 Documenting your solution

Case Manager Builder enables the solution designer to effectively capture requirements directly into the tool. A document can then be generated by using the available solution description generation tool that contains all the configuration data within your solution. This solution description document provides a way to communicate and present the configuration details of the solution for review, discussion, and analysis.

The Solution Document Generator is a command-line tool. For information about how to use it, see the IBM Information Center on Case Manager at:

<http://publib.boulder.ibm.com/infocenter/casemgmt/v5r1m0/topic/com.ibm.casemgmt.installng.doc/acmde006.htm>

Figure 5-3 shows an example of the documentation generated.

Folders				
The following folder structure was defined for this view:				
<table border="1"><tr><td>CC_Complaint Subdirectories:</td></tr><tr><td>Correspondence</td></tr><tr><td>Supporting Documents</td></tr></table>		CC_Complaint Subdirectories:	Correspondence	Supporting Documents
CC_Complaint Subdirectories:				
Correspondence				
Supporting Documents				
Tasks				
The following tasks were created for this case:				
Task	Detail			
Display Name: Verify Complaint Unique Name: CC_VerifyComplaint Description This process is executed to establish whether the complaint is valid or invalid.	Task Starts: automatic Required: true Repeatable: false User-Creatable: false			
Display Name: Review Product Complaint Unique Name: CC_ReviewProductComplaint Description All product complaints are handled with this task.	Task Starts: automatic Required: false Set Name: Complaint Type Set Type: exclusive Property Precondition: CC_Valid = True AND Property Precondition: CC_ComplaintCategory = Product Repeatable: false User-Creatable: false			

Figure 5-3 Generated documentation



Building a simple solution: Part 1

This chapter and the one that follows describe how to build a simple solution. The use case that was introduced in 2.2, “Complaints management example use case” on page 25 is used to illustrate a step-by-step procedure for building your solution.

Later chapters build on the use case to show how to introduce more advanced features in the solution. These features include case splitting, reuse BPM processes, and integration to other systems and components.

This chapter includes the following sections:

- ▶ Building the sample solution overview
- ▶ Artifacts to be implemented in the sample solution
- ▶ Creating the Customer Complaint solution
- ▶ Setting up and configure artifacts for the solution
- ▶ Setting up tasks for the solution

6.1 Building the sample solution overview

The Customer Complaints solution is a simple solution that is intended to show as much IBM Case Manager functionality as possible. However, the solution is kept simple for easier reading and understanding. The Customer Complaints solution demonstrates the following IBM Case Manager functions:

- ▶ Using smaller task-based processes
- ▶ Using container tasks
- ▶ Using automatic, required, and manual tasks
- ▶ Using mutually exclusive task sets
- ▶ Using document-based preconditions
- ▶ Using property-based preconditions
- ▶ Using preconditions based on property updates

Building the simple Customer Complaints solution has the following basic steps:

1. Create a simple Customer Complaints solution by using create solution wizard from Case Manager Builder.
2. Setting up and configure artifacts for the solution:
 - Setting up properties.
 - Setting up roles.
 - Setting up in-baskets
 - Setting up document types.
 - Setting up case type.
3. Set up tasks.
4. Configure tasks process diagrams for the tasks by using Case Manager Builder Step Editor.
5. Save and validate the solution.
6. Deploy and test the solution.

6.2 Artifacts to be implemented in the sample solution

The simple Customer Complaints solution presented in this chapter is based on the Customer Complaints use case described in 2.2.1, “Complaint scenario” on page 26. The process that is used to design a case management solution for the Customer Complaints use case is described in Chapter 3, “IBM Case Manager overview” on page 31.

6.2.1 Case types

The simple Customer Complaints solution has a single case type titled Complaint. The properties for the Complaint case type are specified in Table 6-5 on page 164.

The following sections describe the various tasks, roles, document types, and properties that are required for the Complaint case type.

6.2.2 Tasks

The tasks that are required for the Complaint cases are listed in Table 6-1.

Table 6-1 Customer Complaint solution tasks

Task	Attributes	Description
Upsell Opportunity	Required, Automatic, Container, Property precondition	This container task starts the subtasks contained in it when there is a possibility of an upsell opportunity
Upgrade Product	Required, Automatic, Exclusive, Property precondition	This task starts automatically when there is a product-related upsell opportunity
Upgrade Plan	Required, Automatic, Exclusive, Property precondition	This task starts automatically when there is a non-product related upsell opportunity
Call Customer	Required, Automatic	This task starts automatically when there is an upsell opportunity
Verify Billing	Optional, Automatic, Filing, Property preconditions	This task starts automatically when a customer submits a supporting document for billing related complaints
Verify Complaint	Required, Automatic, Repeatable, Property update precondition	This task starts automatically and is repeated every time the complaint category property is changed
Review Product Complaint	Optional, Automatic, Property precondition	This task starts automatically when a property condition is met to handle all product complaints.
Review Non-Product Complaint	Optional, Automatic, Property precondition	This task starts automatically when a property condition is met to handle all non-product complaints.

Task	Attributes	Description
Investigate Product Safety	Optional, Manual, Property precondition	This task can be started at anytime to perform a product safety investigation.
Send Corresponding Letter	Required, Automatic, Document Filing, and Property preconditions	This task starts automatically when a case worker files a corresponding letter into a case. After the letter is reviewed, it is sent to the customer through a third-party letter generation software.
Close Complaint	Required, Manual	This task starts manually when a complaint needs to be closed.
Investigate Employee	Discretionary	This discretionary task can be started one or more times to investigate one or more employees.
Request Assistance	Discretionary	This discretionary task can be started one or more times to request help to handle the complaint.

6.2.3 Roles

The Roles that are identified for processing the *Complaint* cases are described in Table 6-2.

Table 6-2 Customer Complaint solution roles

Role	In-basket	Description
Contact Center	Contact Center	These workers handle calls and email complaints.
Investigator	Investigator	This role performs detailed investigation on complaints when required.
Manager	Manager	This role is for the manager of the complaint Contact Center workers.
Specialist	Specialist	Members of this role handle product-specific complaints.
Billing Agent	Billing Agent	These workers handle billing-specific complaints.
Sales Agent	Sales Agent	This role is for the sales agents who interact with customers for any sales opportunities.

6.2.4 In-baskets

The in-baskets are specified in Table 6-3.

Table 6-3 In-basket definition

In-basket	Columns	Filter
Complaints	Complaint Received Date (sortable, sort default), Customer Rating (sortable), Complaint Category (sortable), Case Number, Complaint Status, Customer Name, Customer Number	Complaint Category Equal, Complaint Receive Date Less than, Complaint Status Equals, Customer Name Like
Investigator	Case Number	
Manager	Case Number	
Specialist	Case Number	
Billing Agent	Case Number	
Specialist	Case Number	
Sales Agent	Case Number, Customer Name, Customer Address, Customer Telephone, Customer Email, Customer City, Customer Rating	Customer Name Like, Case Number Equal
My Work (Common Personal in-basket)	Complaint Received Date, Case Number, Complaint Status, Complaint Category, Customer Name, Customer Number, Customer Rating	
My Work (Personal in-basket for Manager Role)	Case Number, Complaint Category, Complaint Description, Complaint Received Date, Complaint Status, Customer Name, Customer Number, Customer Rating, Customer Since, Part Number, Safety Check, Upsale Opportunity	Case Number Equal, Customer Name Like, Complaint Received Date Less than

6.2.5 Documents

Documents that are used in the Complaint cases are noted in Table 6-4.

Table 6-4 Customer Complaint solution document types

Document type	Properties	Description
Correspondence	Case Number	Document sent to customer to communicate the complaint status
Supporting Document	Case Number	Documents received from customer to support customer complaints

6.2.6 Properties

The properties that are required to process the Complaint cases are listed in Table 6-5.

Table 6-5 Complaint solution properties

Property	Type	Description
Case Number	String(10)	Auto-generated case number
Case Source	String(64), Choice list: Case Sources	The channel that the complaint was received on.
Complaint Category	String(64), Choice list: Complaint Categories	The type of complaint submitted.
Complaint Description	String(1000)	The description of the complaint submitted.
Complaint Received Date	DateTime	The date and time the complaint is received.
Complaint Status	String(64), Choice list: Case States	The status of the complaint
Customer Name	String(64)	The first and surname of the customer, obtained from the CRM system
Customer Number	String(10)	The customer number used to query the CRM system
Customer City	String(64)	Customer city, obtained from the CRM system

Property	Type	Description
Customer Address	String(200)	Customer street name, obtained from the CRM system
Customer Rating	String(64)	Calculated value that indicates how the customer is rated
Customer Since	Integer	The number of years the customer has been with the company.
Customer State	String(64)	Customer state, obtained from the CRM system
Customer Email	String(64)	The email address of the customer, obtained from the CRM system
Part Number	String(64)	The part number of the product specified in the complaint
Safety Check	Boolean	This property is set to true if a safety check is required.
Total Transaction Amount	Float	The transaction amount that is filed in the complaint
Telephone	String(64)	The customer telephone number, obtained from the CRM system
Valid	Boolean	Set to true if the complaint is a valid complaint.
Upgrade Category	String(64), Choice list: Upgrade Category	The type of upgrade requested
Upsale Opportunity	Boolean	Set to true if there is a possibility of a product or non-product upsell.

6.2.7 Property choice lists

The choice lists for the simple Complaints solution are listed in Table 6-6.

Table 6-6 Property choice list

Choice list	Choice Item
Case Sources	Form, Email, Fax, Letter
Case States	Open, Closed, Processing, Pending, On Hold
Complaint Categories	Product, Billing, Service, Other
Upgrade Category	Product, Service Plan

6.3 Creating the Customer Complaint solution

Complete these steps to add the Customer Complaints solution by using the Case Manager Builder wizard:

1. Log in to Case Manager Builder at `http://server:port/CaseBuilder` as a user with sufficient rights to create the case solution (Figure 6-1 on page 167).

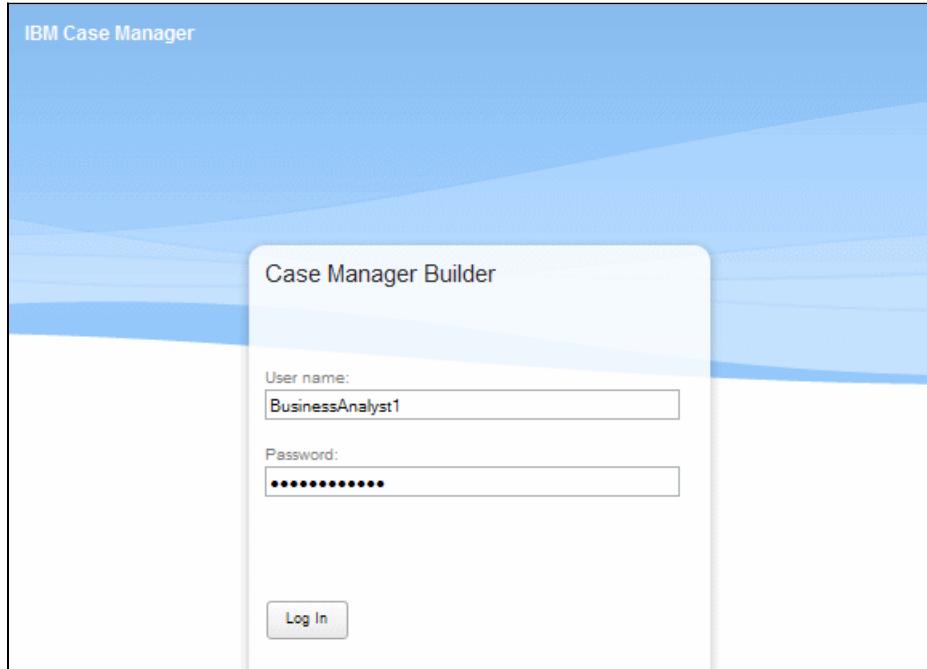


Figure 6-1 Logging in to Case Manager Builder

2. In the Manage Solutions window, select **Use the wizard to define the solution** as shown in Figure 6-2, and click **Add Solution**.

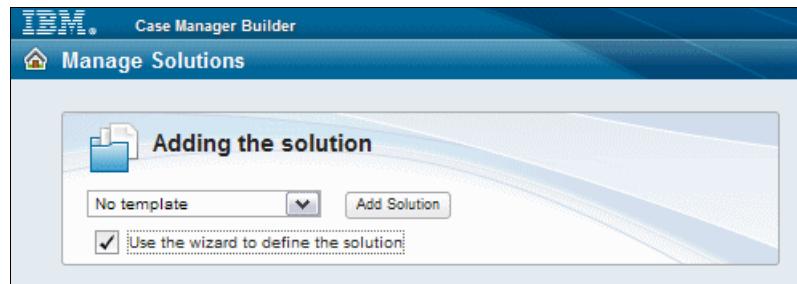


Figure 6-2 Creating a solution by using the wizard

3. Name the solution:
 - Enter the name of the solution Customer Complaints in the **Name** text box.
 - Enter a two to five alphanumeric character prefix that starts with an alphabetic character in the **Solution prefix** text box.

This sample solution uses the prefix CC. A unique solution prefix is required to prefix all object types used in the solution.

- Select the icon for the solution. The solution icon represents the solution type such as Aerospace, Energy, or Financial. IBM Case Manager is bundled with a set of default solution icons.

Figure 6-3 shows the results of the preceding selection.

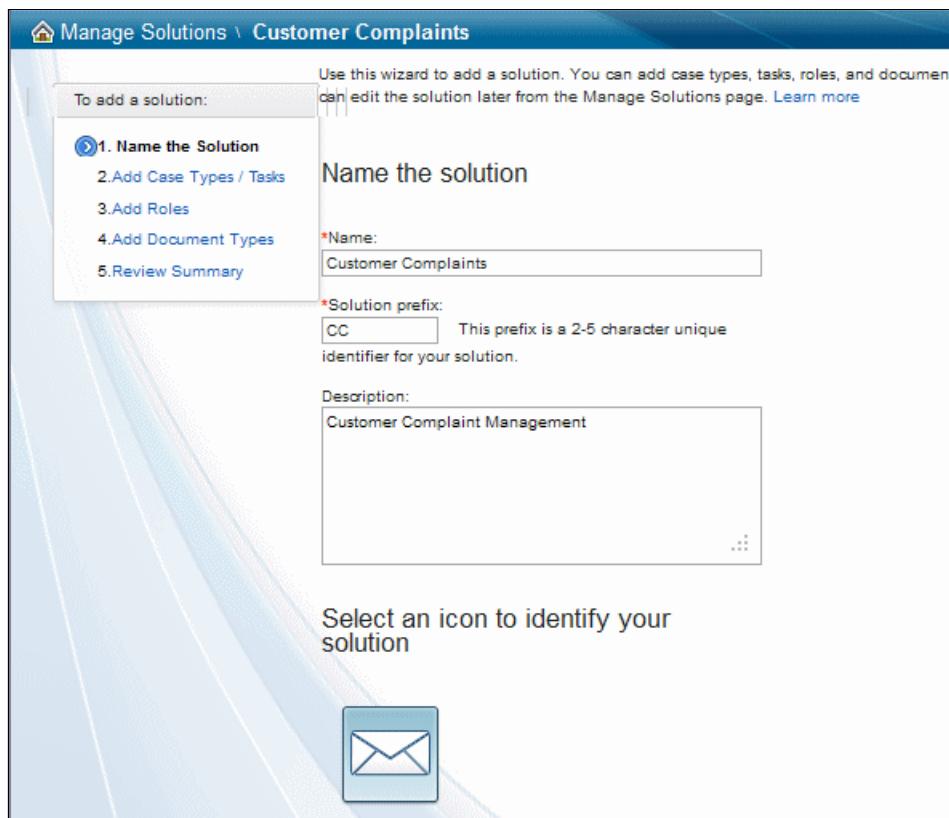


Figure 6-3 Naming the solution

4. Click **Add Case Types**. A warning that the name and prefix cannot be changed after this point is displayed. Click **OK** to continue.
5. Enter case type information:
 - a. Type the name of the Case Type, in the example Complaint, in the **Case type name** text box.
 - b. Enter case type description in the **Case type description** text box.
 - c. Click **Save Case Type**.

Figure 6-4 shows the Complaint case type definition.

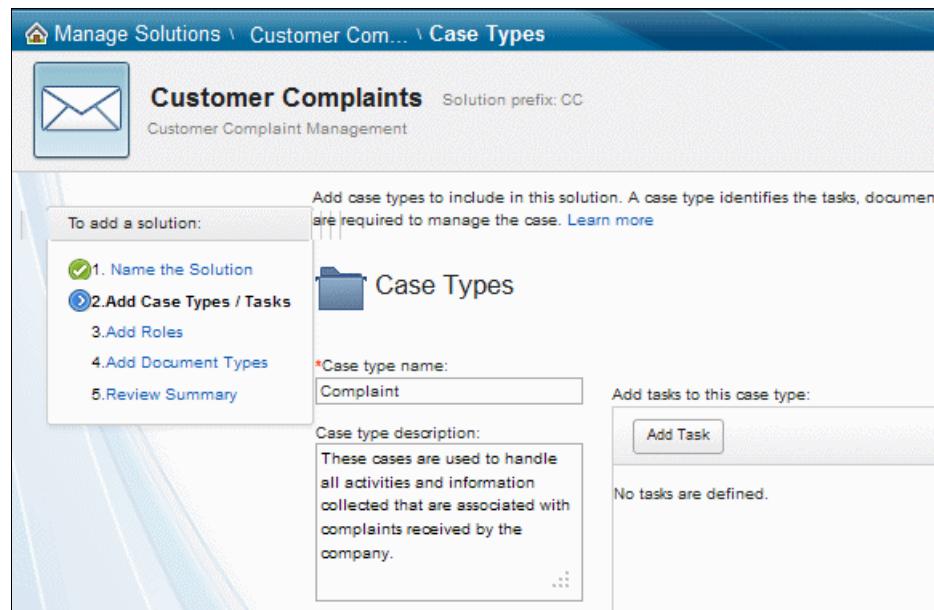


Figure 6-4 Complaint case type definition

6. Add tasks to the Complaint case type of the Customer Complaints solution by using the following steps:
 - a. Click **Add Task** as shown in Figure 6-5.

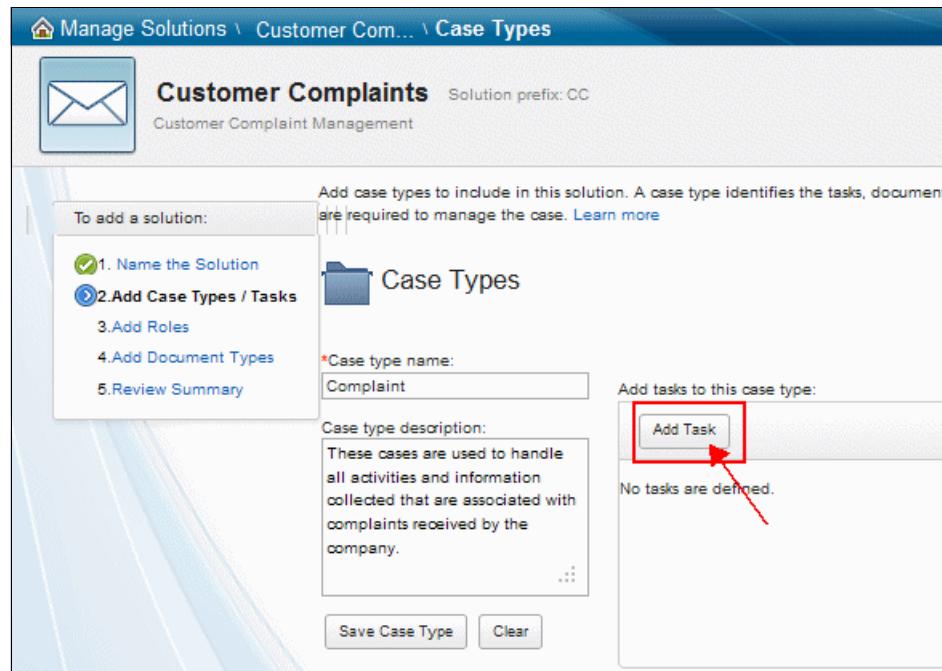


Figure 6-5 Adding tasks to the case type

- b. Enter Close Complaint and a task description in the **Name** and **Description** boxes as shown in Figure 6-6. Click **OK** to add the task.

This is a dialog box titled 'Add tasks to this case type:'. It contains a 'Add Task' button. Below it, there are two input fields: 'Name:' containing 'Close Complaint' and 'Description:' containing 'The task is executed when a complaint is to be closed.'. At the bottom right are 'Add a Comment' and 'OK' buttons.

Figure 6-6 Entering task name and description

c. Click **Add Task** to add extra tasks as shown in Figure 6-7.

The dialog box has a title bar 'Add tasks to this case type:' and a 'Add Task' button. It contains fields for 'Name' (with a required asterisk) and 'Description'. Below these are three task entries: 'Close Complaint' (task is executed when a complaint is to be closed), 'Verify Complaint' (starts automatically), 'Send Corresponding Letter' (starts after review), and 'Request Assistance' (an adhoc task). There are 'Add a Comment' and 'OK' buttons at the bottom.

Figure 6-7 Adding extra tasks

d. Add the following tasks using the information from Table 6-1 on page 161:

- Investigate Product Safety
- Investigate Employee
- Request Assistance
- Review Non-Product Complaint
- Review Product Complaint
- Send Corresponding Letter
- Verify Complaint
- Call Customer
- Upgrade Product
- Upgrade Plan
- Verify Billing

Click **OK** after you add each task. A complete list of tasks like the one in Figure 6-8 is displayed.

The dialog box shows a 'Case Types' section with a 'Case type name' field containing 'Complaint' and a 'Case type description' field with a detailed explanation of what these cases handle. To the right is a 'Add tasks to this case type:' section with the same four tasks listed as in Figure 6-7: 'Close Complaint', 'Verify Complaint', 'Send Corresponding Letter', and 'Request Assistance'. There are 'Save Case Type' and 'Clear' buttons at the bottom.

Figure 6-8 Adding tasks complete

Exception: If you do not click **OK** after each added task, a window is displayed as shown in Figure 6-9. The window asks you if you want to apply or discard your changes. Click **Apply**.

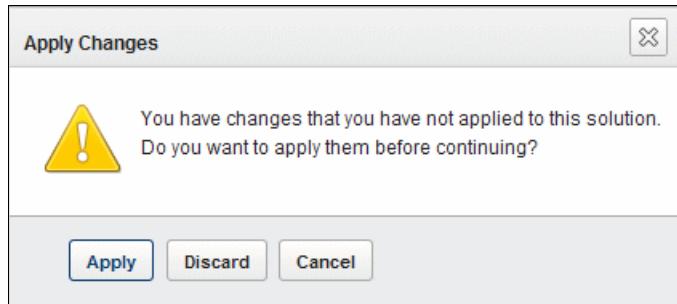


Figure 6-9 Apply Changes window

- e. Click **Save Case Type** to continue.
7. Add roles by completing the following steps:
 - a. Click **Add Roles** as shown in Figure 6-10 on page 173.

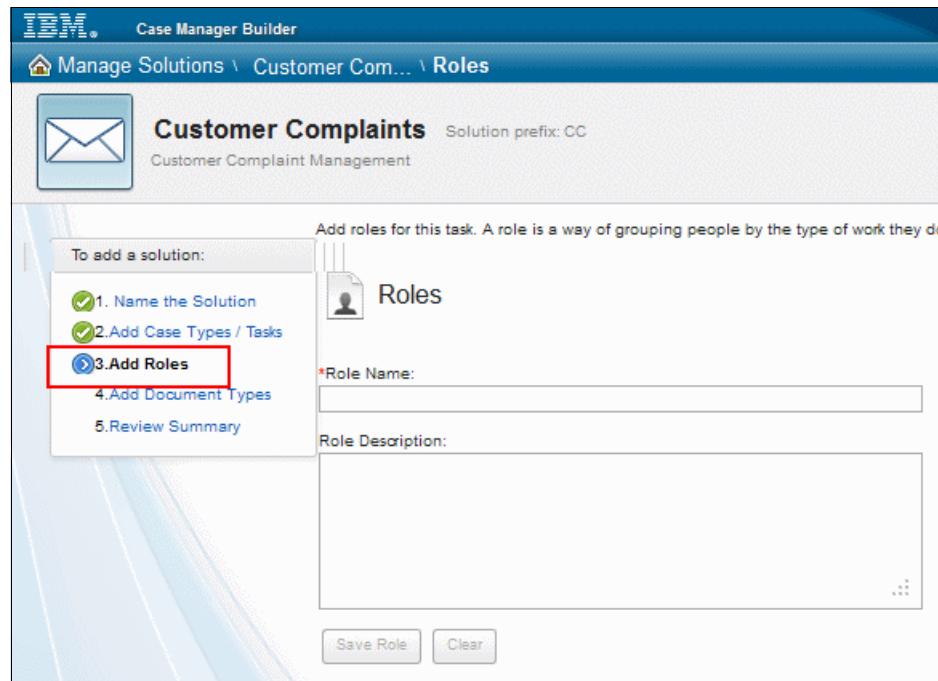


Figure 6-10 Clicking the Add Roles link

- b. Enter Contact Center as the **Role Name** and a description in the **Role Description** text box, then click **Save Role**.

Restriction: The role name must be 55 characters or less, and cannot start with the tilde (~) special character. The role name cannot use the same name as used for a task in the same solution.

- c. Add more roles for the following using the information from Table 6-2 on page 162:
- Investigator
 - Manager
 - Specialist
 - Billing Agent
 - Sales Agent

The complete list of roles looks similar to the list shown in Figure 6-11.

The screenshot shows a software interface for managing roles. On the left, a sidebar lists steps: 'To add a solution:' followed by numbered items: 1. Name the Solution (checkmark), 2. Add Case Types / Tasks (checkmark), 3. Add Roles (radio button selected), 4. Add Document Types, and 5. Review Summary. The main area has a title 'Add roles for this task. A role is a way of grouping people by the type of work they do, f...' and a sub-section titled 'Roles'. It contains fields for 'Role Name:' (with a placeholder 'Role Name') and 'Role Description:' (with a large text area). Below these are 'Save Role' and 'Clear' buttons. To the right, there is a list of four roles with icons: 'Contact Center' (described as handling calls and email complaints), 'Investigator' (described as performing detailed investigation on complaints when required), 'Manager' (described as for the manager of the complaint contact center workers), and 'Specialist' (described as members of this role handle product specific complaints).

Role	Description
Contact Center	These workers handle calls and email complaints.
Investigator	This role performs detailed investigation on complaints when required.
Manager	This role is for the manager of the complaint contact center workers.
Specialist	Members of this role handle product specific complaints.

Figure 6-11 Roles are complete

8. Add document types by completing the following steps:
 - a. Click **Add Document Types** as shown in Figure 6-12.

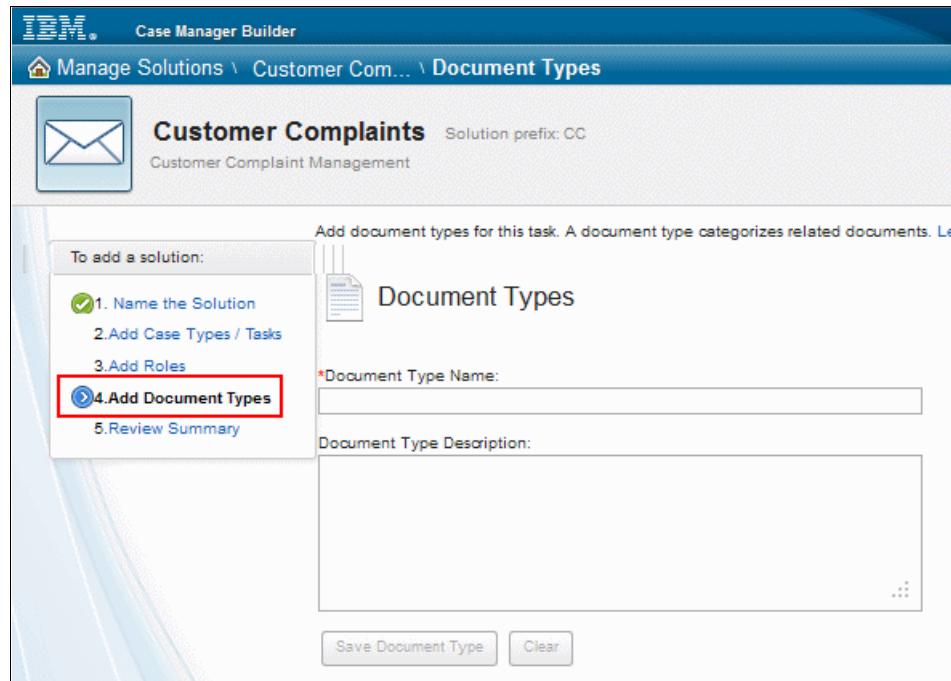


Figure 6-12 Adding document types

- b. Enter Correspondence for the **Document Type Name** as shown in Figure 6-13.

The screenshot shows the IBM Case Manager Builder interface. At the top, it says "IBM Case Manager Builder" and "Manage Solutions \ Customer Com... \ Document Types". On the left, there's a sidebar titled "To add a solution:" with numbered steps: 1. Name the Solution (checkmark), 2. Add Case Types / Tasks, 3. Add Roles, 4. Add Document Types (blue circle with arrow), and 5. Review Summary. The main area is titled "Customer Complaints" with "Solution prefix: CC" and "Customer Complaint Management". It has a sub-section titled "Document Types" with the following fields:

- *Document Type Name: Correspondence
- Document Type Description: Document sent to customer to communicate the complaint status

At the bottom of this section are "Save Document Type" and "Clear" buttons.

Figure 6-13 Adding the Document Type Name

- c. Click **Save Document Type**.
- d. Add these additional document types using the information from Table 6-4 on page 164:
 - Supporting Document

The complete list of document types is shown in Figure 6-14.

The screenshot shows the IBM Case Manager Builder interface. The title bar reads "IBM Case Manager Builder". Below it, the breadcrumb navigation shows "Manage Solutions \ Customer Com... \ Document Types". The main content area is titled "Customer Complaints" with the solution prefix "CC". A sidebar on the left lists steps: "To add a solution:" followed by numbered steps 1 through 5, with step 4 highlighted as "4.Add Document Types". The main panel has a heading "Document Types" with a sub-section for "Document Type Name" and "Document Type Description". Below this, two document types are listed: "Correspondence" and "Supporting Document".

Figure 6-14 Document types now completed

9. Click **Summary** to continue.

10. Review the summary information as shown in Figure 6-15. Use the **Add** links to add any artifacts you missed. If everything is correct, click **Finish**.

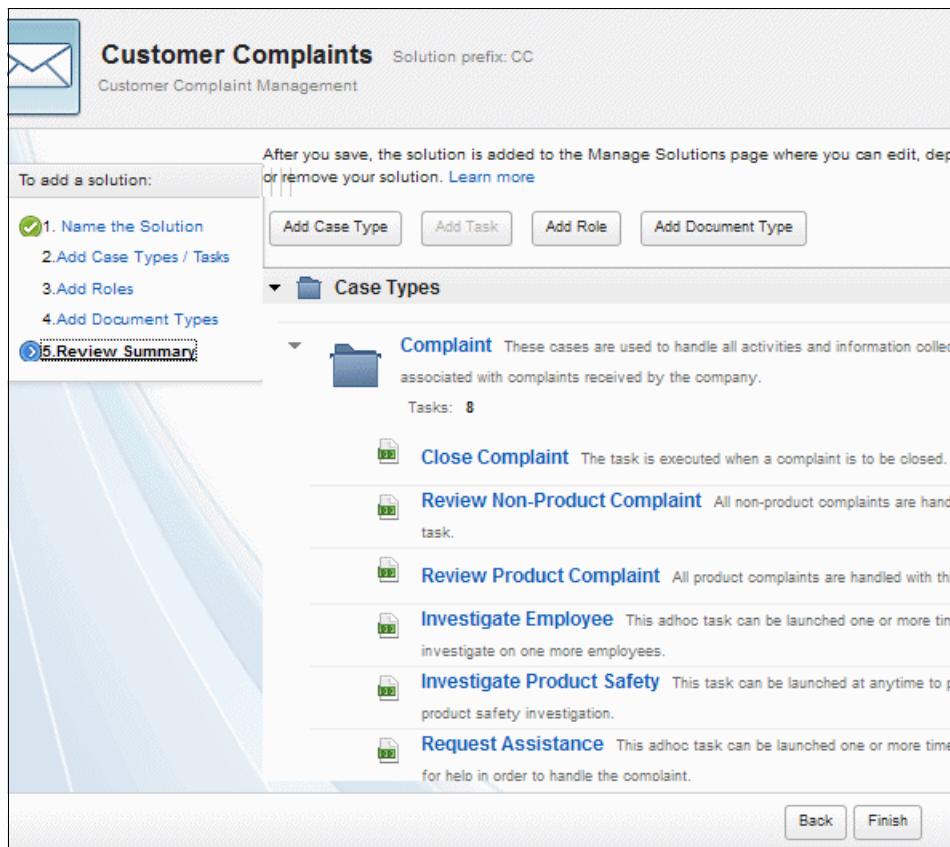


Figure 6-15 Reviewing the summary

6.4 Setting up and configure artifacts for the solution

After you create a solution by using the Create Solution wizard, set up and configure artifacts that are used for the solution. The artifacts include properties, roles, document types, personal in-baskets, and case types.

Tip: While in Edit mode, periodically save your work.

6.4.1 Setting up properties

To set up properties for the solution, complete the following steps:

1. On the Manage Solutions home window, click the **Edit** link that is located under your solution name. Alternatively, you can click the name of the solution directly as shown in Figure 6-16 to open it for editing.



Figure 6-16 Opening the solution for editing

2. Click the **Properties** tab.
3. Click **Add Property** → **New** as shown in Figure 6-17.

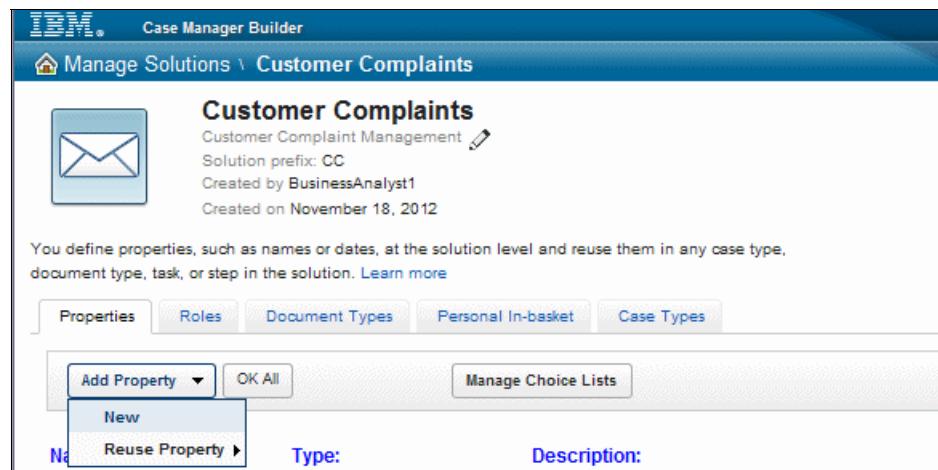


Figure 6-17 Adding properties to the solution

4. Use the information from to Table 6-5 on page 164 to enter the property definition for Address as shown in Figure 6-18.
 - a. For the property **Name** field, enter Customer Address.
 - b. For property **Type**, select **String**.
 - c. Expand **Define Property Values** and enter 200 for **Maximum length**.
 - d. Click **OK** to continue.

Name	Type	Description
*Name:	Type:	Description:
Customer Address	String	Customer street name obtained from the CRM system

Define Property Values

This property can have:	*Maximum length:
<input checked="" type="radio"/> A single value	200
<input type="radio"/> Multiple values	
Default value:	*Unique Identifier
	CC_CustomerAddress

Figure 6-18 Defining the Address property

Tip: The basic property types are string, integer, datetime, float, and Boolean. Properties can be a single value or multiple values. The Customer Complaint solution uses only single values.

Restriction: The Case Manager Builder application does not provide a method for indexing property values. To index a property, you must use the FileNet Enterprise Manager on the target object store to set the index property for simple indexes. For more complex indexes, use the database tool to define them.

Properties that are used as case search criteria typically require indexing to improve performance. Monitor your system during testing and in production to determine the final set of properties to configure for indexing.

For more information, see “Indexing IBM FileNet P8 Content Engine databases” at:

<http://publib.boulder.ibm.com/infocenter/p8docs/v5r1m0/index.jsp?topic=%2Fcom.ibm.p8.performance.doc%2Fp8ppt121.htm>

5. In the same manner, add the rest of the properties that are listed in Table 6-5 on page 164.

For the property with choice list, select your choice in the **Manage Choice Lists Name** text box. For example, define the choice list Case Sources as shown in Figure 6-19 on page 182:

- a. Enter Case Sources in the **Manage Choice Lists Name** text box.
- b. To add a choice, click **Add Choice Item** and enter the choice information. In the example, enter Form for **Display Name** and **Value**.
- c. Repeat these steps for Form, Email, Fax, and Letter.
- d. Click **OK** to complete the choice list.
- e. To add another choice list, click **Add Choice List**.
- f. When you finish creating all the choice lists, click **Close**.

Manage Choice Lists

Add Choice List

Choice List ^

No choice lists are defined.

Choice list type: String

*Name: Case Sources

*Display Name	*Value
Form	Form
Email	Email
Fax	Fax
Letter	Letter

Add Choice Item Clear List

Close

Figure 6-19 Adding a choice list

For a complete list of choice lists, see Table 6-6 on page 166.

- g. After the choice list for the property is defined, select the choice list for the property. In the example, the **Case Sources** choice list is selected for the property Case Source as shown in Figure 6-20.

Name: ^	Type:	Description:
*Name: Case Source	Type: String	Description: The channel that the complaint was received on.
Define Property Values		
This property can have:	<input checked="" type="radio"/> A single value <input type="radio"/> Multiple values	
Default value:	<input type="radio"/> None	
	*Maximum length:	64
	*Unique Identifier:	CC_CaseSource

Figure 6-20 Example of a property with a choice list

Tip: You can select a default value from the choice list. However, the example does not use a default value. Click **Save** after you complete the properties configuration.

The completed property list looks similar to the one shown in Figure 6-21.

Name ^	Type	Description
Case Number	String	Auto-generated case number
Case Source	String	The channel that the complaint was received on
Complaint Category	String	The type of complaint that has been submitted
Complaint Descrip...	String	The description of the complaint that has been submitted
Complaint Receive...	DateTime	The date and time the complaint is received
Complaint Status	String	The status of the complaint
Customer Address	String	Customer street name obtained from the CRM system
Customer City	String	Customer city obtained from the CRM system

Figure 6-21 Completed properties

6.4.2 Setting up roles

After you set up the properties for your solution, create the role and select the type of in-baskets to display for the role. To configure common view for the personal in-basket for the role, complete the following steps (continuing from the previous section).

Remember: When you create a solution by using the wizard, the in-basket name for the role is automatically generated.

On the **Roles** tab, click the **Contact Center** role and select **Personal (Role): Show a custom view for this role** as shown in Figure 6-22.

The screenshot shows a dialog box titled 'Roles'. The 'In-baskets' tab is selected. At the top, there are buttons for 'Add Role' and 'OK All'. Below that, a field labeled 'Role' contains 'Contact Center'. A 'Description' field contains 'These workers handle calls and email complaints'. Under 'In-baskets currently associated with this role:', it lists 'Contact Center, My Work'. A section titled 'Select the type of personal in-basket to display for this role:' contains three radio buttons: 'Personal (Common): Show the common view' (unchecked), 'Personal (Role): Show a custom view for this role' (checked), and 'Do not show common or role personal in-baskets' (unchecked).

Figure 6-22 Setting up in-basket type for a role

Tip: The type of personal in-basket to display for a role can be customized. Selecting **Personal (Common): Show the common view** defines a common in-basket, **My Work**.

6.4.3 Setting up In-baskets

Role in-baskets contain work items that are assigned to a role. To set up the in-baskets, complete the following steps:

1. Click the **In-baskets** tab.
2. Click **Contact Center** in-basket and then click the **In-basket General** tab.

3. Select the solution properties to be included in the in-basket display for the role. These properties are included in Case Manager Client when users access the role in-basket for their role as per the sorting order defined in Figure 6-23.

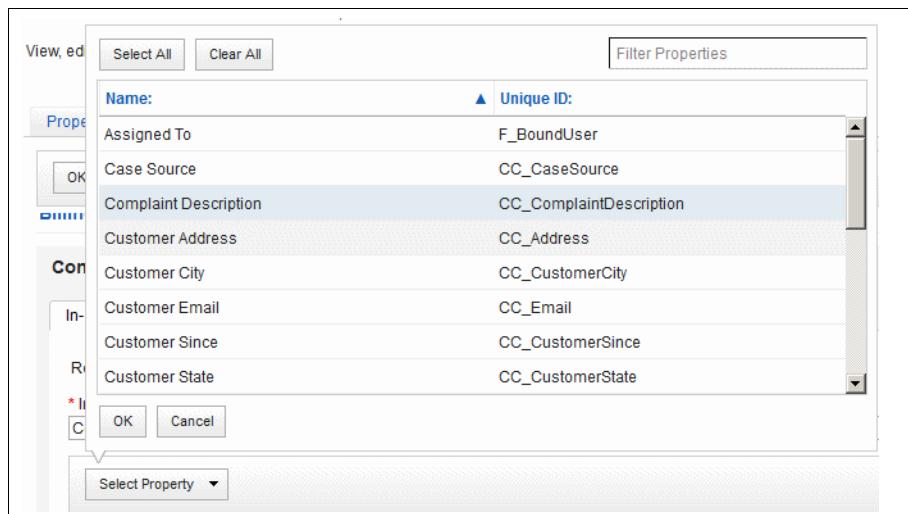


Figure 6-23 Selecting the in-basket properties

4. Set up sort information for the properties:
 - **Complaint Received Date:** Select **Sortable** and set the sort order to **Ascending**.
 - **Customer Rating:** Select **Sortable**.
 - **Complaint Category:** Select **Sortable**.

The results are shown in Figure 6-24.

The screenshot shows a 'Configure Sort Parameters' dialog box. It has a 'Select Property' dropdown at the top. Below is a table with columns: 'Name', 'Sortable', 'Sort Default', and 'Sort Order'. The table data is as follows:

Name	Sortable	Sort Default	Sort Order
Complaint Received Date	<input checked="" type="checkbox"/>	<input checked="" type="radio"/>	Ascending
Customer Rating	<input checked="" type="checkbox"/>	<input type="radio"/>	
Complaint Category	<input checked="" type="checkbox"/>	<input type="radio"/>	
Case Number	<input type="checkbox"/>		
Customer Name	<input type="checkbox"/>		
Customer Number	<input type="checkbox"/>		

Figure 6-24 Configuring the in-basket sort parameters

- Click **Add Filter** on the **In-basket Filters** tab. Select the properties and the associated operators for your filters:
 - Complaint Category** with operator **Equals**
 - Complaint Received Date** with operator **Less than**
 - Complaint Status** with operator **Equals**
 - Customer Name** with operator **Like**

The result looks similar to Figure 6-25.

Property	Operator	Text to Display
Complaint Category	=	Complaint Category
Complaint Received Date	<	Complaint Received Date
Complaint Status	=	Complaint Status
Case Number	is like	Case Number

Figure 6-25 Adding filters for an in-basket

- Click **OK All** to complete the in-basket configuration for the Contact Center role.
- Repeat the steps for the other roles. Assign the properties, sort parameters, and filter as you did for the Contact Center role in-basket. See the Table 6-3 on page 163 for the in-basket configuration.

Tip: It is a good idea to click **Save** after you complete the roles configuration.

6.4.4 Setting up document types

For your solution, set up the necessary document types. For the *Customer Complaints* solution, a document type for customer correspondence is needed. To set up the document type, complete the following steps (continuing from the previous section):

- Click the **Document Types** tab.
- Select the **Correspondence** document type in the left pane by clicking next to the name (not on it) so that the Correspondence document type is highlighted.

- Click **Add Property** and select **Existing**. Click **Case Number** to assign the existing properties to the document type as shown in Figure 6-26.

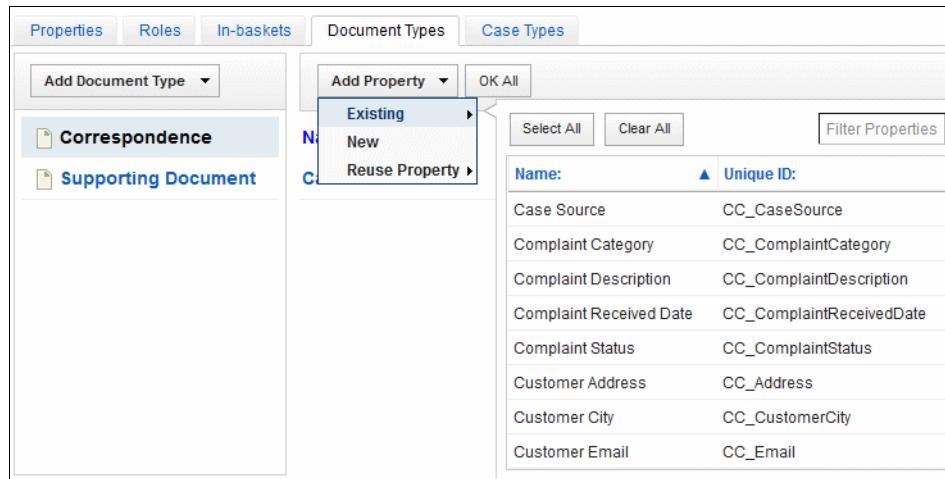


Figure 6-26 Setting document properties

- Click **OK**, then **OK All** to complete the document property configuration.
- Repeat these steps for the other document types. See Table 6-4 on page 164 for the list of all document types and their properties.

Tip: Generally, the properties for documents are significant to the document rather than the case. In this example, the case properties are used for illustrative purposes only.

6.4.5 Setting up case type

To set up the Complaint case type, complete the following steps:

- Click the **Case Types** tab.
- Click the **Complaint** case type as shown in Figure 6-27.

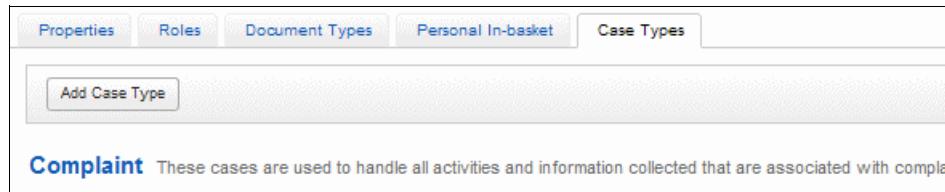


Figure 6-27 Configuring case types

3. The Case Type configuration page is shown in Figure 6-28. The options include configuring the document type so that it adds a case upon document creation. You can also change the default page so it displays the add case, split case, and case details web pages. Because you are creating cases manually for this solution and using the default pages, leave these options unchanged.

The screenshot shows the 'Case Manager Builder' application window. The title bar says 'IBM Case Manager Builder'. The main area is titled 'Manage Solutions \ Customer Com... \ Complaint'. On the left, there's a sidebar with icons for 'Case Type' (selected), 'Properties', 'Views', 'Case Folders', and 'Tasks'. The main content area has a header: 'Specify attributes for the case type that identify the tasks, properties, folders, and views that are required to manage the case. Page layouts control how case workers see the case. Learn more'. Below this, there are several input fields:

- *Case type name:
- *Case type unique identifier:
- Case type description:
These cases are used to handle all activities and information collected that are associated with complaints received by the company.
- Starting document type:
- Default layout for Add Case page:
- Default layout for Split Case page:
- Default layout for Case Details page:
- Override default case details page layout for:

Figure 6-28 Configuring the case type

4. In the left pane, select **Properties** to configure which properties are available within the case. Add all the declared properties to the case by clicking **Add Property** → **Existing** → **Select all** → **OK** (Figure 6-29). Click **OK All**.

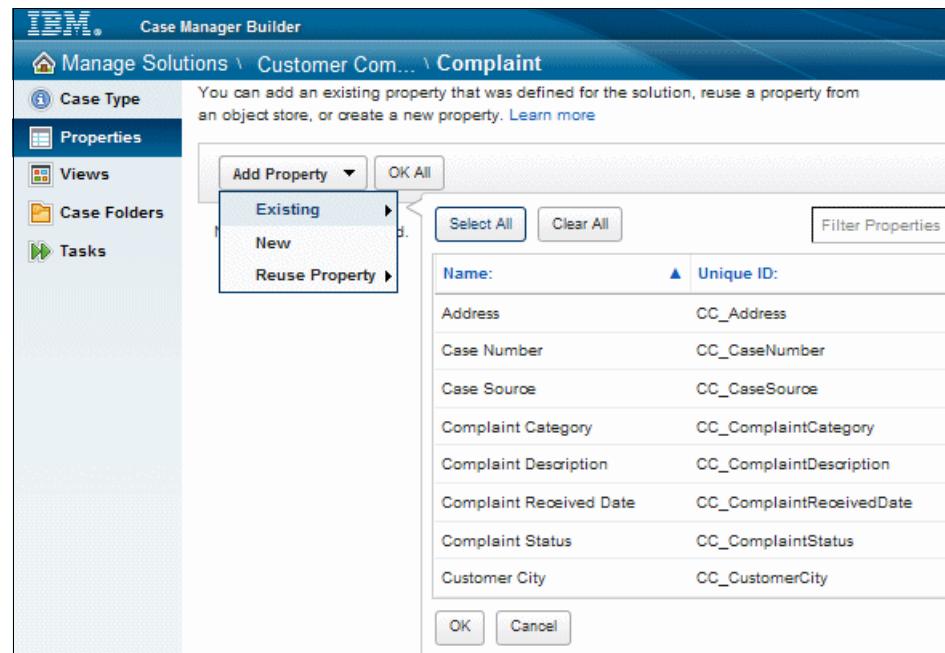


Figure 6-29 Configuring the case properties

5. In the left pane, select the **Views** option as shown in Figure 6-30. The views for the Case Summary, eCase Data, and Case Search are configured on this page, so only the properties for each view are displayed.

The screenshot shows the 'IBM Case Manager Builder' interface. The left sidebar has options: Case Type, Properties, **Views** (which is selected), Case Folders, and Tasks. The main area title is 'Manage Solutions \ Customer Com... \ Complaint'. A sub-section title 'Views' is followed by a description: 'Views define the properties to display to case workers and in what order those properties appear.' Below this is a table:

Available Properties	Properties in the Case Summary view
Case Number	No properties are in this view.
Case Source	
Complaint Category	Add this property to the view.
Complaint Descrip...	
Complaint Receive...	
Complaint Status	
Customer Address	
Customer City	
Customer Email	
Customer Name	
Customer Number	
Customer Rating	

Figure 6-30 Configuring the case views

Tip: To the right of each of the properties in the **Available Properties** field is an arrow. Clicking that arrow to move the property into the **Properties in the Case Summary view** field and back.

The order the properties are displayed in the view is the order that they are displayed in this field. The properties can be moved up or down by clicking the **Move Up** or **Move Down** icons to the right of the property name.

6. Configure the **Case Summary** view as shown in Figure 6-31.

The screenshot shows the IBM Case Manager Builder interface. The left sidebar has a tree structure with nodes: Case Type, Properties, Views (selected), Case Folders, and Tasks. The main area has a title bar: Manage Solutions \ Customer Com... \ Complaint. Below it, a message says: "Views define the properties to display to case workers and in what order those properties appear." A sub-header "Case Title Property: Case ID" with an edit icon follows. Below are three tabs: Case Summary (selected), Properties, and Case Search. The "Available Properties" section contains a list of properties: Complaint Description..., Customer Address, Customer City, Customer Email, Customer Since, Customer State, Customer Telephone, Safety Check, Total Transaction..., Upgrade Category, Upsale Opportunity, and Valid. The "Properties in the Case Summary view" section contains a subset of these properties: Case Number, Case Source, Complaint Status, Complaint Category, Complaint Received Date, Customer Name, Customer Number, Customer Rating, and Part Number.

Figure 6-31 Case Summary view configuration completed

7. Click the **Properties** tab.
8. Configure the **Properties** view as shown in Figure 6-32 on page 192.
 - a. Click **Add Group**, enter Customer Information, and click **OK**.
 - b. Click the arrow next to the property in the **Available Fields** to move the property to the Customer Information group as shown in Figure 6-32 on page 192.

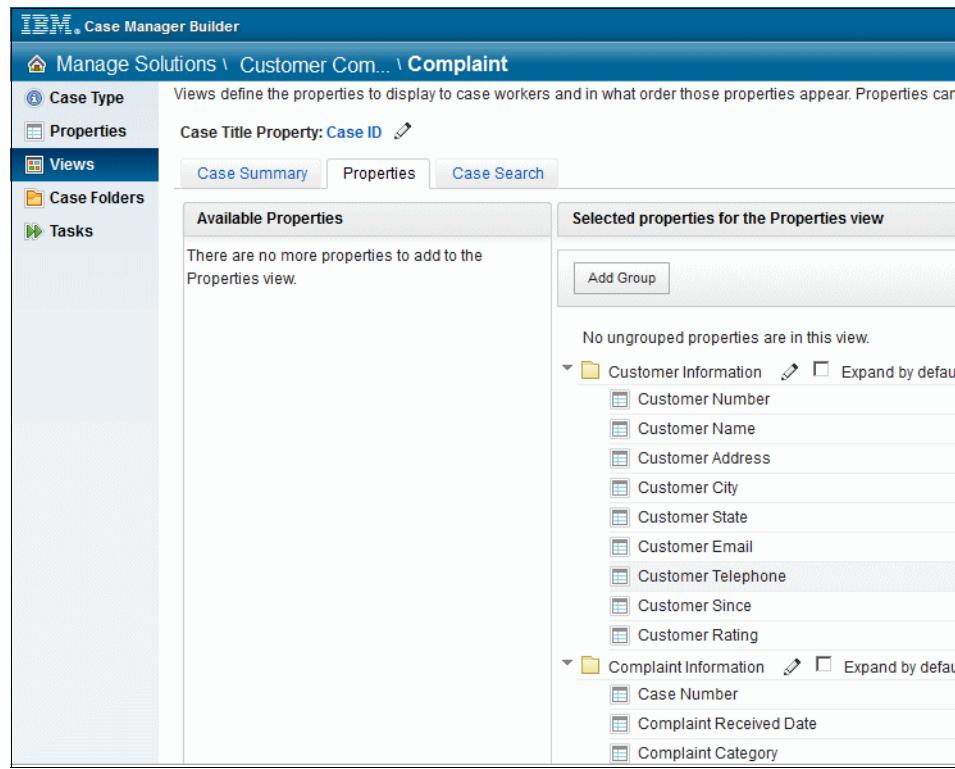


Figure 6-32 Properties view with group

- c. Repeat these steps for the Complaint Information and Flags groups as shown in Figure 6-32.
9. Click the **Case Search** tab.

10. Configure the **Case Search** view as shown in Figure 6-33.

The screenshot shows the IBM Case Manager Builder interface. The left sidebar has options: Case Type, Properties, Views (selected), Case Folders, and Tasks. The main area title is 'Manage Solutions \ Customer Com... \ Complaint'. Under 'Views', it says 'Case Title Property: Case ID'. Below are three tabs: Case Summary (selected), Properties, and Case Search. The 'Case Search' tab has two panes: 'Available Properties' (containing Case Source, Complaint Description, Customer Address, Customer City, Customer Email, Customer Rating, Customer Since, Customer State, Customer Telephone, Safety Check, Total Transaction, and Valid) and 'Properties in the Case Search view' (containing Case Number, Customer Number, Customer Name, Complaint Received Date, Complaint Category, Complaint Status, Part Number, Upsale Opportunity, and Upgrade Category). A large blue arrow points from the 'Available Properties' pane to the 'Properties in the Case Search view' pane.

Figure 6-33 Case Search view configuration completed

11. In the left pane, select the **Case Folders** option to create folders for storing case objects.
12. Click **Add Folder**.
13. Enter Correspondence as the folder name as shown in Figure 6-34, and click **OK**.

The screenshot shows the IBM Case Manager Builder interface. The left sidebar has options: Case Type, Properties, Views, Case Folders (selected), and Tasks. The main area title is 'Manage Solutions \ Customer Com... \ Complaint'. Under 'Case Folders', there is a note: 'Folders provide logical groupings for case workers to organize documents that are required to complete the case.' Below is a button 'Add Folder'. A dialog box is open, showing a tree structure with 'Complaint' expanded, and 'Correspondence' selected in a text input field. There are 'OK' and 'Cancel' buttons at the bottom right of the dialog.

Figure 6-34 Case Folder configuration

14. Repeat the last three steps for Supporting Documents. The complete list of folders are listed as shown in Figure 6-35.

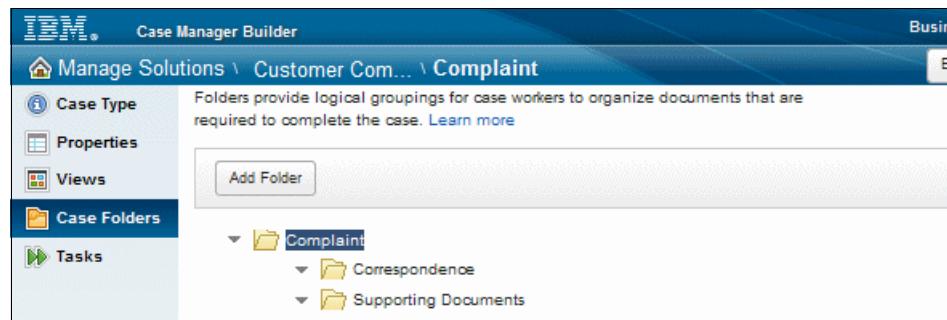


Figure 6-35 Add folders completed

Remember: When you create a case, IBM Case Manager creates a subfolder structure for the case as specified by the “Case Folders” attributes in a case type definition.

15. Click **Save**.

6.5 Setting up tasks for the solution

The Tasks page displays the tasks that were created using the wizard as shown in Figure 6-36. Because these tasks are not yet configured, they are displayed as optional tasks. For the complete list of tasks and their attribute types, see Table 6-1 on page 161.

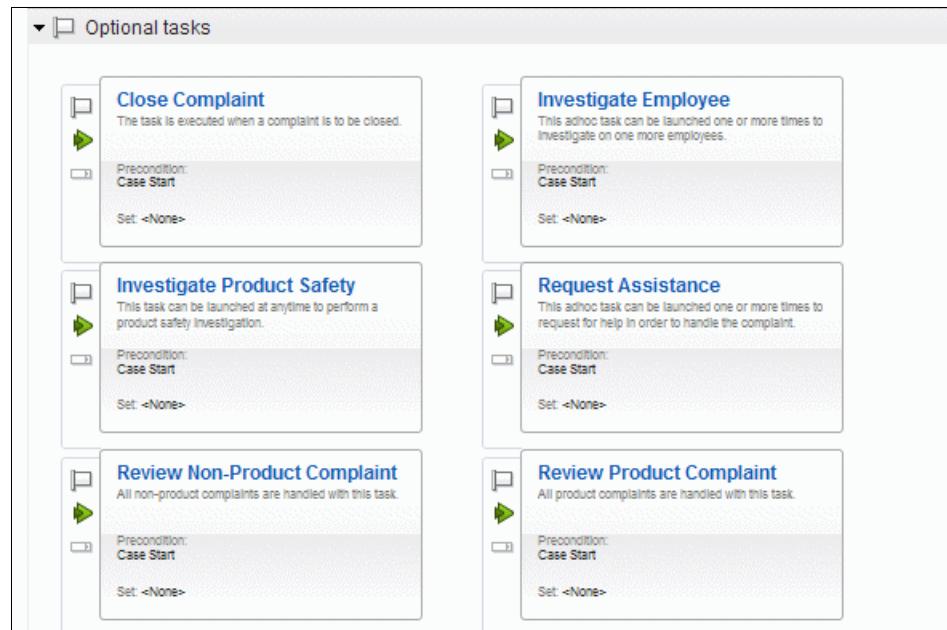


Figure 6-36 Tasks that were created in the wizard

You must create a task set to make two tasks mutually exclusive. To create a task set, complete the following steps:

1. In the left pane, select the **Tasks** option.
2. Click **Manage Sets**.
3. Click **Add Set** and enter Upgrade Options as the name of the managed set.

4. Select **A mutually exclusive set** and click **OK** to complete the creation of the Complaint Type set as shown in Figure 6-37.

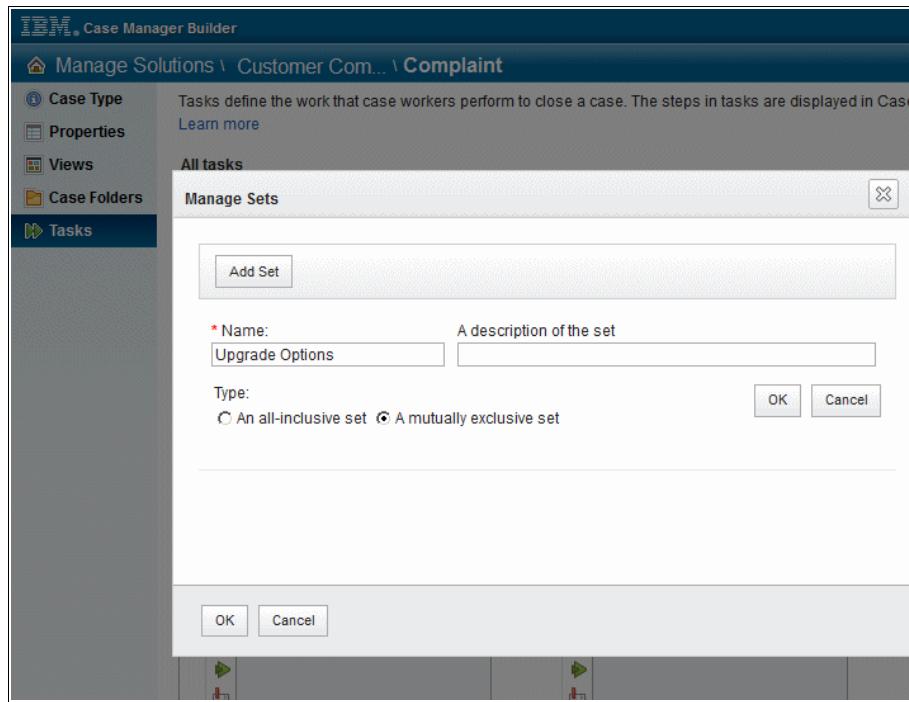


Figure 6-37 Creating a managed task set

5. Click **OK** at the bottom of the Manage Sets window to close the window. The set will be associated with tasks later in this procedure.

6.5.1 Creating the Upsell Opportunity container task

The Call Customer, Upgrade Product, and Upgrade Plan tasks must be completed when there is an opportunity for the company to upsell either its product or plan to the customer. These tasks can be grouped into a new container task that is started when there is an upsell opportunity.

To create a container task for Upsell Opportunity, complete these steps:

1. On the **Tasks** page, click **Add Task**.
2. Select **Container task** as shown in Figure 6-38.

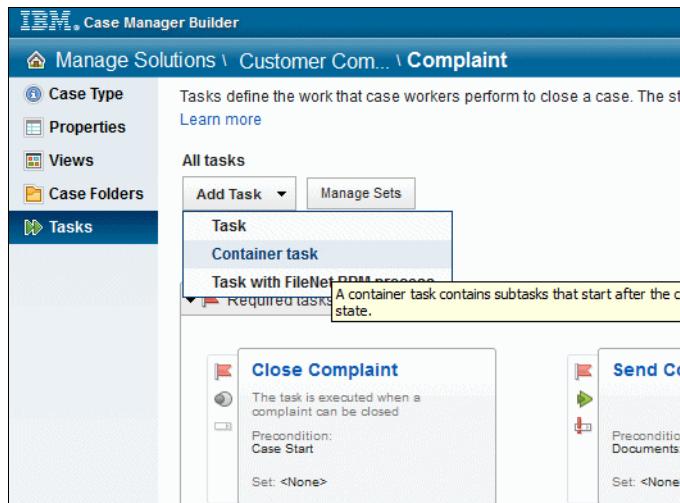


Figure 6-38 Creating the Container task menu option

3. On the **General** tab, configure the task attributes:
 - a. Select **Upsell Opportunity** as the task name.
 - b. Select **Task Starts: Automatically**.
 - c. Leave the **Required** check box cleared because this task is an optional task.
4. On the **Preconditions** tab, configure the preconditions for the task as shown in Figure 6-39 on page 198:
 - a. Select **A property condition is met**.
 - b. Click **Add condition** to configure the first condition:
 - For **Property**, select **Upsale Opportunity**.
 - For **Operator**, select **=**.
5. For **Value**, select **True**

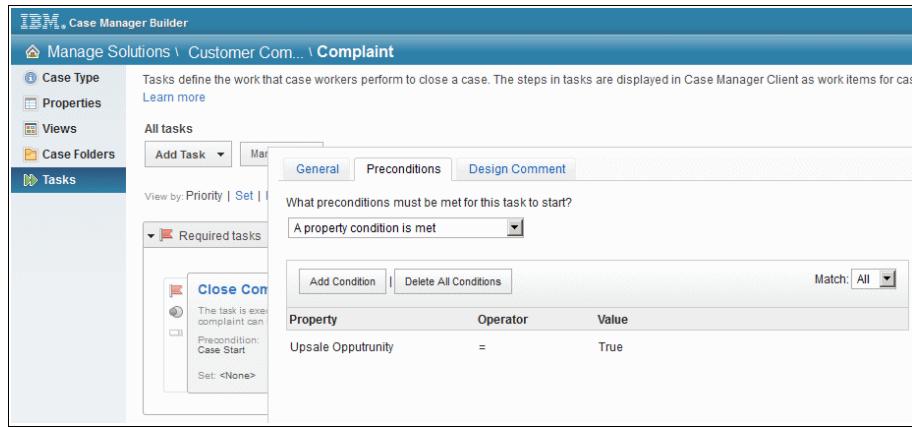


Figure 6-39 Configuring the Upsell Opportunity precondition

6. Click **OK**.

6.5.2 Editing the Upgrade Product task

The Upgrade Product task belongs to the previously defined set Complaint Case Type. This task must be moved as a subtask under Upsell Opportunity container task. Complete the following steps to edit the Upgrade Product task:

1. Click **Move task** as shown in Figure 6-40.

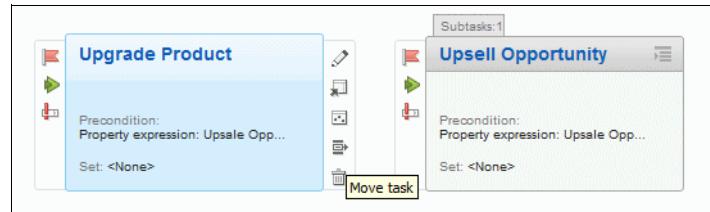


Figure 6-40 Moving the Upgrade Product task option

2. Select **Upsell Opportunity** as the target container task as shown in Figure 6-41.

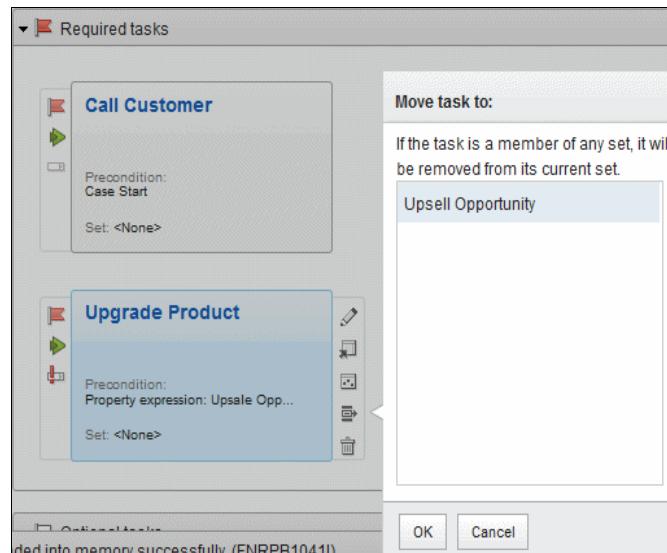


Figure 6-41 Moving the Upgrade Product task to a container task

3. Click **Go to subtasks** of **Upsell Opportunity** to access the subtask as shown in Figure 6-42.



Figure 6-42 Opening the subtasks of Upsell Opportunity

4. Move the cursor to the right of the Upgrade Product task and click **Edit Task**. This icon has an image of a pencil as shown in Figure 6-43.

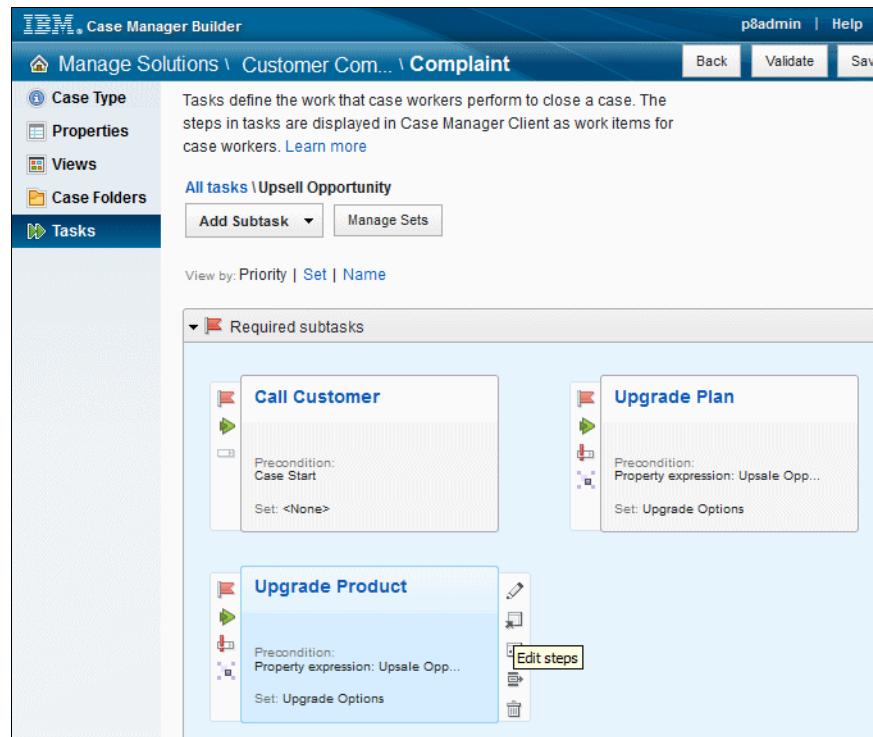


Figure 6-43 Editing the Upgrade Product task

5. On the **General** tab, configure the task attributes:
 - a. Select **Task Starts: Automatically**.
 - b. Select **Upgrade Options** from the **Assign to set** field.
 - c. Leave the **Required** check box cleared because this task is an optional task.

The results are shown in Figure 6-44.

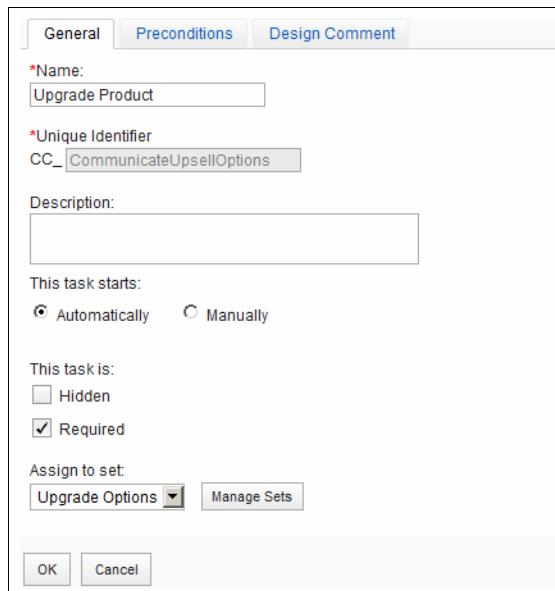


Figure 6-44 Configuring general attributes for Upgrade Product

6. On the **Preconditions** tab, configure the preconditions for the task as shown in Figure 6-45 on page 202:
 - a. Select **A property condition is met**.
 - b. Click **Add condition** to configure the first condition:
 - For **Property**, select **Upsale Opportunity**.
 - For **Operator**, select **=**.
 - For **Value**, select **True**.
 - c. Click **Add condition** to configure the next condition:
 - For **Property**, select **Upgrade Category**.
 - For **Operator**, select **=**.
 - For **Value**, select **Product**.
 - d. Click **OK**.

The screenshot shows the 'Preconditions' tab of a task configuration interface. At the top, there are three tabs: 'General', 'Preconditions' (which is selected), and 'Design Comment'. Below the tabs, a question asks, 'What preconditions must be met for this task to start?'. A dropdown menu shows 'A property condition is met'. Below this are buttons for 'Add Condition' and 'Delete All Conditions', and a dropdown for 'Match' set to 'All'. A table lists the conditions:

Property	Operator	Value	
Upsale Oppportunity	=	True	AND
Upgrade Category	=	Product	

Figure 6-45 Configuring the Upgrade Product preconditions

6.5.3 Editing the Upgrade Plan task

Similar to the Upgrade Product task, the Upgrade Plan task is a subtask of the Upsell Opportunity container task. It has property preconditions and belongs to Upgrade Options set. Use the same procedure that is listed in 6.5.2, “Editing the Upgrade Product task” on page 198 to configure the Upgrade Plan task. The preconditions for the “Upgrade Plan” task are shown in Figure 6-46.

The screenshot shows the 'Preconditions' tab of a task configuration interface. At the top, there are three tabs: 'General', 'Preconditions' (selected), and 'Design Comment'. Below the tabs, a question asks, 'What preconditions must be met for this task to start?'. A dropdown menu shows 'A property condition is met'. Below this are buttons for 'Add Condition' and 'Delete All Conditions', and a dropdown for 'Match' set to 'All'. A table lists the conditions:

Property	Operator	Value	
Upsale Oppularity	=	True	AND
Upgrade Category	=	Service Plan	

Figure 6-46 Upgrade Plan task preconditions

6.5.4 Editing the Call Customer task

Similar to the Upgrade Product task and Upgrade Plan tasks, Call Customer is a subtask of the Upsell Opportunity container task. However, it has no property preconditions and does not belong to any set. Use the same procedure that is listed in 6.5.2, “Editing the Upgrade Product task” on page 198 to configure the Call Customer task.

At this stage, the Upsell Opportunity container task contains the subtasks shown in Figure 6-47.

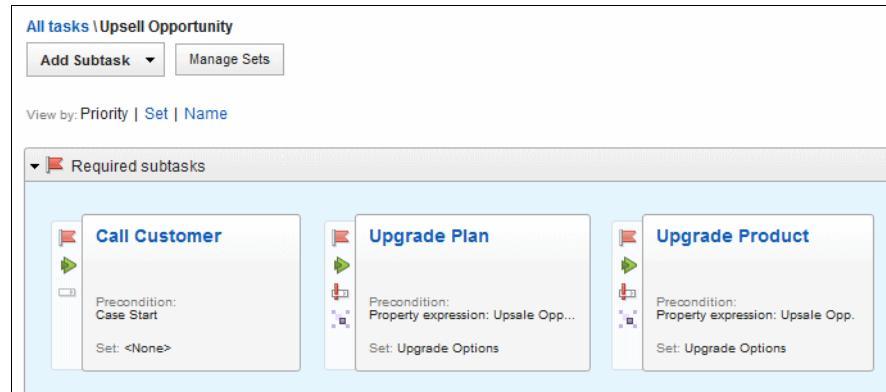


Figure 6-47 Subtasks of Upsell Opportunity

6.5.5 Editing the Verify Billing task

As shown in Table 6-1 on page 161, the Verify Billing is an optional task. It auto-starts when a Supporting document is filed into a case.

Complete these steps to edit the Verify Billing task:

1. Move the cursor to the right of the Verify Billing task and click **Edit Task**.
2. On the **General** tab, select **Task Starts: Automatically** and **Optional**
3. On the **Preconditions** tab, configure the task preconditions:
 - a. Select **A document is added to the case**.
 - b. Select **Document of a type defined for this case**, and select **Supporting Document** from the menu for the document type.
 - c. Click **OK**.

The preconditions are shown in Figure 6-48.

The screenshot shows the 'Preconditions' tab of a configuration dialog. At the top, there are three tabs: 'General', 'Preconditions' (which is selected), and 'Design Comment'. Below the tabs, a question asks, 'What preconditions must be met for this task to start?'. A dropdown menu shows 'A document is filed in the case' as the selected precondition. To the right of the dropdown is a checkbox for 'Task is repeatable' which is unchecked. Below the dropdown is another checkbox for 'Any document type' which is also unchecked. A section titled 'Document types:' contains a dropdown menu with two items: 'Correspondence' and 'Supporting Document', where 'Supporting Document' is currently selected. Below this is a section titled 'The above precondition and the following conditions:' containing a table with two rows. The table has columns for 'Property', 'Operator', and 'Value'. The first row has 'Valid' in 'Property', '=' in 'Operator', and 'True' in 'Value'. To the right of this row is the word 'AND'. The second row has 'Complaint Category' in 'Property', '=' in 'Operator', and 'Billing' in 'Value'. At the bottom of the dialog are 'OK' and 'Cancel' buttons.

Figure 6-48 Configuring the Verify Billing task preconditions

6.5.6 Editing the Verify Complaint task

As shown in Table 6-1 on page 161, the Verify Complaint is an optional task. It auto-starts when the Complaint Category property gets updated every time.

Complete these steps to edit the Verify Complaint task:

1. Move the cursor to the right of the Verify Billing task and click **Edit Task**.
2. On the **General** tab, select **Task Starts: Automatically** and **Required**
3. On the **Preconditions** tab, configure the task preconditions:
 - a. Select **A case property is updated**.
 - b. Select **Complaint Category** from the menu for case properties list.
 - c. Select **Task is repeatable**.
 - d. Click **OK**.

The preconditions are shown in Figure 6-49.

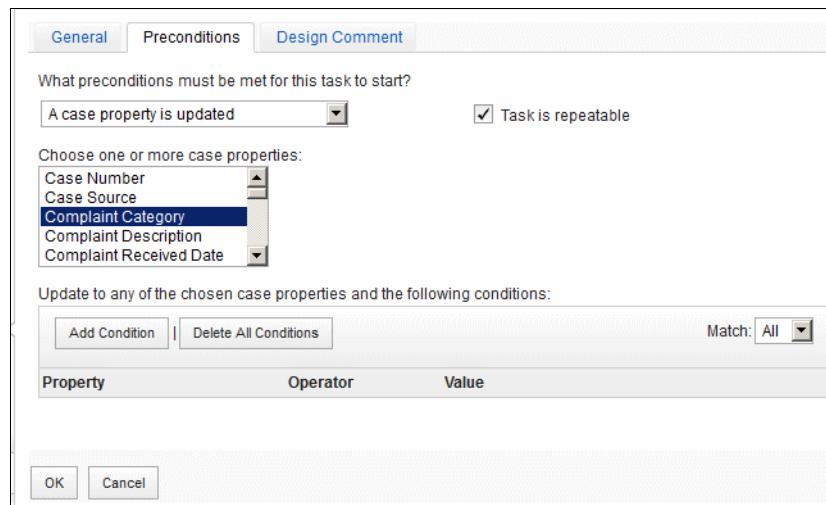


Figure 6-49 Configuring the Verify Complaint task preconditions

6.5.7 Editing the Review Product Complaint task

As shown in Table 6-1 on page 161, the Review Product Complaint task is an optional task that auto-starts when a valid product-related complaint comes in.

Complete the following steps to edit the Review Product Complaint task:

1. Click the **Edit Task** icon of Review Product Complaint task.
2. Select **Task Starts: Automatically** and leave the **Required** check box cleared because this is an optional task. See Figure 6-50.

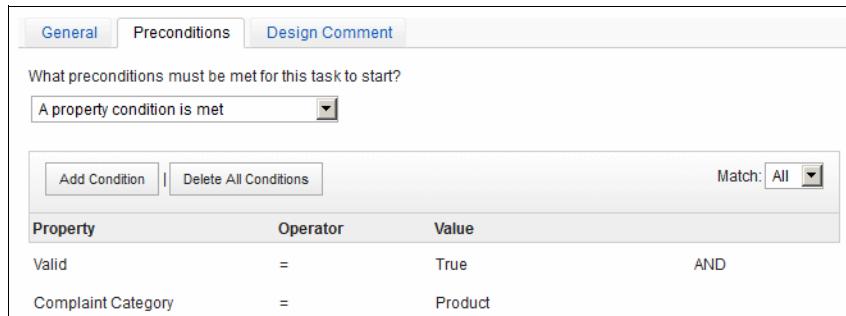


Figure 6-50 Configuring Review Product Complaint task preconditions

6.5.8 Editing the Review Non-Product Complaint task

The Review Product Complaint task is similar to the Review Non-Product Complaint task. Use the same procedure that is listed in 6.5.7, “Editing the Review Product Complaint task” on page 205. The preconditions for the “Review Non-Product Complaint” task are shown in Figure 6-51.

The screenshot shows the 'Preconditions' tab of a task configuration interface. At the top, there are three tabs: 'General', 'Preconditions' (which is selected), and 'Design Comment'. Below the tabs, a question asks, 'What preconditions must be met for this task to start?'. A dropdown menu shows 'A property condition is met'. Below this, there is a table for defining conditions. The table has columns for 'Property', 'Operator', and 'Value'. There are two rows defined:

Property	Operator	Value	Match
Valid	=	True	AND
Complaint Category	not equal	Product	

Figure 6-51 Configuring the Review Non-Product Complaint task preconditions

6.5.9 Editing the Investigate Product Safety task

The Investigate Product Safety task is an optional task that can be started manually by a case worker. The task is created when the value of the *Valid* property is equal to *true* and the value of the *Complaint Category* property is equal to *Product*.

Perform the following steps to edit the Investigate Product Safety task:

1. Click **Edit Task**.
2. From the **General** tab, configure the general task attributes as shown in Figure 6-52 on page 207.
 - a. Select **Task Starts: Manually**.
 - b. Leave the **Required** check box cleared because the task is optional.

Name: Investigate Product Safety

Unique Identifier: CC_InvestigateProductSafety

Description: This task can be launched at anytime to perform a product safety investigation.

Task starts: Manually (selected)

Required:

Assign to set: <None>

Figure 6-52 Configuring the General tab for Investigate Product Safety

- Set the **Preconditions** for the Investigate Product Safety task as shown in Figure 6-53, and click **OK** to finish.

What preconditions must be met for this task to start?

A document is added to the case A property condition is met No precondition, start task.

Add Condition AND the following conditions

Property	Operator	Value
Valid	=	True
Complaint Category	=	Product

OK Cancel

Figure 6-53 Configuring the Investigate Product Safety preconditions

6.5.10 Editing the Send Corresponding Letter task

As shown in Table 6-1 on page 161, the Send Corresponding Letter task is a required task. It auto-starts when a Correspondence document is filed into a case.

Complete the following steps to edit the Send Correspondence Letter task:

- Click **Edit Task** icon of Review Product Complaint task.
- Select **Task Starts: Automatically** and **Required**.

3. On **Preconditions** tab, configure these task preconditions:

- a. Select **A document is added to the same**.
- b. Select **Document Types**: Correspondence.
- c. Click **OK**.

The preconditions are shown in Figure 6-54.

The screenshot shows the 'Preconditions' tab of a task configuration dialog. At the top, there are three tabs: 'General', 'Preconditions' (which is selected), and 'Design Comment'. Below the tabs, a section asks 'What preconditions must be met for this task to start?'. A dropdown menu shows 'A document is filed in the case' and an unchecked checkbox for 'Task is repeatable'. Another checkbox for 'Any document type' is also present. A list of 'Document types:' includes 'Correspondence' (selected) and 'Supporting Document'. Below this, a section for 'The above precondition and the following conditions:' includes 'Add Condition', 'Delete All Conditions', and a 'Match' dropdown set to 'All'. A table below has columns for 'Property', 'Operator', and 'Value'.

Figure 6-54 Configuring Send Correspondence Letter task preconditions

6.5.11 Editing the Close Complaint task

The Close Complaint task is a required task that can be started manually by a case worker. The configured task is shown in Figure 6-55.

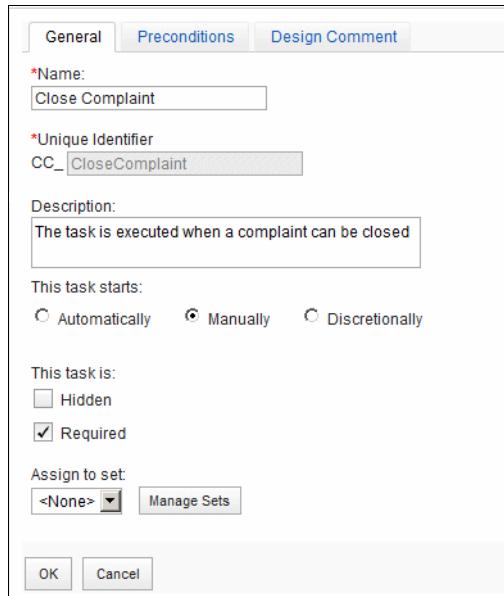


Figure 6-55 Configuring the Close Complaint task

6.5.12 Editing the Investigate Employee task

As shown in Table 6-1 on page 161, the Investigate Employee task is a discretionary task. A case worker can create this task at any time during a case processing.

To configure this task, complete the following steps:

1. Move the cursor to the right of the Investigate Employee task and click the **Edit Task** icon.
2. Select **Task Starts: Discretionally**.
3. Click **OK**.

6.5.13 Editing the Request Assistance task

The Request Assistance task is another discretionary task. Use the procedure that is listed in 6.5.12, “Editing the Investigate Employee task” on page 209 to configure the Request Assistance task.

6.5.14 Editing task summary

After you edit the attributes for all the tasks, the simple Customer Complaints solution has required tasks, optional tasks, and discretionary tasks.

Figure 6-56 shows all the required tasks for the Customer Complaints solution.

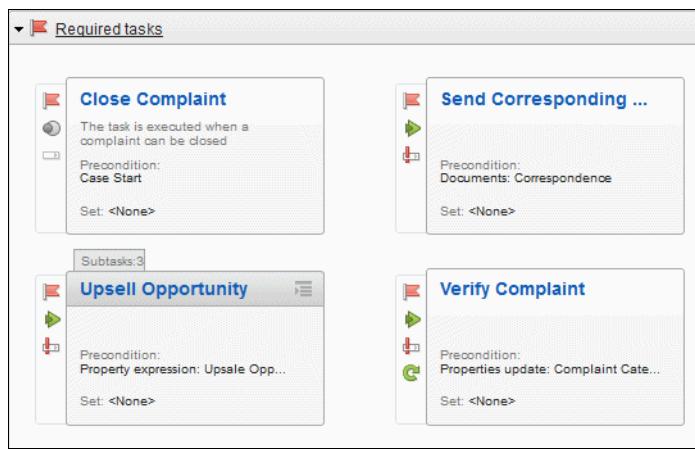


Figure 6-56 Required tasks for the Customer Complaints solution

Figure 6-57 shows all the optional tasks for the Customer Complaints solution.

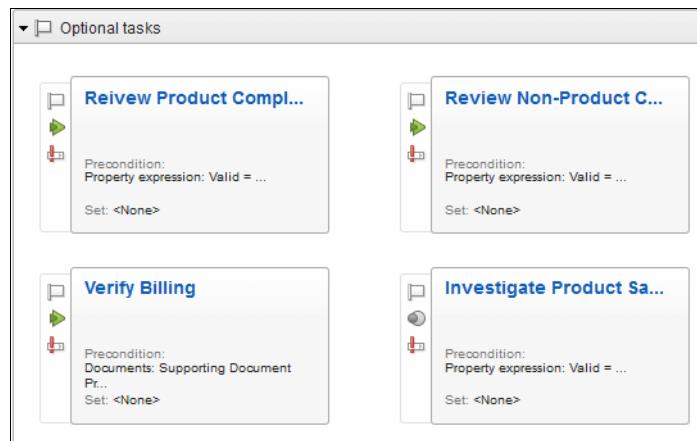


Figure 6-57 Optional tasks for the Customer Complaints solution

Figure 6-58 displays all the discretionary task for the simple Customer Complaints solution.

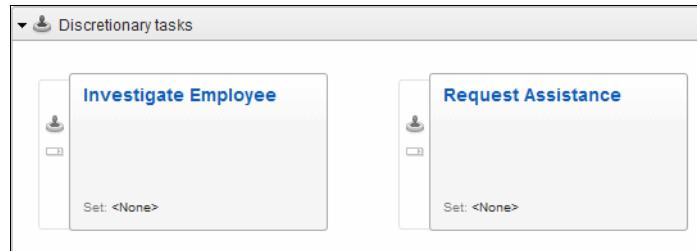


Figure 6-58 Discretionary tasks for the Customer Complaints solution

Figure 6-59 displays all of the tasks sorted by task name.

All tasks				
Add Task		Manage Sets		
View by: Priority Set Name				
Task Name:	Precondition	Set	Settings	
Investigate Em...		<None>		
Investigate Pr...	Property expression: Valid = True AND...	<None>		
Request Assis...		<None>		
Review Non-P...	Property expression: Valid = True AND...	<None>		
Reivew Produ...	Property expression: Valid = True AN...	<None>		
Verify Compla...	Properties update: Complaint Catego...	<None>		
Send Corresp...	Documents: Correspondence	<None>		
Upsell Opport...	Property expression: Upsale Opputru...	<None>		
Verify Billing	Documents: Supporting Document P...	<None>		
Close Compla...	Case Start	<None>		

Figure 6-59 All tasks sorted by task name



Building a simple solution: Part 2

This chapter and the previous chapter describe how to build a simple solution. The use case that was introduced in Chapter 2, “Typical case management applications” on page 21 is used to provide step-by-step procedure to build the solution. These two chapters show building a simple case. Later chapters build on this use case to show how to build more advanced features into the solution. These features include case splitting, reusing BPM processes, and integration with other systems and components.

This chapter covers the following topics:

- ▶ Configuring workflow diagrams by using Step Editor
- ▶ Saving and validating the solution
- ▶ Deploying the Customer Complaints solution
- ▶ Testing the Customer Complaints solution
- ▶ Improving the solution

7.1 Configuring workflow diagrams by using Step Editor

Use the Case Manager Builder Step Editor to create workflow diagrams for the tasks that are listed in Table 6-1 on page 161:

- ▶ Upgrade Product task
- ▶ Upgrade Plan task
- ▶ Call Customer task
- ▶ Verify Billing task
- ▶ Verify Complaint task
- ▶ Review Product Complaint task
- ▶ Review Non-Product Complaint task
- ▶ Investigate Product Safety task
- ▶ Send Corresponding Letter task
- ▶ Close Complaint task
- ▶ Investigate Employee task
- ▶ Request Assistance task

7.1.1 Creating the Upgrade Product task diagram

To create a workflow diagram for the Upgrade Product task, complete the following steps:

1. Click the **Go to subtasks** icon on Upsell Opportunity container task.
2. Click the **Edit steps** icon to the right of the Upgrade Product task, which is located just below the **Edit Task** icon, as shown in Figure 7-1.

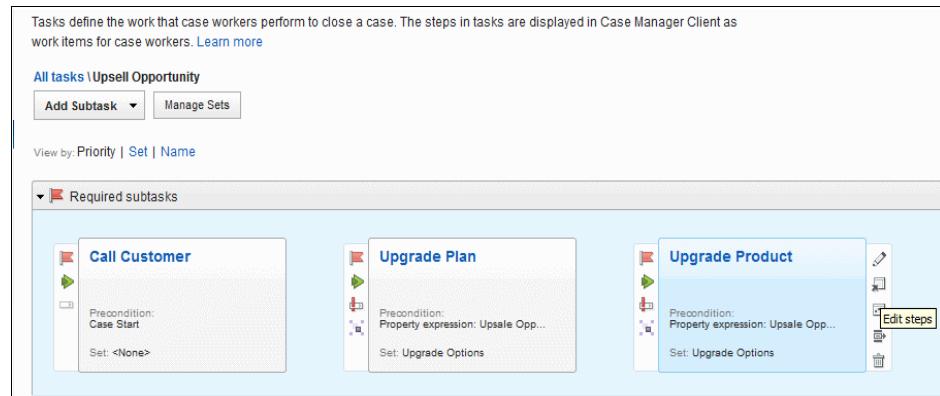


Figure 7-1 The Open Step Editor (Edit steps) icon

3. The Step Editor interface is displayed as shown in Figure 7-2. The Step Editor window consists of horizontal sections known as *swimlanes*. Other than the **System** and **Undefined Participants** swimlanes, each new swimlane represents either a user role or workflow group. Steps are added to the swimlanes and the work is routed between the steps by *connectors*.

Troubleshooting: If the swimlanes are missing when you open the Step Editor, ensure that the current version of your browser is supported by IBM Case Manager.

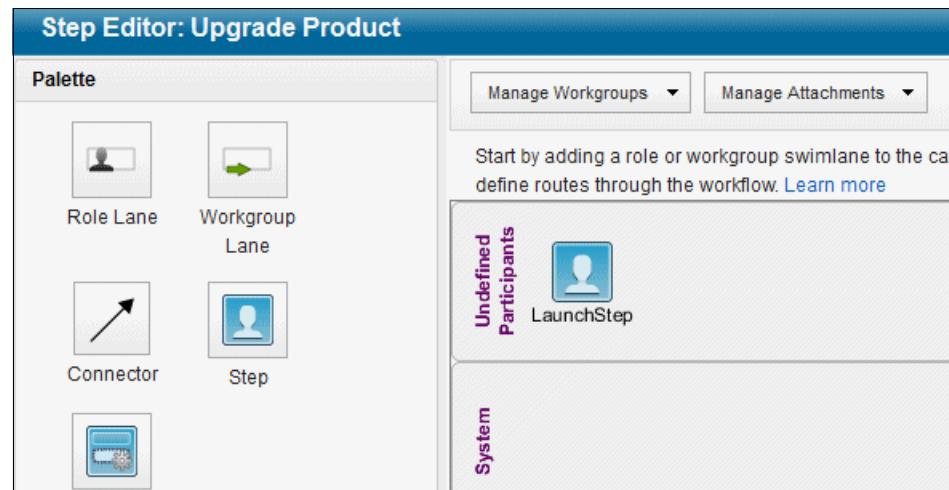


Figure 7-2 Using the Step Editor to design the Upgrade Product workflow task

4. Click the **Role Lane** icon at the upper left of the Step Editor, and drag it onto the unused white space in the workflow pane. A role swimlane is added as shown in Figure 7-3.

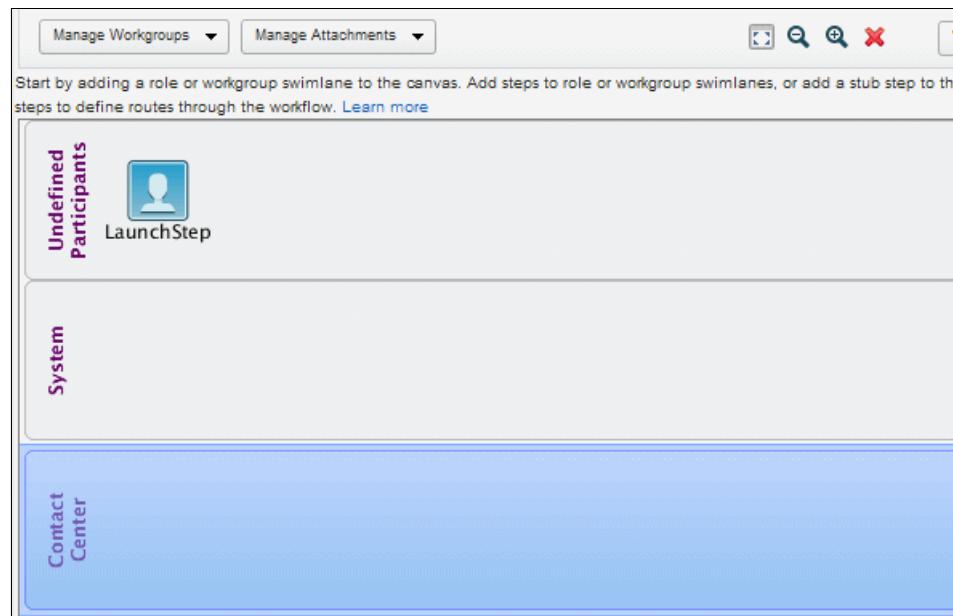


Figure 7-3 Adding a role swimlane

5. The Step Editor assigns the default role to the new swimlane. To select a different role, complete these steps:
 - a. In the left pane, under the **Role Property**, select the role from the **Role** menu. For this simple Customer Complaints solution, select **Sales Agent** as shown in Figure 7-4.
 - b. Click **OK** to update the role assignment.

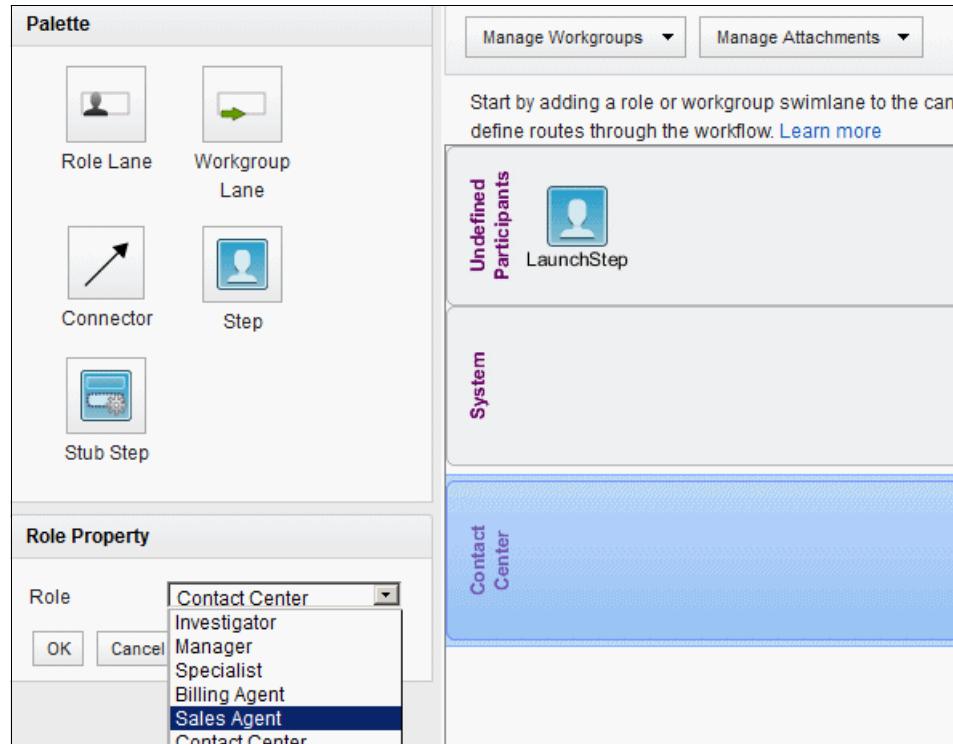


Figure 7-4 Selecting the role for a swimlane

6. Drag a **Step** icon onto the Sales Agent swimlane. The result is shown in Figure 7-5.

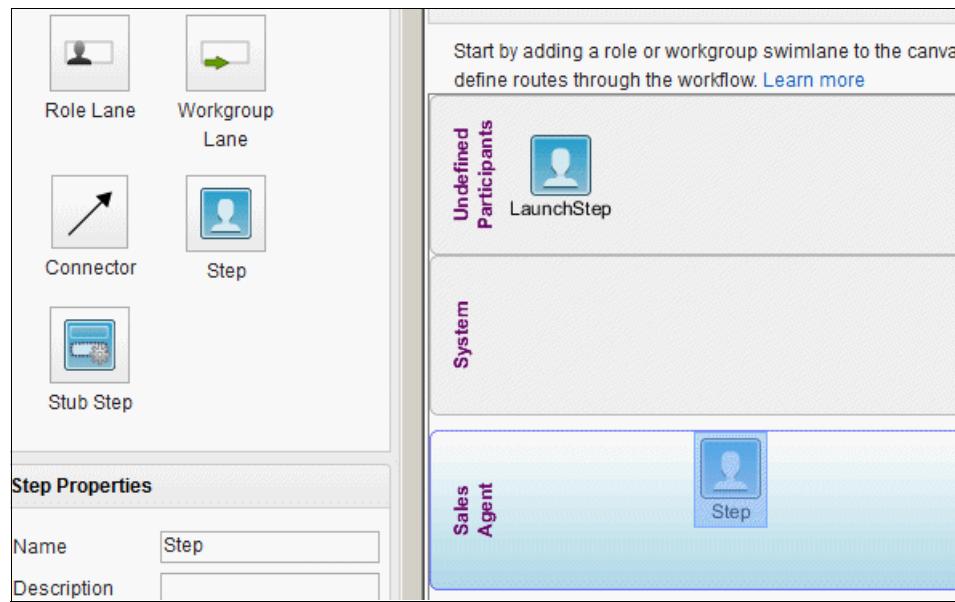


Figure 7-5 Adding a step to the swimlane

7. In the Step Properties pane (on the left), enter Communicate Upgrade Options in the Name text box. This process sets the step name.
8. Click the **Edit** icon to the right of the Case Properties field.
9. On the **Case Properties** tab, click **Select Property**.

10. Click **Select all** to choose all the case properties as the step parameters as shown in Figure 7-6, and click **OK**.

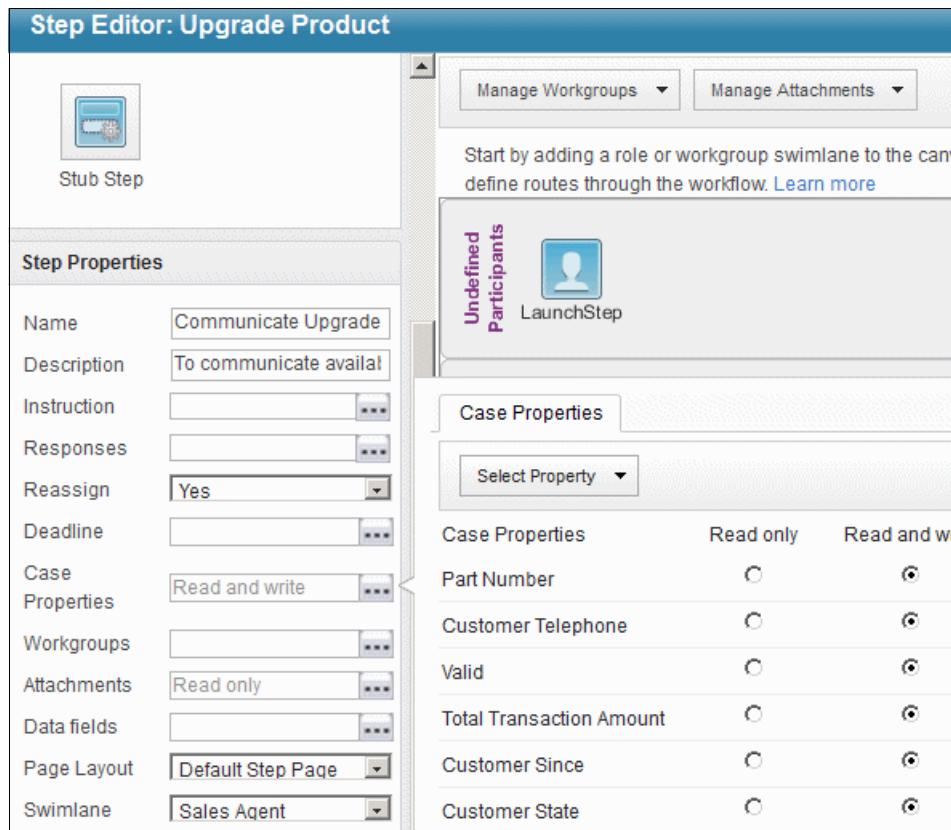


Figure 7-6 Editing step properties

Remember: This example maps all the case properties to data fields in the task (workflow). However, only those fields required by the task must be mapped.

- 11.In the Step Properties pane (on the left), click **OK** to save the step properties.
The step in the swimlane is updated with the new step name as shown in Figure 7-7.

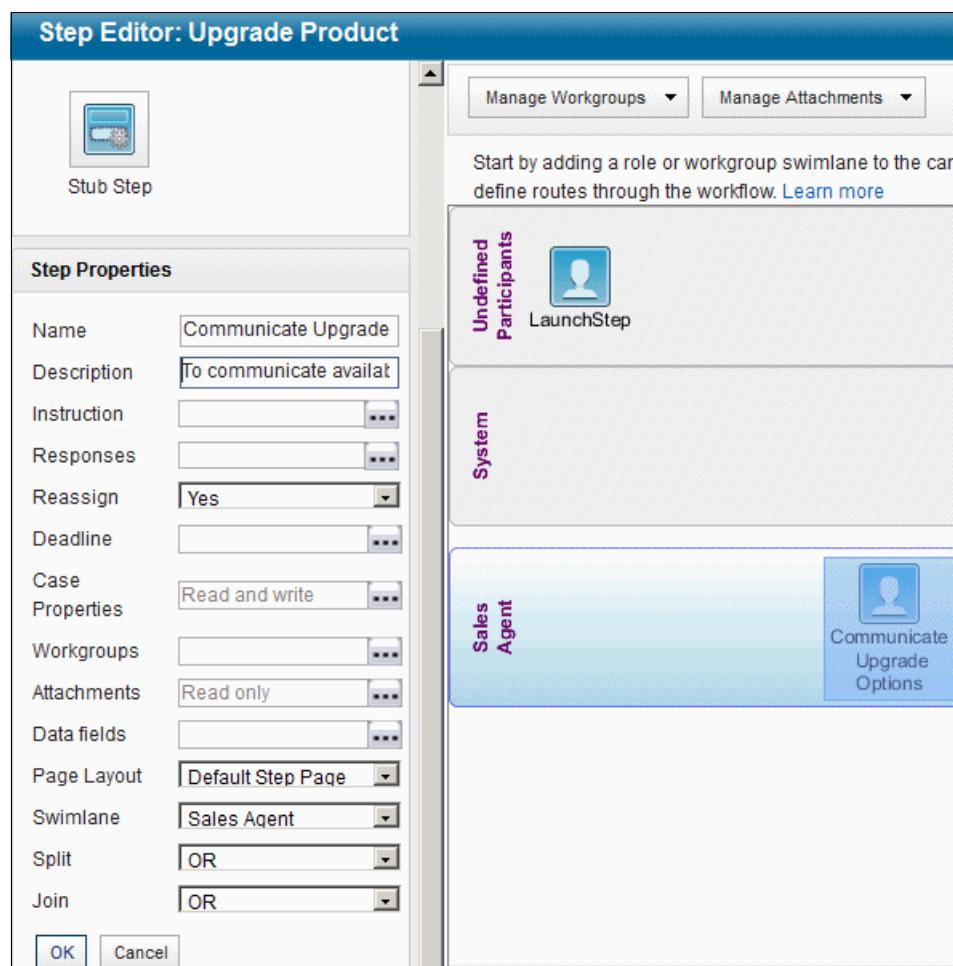


Figure 7-7 Completing the step properties editing

12. Click the **Connector** icon in the Palette pane as shown in Figure 7-8. Click and drag the **LaunchStep** to the center of the **Communicate Upgrade Options** step to connect the two steps.

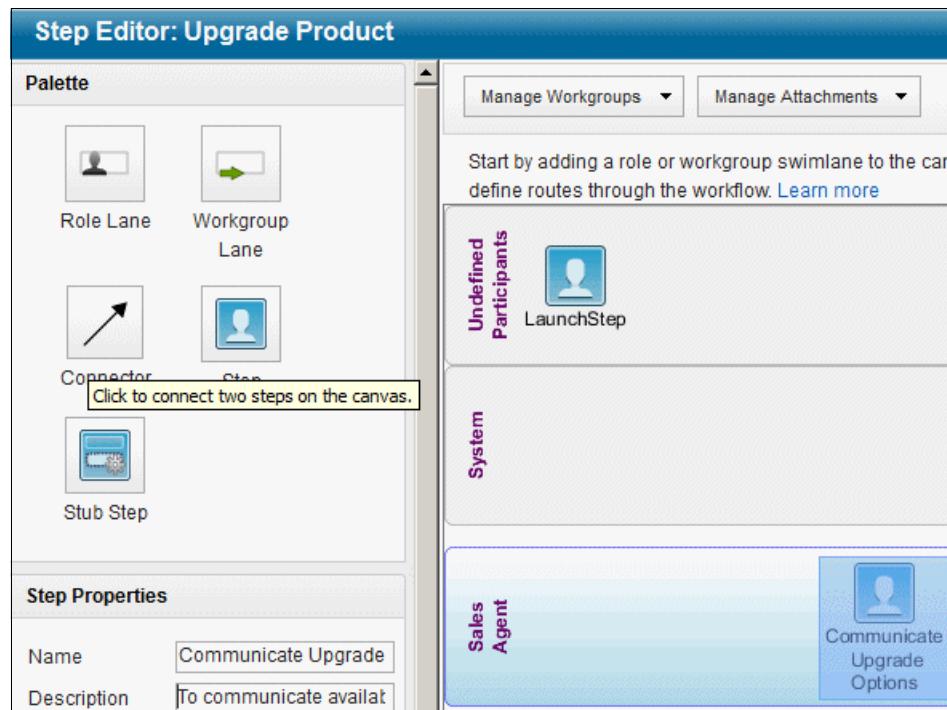


Figure 7-8 Selecting the connector icon

The Case Manager Builder Step Editor uses an elastic band effect when you connect the icons. The graphic then becomes a series of curves that connect the two steps as seen in Figure 7-9.

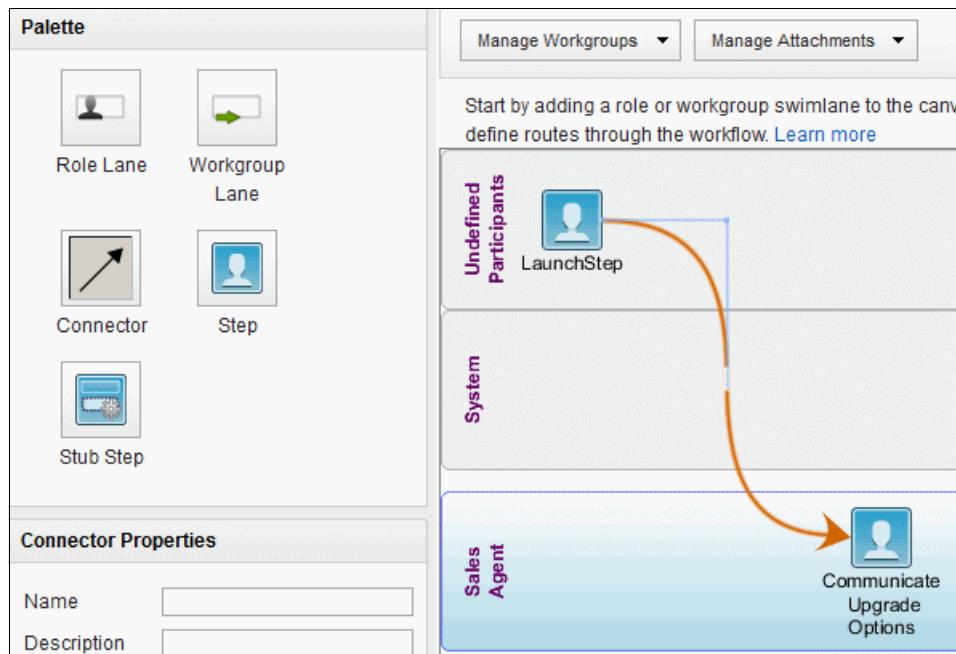


Figure 7-9 Connecting the two steps

13. Click **Validate**. If there are any validation errors, they are displayed in the lower left of the window.
14. Click **Save** and then **Close** to complete editing of the task workflow.
15. Click **Save and Close** to save and close the solution.

Tip: Save and Close the solution often to save the solution artifacts in the design object store.

7.1.2 Creating the Upgrade Plan task diagram

Use the same procedure in 7.1.1, “Creating the Upgrade Product task diagram” on page 214 to create the task diagram for the Upgrade Plan task:

1. Select **Sales Agent** for the swimlane role.
2. Enter **Communicate Plan Upgrade Options** for the step name.

3. Select all case properties as the step parameters.

Figure 7-10 shows the task diagram for the Upgrade Plan task.

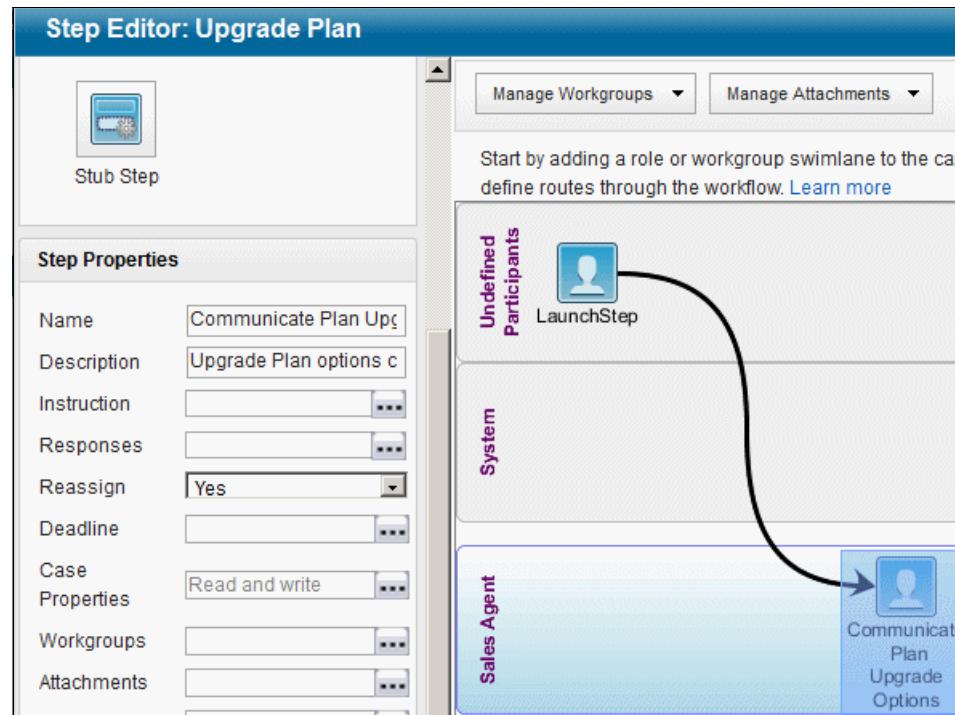


Figure 7-10 Upgrade Plan task diagram

7.1.3 Creating the Call Customer task diagram

Use the same procedure in 7.1.1, “Creating the Upgrade Product task diagram” on page 214 to create the task diagram for the Call Customer task:

1. Select **Sales Agent** for the swimlane role.
2. Enter Call Customer for the step name.

3. Select all case properties as the step parameters as shown in Figure 7-11.

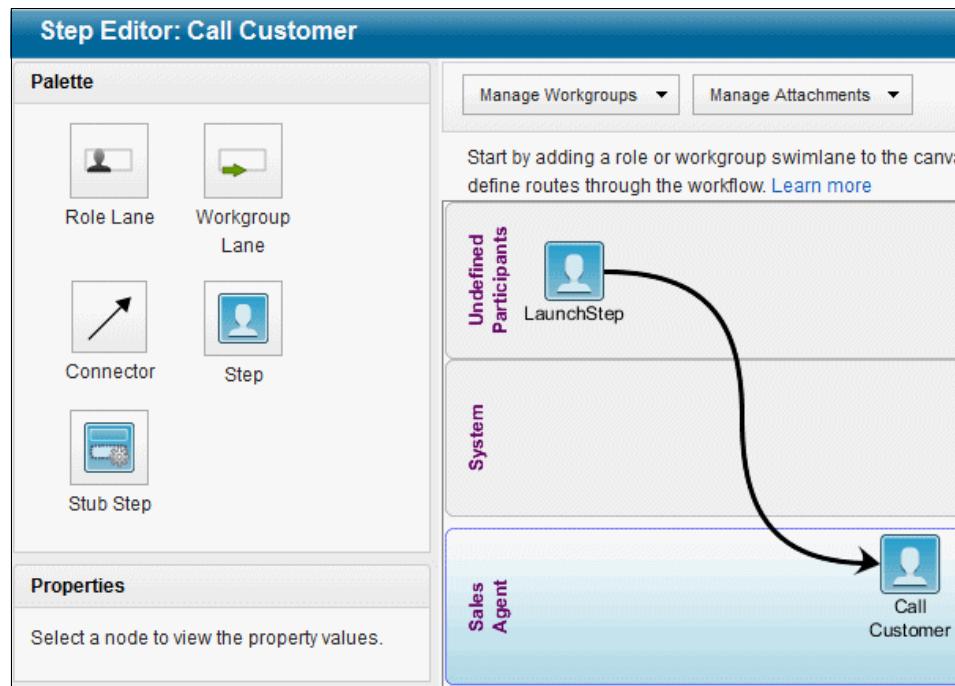


Figure 7-11 Call Customer task diagram

7.1.4 Creating the Verify Billing task diagram

Use the same procedure in 7.1.1, “Creating the Upgrade Product task diagram” on page 214 to create the task diagram for the Verify Billing task:

1. Open the solution for editing.
2. On the Case Type tab, start the Step Editor for this task.
3. Select **Billing Agent** for the swimlane role.
4. Enter Verify Supporting Doc for the step name.

5. In the Step Properties pane, click **Responses**. Add Valid Billing and Invalid Billing as the responses for Verify Supporting Doc step as shown in Figure 7-12, and click **OK**.

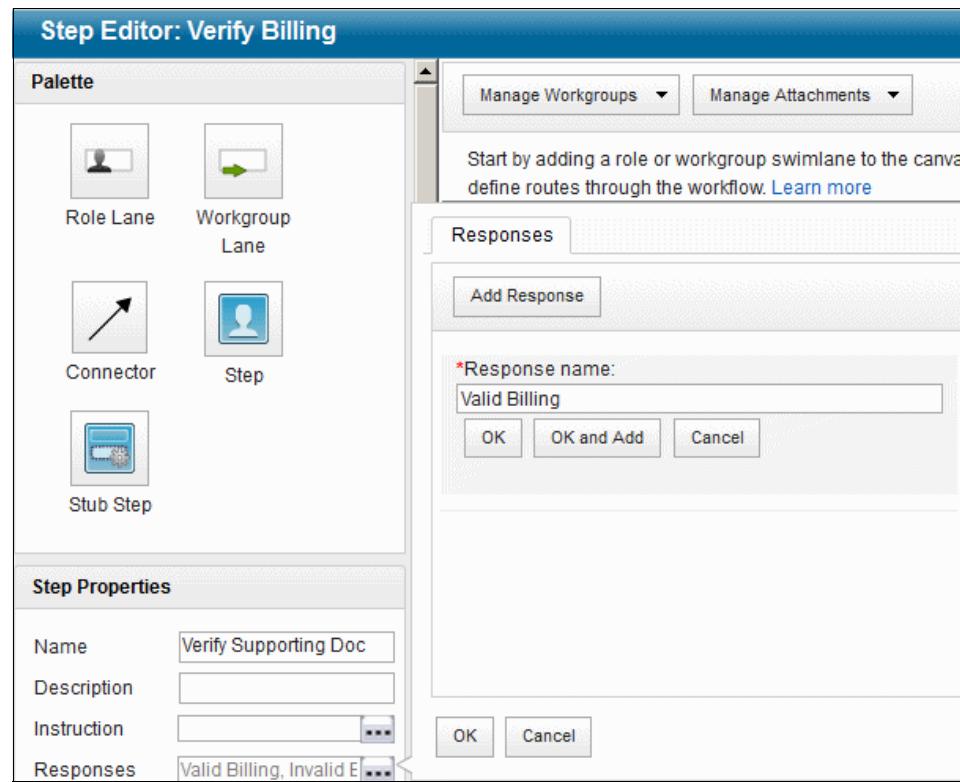


Figure 7-12 Adding response to a step

6. Select all case properties as the step parameters, and click **OK**.
7. Drag the connector from the **Launch Step** to the **Verify Complaint** step.
8. Create another Billing Adjustment step in the **Billing Agent** swimlane.
9. Drag the connector from the **Verify Supporting Doc** to the **Billing Adjustment**.

10. Click the connector and select **Valid Billing** as the response from the Connector Properties pane as shown in Figure 7-13.

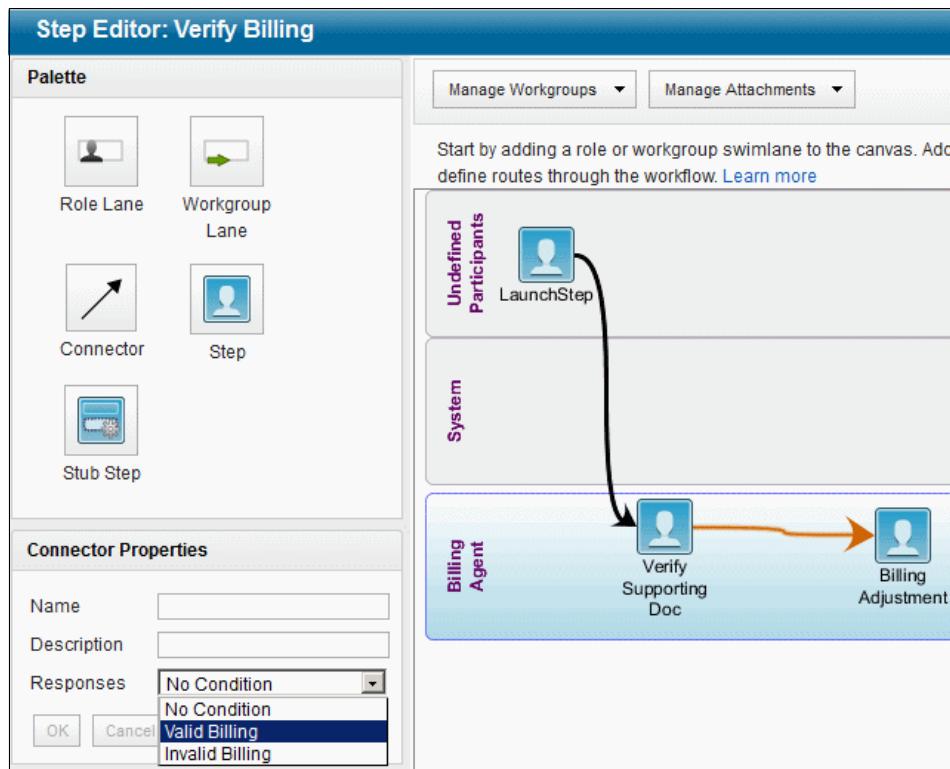


Figure 7-13 Select a response for a connector

11. Create **Contact Center** swimlane and add the Send Correspondence step to it.

12. Connect **Verify Supporting Doc** and **Send Correspondence**, and set the response to **Invalid Billing**. The final task diagram for Verify Billing task is shown in Figure 7-14.

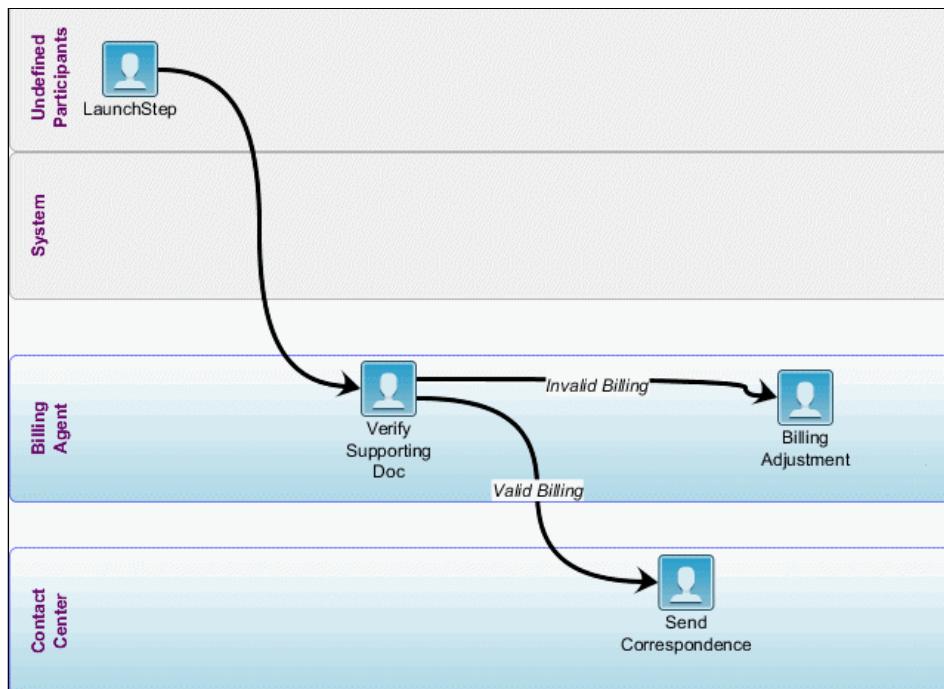


Figure 7-14 Verify Billing task diagram

13. Validate, save, and close.

7.1.5 Creating the Verify Complaint task diagram

Use the same procedure in 7.1.1, “Creating the Upgrade Product task diagram” on page 214 to create the task diagram for the Verify Complaint task:

1. Open the solution for editing.
2. On the Case Type tab, start the Step Editor for this task.
3. Select **Contact Center** for the swimlane role.
4. Enter **Verify Complaint** for the step name.
5. Select all case properties as the step parameters, and click **OK**.
6. Drag the connector from the Launch Step to the Verify Complaint step.
7. Validate, save, and close.

Figure 7-15 displays the task diagram for the Verify Complaint task.

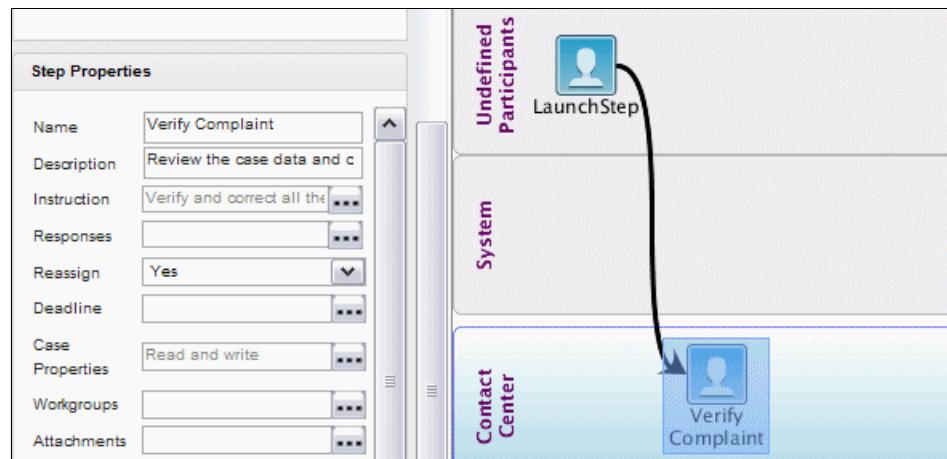


Figure 7-15 Verify Complaint task diagram

7.1.6 Creating the Review Product Complaint task diagram

Use the same procedure in 7.1.1, “Creating the Upgrade Product task diagram” on page 214 to create the task diagram for the Review Product Complaint task:

1. Select **Specialist** for the swimlane role.
2. Enter Review Product Complaint for the step name.
3. Select all case properties as the step parameters.

Figure 7-16 displays the task diagram for Review Product Complaint task.

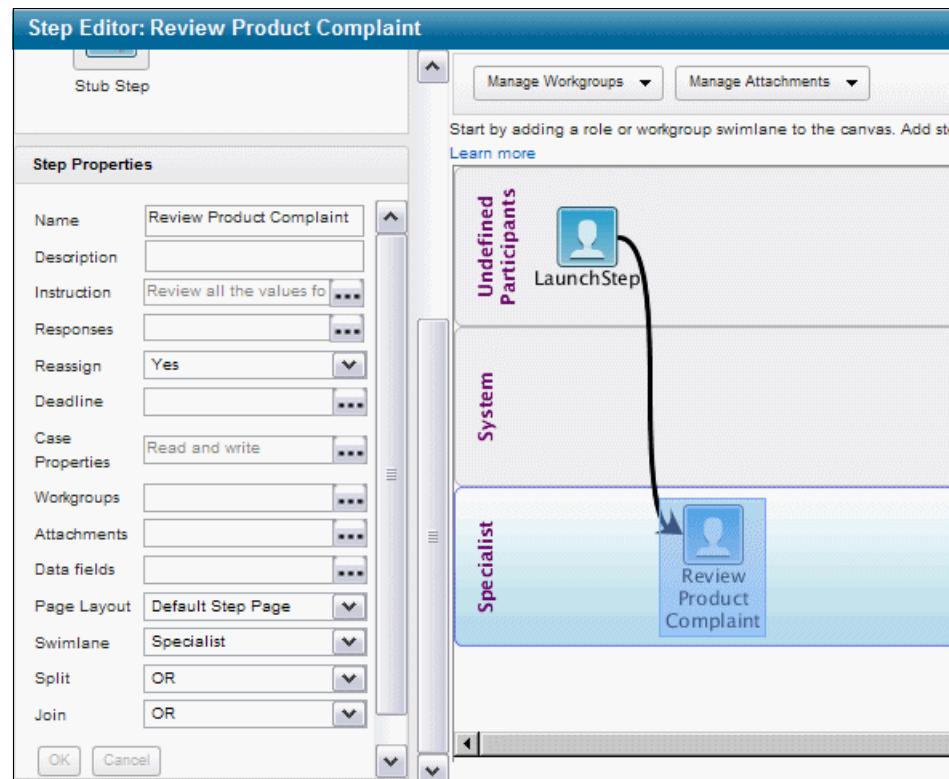


Figure 7-16 Review Product Complaint task diagram

7.1.7 Creating the Review Non-Product Complaint task diagram

Use the same procedure in 7.1.1, “Creating the Upgrade Product task diagram” on page 214 to create the task diagram for the Review Non-Product Complaint task:

1. Select **Specialist** for the swimlane role.
2. Enter Review Non-Product Complaint for the step name.
3. Select all case properties as the step parameters.

Figure 7-17 shows the task diagram for Review Non-Product Complaint task.

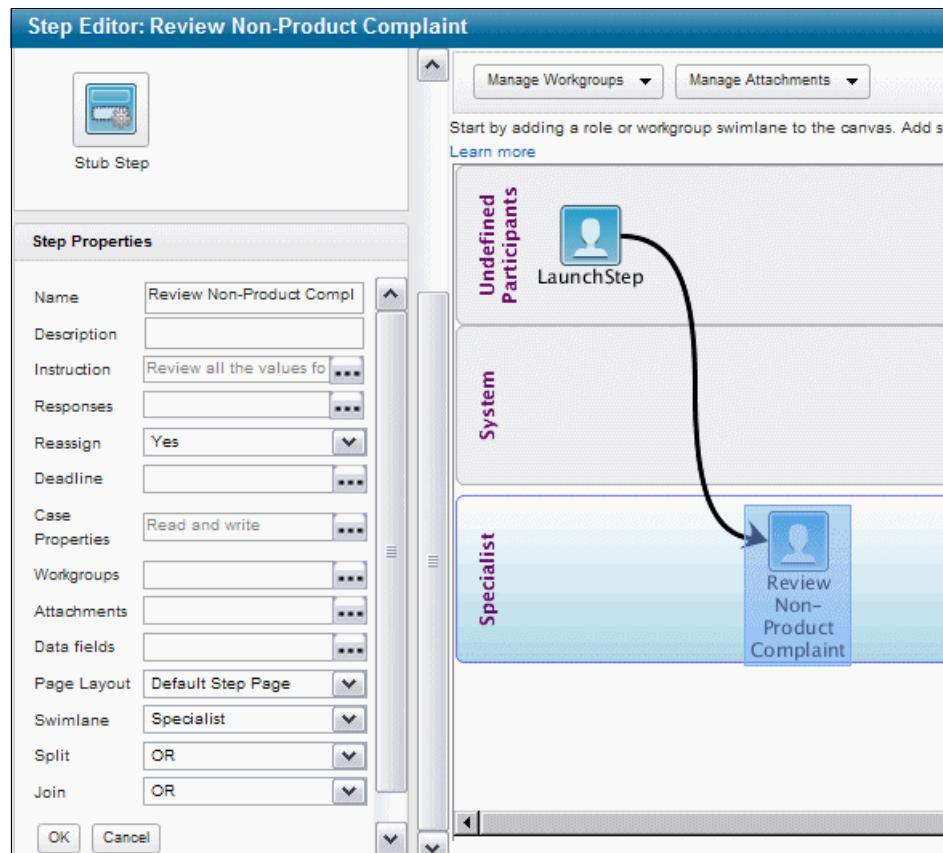


Figure 7-17 Review Non-Product Complaint task diagram

7.1.8 Creating the Investigate Product Safety task diagram

Use the same procedure in 7.1.1, “Creating the Upgrade Product task diagram” on page 214 to create the task diagram for the Investigate Product Safety task:

- ▶ Select **Investigator** for the swimlane role.
- ▶ Enter Investigate Product Safety for the step name.
- ▶ Select all case properties as the step parameters. Change the access attribute to **Read only** for the following properties:
 - Customer Telephone
 - Customer Email
 - Customer State
 - Customer Rating

- Customer Number
- Customer Name
- Customer City
- Complaint Received Date
- Case Number
- Customer Address

The changed properties are shown in Figure 7-18.

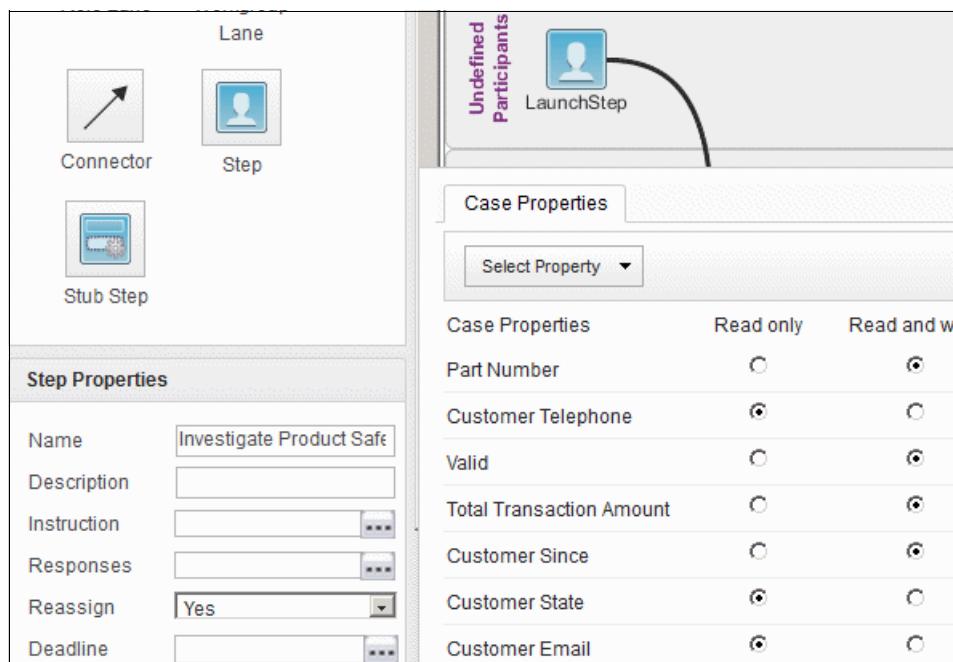


Figure 7-18 Access attribute for the case properties in a step

Figure 7-19 displays the task diagram for the Investigate Product Safety task.

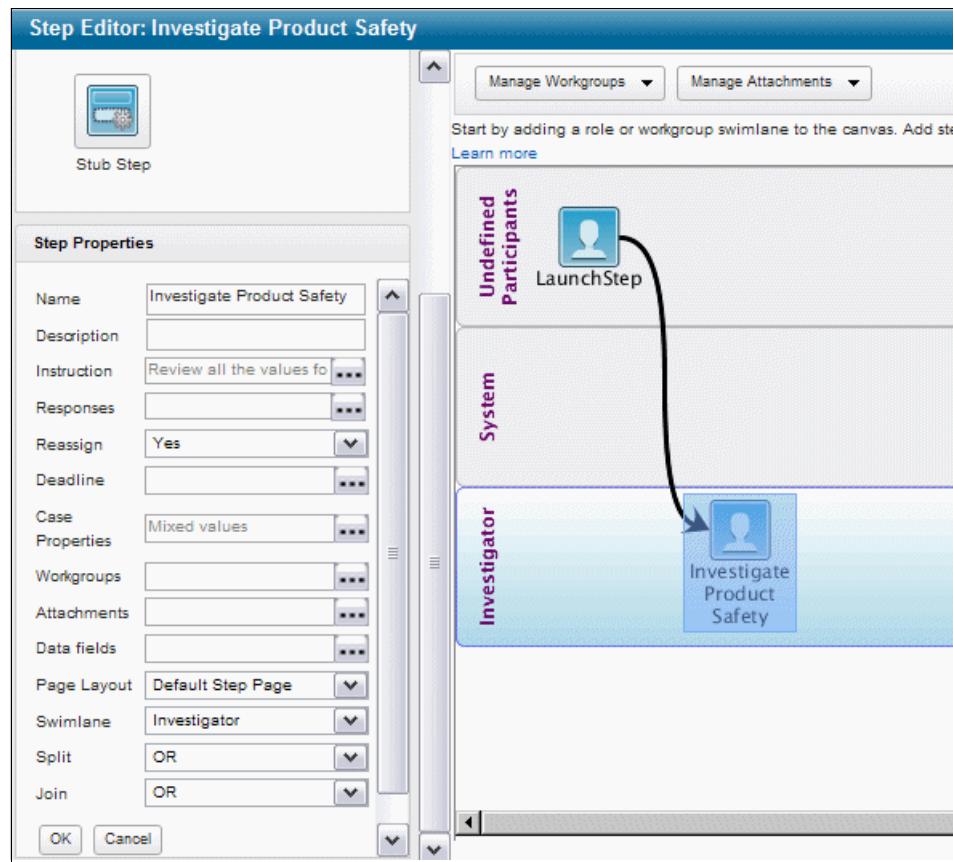


Figure 7-19 Investigate Product Safety task diagram

7.1.9 Creating the Send Corresponding Letter task diagram

This task is started when a document of type Correspondence is filed in the case. The document must be attached to the task workflow.

1. Create the attachment that initiates the task as a property in the workflow:
 - a. Start the **Step Editor** for the Send Corresponding Letter task.
 - b. Click **Manage Attachments** at the top of the Step Editor.
 - c. Click **Add Attachment**.
 - d. In the **Attachment Name** field, enter CorrespondingLetter. Do not include any spaces because IBM FileNet Process Engine does not allow spaces in attachment name.

- e. In the **Prompt** field, enter Corresponding Letter. Unlike the Attachment Name, the Prompt field can have spaces. It represents the display name for an attachment.
- f. Click **OK**.

Figure 7-20 shows the attachment creation window.

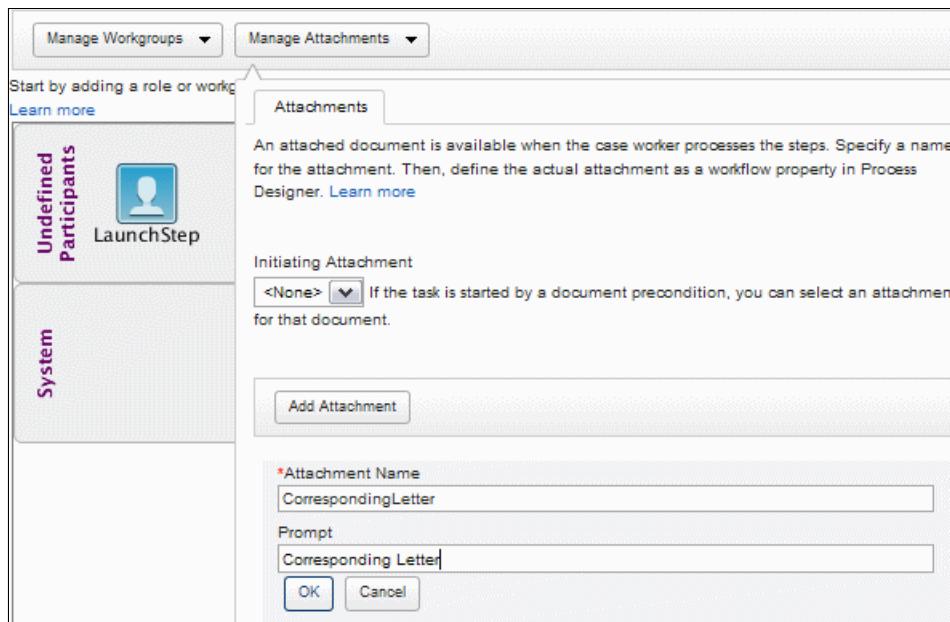


Figure 7-20 Creating the attachment so it is available as a parameter to the step

- g. Select **CorrespondingLetter** from the **Initiating Attachment** field and click **Close** as shown in Figure 7-21. This process sets the **CorrespondingLetter** attachment as an initiating attachment for the task workflow.

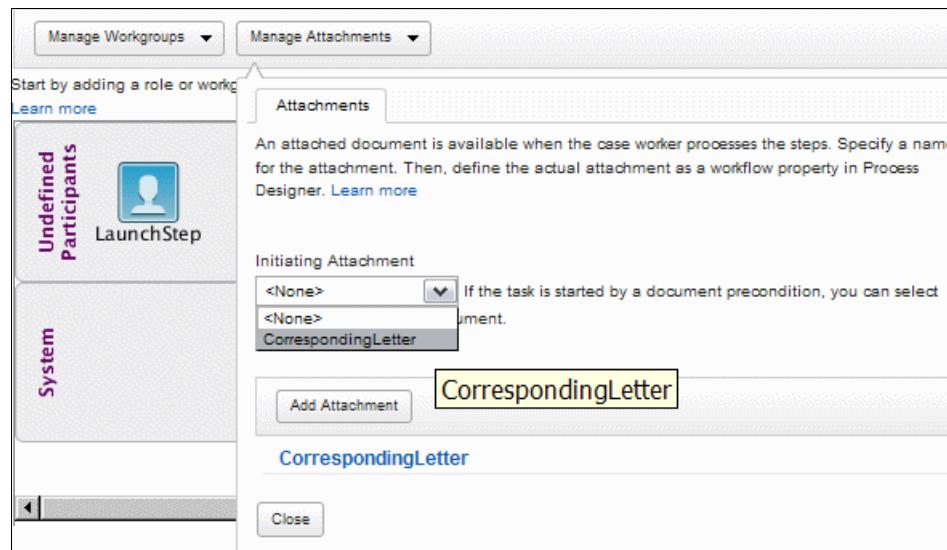


Figure 7-21 Designating an attachment as a task workflow initiating attachment

2. Use the procedure in 7.1.1, “Creating the Upgrade Product task diagram” on page 214 to add a swimlane role and step, and edit the step properties:
 - a. Select **Contact Center** for the swimlane role.
 - b. Enter Review Corresponding Letter for the step name.

- c. Select all case properties. Select **Read only** for all the case properties that were added as step parameters (Figure 7-22).

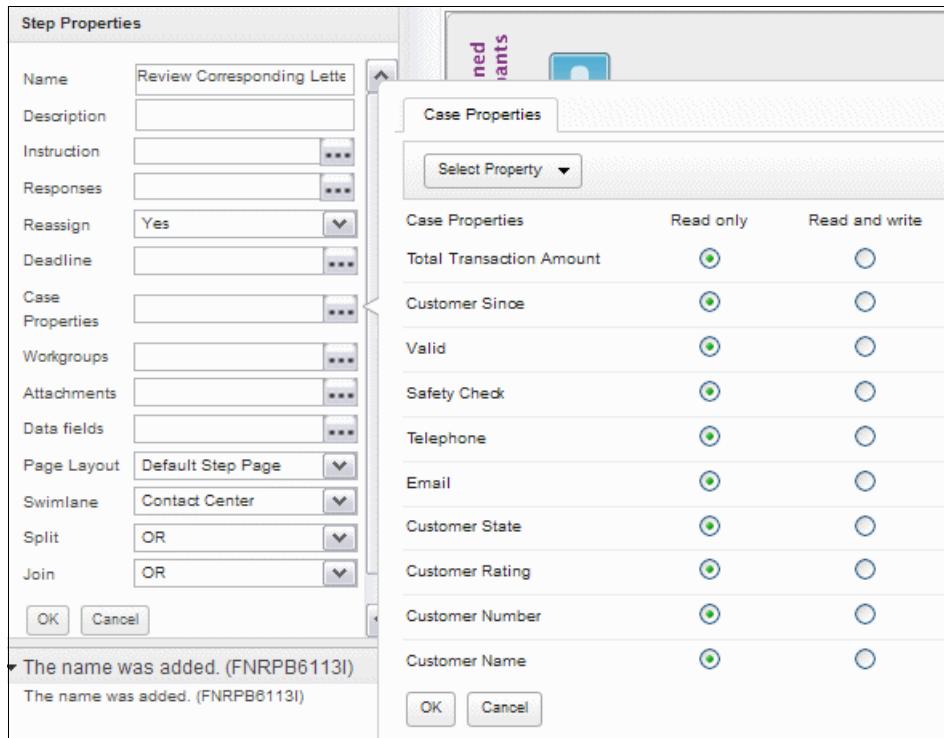


Figure 7-22 Adding case properties to the step in Read only mode

3. In the Step Properties pane, click the **Edit** icon to the right of the **Attachments** field.

4. Click **Select Attachment** → **CorrespondingLetter** as shown in Figure 7-23.
Click **OK** to accept the selection.

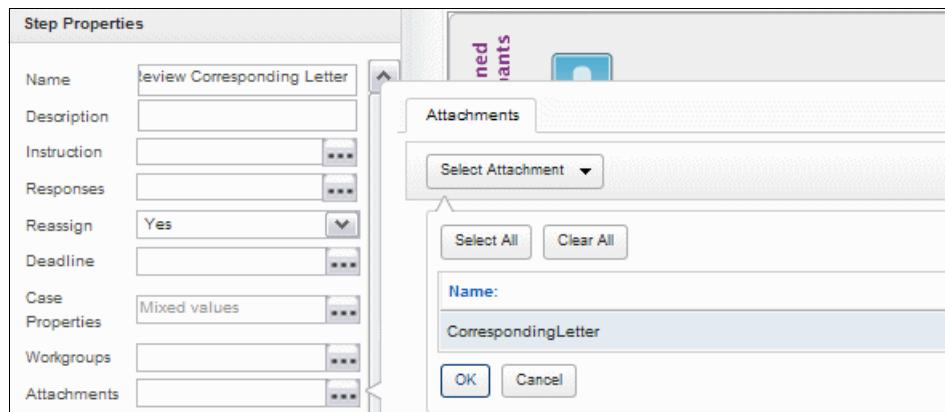


Figure 7-23 Selecting an attachment for a step

5. Click **OK** to finish selecting attachments.
6. In the Step Properties pane, click **OK** to save the step properties.
7. Draw the connector from the Launch Step to the Review Corresponding Letter step.

Figure 7-24 shows the task diagram for the Send Corresponding Letter task.



Figure 7-24 Send Corresponding Letter task diagram

The Send Corresponding Letter task can also contain a step that generates a customer letter. This step uses a third-party tool to generate the letter to be sent to the customer.

7.1.10 Creating the Close Complaint task diagram

Use the same procedure in 7.1.1, “Creating the Upgrade Product task diagram” on page 214 to create the task diagram for the Close Complaint task:

1. Select **Specialist** for the swimlane role.
2. Enter Review and Close for the step name.
3. Select all case properties as the step parameters.

Figure 7-25 shows the task diagram for Close Complaint task.

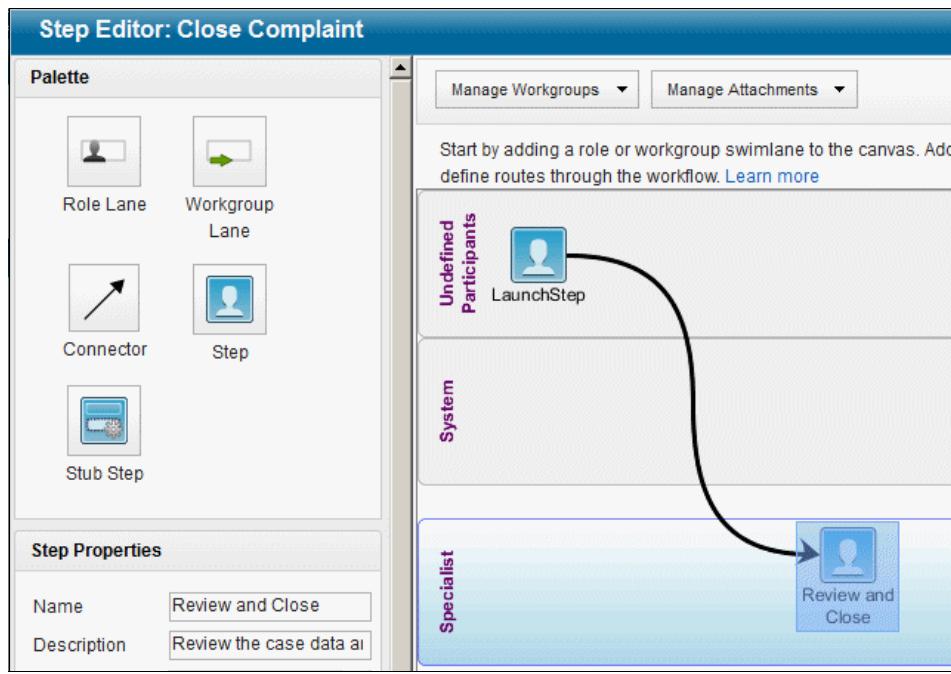


Figure 7-25 Close Complaint task diagram

7.1.11 Creating the Investigate Employee task diagram

The Investigate Employee task is a user-created task. When a case worker creates this task, Case Manager Client presents the case worker with the Launch Step page. The case worker can then enter the values for the properties and attachments. These values are used to initialize the task workflow data fields and attachments when the workflow is started.

1. Use the following the procedure to define the attachments for the task workflow (Figure 7-26 on page 239):
 - a. Click **Manage Attachments** at the top of the Step Editor.
 - b. Click **Add Attachment**.
 - c. In the **Attachment Name** field, enter CorrespondingLetters (no spaces allowed).
 - d. In the **Prompt** field, enter Corresponding Letters.
 - e. Click **OK**.
 - f. Repeat these steps for the attachment SupportingDocuments.

g. Click **Close**.

Requirement: For a case worker to see the attachment when opening a work item, you must define the attachment by using **Add Attachments**. After you define it, use step properties to add the attachment to the step. The attachment must be created in each task that contains steps where the attachment is exposed.

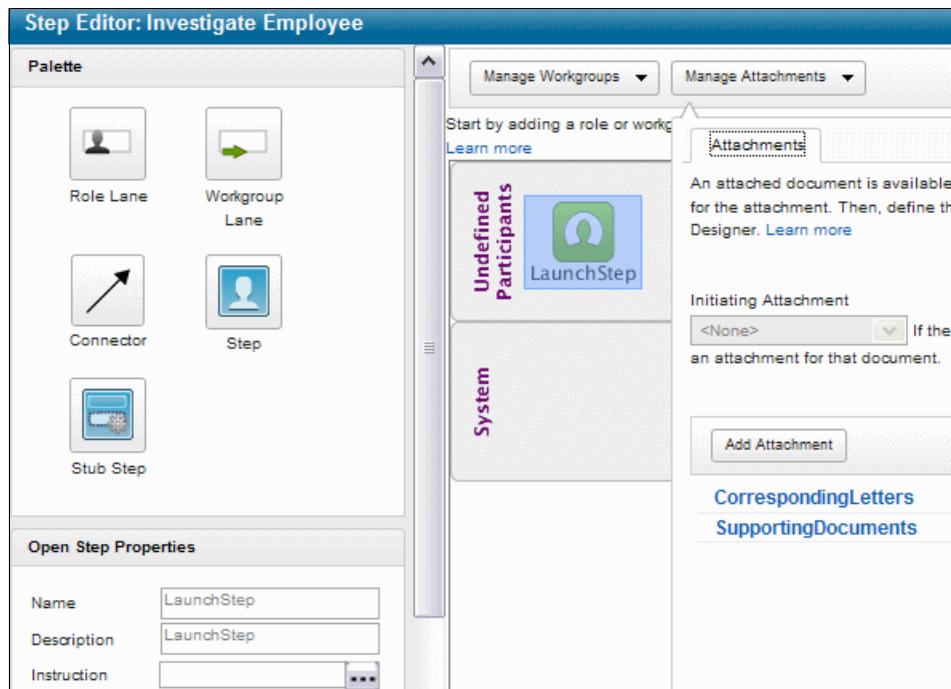


Figure 7-26 Defining attachments to use in the Investigate Employee task diagram

2. Update step properties for the Launch Step by completing these steps:
 - a. Select the **Launch Step**.
 - b. In the Step Properties pane, click the **Edit** icon to the right of the **Attachments** field.

- c. Select **CorrespondingLetters** and **SupportingDocuments** from the Select Attachment menu as shown in Figure 7-27. Click **OK** to accept the selection.

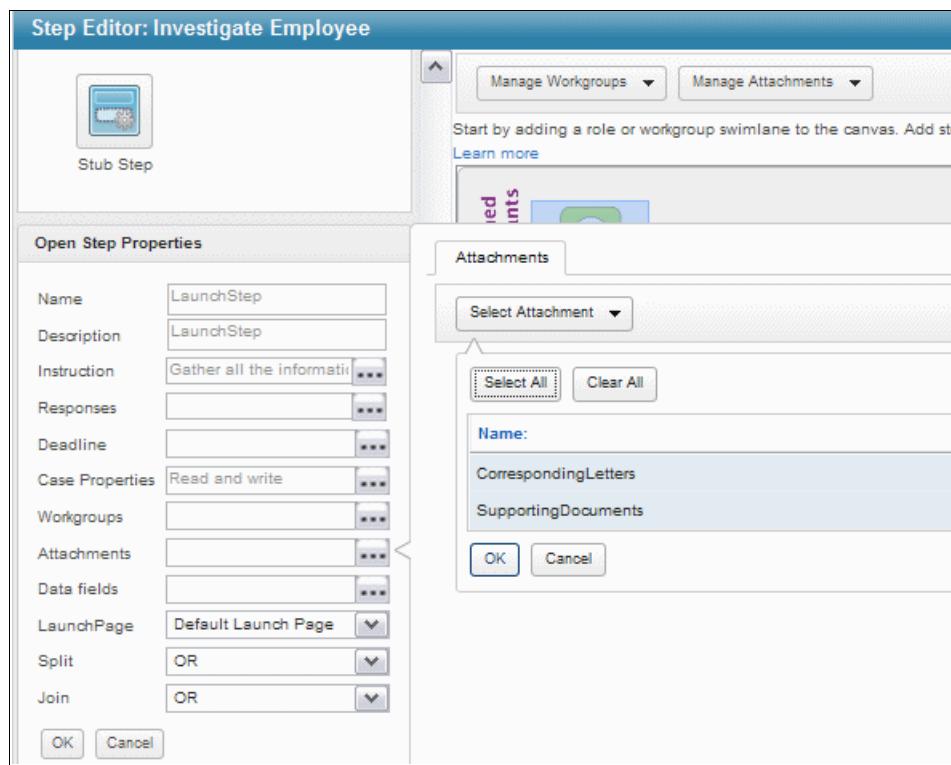


Figure 7-27 Adding attachments to the Launch Step

- d. Click **OK** to finish adding attachments for the Launch Step.
- e. In the Step Properties pane, click the **Edit** icon to the right of the **Case Properties** field.
- f. On the **Case Properties** tab, click **Select Property**.
- g. Click **Select all** to choose all the case properties as the step parameters, and click **OK**.
- h. In the Step Properties pane, click **OK** to save the changes for the Launch Step.
3. Create role swimlane and step by using the procedure in 7.1.1, “Creating the Upgrade Product task diagram” on page 214:
- Select **Manager** for the swimlane role.
 - Enter **Investigate Employee** for the step name.

- Select all case properties as the step parameters.
- Draw the connector from the Launch Step to the Investigate Employee step.

Figure 7-28 shows the task diagram for the Investigate Employee task.

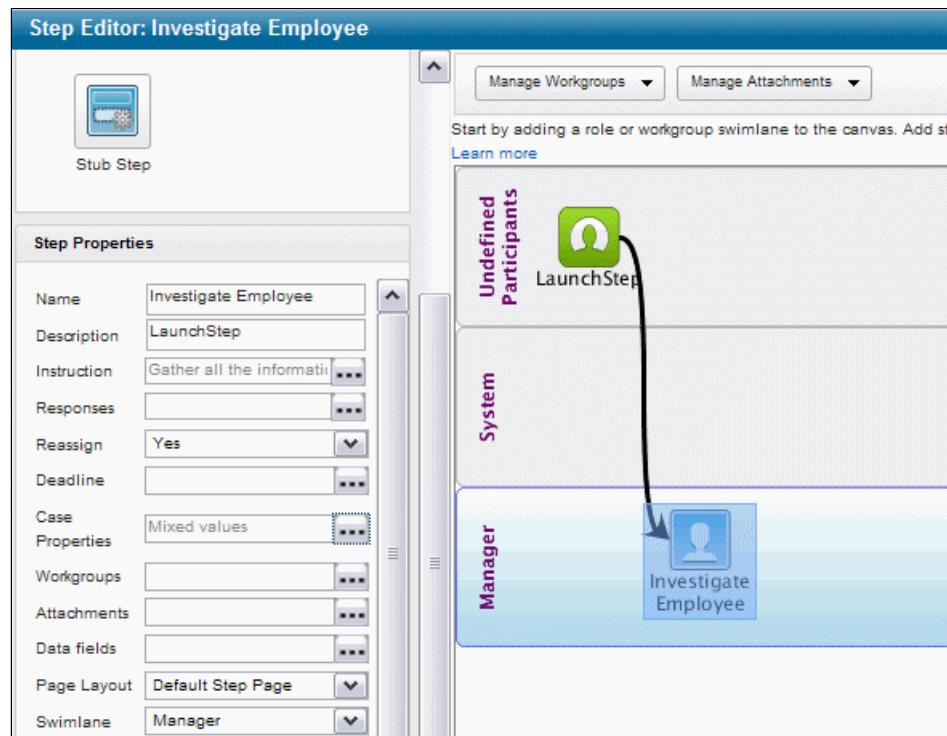


Figure 7-28 Investigate Employee task diagram

7.1.12 Creating the Request Assistance task diagram

The Request Assistance is another user-created task. Use the procedure in 7.1.11, “Creating the Investigate Employee task diagram” on page 238 to configure the Launch step and create the task diagram:

- For attachments, select **SupportingDocuments**.
- For first role swimlane, select **Contact Center**.

3. For the step on the Contact Center:
 - a. Enter Request Help as its step name.
 - b. For the case properties, select all properties.
 - c. For the attachment, select **SupportingDocuments**.
4. For the second role swimlane, select **Specialist**.
5. For the step on the Specialist swimlane:
 - a. Enter Process Request as its step name.
 - b. For the case properties, select all properties.
 - c. For the attachment, select **SupportingDocuments**.

The Request Assistance task is shown in Figure 7-29.

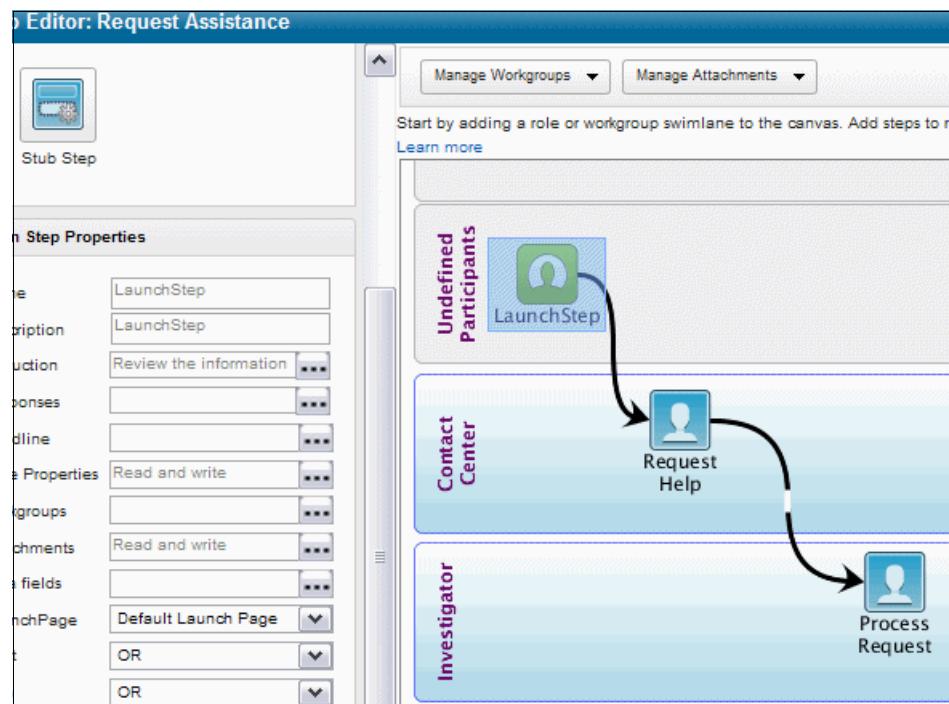


Figure 7-29 Request Assistance task diagram

7.2 Saving and validating the solution

To complete the solution, save and validate it using the following steps:

1. From the Case Type window, click **Save** as shown in Figure 7-30.

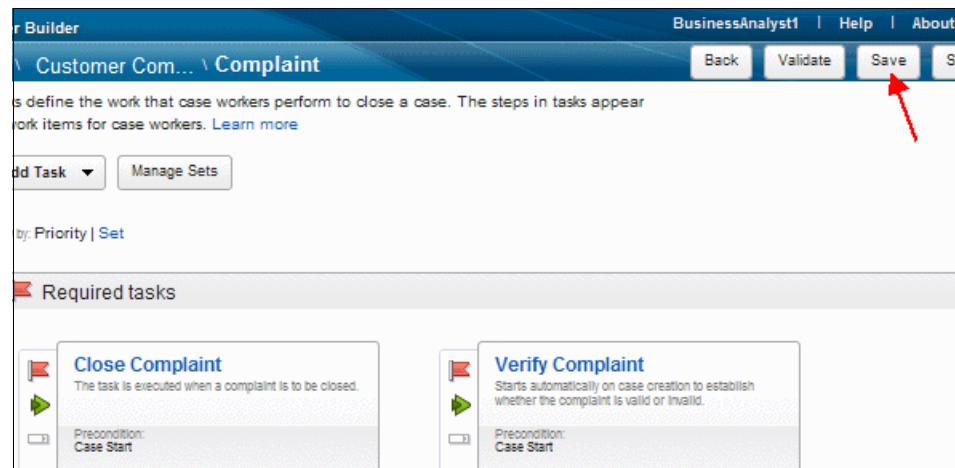


Figure 7-30 Saving the tasks

2. Click **Validate** as shown in Figure 7-31.

Remember: This validation is only for this Case type and its collection of Tasks. This process does not validate the entire solution.

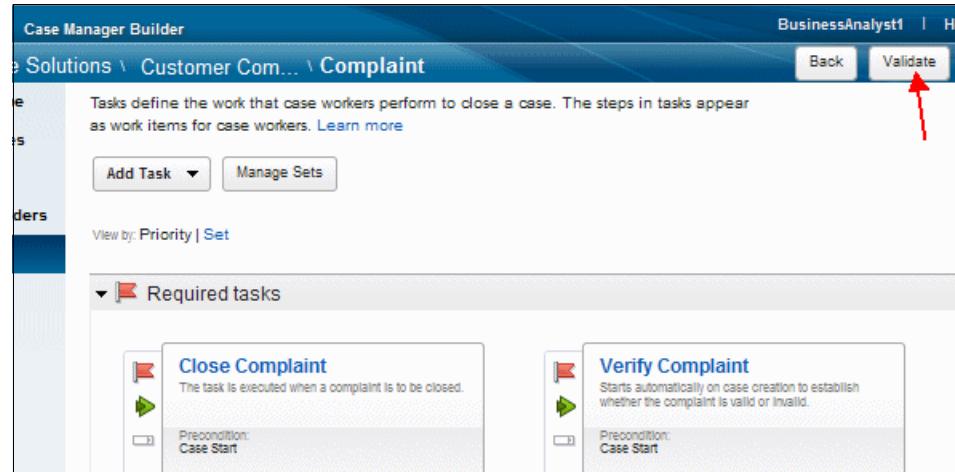


Figure 7-31 Validating the case type

3. Click **Save and Close** to save the solution.

The solution is now complete. After you build the solution, you must deploy it in an environment before you can use it.

7.3 Deploying the Customer Complaints solution

In the development environment, use Case Manager Builder to deploy the solution by completing the following steps:

1. On the **Manage Solutions** page, select the **Customer Complaints** solution.
2. Click **Deploy** as shown in Figure 7-32.

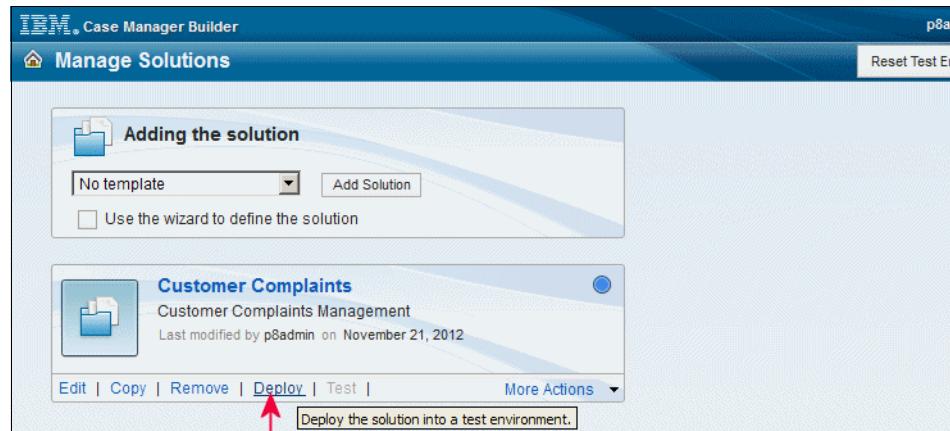


Figure 7-32 Deploying the Customer Complaints solution

3. After the deployment is completed, Case Manager Builder displays the deployment status for the solution on the **Manage Solutions** page. For the simple Customer Complaints solution, Case Manager Builder displays the successful deployment as shown in Figure 7-33.

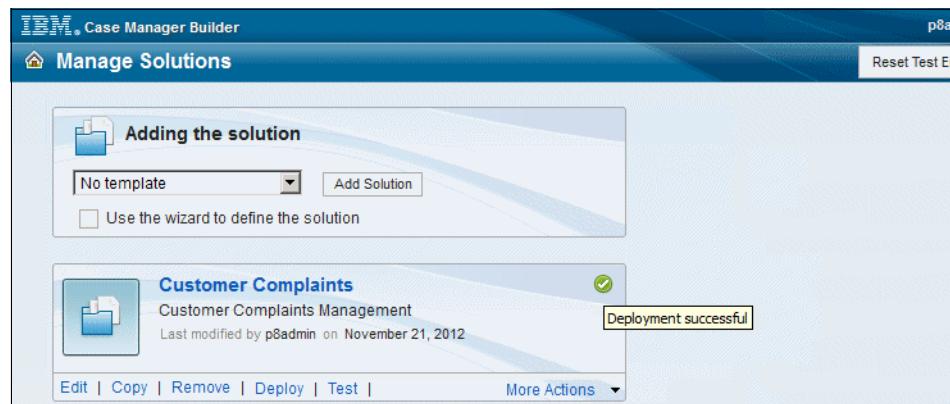


Figure 7-33 Successful deployment message

Troubleshooting: If the deployment is not successful, check the deployment detail and deployment error logs from the **More Actions** menu (Figure 7-34).

The deployment might take a while. The status bar shows the deployment status.

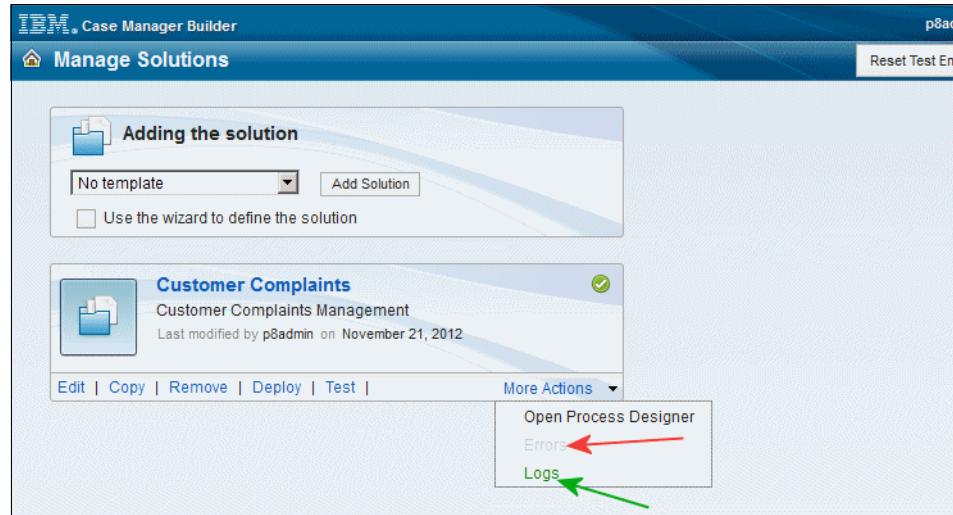


Figure 7-34 Locating the deployment logs

7.4 Testing the Customer Complaints solution

The users Bill, Carl, Ian, Isabella, Mary, Steve, and Sally are used to create cases and process the work for the roles specified in the sample solution. Make sure to add these users to your LDAP server before you begin. If you cannot add these users to your LDAP server, replace them with the appropriate users from your LDAP.

Requirement: Follow the guidelines in the “Planning for IBM Case Manager security” section in IBM Case Manager Information Center to give users access to the cases and tasks.

After the successful deployment, test the solution by completing these steps:

1. From Case Manager Builder, click **Test** for the Customer Complaints solution as shown in Figure 7-35.



Figure 7-35 Testing the solution

2. After Case Manager Client displays the solution space page, click **Manage Roles** to assign the users and groups to the roles. The Manage Roles window is displayed for the solution as shown in Figure 7-36.

The screenshot shows the 'Customer Complaints' solution space in Case Manager Client. On the left, there's a sidebar with 'Work' and 'Cases' tabs, 'Add Case', and 'In-basket' which says 'Displays the in-basket'. The main area is titled 'Manage Roles' and lists several roles: 'Billing Agent', 'Contact Center' (which is selected and highlighted in blue), 'Investigator', 'Manager', and 'Sales Agent'. To the right, there's a panel with 'Add Members' and 'Delete' buttons, and a section titled 'Contact Center Members' which is currently empty.

Figure 7-36 Manage Roles window

3. In the **Manage Roles** window, click **Add Members** as shown in Figure 7-37.

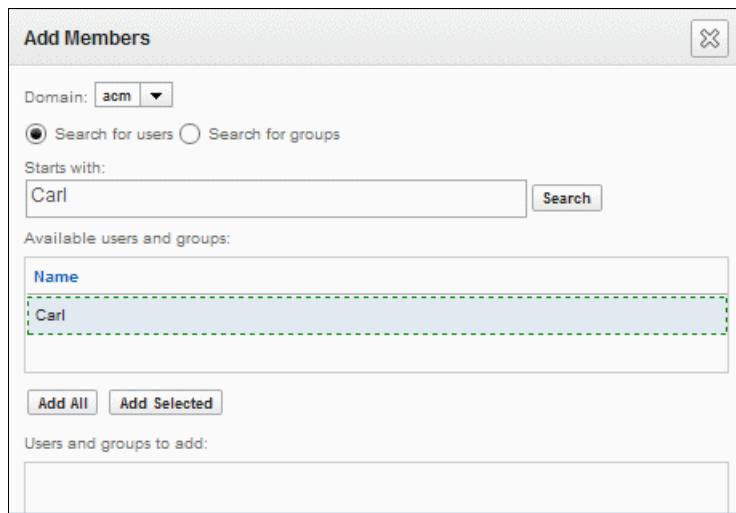


Figure 7-37 Adding Carl to the Contact Center role

4. Add the members by completing these steps:
 - Highlight **Contact Center**.
 - Click **Add Members**.
 - Enter **Carl** for the **Start with** field, then click **Search**.
 - Highlight **Carl**, then click **Add Selected**.
 - Click **OK** to add Carl to the Contact Center role.
 - Repeat these steps for the other roles:
 - For the Billing Agent role, add Bill
 - For the Investigate role, add Ian and Isabelle.
 - For the Manager role, add Mary.
 - For the Specialist role, add Steve.
 - For the Sales Agent role, add Sally
 - Click **OK** after you add the members to all roles. The **Manage Roles** pane is closed.
5. Log out from Case Manager Client.
6. Log in as Carl to create a case.

If you get the “Welcome to case management” window, go to the Customer Complaint solution from the spaces menu:

- a. Click the link at the top **Go to Spaces**.
- b. Select the **Customer Complaints Solution**. The Contact Center Work page is displayed.

If you use the same user to log in, you go back to the same solution page.

Tip: If you use the same browser session as a different user login, clear the browser cache before you log in.

7. Click **Add Case → Complaint** as shown in Figure 7-38.

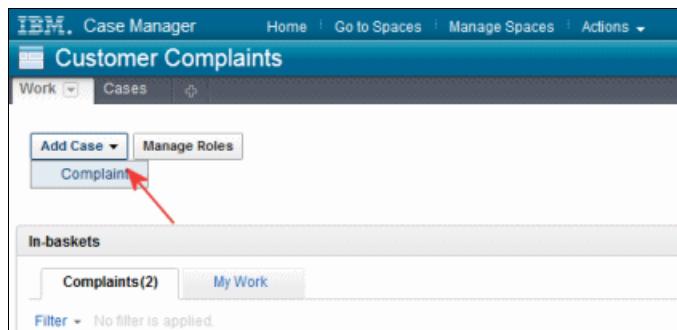


Figure 7-38 Selecting the Complaint case type for add case

Figure 7-39 displays the Add Case page for the Complaint case type.

A screenshot of the "Add Case" window for the "Complaint" case type. The window has a title bar with "Work", "Cases", and "Add Case" buttons. The main area is titled "Complaint". It contains a "Case Data" section with three expandable sections: "Customer Information", "Complaint Information", and "Flags". At the top right of the window are "Add" and "Cancel" buttons.

Figure 7-39 Add Case window

8. Expand **Customer Information** and enter the customer information.
Figure 7-40 displays sample customer information.

The screenshot shows a software interface for managing cases. At the top, there's a navigation bar with tabs for 'Work', 'Cases', and 'Add Case'. Below the navigation bar, the title 'Complaint' is displayed. On the right side of the interface, there are two buttons: 'Add' and 'Cancel'. The main area is titled 'Case Data' and contains a section titled 'Customer Information'. This section includes fields for Customer Number (1234), Customer Name (Customer One), Address (1 Manager Lane), Customer City (Blossom City), Customer State (CA), Email (CustomerOne@his.com), Telephone (555-123-4567), Customer Since (2,000), and Customer Rating (Premium). There are also sections for 'Complaint Information' and 'Flags' which are currently collapsed.

Figure 7-40 Entering the customer information

9. Do likewise for Complaint Information and Flags. For this example,
 - a. Set the Category to Product.
 - b. Set Valid flag to true.
 - c. Set Upsale Opportunity flag to true.

10. Click **Add** to add the case. The tasks for the new case are shown in Figure 7-41:

- a. The Send Corresponding Letter task is in the Waiting state waiting for a document of type Correspondence to be filed in the case.
- b. The Upsell Opportunity container task is started automatically when the condition Upsale Opportunity = true is met.
- c. The Call Customer subtask of the Upsell Opportunity container task is started automatically.
- d. The Upgrade Plan subtask is disabled because it belongs to an exclusive set. The Upgrade Product subtask is already started when the preconditions Upsale Opportunity = true and Upgrade Category = Product are met. See Table 6-1 on page 161 for the list of exclusive tasks. For more information about the procedure to specify tasks in the exclusive set, see “Setting up tasks for the solution” on page 195.
- e. The Review Product Complaint task is started because the conditions Product Category = ‘Product’ and Valid = true are met.

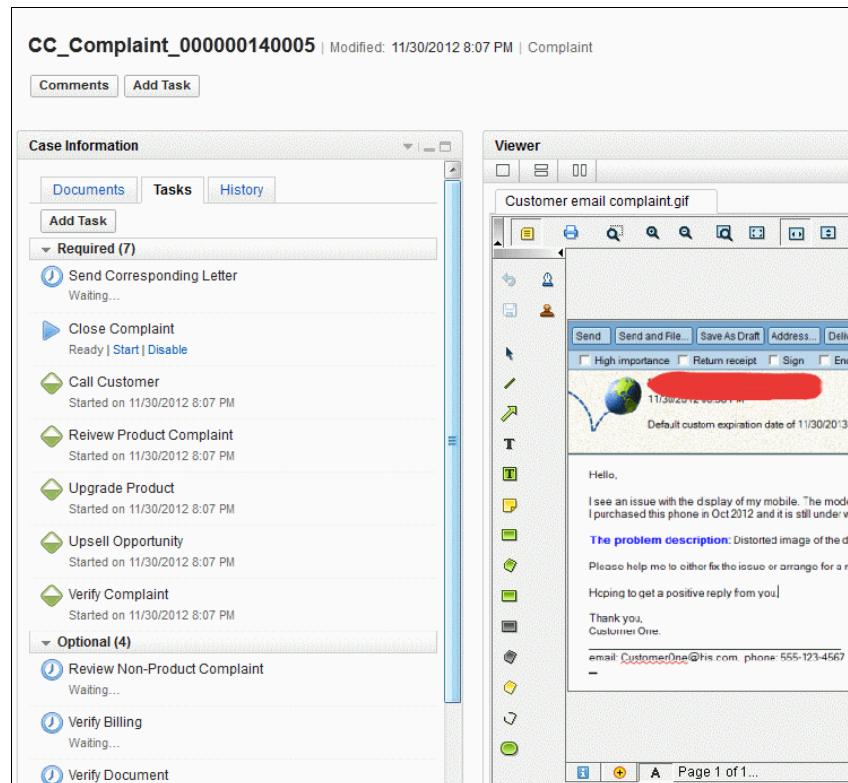


Figure 7-41 List of tasks for a running case

11. The Close Complaint task is created but not started because it is a manual task. Similarly, the Investigate Product Safety task is created but not started because it is a manual task.
12. Because Carl is logged in, Carl can process the work item in the Contact Center in-basket, Complaint. Figure 7-42 shows the work item in the Complaints in-basket. Carl opens the work item in the Complaints in-basket, sets the Complaint Status to **Processing**, and completes the work.

In-baskets			
Complaints(1)		My Work	
Filter ▾ No filter is applied.			
Items 1 - 1			
Step Name	Time Created	Subject	▲ Complaint Receive Customer Ra
Verify	11/30/2012 7:00 PM	Verify Complaint	11/15/2012 12:00 AM Premium

Figure 7-42 In-basket for Contact Center role

13. Similarly, Steve logs in to process the work for the Review Product Complaint task. Steve determines that the complaint is valid, and completes the work.

The case now has two completed tasks as shown in Figure 7-43.

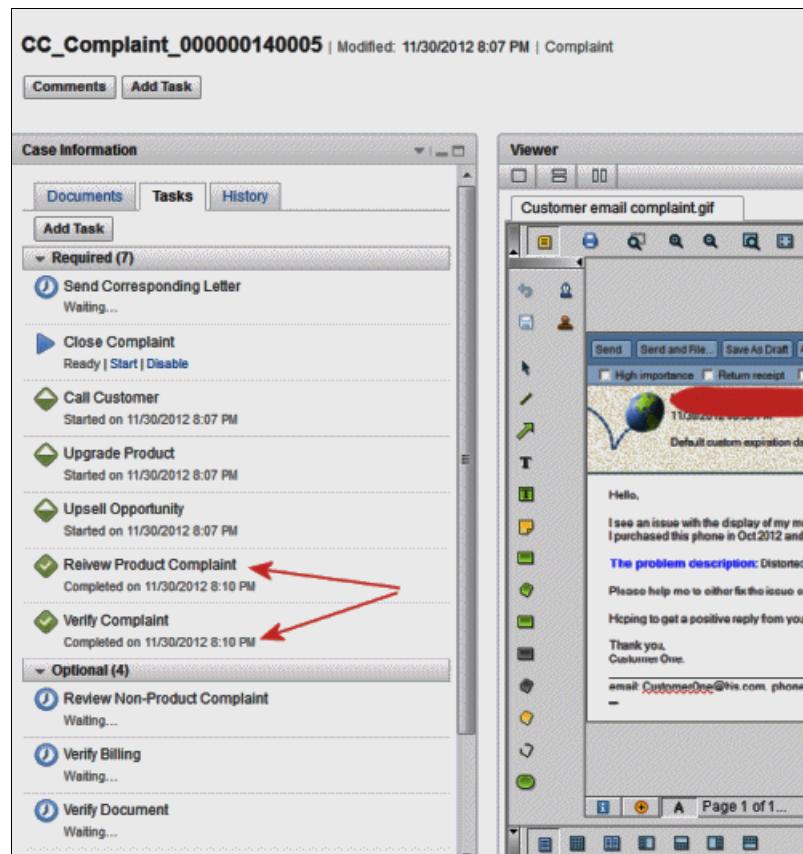


Figure 7-43 Current completed tasks

14. The Contact Center creates a letter to inform the customer of the case status in the Letter Templates folder and files it in the current case. File the document to the subfolder *Correspondence* of the case instance. See step 11 on page 193 in “Setting up case type” on page 187 for setting case folder structure for a case type. To file the letter, follow these steps.
- Click the **Documents** tab of the **Case Information**.
 - Double-click the **Correspondence** folder.

- c. Click **Add** → **Add Document** as shown in Figure 7-44.

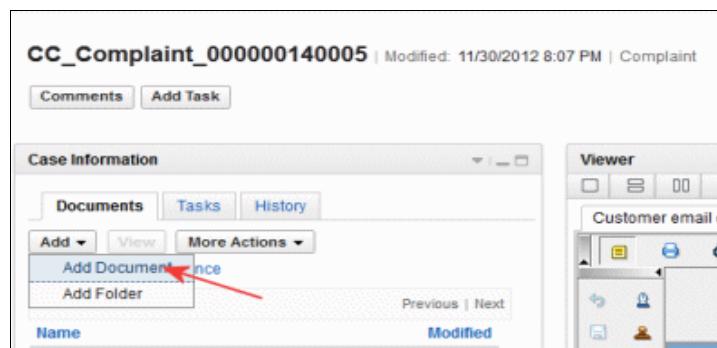


Figure 7-44 Selecting Add Document

- d. From **Add Document**, select **The Repository** and click **Next**.
- e. For **Search in**, browse to the location of the document. In this example, the letter is in the Letter Templates folder.
- f. For **Class**, select **Correspondence**.
- g. Enter a few characters for the document for the **Document title starts with** to search for the document.
- h. After you select the document, click **Search**. See Figure 7-45.

A screenshot of the 'Select Document' dialog. On the left is the 'Case Information' panel from Figure 7-44. On the right is the 'Select Document' dialog. It has sections for 'Search Criteria' and 'Search Results'. The results show 'Items 1 - 1' for 'Name'. Below it is a 'Version Information' section for 'Correspondence Letter to Customer One.odp', showing it was modified on '11/30/2012 7:39 PM' by 'p8admin'. There are also 'Previous' and 'Next' buttons at the bottom of both panels.

Figure 7-45 Selecting a document of the correct document type

- i. Click **Add** and then click **OK**. The Send Corresponding Letter task is started as shown in Figure 7-46.

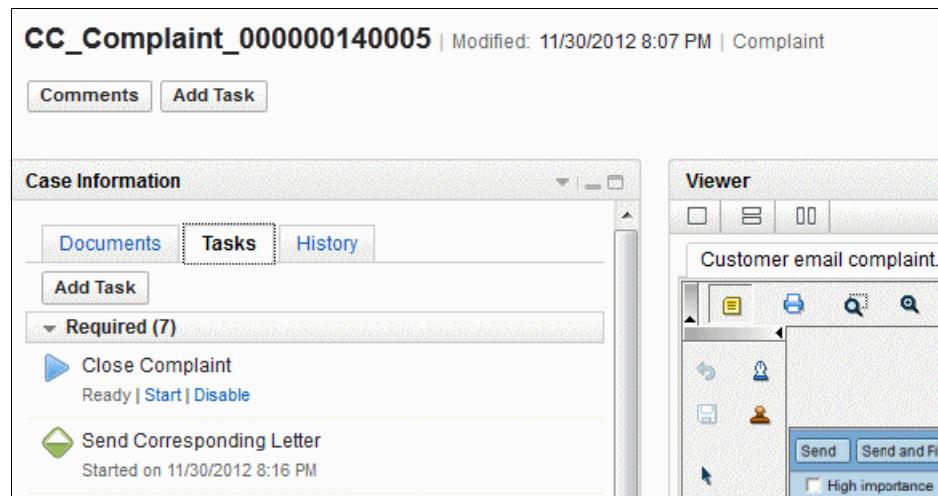


Figure 7-46 Send Corresponding Letter task started

15. Carl processes the work for the Send Corresponding Letter task.
16. Sally processes the work for the Call Customer and Upgrade Product tasks, closing the possible upsell opportunity. Processing these two tasks completes the Upsell Opportunity task.
17. Steve reviews the case information, then sets the Complaint Status to **Close**. This step completes the work for the Close Complaint task.

All the required tasks for the current case are completed. See Figure 7-47.

The screenshot shows the IBM Case Manager interface for a case titled "CC_Complaint_000000140005". The top navigation bar includes "Comments" and "Add Task" buttons. Below this is a "Case Information" panel with tabs for "Documents", "Tasks" (which is selected), and "History". The "Tasks" tab has a sub-section titled "Required (7)" which lists seven completed tasks:

- Call Customer (Completed on 11/30/2012 8:12 PM)
- Close Complaint (Completed on 11/30/2012 8:19 PM)
- Review Product Complaint (Completed on 11/30/2012 8:10 PM)
- Send Corresponding Letter (Completed on 11/30/2012 8:18 PM)
- Upgrade Product (Completed on 11/30/2012 8:13 PM)
- Upsell Opportunity (Completed on 11/30/2012 8:13 PM)

To the right of the tasks is a "Viewer" panel displaying an email message from a customer. The email subject is "Customer email complaint.gi". The message body contains a greeting, a statement about a product issue, a problem description, and a request for help.

Figure 7-47 List of all completed tasks

Figure 7-48 shows the case is completed.

This screenshot shows the details of the completed case "CC_Complaint_000000140005". The case status is listed as "Closed". The "Case state" field is highlighted with a red arrow and shows the value "Complete". Other fields visible include Case Number, Case Source (Email), Complaint Status (Closed), Complaint Category (Product), Complaint Received Date (11/16/2012 12:00 AM), Customer Name (Customer One), Customer Number (1234), Customer Rating (Premium), Part Number (A123), Case Type (Complaint), Modified By (p8admin), and Date Modified (11/30/2012 8:21 PM). A "Case ID" link is also present.

Figure 7-48 Case completed window

7.5 Improving the solution

What is presented in this chapter and the previous one is a simple solution. Solution development usually takes multiple iterations. The next iteration of solution development can improve the following areas:

- ▶ Solution icon
- ▶ Property choice list
- ▶ Solution properties
- ▶ Document properties
- ▶ In-basket properties

It is beyond the scope of this book to provide detail implementation steps for these areas. These potential improvements are mentioned to showcase the improvements you can build into the solution for future iterations.

7.5.1 Solution icon

The current default icon (Figure 7-49) might not depict the solution properly. You might want to change it to another icon by selecting one of the icons that are bundled with IBM Case Manager.



Figure 7-49 Current solution icon

7.5.2 Property choice list

Table 6-6 on page 166 lists the following case states: Open, Closed, Processing, Pending, and On-Hold. To produce a case status report for the number of invalid cases, you might want to have more choices such as Invalid. This choice allows you to differentiate between cases that are closed because the case workers resolve them, and those that the case workers determined were invalid.

7.5.3 Solution properties

You can handle the solution properties for the customer address better by breaking up the address into smaller pieces. Instead of Address, Customer City, and Customer State, you can have the properties Customer Street Address, Customer City, Customer State, and Customer Zip Code. You can then construct the customer address by using these properties. Also, you might want to design

the properties for the customer address to handle the format differences between nationalities.

7.5.4 Document properties

Currently, a document associates only with a single case property, Case Number. To search for the documents with different attributes such as document received date and case priority, you must add more properties for the documents.

7.5.5 In-basket properties

If you want to sort the in-basket by Customer Number and Case Number, you must modify the in-basket definition to sort by Customer Number and Case Number.



Solution deployment

After a solution is developed and tested, it will typically be necessary to migrate the solution from a development environment to a testing or production environment. This process can be involved, and can incorporate a number of tools and techniques. This chapter describes the deployment process.

This chapter includes the following sections:

- ▶ Deploying a solution
- ▶ Prerequisite steps
- ▶ Identification
- ▶ Migrating external assets and processes
- ▶ Migrating non-Case-Manager FileNet-based assets
- ▶ Migrating a Case Manager solution

8.1 Deploying a solution

To deploy an IBM Case Manager solution, there are several types of assets that must be exported, manipulated, and imported into the target environment. These combined streams of data recreate the solution in the target environment. The following are the types of assets involved:

- ▶ Case Manager solutions (migrated by using Case Manager Administration Client)
- ▶ Non-Case Manager specific and shared assets (migrated by using FileNet Deployment Manager) such as:
 - Classes, properties, and choice lists
 - Workflow definitions
 - Content-based objects (forms, search templates, and form templates)
- ▶ External assets and procedures that must be migrated outside of FileNet Deployment Manager or Case Manager Administration Client, such as:
 - Rules
 - Custom widgets
 - Deployable code, such as services and Process Engine components

Figure 8-1 provides a visual overview of the process migration process, which is broken up into three classes of assets. The diagram is also broken into the source and target environments, and shows the steps relative to each other. The migration process often occurs in parallel, with each of the classes of assets being migrated separately.

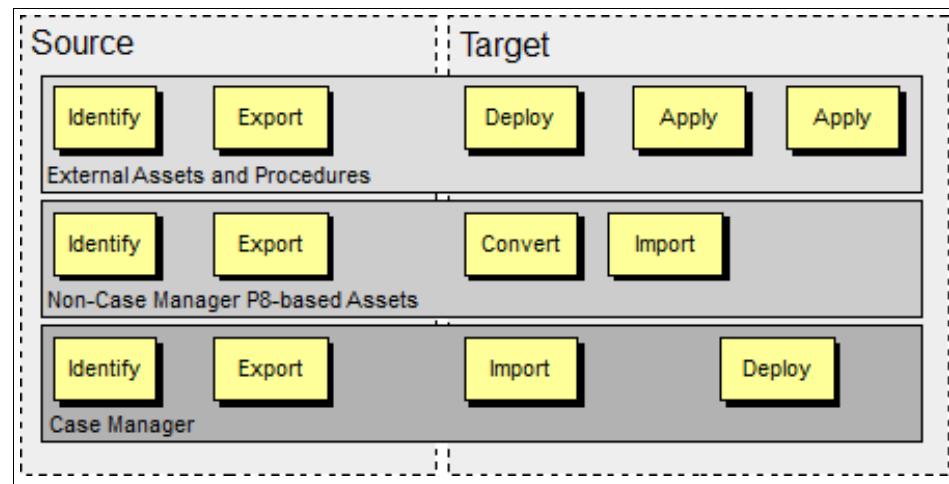


Figure 8-1 Migration timeline

Some elements will need to be deployed before others. In the figure, they are indicated by gaps in the deployment timeline. Specifically, before a Case Manager solution can be deployed, many of the supporting assets must be in place. In addition, there are often post deployment procedures that must be completed.

Important: Although some items can be done in parallel, deploy all code and non-Case Manager assets to the target environment before you import and deploy the Case Manager solution.

8.1.1 Tools for migration

For migrating a solution, Case Manager provides two key tools to move the various assets and configurations that are created within the platform. These tools are used to migrate new solutions and changes to existing solutions between environments. They can also be used to export solutions for storage in an external source code management system such as IBM Rational Jazz™.

- ▶ Case Manager Administration Client

Case Manager Administration Client (CMAC) is a ready-for-use tool that is used to both install a Case Manager system and to help migrate solutions between environments. For migration, CMAC can do the following tasks:

- Export and import solution packages
- Deploy a solution to the production target object store
- Copy solutions

- ▶ FileNet Deployment Manager

FileNet Deployment Manager ships with the base IBM FileNet P8 Platform, and is used to migrate Content Engine and certain Process Engine elements from one environment to another. For more information about FileNet Deployment Manager, see the Information Center page at:

http://pic.dhe.ibm.com/infocenter/p8docs/v5r1m0/topic/com.ibm.p8.common.deploy.doc/overview_intro.htm

In addition to the tools listed above, the following tools can also be useful in both gathering information and solution migration:

- ▶ Solution Document Generation

The Solution Document Generation tool creates a report of the Case Manager native assets that are used in a solution. It reads the solution definition and Process Engine configuration settings objects in the design object store, and reports the assets and tasks that are used in the solution.

For documentation about the Solution Document Generation tool, see the “Creating a solution description document” in the IBM Case Manager Information Center at:

<http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/topic/com.ibm.casemgmt.design.doc/acmdt018.htm>

► Other tools

In addition to the tools listed above, a deployment effort will likely incorporate more tools in the IBM FileNet P8 suite, and other tools for other products and solutions. These tools can include:

- Process Task Manager
- Process Configuration Console
- IBM WebSphere Administration Console
- Platform-specific tools for migrating web services, rules, and other third-party integration components

8.1.2 Effects of redeployment

A solution can be redeployed to a target object store. Deploying a solution for the first time creates various objects in the FileNet Content Engine, Process Engine, and Business Space. Understanding the redeployment effects on existing objects is critical to ensure accurate changes are made to the run time.

For a detailed list of redeployment effects, see “Redeployment restrictions for modifying a solution” in the IBM Case Manager Information Center at:

<http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/topic/com.ibm.casemgmt.design.doc/acmdc024.htm>

In terms of security, any object that is already created and has security configured maintains that security configuration. If inherited security is used and properly configured, no further security configuration should be necessary for new objects added to the solution.

8.2 Prerequisite steps

Before you deploying a solution, complete the following prerequisite steps to prepare the deployment environment and configure the tools:

- ▶ Configuring the environments within FileNet Deployment Manager.
- ▶ Configuring the profile in Case Manager Administration Client, if a CMAC-based solution deployment is necessary.
- ▶ Identifying shared locations for solution deployment assets (such as compressed files) in the case of a disconnected deployment.

Important: In addition to these steps, ensure that the source and target versions of Case Manager are the same. Version mismatches can lead to migration issues and unexpected functionality changes.

8.2.1 FileNet Deployment Manager

To accomplish a migration, FileNet Deployment Manager works on a source-target pair where the data is extracted from the source and imported into the target. FileNet Deployment Manager is also capable of mapping security principals, object stores, and web services links. This mapping is accomplished by preparing two half maps, described in 8.5.1, “Half-maps” on page 271.

A migration can be accomplished in one of two ways:

- ▶ A connected deployment where one FileNet Deployment Manager instance is able to connect to both environments at the same time.
- ▶ A disconnected deployment where one FileNet Deployment Manager instance is used to extract and a second FileNet Deployment Manager instance, on a separate computer, is used to convert and import.

Regardless of which process you use, the steps are basically the same. Any differences will be noted in the following descriptions.

Initial FileNet Deployment Manager configuration

Before beginning, a profile must be created. By default, this profile is created in the default locations that are listed below. However, this location can be changed by using the window in Figure 8-2 on page 264.

- ▶ Windows: <install directory>\IBM\FileNet\ContentEngine\tools\deploy
- ▶ UNIX: <install directory>/IBM/FileNet/ContentEngine/tools/deploy

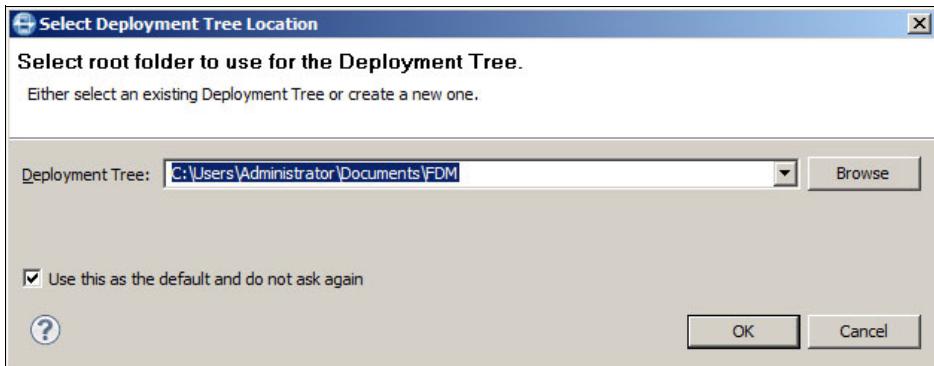


Figure 8-2 Creating a Deployment Manager deployment tree

Remember: Regardless of where the profile is created, make note of the path. This information is necessary if you need to review the log files that are created by a deployment step.

Creating an environment

After a profile is created, one or more environments must be created. These environments are used to gather the following information:

- ▶ Object Stores
- ▶ Security Principals
- ▶ Service Data (such as web services links)
- ▶ Export Manifests (for source systems)

Tip: When working in disconnected mode, the source system needs only one environment, but the target system needs both a source and target. For more information about expanding an extract package in a disconnected environment, see 8.5.7, “Disconnected deployment” on page 284.

After the profile is created or loaded, create new environments by completing the following steps:

1. Create the environment as shown in Figure 8-3.



Figure 8-3 Creating a new environment

2. Provide a name for the environment, ensuring the name is unique and descriptive as shown in Figure 8-4.



Figure 8-4 Naming an environment

3. Double-click the environment that you just created to edit connection settings as shown in Figure 8-5. Enter the connection settings and credentials, including user name, password, host, port, and URL.

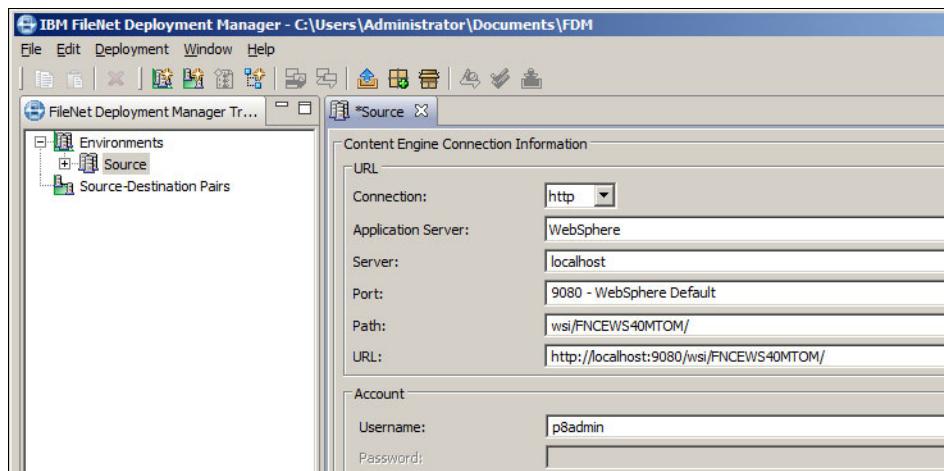


Figure 8-5 Configuring credentials and connection settings

This procedure must be completed for each environment that you will use during the migration. If the migration is to be completed in disconnected mode, environment creation must be done on each of the systems.

8.2.2 Case Manager Administration Client (CMAC)

Similar to FileNet Deployment Manager, CMAC needs a profile to store its settings. However, CMAC needs a unique profile for each environment. If it runs from the Case Manager server, only one profile must be created, and likely will have been created during installation. If not, complete the following steps to create a CMAC profile:

1. Create the profile.
2. Step through the wizard.
3. Select the **Production environment profile** and name the environment.
4. Configure the WebSphere settings and test the connection. This step is necessary for creating the profile, not for actual deployment.
5. Configure the connection to the Content Engine.
6. Click to clear every option except **Deploy Solution** and optionally **Register External Data Service**.

This process results in a profile appropriate for doing an IBM Case Manager solution deployment as shown in Figure 8-6.

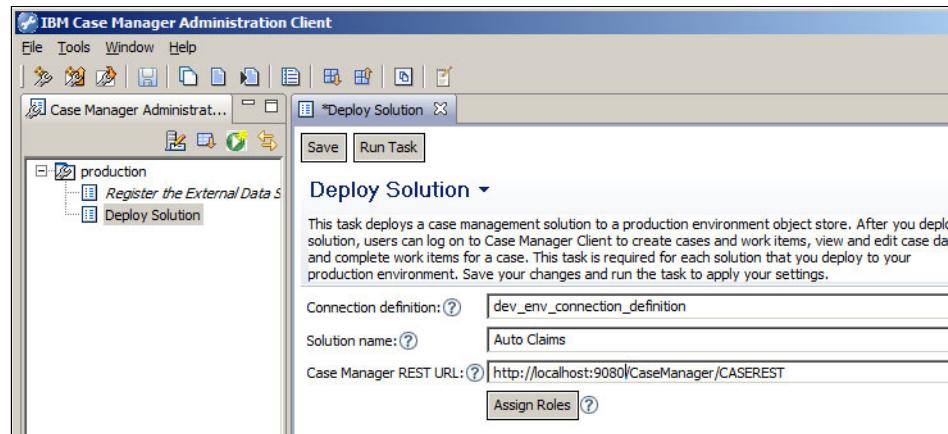


Figure 8-6 Case Manager Administration Client profile

Tip: For a deployment profile, the production deployment option allows deploying a solution from the design to the target environment without having to deploy Case Builder Client.

8.2.3 Content Engine

If you intend to persist key system fields on your objects, such as those listed below, use FileNet Enterprise Manager to enable the “Modify Certain System Properties” permission for any object stores that FileNet Deployment Manager will be importing assets into. This configuration allows the migration of the following system-level properties:

- ▶ Creator
- ▶ Create Date
- ▶ Last Modify User
- ▶ Modify Date

Modification of the permission is shown in Figure 8-7.

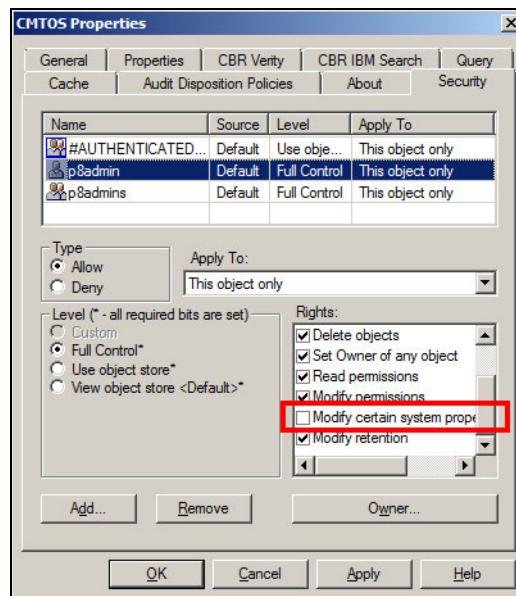


Figure 8-7 Modifying system properties for object stores

When you determine whether to migrate the system fields or not, you must understand how you plan on moving the elements. FileNet Deployment Manager has three modes for applying changes during a migration:

- ▶ Always update
- ▶ Never update
- ▶ Update if newer

The always update option ensures that the objects are in sync regardless of date that they are created. Update if newer updates only objects that have been changed. CMAC operates in an always update mode, meaning the whole solution is always pushed across with each update.

For non-Case Manager managed assets, you have the option of picking always update or update if newer. If you choose update if newer, you must enable the system property migration as described above.

8.3 Identification

The first step in doing a migration is to identify the assets to be migrated. This process includes creating a list of the following assets:

- ▶ Case Manager solutions.
- ▶ Non-Case Manager-specific assets such as:
 - Reused Classes
 - Reused Property Templates
 - Choice Lists
 - Form Templates
 - Search Templates
- ▶ Any non-IBM FileNet P8-based assets:
 - Rules
 - Custom code, such as custom services and Process Engine components
- ▶ Any manual changes that must be applied.

Use Table 8-1, Table 8-2, and Table 8-3 on page 270 to track the changes.

Table 8-1 Sample Non-Case Manager Assets worksheet

Type ^a	Object Store	Name, Path or ID ^b	Restrictions ^c

a. Type can be a folder, document, class, choice list, etc.

b. Make this either a path, name or GUID so the deployer can easily identify what needs to be migrated.

c. These restrictions can be to include/exclude subfolders, contents of folders, follow object valued property links, etc

Table 8-2 Sample Case Manager list

Solution Name	Notes

Table 8-3 Sample Non-FileNet Asset list

Target	Directions	

8.4 Migrating external assets and processes

When you migrate a solution, a number of non-Case Manager and FileNet P8-based assets and processes might need to be migrated to the new environment. These items have their own tools and procedures for migration, both automated and manual. The team responsible for deployment of the solution must work with each of the development teams to determine the appropriate method and order for migrating and deploying these assets. For more information about roles, see 9.1.1, “Roles” on page 290.

8.5 Migrating non-Case-Manager FileNet-based assets

Non-Case-Manager-specific assets can be migrated either at the same time as or after the migration of external assets. These assets are migrated by using FileNet Deployment Manager. This process is broken into a number of steps, which are divided by section in FileNet Deployment Manager:

- ▶ Exporting the assets from the source environment:
 - Build the object store half maps
 - Build the export manifest
 - Perform the export
 - Build the security principal half map from the exported data
 - Build the service data half map
 - Create an export package (optional)
- ▶ Building the target half map:
 - Build the object store half map
 - Build the security principal half map
 - Build the service data half map
- ▶ Create or update the source-destination pair:
 - Map the object stores
 - Map the principals
 - Map the services

- ▶ Migrate the assets:
 - Convert the assets
 - Perform a change impact analysis (optional)
 - Import the assets
- ▶ Validation:
 - Review the logs
 - Perform any post migration burn tests

8.5.1 Half-maps

FileNet Deployment Manager works by combining three sets of half-maps to create a single map for converting assets. These are used to map the following assets:

- ▶ Object stores
- ▶ Security principals
- ▶ Service maps

Each half map defines the various items that must be converted when you migrate from the source to target. To simplify the process of creating the final map, FileNet Deployment Manager allows all items to be labeled, providing a common identifier between the source and target environments.

When you create the final map, FileNet Deployment Manager attempts to automatically map the object stores, principals, and so on). In this example, however, FileNet Deployment Manager does not know that two object stores with different names are related. You can use labels to create a common name that is used to automatically map these items. Figure 8-8 shows how the target object store has different symbolic names (DevTargetOS in development and ProdTargetOS in production), but share a label of Target.

Label	Name	Symbolic Name
Design2	CMDEV	CMDEV
Design1	CMDOS	CMDOS
Target	CMTOS	DevTargetOS
Other	ECM	ECM
FilePlan	FPOS	FPOS

Label	Name	Symbolic Name
Design2	CMDEV	CMDEV
Design1	CMDOS	CMDOS
Target	CMTOS	ProdTargetOS
Other	ECM	ECM
Fileplan	FPOS	FPOS

Figure 8-8 Using labels in FileNet Deployment Manager

8.5.2 Exporting the assets

After you create the environment as described in “Creating an environment” on page 264, complete the following steps to export the assets:

- ▶ Build the half-maps
- ▶ Build the export manifests
- ▶ Export the assets
- ▶ Create a deployment data set (optional)

Building the half maps

When you export assets from a source system, a half map must be created. This half map provides the system-specific items included in the export, including the following data about the source system:

- ▶ Object store names and IDs

Object stores are retrieved directly from the Content Engine by clicking the **Retrieve Data**, or right-clicking the object store node in the source environment and selecting **Retrieve Data**.

When retrieving, you are given the option to merge or overwrite. Most of the time, use the merge option because this option only updates the half map with new object stores. Select overwrite if the current half map must be rebuilt.

After you retrieve the object stores, label them to simplify mapping. See Figure 8-8 on page 271 for an example of labeling object stores.

- ▶ Security principals

If you are migrating security, you must retrieve the security principals that are used in the source system. They can be retrieved directly from the FileNet Content Engine LDAP repository, or from an exported data set.

The exported data set generally contains fewer principals to map than retrieving directly from LDAP because FileNet Deployment Manager imports only the principals that are included on the Access Control Lists for the objects being exported. For more information about how to import principal data from an exported data set, see “Importing the security principals from the export” on page 275.

Also, just like the object stores, you are given the option of merging or overwriting. Only use overwrite if the existing collection of principals has been corrupted.

- ▶ Any service data used in the solution, such as web service URLs in forms

The service data is always retrieved from an export data set. This step can be skipped if no service URLs are in use in the solution or objects that are being exported. This might occur when the export only contains properties and classes.

Building the export manifests

When you perform a migration by using FileNet Deployment Manager, you must create one or more export manifests. An export manifest contains a list of all assets to be exported and the options for exporting those items.

To build an export manifest, complete the following steps:

1. Create and name new export manifest.
2. Open the newly created Export Manifest.
3. Use the add assets tool to add any items to migrate. Figure 8-9 shows how to add assets to an export manifest.

Tip: When you select items to add, they must be selected in the right pane.

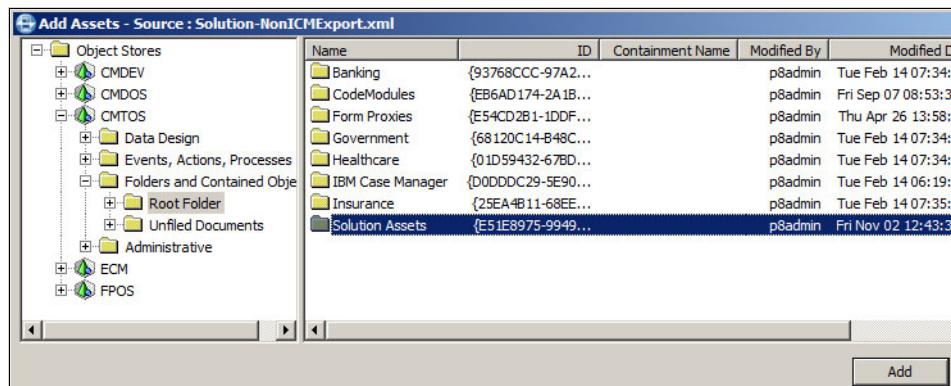
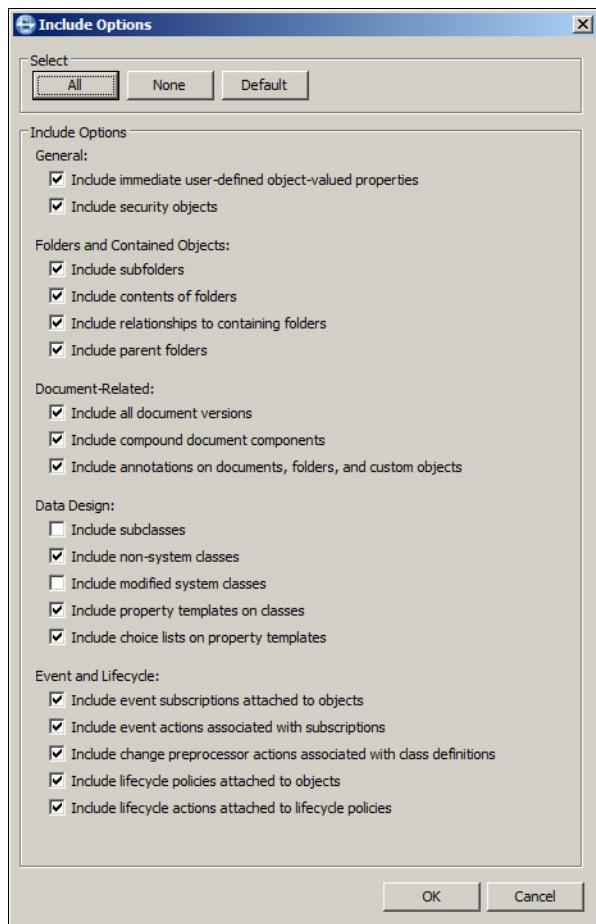


Figure 8-9 Adding assets to an export manifest

4. For each asset, modify the export options if necessary by double-clicking the asset or selecting several and right clicking them and selecting **Include Options**. Select (or click to clear) the appropriate export options as shown in Figure 8-10.



- Include subfolders (when a folder is to be moved, but not the contained folders)
- Include contents of folders (when a folder contains more objects than need to be moved)

For a full description of the options, see “Specify the include options of an asset” on the IBM FileNet P8 5.1 Information Center at:

http://pic.dhe.ibm.com/infocenter/p8docs/v5r1m0/topic/com.ibm.p8.common.deploy.doc/prep_data_ce_include_options.htm

In addition to the Information Center, there is a technote that further explains the export options and which options to select when you migrate a Case Manager solution. The technote is available at:

<http://www-01.ibm.com/support/docview.wss?uid=swg21612959>

5. Run the export by selecting the export manifest that contains the assets you want to migrate. This process extracts the definitions and content (if there is any) to the file system. Figure 8-11 shows the options for performing an export.

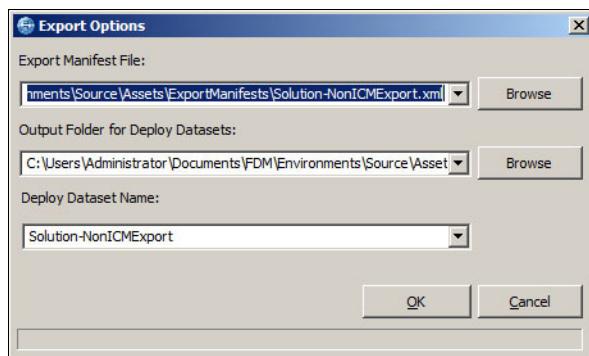


Figure 8-11 Exporting the assets

Remember: To complete this process, you must select the export manifest XML file by browsing for it.

Importing the security principals from the export

To map the security principals from the source to the target, read them from the extract (or from the Content Engine LDAP provider). Importing the users from the extract means that only principals included in the migration set must be mapped. Importing from LDAP results in a larger set of principals to manage.

To import the security principals from the export, complete the following steps:

1. Expand the environment node and right-click the **Security Principal Data** node.
2. Select **Retrieve Data**.
3. Select **Deploy Dataset File** as shown in Figure 8-12.

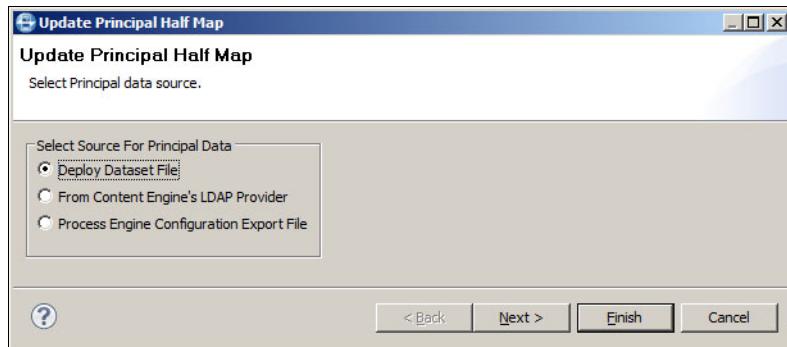


Figure 8-12 Importing security principal data from a deploy dataset

4. Select the extracted dataset folder.
5. Label the principals to help with automatic mapping as shown in Figure 8-8 on page 271.

Importing service data

Similar to importing the security principal data from an exported dataset, complete these steps to import the service data from that same dataset:

1. Select retrieve data for the service data.
2. Select the deploy dataset option.
3. Select the deploy dataset to extract from.
4. Open the imported data and label the services (optional).

Creating the deploy package

If wanted, FileNet Deployment Manager can create a deploy package. A deploy package is single compressed file that contains the environment definition and exported assets. This package can be:

- ▶ Stored in a source code management system like IBM Rational® Jazz and tied to a specific release.
- ▶ Moved to a separate FileNet Deployment Manager environment and used to recreate the environment on that system for disconnected deployment scenarios.
- ▶ Added to the Case Manager solution folder in Content Manager to allow migrating a single file that contains all assets for a solution.

For more information about how to expand the deployment package into an existing environment, see 8.5.7, “Disconnected deployment” on page 284.

Tip: If the complete deployment is being performed on a single FileNet Deployment Manager instance, this step is not required.

To create a deployment package, complete these steps:

1. Click **File → Deploy Package → Create Deploy Package**.
2. Browse for the extracted data set (the Deploy Dataset folder that is listed in the export completion window).
3. Select the target location for the deployment package.

The resulting file contains the definitions for the objects that are exported, the contents of any objects in the dataset, and the half maps created in the environment. All of these items allow you to recreate the environment on a separate FileNet Deployment Manager instance.

8.5.3 Building the target half map

Before you map and import the data, the destination environment must be created. This process involves the following steps:

- ▶ Retrieving and labeling the object store list
- ▶ Retrieving and labeling the principals
- ▶ Retrieving and labeling the service data

Retrieving object stores

Retrieving the object stores is the same process as source. For more information, see “Creating an environment” on page 264.

Retrieving security principals

Unlike the source environment, the security principals must be retrieved from a source other than the exported dataset. Typically the principals are retrieved from the FileNet Content Engine LDAP provider. Figure 8-13 on page 279 shows the update principal half map from LDAP dialog.

Remember: When you retrieve the principals from LDAP, be aware that improper settings can result in the complete contents of the LDAP repository being retrieved.

To retrieve the security data from LDAP, complete the following steps:

1. Load the security principals for the target environment (right-click the environment's security list and select **Retrieve Data**).
2. Import the security data from the FileNet Content Engine LDAP provider.
3. Retrieve and select the appropriate realm.
4. Determine which filter (if any) to apply. The following filters are available:
 - An environment's half map, which is useful if the user identifiers (short names) are shared between the two environments.
 - A file with a list of user IDs to retrieve, called a label file.

5. Label the users for simplified mapping as shown in Figure 8-13.

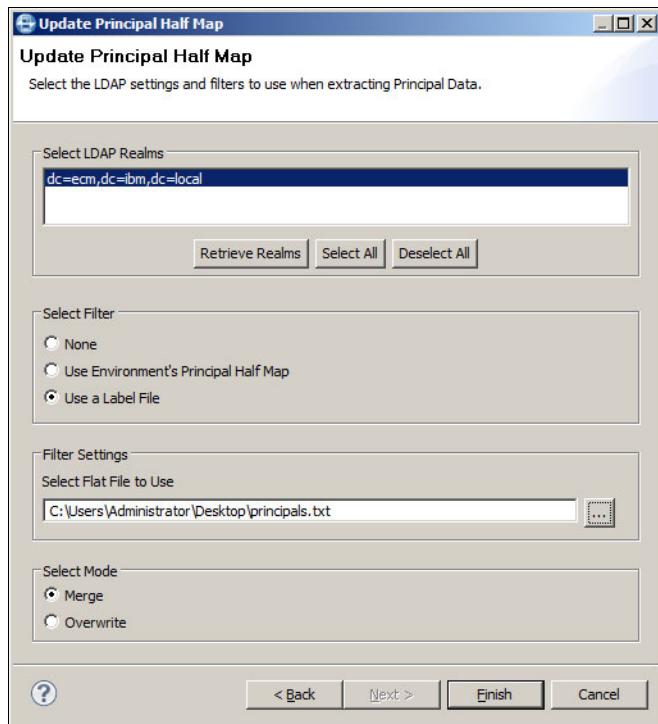


Figure 8-13 Update security principals half map with a label file filter applied

8.5.4 Creating the source-destination mappings

After you create and populate the source and destination, a source-destination pair can be created (or updated). Creating this mapping merges the two half-maps for the environments. This process creates the final map that translates the assets and objects from the source environment to the target environment.

Creating the source-destination pair

If you do not already have a source-destination pair, create one by either right-clicking the Source-Destination pairs node or by clicking **File** → **New** → **Source-Destination pair**. See Figure 8-14 for an example of creating a source-destination pair.

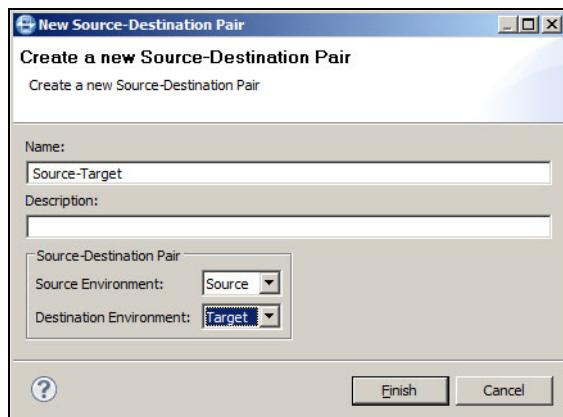


Figure 8-14 Creating a source-destination pair

Building the data maps

FileNet Deployment Manager uses the combination of the two half maps to complete the migration. This map defines how system-specific elements are translated during the migration process. This configuration is done by creating the maps for the following items:

- ▶ Object stores
- ▶ Security principals
- ▶ Service data

An example map is shown in Figure 8-15.

	Source Name	Destination Name	Source Symbolic Name	Destination Symbolic Name
1	CMDOS	CMDOS	CMDOS	CMDOS
2	CMDEV	CMDEV	CMDEV	CMDEV
3	ECM	ECM	ECM	ECM
4	CMTOS	CMTOS	DevTargetOS	ProdTargetOS
5	FPOS	FPOS	FPOS	FPOS

Figure 8-15 An object store map in FileNet Deployment Manager

Consideration: In each of the following steps, if the sources were labeled properly, creating or updating the maps automatically maps objects that share a label. Anything that has not been labeled or does not contain a matching label must be manually mapped.

8.5.5 Importing the assets

After the map is built, you can perform the following steps for importing the assets:

- ▶ Convert the assets
- ▶ Perform a change impact analysis
- ▶ Import the assets

Converting the assets applies the mapping changes created in the source-destination pair. This process prepares the assets to be imported into the destination environment. It is the first chance to catch errors related to mappings or data issues. To convert the exported assets, complete these steps:

1. Right click the source-destination pair and select **Convert Assets**.
2. Select **Content Engine Deploy Dataset**.
3. Select the folder that contains the extracted dataset. This folder is defined either by the export location in a connected deployment or from the deployment package extract.
4. Select the folder for the converted assets. Generally, the default conversion location is acceptable. The default is to create a folder in the target environment named <name of source extract>.converted.
5. Run the conversion. If necessary, correct any errors that you found during the conversion, and then reconvert.

8.5.6 Analyze the dataset

Analyzing the dataset indicates what operations and objects will be imported and updated in the target environment. It is an extra place to catch mapping errors. This task is optional, but can provide useful information for auditing and ensuring that only the wanted items are moved and modified. Analyzing the dataset involves the following steps:

1. Right click the source-destination pair and select **Analyze**.
2. Select the converted asset folder.

3. Select the options for the analysis. Typically the default of **Only update** is sufficient.
4. Run the analysis, and review the logs and the analysis report.

The analysis process creates an XML file of the results. By default, the process also writes an XSLT file that you can use to view the report in a web browser. You can also parse the resulting report file if there are automated processes necessary. Figure 8-16 shows a sample Change Impact Analysis report.

Change Impact Analysis - Passed Assets Report #1

[Return to main report](#)

Assets That Passed Analysis

Due to web browser constraints, when viewed using a web browser, the following table lists only summary information for each object that passed analysis. When the same passed asset XML file is viewed with a text editor, additional information is available including an estimated size breakdown for each property of an object. To view the additional information for a specific object in this file, open the file in a text editor and search for the object by its name or ID (as provided in the Name or ID field of the following table).

Name	Class	ID	Analysis Status	Import Operation	Estimated Size	Comments
Solution Assets	Folder	{E51E8975-9949-4E91-8F4E-3B591BFB66A1}	Passed	Skip	0.15 KB	Object to be imported has a Last Modified date that is not newer than the existing object in the target environment. This object would be skipped if imported: Name='Solution Assets', ID={E51E8975-9949-4E91-8F4E-3B591BFB66A1}, Class=Folder
Auto Claims	Folder	{164F8189-BC6A-4F95-8234-6464FA7889DD}	Passed	Skip	0.14 KB	Object to be imported has a Last Modified date that is not newer than the existing object in the target environment. This object would be skipped if imported: Name='Auto Claims', ID={164F8189-BC6A-4F95-8234-6464FA7889DD}, Class=Folder

[Return to main report](#)

Figure 8-16 Sample change impact analysis report

Importing the dataset

The dataset can now be imported into the destination environment. At the end of this process, a log is created. If there are errors, review the log and attempt to fix any issues. Then, re-execute the import. FileNet Deployment Manager continues to apply changes and reports success if everything is completed.

To perform the import, complete these steps:

1. Right click the source-destination pair and select **Import**.
2. Select the import options. Figure 8-17 shows the options that are available for importing.

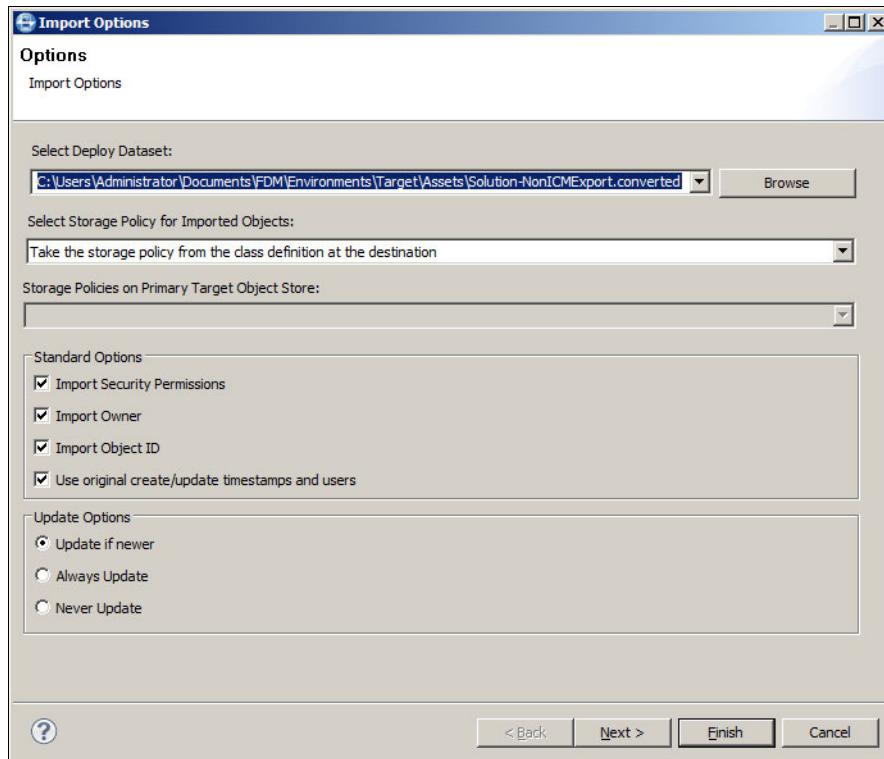


Figure 8-17 Import settings dialog

For the import **Standard Options**, generally select importobject ID to allow all existing references to the assets that are being imported to persist. These references include links to objects in widgets, such as the Form Data widget and hardcoded attachments in a workflow.

The other options depend on what type of tracking you want to do. Importing the owner and original modification information allows you to track who last modified a migrated asset and when. However, you might want to have the create date in the target system be the date that the object was migrated. If that is the case, then click to clear **Use original modify/update timestamps and users**.

3. Enable any required scripts for pre/post-import. This process automatically applies changes that cannot be migrated directly, such as applying events and creating items that depend on objects that are environment-specific.
4. Run the import, and review the logs.

8.5.7 Disconnected deployment

For a deployment where the FileNet Deployment Manager workstation cannot connect to both the source and target systems, a disconnected deployment is necessary. In this case, create, copy, and expand a deployment package. Doing so recreates the source environment on the target FileNet Deployment Manager instance. For more information about creating the deploy package, see “Creating the deploy package” on page 277.

You can do so by expanding a deployment package and applying the changes to an existing environment, or by creating an environment. See Figure 8-18 for an example of expanding a package into an existing environment.

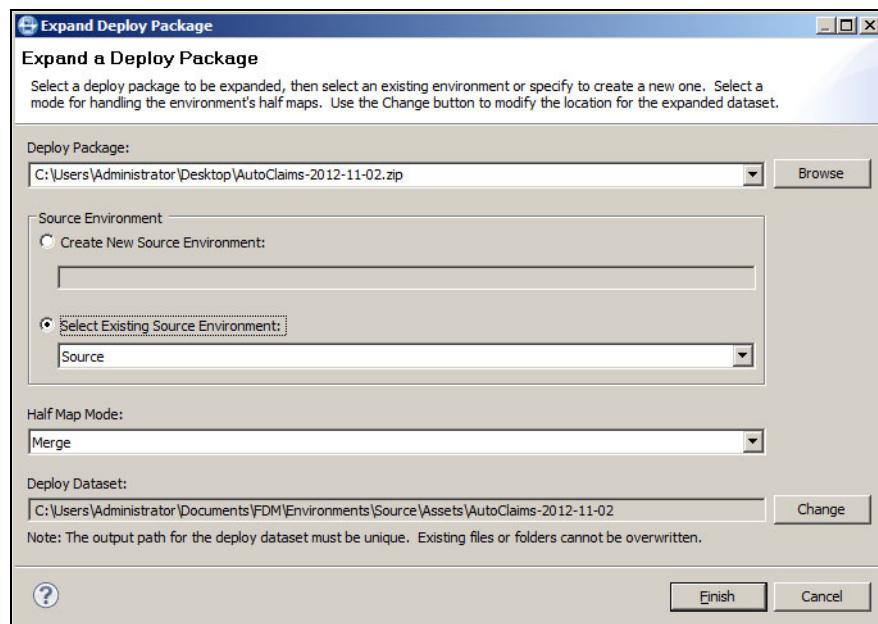


Figure 8-18 Expanding a deployment package into an existing environment

8.6 Migrating a Case Manager solution

A Case Manager solution is migrated by using the Case Manager Administration Client. This process involves these steps:

1. Exporting the solution from the source environment's design object store.
2. Importing the solution into the destination environment's design object store.
3. Deploying the solution into the destination environment's target object store.
4. Apply any post-deployment changes and updates.

8.6.1 Exporting solution package

The first step in migrating a solution is to export it from the source system. Complete the following steps to export a solution package:

1. Select **Tools** → **Export Solution Package**.
2. Step through the wizard, entering the connection settings and selecting the Case Manager design object store.
3. Select the solution to export and enter the target file name for the export package as shown in Figure 8-19.

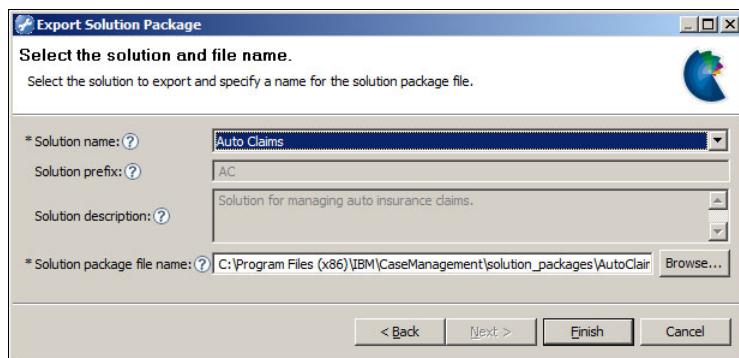


Figure 8-19 Selecting the solution to export

Tip: If you are using a source code management tool to track solution assets, the file that is generated by this process can be stored with the tag or branch that represents your release.

8.6.2 Importing the Case Manager solution

Copy the extracted file to the destination environment and import the solution into the target Case Manager environment.

Important: All widgets and custom Business Space elements must be imported before this process happens.

Complete the following steps to import a solution package into Case Manager:

1. Import solution using CMAC by clicking **Tools** → **Import Solution Package**.
2. Enter the credentials and connection settings.
3. Select the solution package.
4. Select project area, map the object stores, and perform any service mappings that are required as shown in Figure 8-20.

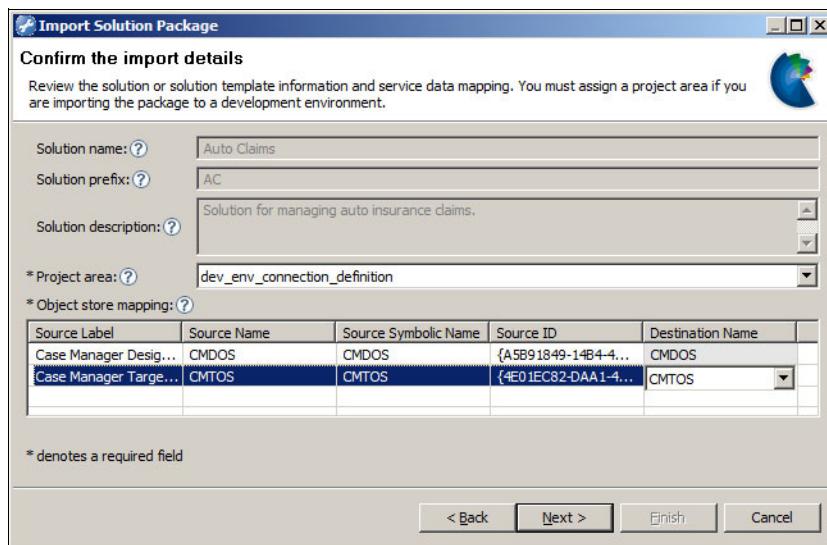


Figure 8-20 Configuring the import settings and mappings

5. Review any error messages.

8.6.3 Deploying the solution to the target object store

After the solution is imported and any pre-deployment changes are applied, the solution can be deployed to the target environment. For environments that have Case Builder deployed, you can deploy the solution from there. Otherwise,

CMAC is used to deploy the solution. One of the tasks in a production profile is deploying a solution. See Figure 8-21 for an example.

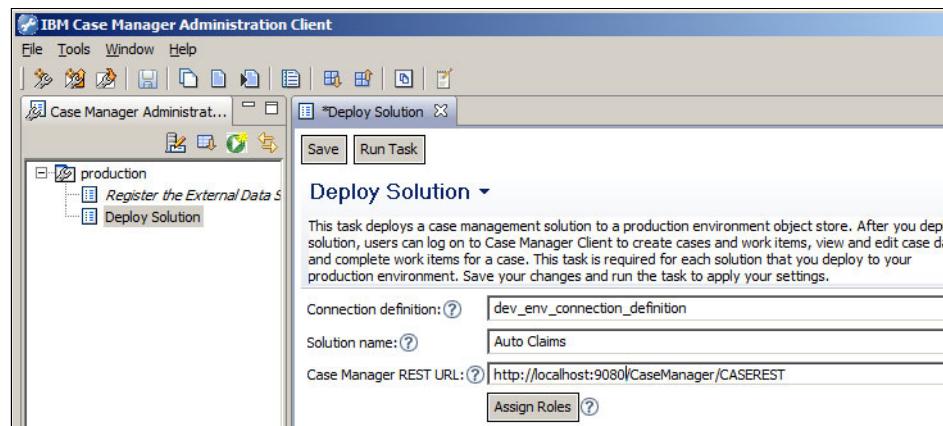


Figure 8-21 Deploying a solution from CMAC

Remember: Before you run this operation, any pre-deployment operations as defined by the Case Manager developers must be run. In addition, all custom widgets and target web services must be deployed for a successful deployment to occur.

8.6.4 Post deployment changes

After the solution is deployed, extra steps might need to be taken. These should be provided by your Case Manager solution developer, and can include:

- ▶ Assigning security principals to roles in Case Client.
- ▶ Setting the security, and default instance security, on the case folder and any document classes that are used in the solution.
- ▶ Assigning security to the case type folders.



Solution development

When you develop an IBM Case Manager solution, there are a number of factors you need to bear in mind. These factors affect design, development, and the deployment process, and ultimately affect how successful the final deployment is. This chapter addresses the factors that pertain to solution development.

This chapter includes the following sections:

- ▶ Solution development
- ▶ Security considerations
- ▶ Solution change and configuration management
- ▶ Sandbox options

9.1 Solution development

When you develop a Case Manager solution, a number of elements might need to be created and managed. The potentially wide ranging nature of a Case Manager solution can require the integration of a number of systems and tools. Therefore, the team can include a number of subject matter experts, domain-specific developers, and business specialists. In addition, it can require tracking and deployment of multiple asset types, ranging from the Case Manager solution itself to forms and rules, and custom integration code.

This section describes the various roles that are involved in designing a solution. It also describes a method for tracking the various assets of a solution. This section does not describe the mechanics of developing a solution. For more information about these topics, see Chapter 5, “Designing case management solutions” on page 123 and Chapter 11, “Development topics” on page 413.

9.1.1 Roles

Solutions can range in complexity from one person building a solution entirely in Case Manager Builder to many people designing and developing the various parts of a solution. At minimum, employ one solution designer. However, more roles might be required. Understanding the designer and developer roles is useful when you are determining how to divide the tasks that are required to build a solution.

In addition to helping divide the work between various team members or groups, the following role definitions can be useful when you deploy a solution. Each of the assets that are created by the various roles can have different methods for migration. These differences must be factored into the deployment plans and asset tracking as described in these places:

- ▶ 9.1.3, “Tracking and packaging solution assets” on page 294
- ▶ Chapter 8, “Solution deployment” on page 259

Depending on the needs of your solution, consider the following roles when you are preparing to build a solution:

- ▶ Business analyst

Interfaces with the users and business principals, and determines the basic functional requirements for a Case Manager solution. This role can also share responsibilities with the solution designer by working directly in Case Builder to design tasks and other elements of the case solution.

- ▶ Solution designer

Responsible for the solution from an end-to-end perspective. Works with the other subject matter experts as needed to drill into particular parts of a solution such as:

 - Page designers to create page layout widgets
 - Developers to help define any custom integration code
 - Forms and user interface specialists to define the user interface and layout
- ▶ Page designer

Works with the mashup container to configure widgets on pages and layout. Members of this role must understand widget and mashup concepts.
- ▶ Custom widget developer

Works with general development tools to create widgets and supporting resources, if necessary. The widget designer must know JavaScript and basic XML, widget concepts, and the Dojo framework. In addition, a widget developer must understand the Case REST, Process REST, and CMIS APIs.
- ▶ Integration developer

Works with general development tools to develop custom components and Process Designer to configure the system steps into a workflow. This developer must know business process management practices, as well as Java if developing custom Process Engine components, and FileNet Process Designer for highly complex workflows. Members of this role might also need to understand the underlying Content Engine and Process Engine APIs.
- ▶ Rule designer

Works with Rules Studio to create business rules. This designer must know rule design practices and Rules Studio capabilities for highly complex rules.
- ▶ Form designer

Works with forms designer tools to create forms. This designer must know both form design practices and the capabilities of the chosen electronic forms tool.

9.1.2 Team collaboration

If a team is formed to build a solution, they need tight collaboration. For teams that consist of more than a few people, you might want a project lead or project manager in addition to the roles defined in 9.1.1, “Roles” on page 290. The project lead or manager facilitates collaboration and asset tracking.

Parallel work

If your solution requires multiple people across multiple roles, you must think about how to organize parallel work. Most work when building a solution can happen in parallel. However, there are some considerations when you organize parallel work:

- ▶ Solution designer and integration developer

Only one user can edit a solution at any time. This concerns solution objects that are configured in Case Manager Builder and Process Designer within a single solution.

Multiple solutions can be built in parallel. For more information about building a solution with parallel effort, see 9.4, “Sandbox options” on page 317.

- ▶ Page designer

If there are multiple people who are designing pages, they can collaborate on the page designs. But assign only one person to edit any specific page to avoid conflicts.

If you expect multiple developers to redeploy the same solution, ensure that all those developers have administrator rights for Business Space in WebSphere Application Server. Doing so prevents problems with different users who redeploy solutions.

An alternative approach is to delegate coordination of solution redeployment to one team member such as the project lead.

- ▶ Custom widget developer and integration developer

Follow standard development practices by using a code repository to coordinate parallel code development for custom widgets, servlets, and components.

- ▶ Rule designer and form designer

The software to design rules and forms limits access to individual rule packages and forms to specific individuals. However, multiple rule packages and multiple forms can be developed in parallel.

Incremental changes

As a solution is being built, the incremental changes in parts of a solution can affect the full solution. Generally speaking, new assets can be created without consequence, but modifications and deletions might require extra actions in other areas of a solution. For more information about how a solution modification can affect the redeployment of a solution, see the Information Center at:

<http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/index.jsp?topic=%2Fcom.ibm.casemgmt.design.doc%2Facmdc024.htm>

The following is an overview of what coordination might be required within the solution development team because of modifications:

► **Solution design**

If a custom widget refers directly to a solution object by ID, such as a task or a case property, the custom widget developer must be notified. The custom widget might need to be modified to reference the new or changed ID.

If a property is mapped to a business rule and the property is modified or removed from the case type, the integration developer and rule designer must be notified. Changes to system steps or rules might be needed.

If a property is mapped to a form and the property is modified or removed from a step or case type, the forms designer must be notified. Changes to a form might be needed.

Solution changes can be made and deployed freely, subject to any solution redeployment restrictions.

► **Solution pages**

If a page is to be deleted and it is in use by a step, the solution designer must be notified to change that step page.

Pages can be created and redesigned freely.

► **Custom widget**

If a widget event changes and the widget is in use, the page designer must be notified. The designer can then make necessary modifications on pages that contain the widget.

If a widget is to be deleted and the widget is in use, the page designer must be notified. The designer can then remove the widget from all pages.

Changes to widget code and creating new widgets can be done freely.

► **Integration**

If a work step intended for human processing requires a change, keep the solution designer informed of the change.

Adding or modifying system steps and workflow properties in any task can be done freely.

Custom components can be developed and applied freely.

► **Rule designer**

If a rule is to be deleted and is in use by a solution or form, the integration developer or forms designer must be notified.

If an input or output parameter changes for a rule that is used by a task, the integration developer must be notified. Changes to the task workflow might be required.

If an input or output parameter changes for a rule that is used by a form for validation, the form designer must be notified. Changes to the form might be required.

Changes to business rules and creating new rules can be done freely.

- ▶ Form designer

If a form is to be deleted and is used as a template, the page designer must be notified. The designer can then remove the form from any form widgets that use it.

If a form is to be deleted and is used as an attachment in task workflows, the integration developer must be notified. The developer can then remove the specified form from any attachments.

Before you deploy a new version of a form, ensure that everywhere this form is referenced is either referenced by specific version or that new functionality is compatible with existing tasks if they are referenced by current version.

New forms can be deployed freely.

9.1.3 Tracking and packaging solution assets

You can add various assets when you are building a solution. Establish a solution development methodology that takes all assets into consideration. This method allows you to end up with an exported solution package and supplemental material that represent a snapshot of your entire solution. This methodology should include procedures to both track and package all assets in a solution application.

Examples of packaging assets

Assets for simple solutions that are built using only Case Builder features are fully contained within the solution folder. The solution package that is created using the Case Manager Administration Client includes all solution assets.

Complex solutions might have the following extra assets to consider:

- ▶ Content Engine object definitions
- ▶ Process Engine configuration elements
- ▶ eForms and Forms document objects
- ▶ Policy and other text document objects
- ▶ Logos, scans, and other image objects
- ▶ Pages and layouts
- ▶ Widget definition XML and JavaScript code
- ▶ Process Engine components
- ▶ Custom servlets and web services
- ▶ Business rules
- ▶ Reporting templates and configuration

Any assets that are configured in Case Manager Builder, Process Designer, or Business Space are readily available within the solution folder. An exported solution package contains these assets at a minimum.

Assets that are created using FileNet Enterprise Manager or Administration Client for Content Engine are stored in the Content Engine and exported using FileNet Deployment Manager.

Code that is developed for widgets, servlets, or custom components typically is managed in a code versioning system. Other external content and configurations might require more considerations when they are associated with a solution package. Business rules or reporting configurations, for example, have their own export and backup procedures. Specific export or backup procedures might also apply to extra systems that you integrate with IBM Case Manager.

In these instances, if you export your assets, consider importing them into either into a source code management system or into the solution folder in Content Engine for consistency. If you cannot export those assets, note the backup and restore procedures of those systems, and record any backup content in association with a solution package.

Follow a solution development methodology that takes all assets into consideration. This method allows you to end up with an exported solution package and supplemental material that represent a snapshot of your entire solution.

Examples of tracking assets

One key to successful creation and deployment of a solution includes tracking all assets that are involved in a solution. To demonstrate asset tracking, consider the following hypothetical IBM Case Manager solution:

- ▶ The solution has the base solution configuration that is done by a solution designer.
- ▶ A custom widget developer develops two widgets: Hello World and Another Widget.
- ▶ A custom widget developer develops a web resource that is called User Information to support the widget called Another Widget.
- ▶ The widget that is called Another Widget also references a REST API provided by an older system.
- ▶ The widget code and web resource code are stored in separate deployment packages.
- ▶ A single custom business space page is configured to contain the two custom widgets. It is designated as a step page.

- ▶ A form is created to display case properties.
- ▶ A rule is created to evaluate priority.
- ▶ A proxy is configured so that the custom widgets can reference custom and older resources without cross-domain issues.

Remember: The following is not representative of best practices and is not all-inclusive. It is intended to help you think about the type of information a project needs to track. Format, expand, and modify this example as needed for your project.

Solution

The sample solution consists of the following configuration:

- ▶ Name: Solution1
- ▶ Prefix: S1
- ▶ Properties:
 - Prop1 (sym{S1_Prop1}, String, size 25)
 - Prop2 (sym{S1_Prop2}, Integer, size 10)
 - Prop3 (sym{S1_Prop3}, Integer, min 100, max 999)
 - Prop4 (sym{S1_Prop4}, String, multi-choice [“First”, “Second”, “Third”])
 - Prop5 (sym{S1_Prop5}, Boolean, default false)
- ▶ Roles:
 - Role1 (columns [Prop1, Prop4])
 - Role2 (columns [F_StepName, Prop1, Prop2], sort[Prop2])
 - Role3 (columns [F_Subject, Prop4], filter[Prop4])
- ▶ Case Type:
 - Name: Case Type 1
 - Symbolic Name: S1_CaseType1
 - Initiating document: none
 - Folder 1: /Supported Docs
 - Folder 2: /Communication
 - Folder 2.1: /Communication/Outbound
 - Folder 2.2: /Communication/Inbound
 - Task1 (automatic)
 - Task2 (manual)
 - Task3 (automatic, precondition [Prop4=“First”])
 - Task4 (user-created)
- ▶ Case Type:
 - Name: Case Type 2
 - Symbolic Name: S1_CaseType2

- Initiating document: none
- Task1 (automatic)

Business Space page

For the sample solution, the business space page is set up in this way:

- ▶ Page name: Sample page
- ▶ Registered name: SamplePage
- ▶ Type of page: Step page
- ▶ Widgets on page:
 - Command
 - Case Information
 - Toolbar
 - Hello World
 - Another widget

Widget

The sample solution has a widget that presents a basic sentence when loaded. If it handles a World event, it triggers a Wave or Greet event, depending on the population of the world. The widget has the following setup:

- ▶ Name: Hello World
- ▶ Scope: com.ibm.custom.widgets.helloworld
- ▶ Resource file: HelloWorld.xml
- ▶ Resource path: /helloworld
- ▶ Resource application: custom_widgets.ear
- ▶ Outbound events:
 - Name 1: Wave
 - Payload 1: {method: string, cycles: integer}
 - Name 2: Greet
 - Payload 2: {message: string, repeat: integer, shake: boolean}
- ▶ Inbound events:
 - Name 1: World
 - Payload 1: {population: integer, name: string}

Web resource

The solution uses a web resource (service) that provides information about personnel. It returns a hierarchy of people in a department that is based on either department code or manager name. It also returns a list of responsibilities based on a user name.

The web resource is set up in this way:

- ▶ Name: User Information
- ▶ Package: com.ibm.custom.service.userInfo
- ▶ Resource path: /userinfo
- ▶ Resource application: custom_resources.ear
- ▶ Servlets:
 - Servlet 1: DepartmentStructure
 - Parameters 1: deptCode (integer), managerName (string)
 - Servlet 2: Responsibilities
 - Parameters 2: userName (string)

Application packages

The sample solution has the following application packages:

- ▶ File name: custom_widgets.ear
- ▶ Context Root: /MyWidgetRoot
- ▶ Host: myserver:9080
- ▶ Provides: widgets
- ▶ Key resources:
 - /helloworld/HelloWorld.xml
 - /anotherwidget/AnotherWidget.xml
- ▶ File name: custom_resources.ear
- ▶ Context Root: /MyServicesRoot
- ▶ Host: myserver:9081
- ▶ Provides: web resources
- ▶ Key resources:
 - /userinfo/DepartmentStructure
 - /userinfo/Responsibilities
 - /payments/ProcessPayment
 - /payments/PaymentDetails

Form

The sample solution uses a form with the following configuration:

- ▶ Name: Form1
- ▶ Repository location: CMTOS:/Form Templates/

- ▶ Properties mapped:
 - Prop1
 - Prop3
 - Prop4

iLog rule

The sample solution uses iLOG rule configured in this way:

- ▶ Name: Department priority
- ▶ WSDL path: proxy:80/Rules/Path/To/Rule/1.0/WSDL
- ▶ Tasks starting rule: S1_CaseType1.Task4, S1_CaseType2.Task1
- ▶ Parameters passed:
 - Department name = S1_Prop1
 - Item number = S1_Prop2
 - Department code = S1_Prop3
- ▶ Parameters received:
 - Department priority = S1_Prop4

Proxy

The sample solution has the following proxy setup:

- ▶ Instance name: webproxy1
- ▶ Host name: proxy
- ▶ Port: 80
- ▶ Type: IBM HTTP Server
- ▶ Resources:
 - /mum* (myserver:9080)
 - /CaseManager* (myserver:9080)
 - /MyWidgetRoot* (myserver:9080)
 - /MyServicesRoot* (myserver:9081)
 - /LegacyService* (legacyserver:80)
 - /Rules* (ruleserver:9080)

9.2 Security considerations

One key to consider when you are building and deploying a Case Manager solution is security. Appropriate security can ensure that cases and case documents are routed to and secured for the appropriate members of the team. However, improper security can result in cases being open for viewing and

modification by unauthorized members of the team. It can also result in team members not being able to create new cases or update cases they are authorized to work on.

Although security can be modified after the solution is live, it is important to apply security correctly from the start. Doing so ensures that all objects in the solution are correctly secured. Consider these objects when applying security:

- ▶ The target object store that hosts the deployed solution application and workflow system, including the class definitions that are created when the solution is deployed
- ▶ Case folders and subfolders
- ▶ Documents
- ▶ Comments
- ▶ Tasks
- ▶ Queues
- ▶ Application Spaces
- ▶ Business Spaces and Pages

Initial security on the object stores must allow all users of the solutions that are deployed to that object store at least read access. After an object store is created, any changes to the security of the object store must be propagated using the Security Wizard in FileNet Enterprise Manager. An example of such a change is adding a group to access the object store.

Tip: The Content Engine virtual group #AUTHENTICATED-USERS provides broad security rights to items in an object store. For a production deployment, this will typically be too broad. Instead, create an LDAP group for users that need access to that object store instead of #AUTHENTICATED-USERS. For more information about using a master group, see 9.2.2, “Security example” on page 308.

For more information about planning security for a Case Manager system and deployment, see the Case Manager Information Center. Click **Planning your case management system → Planning for IBM Case Manager security → Planning for security in the production environment.**

9.2.1 Security concepts

Case Manager is built on three key components: Content Engine, Process Engine, and Business Space. A case solution derives its security from the

combination of the three systems. It is important to understand some of the core concepts of how FileNet Content Engine, FileNet Process Engine, and Business Space implement security. This section addresses these key concepts and provides links to the Information Center where you can read more about these concepts and how they are implemented.

When you are determining the appropriate security for each element in the three components, keep the following categories of users in mind. A case solution typically starts with these categories of users and refines them as needed:

- ▶ Case Viewers: The users in this group need to have read-only access to a case.
- ▶ Case Initiators: The users in this group need to be able to view and create new cases.
- ▶ Case Workers: The users in this group need to be able to view and update cases, and create and update case-related objects. They often cannot create cases, delete access, or change the permission or the owner of case-related objects.
- ▶ Case Administrators: The users in this group need to have full control over case-related objects (the case folder, subfolders, comments, and tasks).
- ▶ System virtual case workers: If a solution has a task that uses the external service wizard to interact with an external data service, external business process, or other custom component accessed with an external service, the component manager/integration user account configured to process the component queue can be considered a virtual case worker. That virtual case worker is acting on behalf of behind-the-scenes system operations.

The examples in the following sections generally refer to these high-level categories of users.

Content Engine

The Content Engine provides access control for all content-based elements within Case Manager, including security for a case folder and documents in a case. The following are key concepts within the Content Engine security system to keep in mind when you build a Case Manager solution:

- ▶ Access Control Lists

Each securable item within the Content Engine has an access control list (ACL). The ACL defines who does and does not have access to an object, and what rights they are granted or denied.

For Case Manager, that applies to the following objects:

- Case folders and subfolders
- Documents in a case

- Case comments (annotations)
- Classes, such as case folder and document classes

For more information about access rights, see the IBM FileNet P8 Information Center at:

<http://pic.dhe.ibm.com/infocenter/p8docs/v5r1m0/topic/com.ibm.p8.security.doc/p8psa050.htm>

► Default instance security

When an object is first created, its security comes from the default instance security for the class it is an instance of. Security can then be changed directly, by applying inheritance, security proxy objects, and marking sets. Regardless of the final security, the starting point is the class's default instance security. In a Case Manager context, default instance security can apply to the following objects:

- Documents added to a case
- Case comments
- Case folders, defining who has access to a case after it is created

Consideration: In a Case Manager context, default instance security is useful to review, but the appropriate starting point for setting security for cases is using inherited security.

For more information about default instance security in the Content Engine, see the IBM FileNet P8 Information Center at:

<http://pic.dhe.ibm.com/infocenter/p8docs/v5r1m0/topic/com.ibm.p8.security.doc/p8psa002.htm>

► Object store security

Before a user can create or use any objects in an object store, they must have rights to use the object store itself. The following are the primary rights for the object store:

- Use object store
- Create objects

These two rights control who can use the object store and the objects that are contained within, and who can create new objects. In a Case Manager context, that controls who can view and search for cases, and who can create new cases.

For more information about object store security, see the IBM FileNet P8 Information Center at:

<http://pic.dhe.ibm.com/infocenter/p8docs/v5r1m0/topic/com.ibm.p8.security.doc/p8psa011.htm>

► Class definition security

Each class definition in the Content Engine is secured using an ACL. This ACL defines who can modify the class itself, but it also defines who can create new instances of that class. Each class object has a security right called “Create Instance”. If granted this right, a user is allowed to create instances of that type of object.

In Case Manager, a user must be granted create instance rights on the case folder class for that type of case to create cases. See Figure 9-1 for an example.

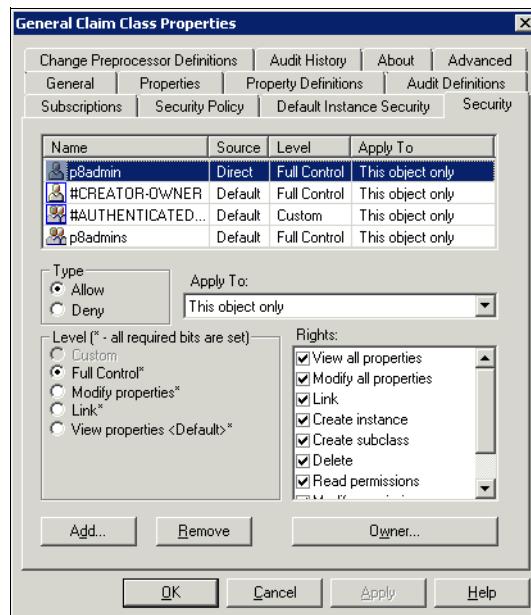


Figure 9-1 Creating instance security rights for a case folder class object

For more information about class security in the Content Engine, see the IBM FileNet P8 Information Center at:

<http://pic.dhe.ibm.com/infocenter/p8docs/v5r1m0/topic/com.ibm.p8.security.doc/p8psa002.htm>

► Inherited security

The Content Engine itself, and by extension Case Manager, supports unique security, or ACLs, for individual objects. However, to simplify security, Case Manager is more easily configured using inherited security. The administrator sets the security on the case type folders in a deployed solution. All cases created in that folder then receive the same security.

One key advantage to this type of deployment is flexibility. If a change in business requires a change in security, the administrator only has to change the security on a single folder, rather than needing to update each existing case individually.

Tip: In addition to inherited security, Content Engine offers additional options for restricting or overriding inherited security for sensitive cases where only a subset of users are allowed access.

- ▶ Case Type Folder

The Case Type Folder is the root folder that contains all instances of a case. To properly secure a deployed case solution, set the security on the Case Type Folder and allow all children to inherit that security. This configuration secures the cases themselves and the other case elements like documents, tasks, comments, and case subfolders.

When you apply security to each of these objects, vary the security rights that are granted by user group. Table 9-1 provides recommended security rights for key Content Engine objects.

Table 9-1 Recommended case folder and object security

Object	Admins	Initiators	Workers	Viewers
Case type folder	Full Control	View level + create sub folders	Modify Level	View
	Apply to: this object and all children	Apply to: this object and all children	Apply to: this object and all children	Apply to: this object and all children
Case class definitions	Grant Create Instance permission			Deny create instance permission
Documents	Can vary based on security rights necessary for each document type. Set by using Default Instance Security.			
All other items	Inherited from case type folder.			

Process Engine

As the workflow engine, the Process Engine controls access to any workflow-based processes within a Case Manager solution. In a Case Manager context, these are primarily roles and in-baskets. These two items are configured

within the Process Engine and can be affected by other elements in the Process Engine itself.

► Queues

Within Case Manager, a role is mapped to a queue in Process Engine. Case Manager users access their work items through in-baskets. In-baskets are filtered views to a Process Engine queue. As such, for case workers to be able to access their work items, they must be able to access the appropriate queue.

If you deploy Workplace XT, you might need to secure the queue itself. Although Case Manager only provides access to the work items their role is assigned to, Workplace XT offers access to the queues themselves. Securing the queues ensures that the members of the roles cannot access queues that contain work that they are not authorized to complete. Queue security is defined by using Process Configuration Console.

For more information about queue security, see the IBM FileNet P8 Information Center at:

<http://pic.dhe.ibm.com/infocenter/p8docs/v5r1m0/topic/com.ibm.p8.security.doc/p8psa012.htm>

Remember: Any virtual case worker user accounts must be granted appropriate rights to the case objects to complete the operations in the component queue. For example, the virtual case worker user will need to be added to the Case Type folder security inheritance model with Modify Properties rights inheriting to all children. You might be able to use one of the existing roles, such as Case Worker or Initiator, for this virtual case worker.

► Application Spaces

The Process Engine organizes itself into sets of resources using Application Spaces. For Case Manager, each solution uses a unique application space within the Process Engine. It is this space that defines the roles and which in-baskets are assigned to a role. You can secure an Application Space if necessary. Application space security is defined in Process Configuration Console.

For more information about managing Application Spaces, see the IBM FileNet P8 Information Center at:

<http://pic.dhe.ibm.com/infocenter/p8docs/v5r1m0/topic/com.ibm.p8.pe.configui.doc/bpfcc053.htm>

► Roles

In a Case Manager context, it is the role that primarily provides access control to the workflow items in a solution. A Process Engine role defines who has access to which in-baskets. Roles are defined within the Case Manager solution's application space and secured using Case Client.

Table 9-2 provides the recommended security rights for each of the primary Process Engine elements. For more information about the rights available in Process Engine, see the IBM FileNet P8 Information Center at:

<http://pic.dhe.ibm.com/infocenter/p8docs/v5r1m0/topic/com.ibm.p8.security.doc/p8psa012.htm>

Table 9-2 Recommended security rights for the common user groups

Element	Case Administrators	Case Initiators	Case Workers	Case Viewers
Queues	Typically given query rights and can be given process rights to troubleshoot work items.	Query and process only if initiators need to view and process work items.	Query and process rights are necessary to handle work items.	None
Roster	Query rights allow admins to troubleshoot work items and create rights if they need to start tasks	Create rights are necessary to allow initiators to create cases and the tasks related to them.	Create rights are necessary if case workers need to create new tasks.	None
Application Space	Case administrators can assign more members to roles to handle work item backlogs.			

Business Space

Case Client is the primary user interface for Case Manager. It is deployed into Business Space, and consists of a collection of widgets and pages for each solution. Business space provides two primary securable items: spaces and pages:

► Spaces

A space is a collection of pages that represent part of a Case Manager solution. Each Case Manager solution is built of three primary spaces: Solution Space, Case Pages Space, and Step Pages Space. Users see

Spaces that they have access to. For spaces, there are three primary access rights:

- Viewers have rights to view a space, but cannot make changes to the space.
 - Editors can modify the attributes of a space, including name and description. They can also add and remove pages.
 - Owners can set access rights, referred to as sharing, on a space. They also have edit rights to the space. By default, the user who created the space is the owner. Ownership can be transferred to another user.
- Pages
- Pages within a space are displayed as tabs. Each page contains one or more widgets. For pages, users can have one of three primary access rights, similar to spaces:
- Viewers have rights to view a page, but cannot make changes to the page.
 - Editors can modify the page's attributes, widgets, and layout.
 - Owners can set access rights, referred to as sharing, on a page. They also have edit rights to the page. By default, the user who created the page is the owner. Ownership can be transferred to another user.

Table 9-3 shows the recommended access rights for spaces and pages.

Table 9-3 Recommended access rights for spaces and pages

	Case Admins	Case Initiators	Case Workers	Case Viewers
Solution Space	View	View	View	View
Work Page	Inherit from solution			
Case Page	Inherit from solution			
Case Pages Space	All Authenticated Users and apply “Hide from list view”			
New Case page	View	View	No access	No access
Case Details page	View	View	View	View
Step Pages Space	All Authenticated Users and apply “Hide from list view”			
New Task Page	View	View	View	No access
Work Item page	View	View	View	No access
Work eForm Page	View	View	View	No access

9.2.2 Security example

This section provides a practical security example for a Case Manager deployment. It reviews the settings for Content Engine objects, key Process Engine elements, and Business Space security.

Content Engine

This section provides an example of how you can configure a Case Manager solution using a master group that provides access to the Case Manager target object store. This method provides a more restrictive alternative to using the #AUTHENTICATED-USERS virtual group that is provided by the Content Engine.

In this example, a group referred to as the master group is used to provide access to the object store and Case Manager. This group can include all members of the primary roles of Case Administrators, Case Workers, Case Initiators, and Case Viewers. This section also shows how this group applies to key objects from the object store down.

- ▶ Object Store

To ensure the correct propagation of permissions in IBM Case Manager for both standard and user-defined metadata and folder objects in Case Manager solutions, add users and groups into the object store creation wizard at create time.

Add the master group to the object store creation wizard's User list, which allows all class definitions, property templates and Case Manager metadata objects to have the required usage permissions that are needed for viewing and searching cases and case-related objects. Extra security permissions or constraints are applied to the actual case folders. For more information, see "Case type folders" on page 312.

Remember: If you do not add the master group to the object store at create-time, case workers that are members of other case groups not added when the object store is created will not be able to search for or view cases without more extensive, supplemental security configuration.

When you create an object store, you have the chance to define two sets of principals for the new object store: object store users and object store administrators. Add the master group when you assign the object store users. The following steps walk you through this process:

- a. Right-click the Object Stores node in FileNet Enterprise Manager (FEM).
- b. Follow the prompts to enter the required data in the wizard, and click **Next** to proceed through to the administrative and user data pages.

- c. Add the administrative users and groups into this list, then click **Next**.
- d. Add the master group, and any other required user groups, into the step as shown in Figure 9-2.

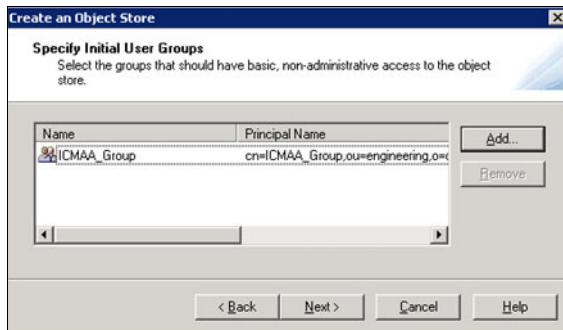


Figure 9-2 Adding the master group

- e. Click **Next** through the AddOns selector, verify the values, then click **Finish** to create the object store.
- Solution deployment
After the target object store is configured for Case Manager using Case Manager Administration Client (CMAC), export the solution packages from the development environment's design object store. Then import the solution packages into the staging area (design) object store of the production environment. Next, use the CMAC Deploy Solution task to install the solution into the production target object store. While deploying, the security for the solution's metadata and folder objects picks up the user groups that were pre-added to the target object store at creation time.
- Solution's document type classes
In the IBM Case Manager security model, case documents do not inherit security from their case folder. Traditional document class Default Instance Security guidelines are used instead.

When a solution is deployed, the Case Manager solution's document types, subclasses of the base Document class definition, have default security settings. These setting depend on which groups are added during the Object Store creation wizard. In the sample of object store creation, the subclasses of the base Document class definition by default contain the following on the Default Instance Security tab:

- #CREATOR-OWNER: Full Control
- Master Group: View properties, View Content, Read Permissions
- Admin users and groups: Full Control

For each deployed solution, have the Case Manager security administrator review each document class's Security and Default Instance Security tabs to ensure that the correct permissions are selected. Also, review the groups in the Security tab to determine who has create instance rights. These rights control who can create documents.

For the groups that need document versioning rights, complete the following steps:

- a. Right-click each document class for the solution and select **Properties**.
- b. Click the Default Instance Security tab.
- c. Add each group that requires versioning rights.
- d. For each group, select the **Major Versioning** and **This object only** options.
- e. Click **Apply**.
- f. Click **OK** when complete.

Consideration: You can add these groups to the base Document class's Default Instance Security tab. The security settings will propagate only to new Document subclasses that are added (e.g. from deployed solutions) after these changes are saved.

You might want to downgrade the permissions for the #CREATOR-OWNER user for each solution's document classes unless otherwise required. This procedure removes the Delete, Modify Permissions, Modify Owner, and Publish privileges. Activate these permissions individually as required.

- a. Click the document class's Default Instance Security tab.
 - b. Highlight the **#CREATOR-OWNER** entry.
 - c. Select **Major versioning**.
 - d. Click **Apply**.
- Form document classes
- Navigate to the following Document subclass **Form Template** → **ITX Form Template**. Normally the object store administrative user checks in the data-entry form document objects into Workplace XT. If not, complete the following steps to allow extra users to check in data-entry form document objects:
- a. Open the **ITX Form Template** class's property sheet.
 - b. Add the user to perform the form document check-in function to this document class's Security tab with a minimum of **View Properties** and the **Create Instance** permissions.

- c. Click the Default Instance Security tab and make sure the **#CREATOR-OWNER** entry has a minimum of **Major Versioning** permission.
- d. The master group has **View Properties** permissions on this object, so the case workers are automatically able to retrieve this object for case creating, viewing, and modifying.

Tip: If after performing the preceding procedures a new subclass of **ITX Form Template** is created, review the **Security** and **Default Instance Security** tabs of the new subclass to verify the settings are correct.

► Class Definitions

By adding the master group to the object store at creation time, all case workers receive the **View Properties** and **Create Instance** permissions. This means that any users that are a member of the master group can create cases. If users that are members of roles should not have this ability, such as Case Viewers, perform one of the following operations:

- On the Security tab, remove the **Create Instance** permission from the master group. Then add the group that needs create instance with the **View Properties** and **Create Instance** permissions selected.
- Add the user or group that you do not want able to create cases the Security tab with an explicit deny of the **Create Instance** permission.

For more information about setting security on other elements of a case solution, see the Information Center article at:

<http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/topic/com.ibm.casemgmt.design.doc/acmdc003.htm>

► ICM form document objects

The master group will already have **View Properties** permissions on this object. Therefore, the case workers can automatically retrieve this object (which is used in Case Client form pages) for case creating, viewing, and modifying.

► ICM event handler

The CmAcmeventHandler document object that contains the ICM event handler Java classes is in the following folder: **Object Store → Root Folder → CodeModules**.

The master group will already have **View Properties** and **View Content** permissions on this object. However, to allow users outside the master group permission to run Case Manager events, you must add the internal

#AUTENTICATED-USERS group to this object's Security tab with the **View Content** permission option.

- ▶ Case type folders

Navigate to the Case Manager root folder and expand the folder tree to display the case types folders. It is at this level that you apply the security as defined in Table 9-1 on page 304.

Case Administrators must be granted full control of the case type folder and all subfolders. This capability allows them to administer the cases in the system.

Case Initiators by default are granted view property permissions on the case folders. To create cases, they must be granted create subfolder rights that apply to the case type folder and all children.

Case Workers must modify property values on any case. To enable this right, the members of this group must have modify level permissions granted to this object and all children.

Case Viewers by default have the appropriate view property rights because they are members of the master group.

- ▶ IBM Case Manager folder tree

The entire folder tree uses inheritance so that the master group gives all case members View Properties permissions across all solutions and their cases. If this is not desirable, you can determine at what level you want to disable this access. For example, you might have certain solutions or specific case types that should not provide access to a certain group. However, remember that if you modify this default configuration, you must manually add the specific groups at the wanted level to reactivate the inheritance. For example, if you want to include only specific groups for a specific case type, remove the **View Properties** rights for the master group. Then add the specific groups at the case type folder level with the **View Properties** and **This object and all children** options.

- ▶ Object Store Security Script

This script applies security for two different group types: Admin (Full control) and User (Use object store). It adds these users and groups across all object store metadata, and every folder under the Root folder, including all case instance folders and their subfolders.

Exception: Due to the potential for a large number of objects in the Root folder, do not run this script on a production target object store.

- ▶ Runtime Security Changes

If there is customized event-driven code that attempts to modify the security of objects in the production target object store, the objects are subject to inherited permissions. If you want to demote access rights, have the custom code add to the parent object, such as the Case instance folder, an updated ACL entry. This entry must have an explicit deny for the specific permission attributes to override the allow permissions on the parent object. If the code attempts to update the child object's existing user/group ACL entry, it generates an exception because of the read-only nature of the AccessMask property.

Process Engine

After a solution is deployed in Case Manager, you can connect to the Process Configuration Console (PCC) through Workplace XT to apply the security rights that are defined in Table 9-2 on page 306.

- ▶ Queues

To set the security on the queues, click **Connection Point → Work Queues** in PCC. Then, select each queue for the solution, right-click, and select get properties. In the security tab, add the appropriate groups with the correct query and process rights.

- ▶ Rosters

In the same manner as queues, expand the rosters section, and select the roster for the solution. The roster has the same name as your solution. Right-click the roster and click **Properties**. In the window, add the groups with the appropriate query and create rights.

- ▶ Application Space

To add users and groups to a role in Case Manager, grant modify rights to the space to the Case Administrators group.

Requirements: For users to process a work item, they must be both a member of the role, and have access to the queue that contains the work item to be processed.

Business Space

The last step to securing a Case Manager solution is to modify the access permissions for the pages and spaces in Business Space. 9.2.2, “Security example” on page 308 defines the recommended access levels for the four primary user types.

- ▶ Spaces

To set the security on the space, select **Manage Spaces** in Business Space. Then, for each of the three primary spaces, select **Actions → Share**. In the share dialog, assign the appropriate users view and edit rights.

- ▶ Pages

After the spaces are secured, set the security on each of the pages. Click **Actions → Share** on each page in the primary spaces. In the security window, add users as required.

9.3 Solution change and configuration management

The lifecycle of a Case Manager solution requires deployment and incremental changes to the solution. As business needs change over time, the solution must also change. To effectively manage these deployments, you need a change management process. This section describes the two major components of a successful strategy:

- ▶ Solution documentation
- ▶ Code and release management

The intent of the solution packaging is to facilitate the migration and deployment of a solution application from the development environment into more environments. The packaging described here relates to employing industry standard software development practices and Software Configuration Management (SCM) tools to provide for change management and version control.

9.3.1 Solution documentation

As you are deploying a Case Manager solution, you must document the solution. This documentation is needed both during the deployment phase and for future changes. When you document the solution, make note of the following items:

- ▶ A description of the Solution Components included in Solution Package, broken into two sets of documents:
 - Solution assets, which can be generated by using the Solution Document Generator. For more information, see in 8.1.1, “Tools for migration” on page 261.
 - Extra assets that are stored in Content Engine, Process Engine, or other integrated components directly, and that are managed outside the Case Manager context.
- ▶ Order of steps to follow after you extract the Solution Package in Case Manager Administration Client (CMAC).
- ▶ FileNet Deployment Manager instructions, including any deviations from standard procedure or extra steps necessary to complete the migration.
- ▶ Pre-requisite steps before deploying the Solution in Case Builder or CMAC.
- ▶ Post-requisite steps after you deploy the Solution in Case Builder or CMAC.
- ▶ Security configuration details.
- ▶ Assets that are outside of Content or Process Engines, such as rules or custom services.
- ▶ Screen captures.
- ▶ Extra information to support production variances.
- ▶ Post migration test plan for the Deployed Solution.

Tip: Although this documentation can be created as the deployment process begins, it is more efficient and produces more complete documentation if the developers and designers complete the documentation as part of their design process.

9.3.2 Source code and release management

Given the wide range of assets and all-encompassing nature of a Case Manager deployment, many implementers choose to use an SCM system to track code and assets in the solution.

If the intent is to keep all assets in an SCM, such as IBM Rational Jazz, then solution packaging is important. To manage a Case Manager solution with an SCM, the deployment and development team must extract and store two basic categories of Case Manager-specific assets:

- ▶ Case Manager solution packages

A Case Manager solution package includes all the defined case details such as properties, document types, tasks, and other items that are defined in Case Builder. Packages can be used to deploy the solution into a different environment. For more information about creating a Case Manager solution deployment package, see 8.6.1, “Exporting solution package” on page 285.

- ▶ Non-Case Manager assets

Depending on how the solution is designed and deployed, there might be more assets that are IBM FileNet P8-based, but that are not created using Case Builder. These assets can include:

- Shared classes, properties, and choice lists
- IBM Forms or FileNet eForms
- Documents that are stored in Content Engine that are used in a solution, but that are not stored in a single case folder
- Stored Searches and Entry Templates

These items must be exported as a deployment package and stored in the SCM. To create a deployment package suitable for inclusion in an SCM, see “Creating the deploy package” on page 277.

The Content Engine and Case Manager handle version management, tracking the various versions of the objects. However, as part of a release, you might want to do a final export of the elements that are listed above. Store them in the SCM along with the specific branch or tag that represents a specific version of your solution. Doing so allows you to retrieve the specific version of all code and assets. You can then create a deployment of that version if needed for bug fixes or regression testing on a specific version of a solution.

Consideration: This list applies only to assets stored in Case Manager directly, specifically in the Content and Process Engines. You must ensure that other assets, such as rules and custom code, are also stored and versioned in the SCM.

9.4 Sandbox options

Because of the wide range of potential developers and designers that are involved in a building a solution, you might need to create a sandbox area for each developer within the Case Manager development environment. Case Manager offers two mechanisms for handling the creation of sandboxes:

- ▶ Solution copies
- ▶ Solution templates

In a development environment, a solution copy is useful. As a solution is being developed, you might want to test a few ideas before you incorporate the changes into the primary solution. You can use a copy to create a sandbox solution for isolated development and testing without affecting the primary solution. When you are satisfied with changes, you can add the changes to the primary solution.

In a large implementation with many projects, a default set of solution artifacts might need to be available. A solution template is useful for this use case. For example, if you have default properties or document types in the object store, you can create a solution template that reuses these objects. Any solution that is created from such a template contains all the predefined reusable assets.

In addition, a solution template can provide a starting point for new solutions. If your organization has several similar lines of business, you can create an “80% solution” encompassing most of the baseline functionality. This baseline solution can be converted into a solution template and be used when you create a solution.

9.4.1 Copying a solution

There are two methods to copying a solution: Using Case Manager Builder and using IBM Case Manager administration client. More options are available when you use the IBM Case Manager administration client to perform a solution copy.

9.4.2 Copying a solution by using Case Manager Builder

Perform these steps to copy a solution in Case Manager Builder:

1. Log in to Case Manager Builder at:

`http://<server>:<port>/CaseBuilder`

2. Select the solution you want to copy to activate solution actions as seen in Figure 9-3.

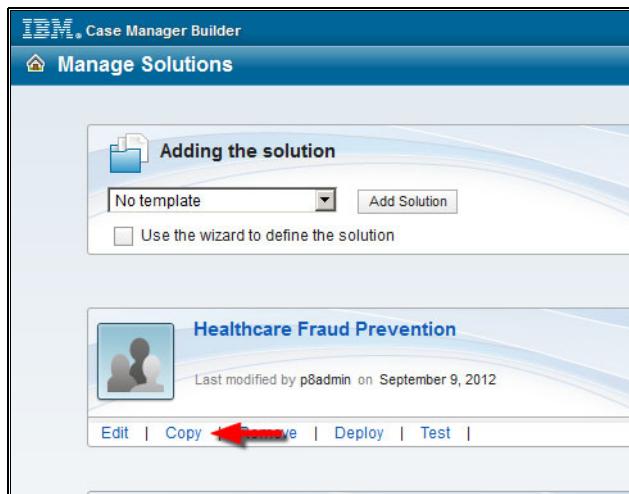


Figure 9-3 Copying a solution

3. Enter the name and solution prefix as shown in Figure 9-4.

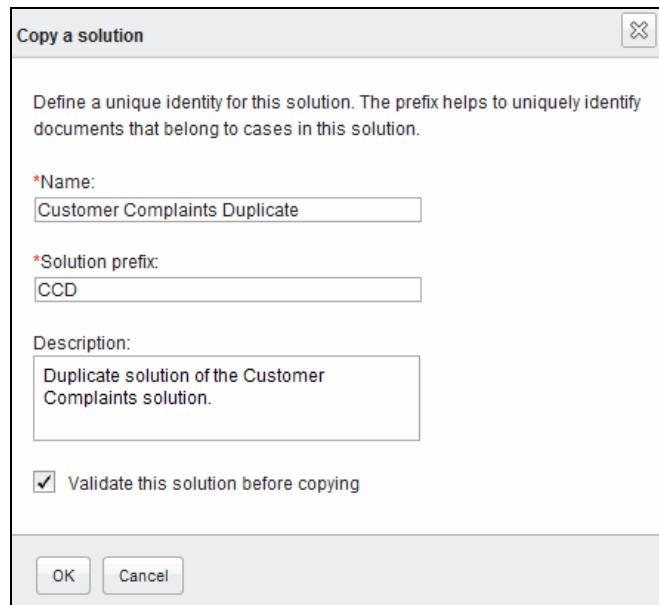


Figure 9-4 Copy a solution window

In this example, the name is Customer Complaints Duplicate, and the solution prefix is CCD.

4. Verify that **Validate this solution before copying** is selected to ensure that the new solution begins in a validated state, and click **OK**.

Restriction: You cannot copy a solution that has files checked out in the solution package if you select the **Validate this solution before copying** option.

Requirement: The solution prefix must be unique. A unique prefix ensures that the artifacts created in each solution are created as unique objects when deployed. Although this is not enforced when you create or copy solutions, you will get an error when you deploy a solution with a duplicate prefix.

The solution is then copied and is available for editing within Case Manager Builder as shown in Figure 9-5.

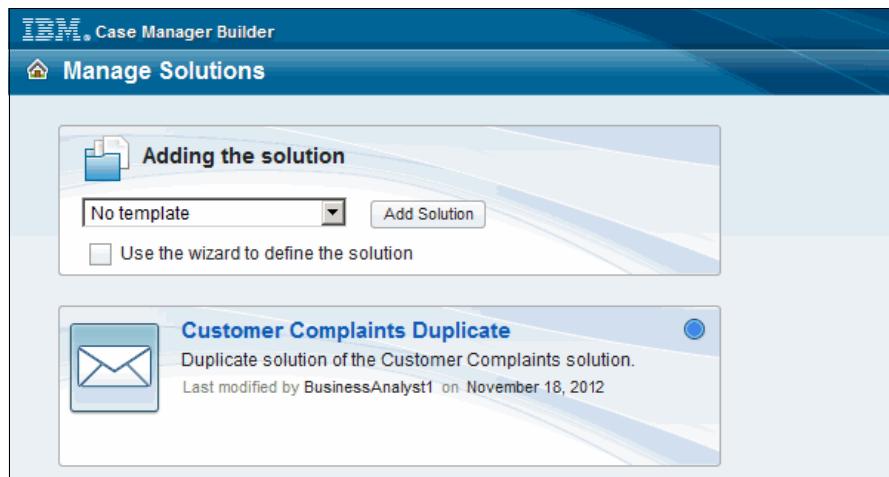


Figure 9-5 Copied solution

9.4.3 Copying a solution by using CMAC

An alternative to using Case Builder to copy a solution is to use the Case Manager Administration Client (CMAC). To use CMAC to copy a solution, complete the following steps. These steps are typically done by an IBM Case Manager administrator.

1. Start IBM Case Manager administration client:

```
<install_directory>\CaseManagement\configure\configmgr.exe  
<install_directory>/CaseManagement/configure/configmgr.sh
```

2. Click **Tools** → **Copy a Solution** as shown in Figure 9-6.

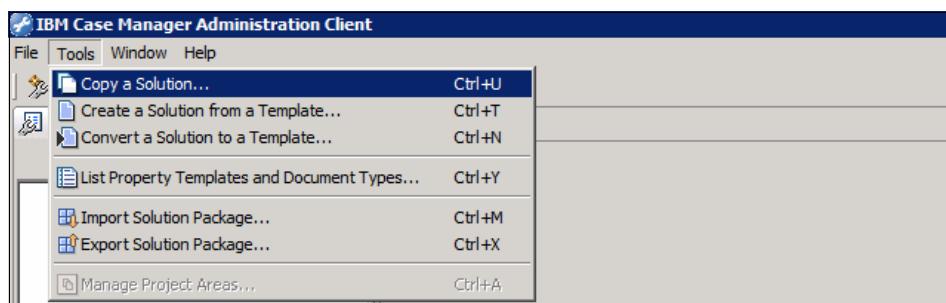


Figure 9-6 Accessing the **Copy a Solution** option in CMAC

3. Review the hints on the first window of the wizard, and click **Next**.
4. Enter the information for your environment and click **Next**.

For the example, the IBM FileNet Content Engine server name is `icmlab1`. The default port number is used, and `p8admin` is used for the Content Engine domain user name as shown in Figure 9-7.

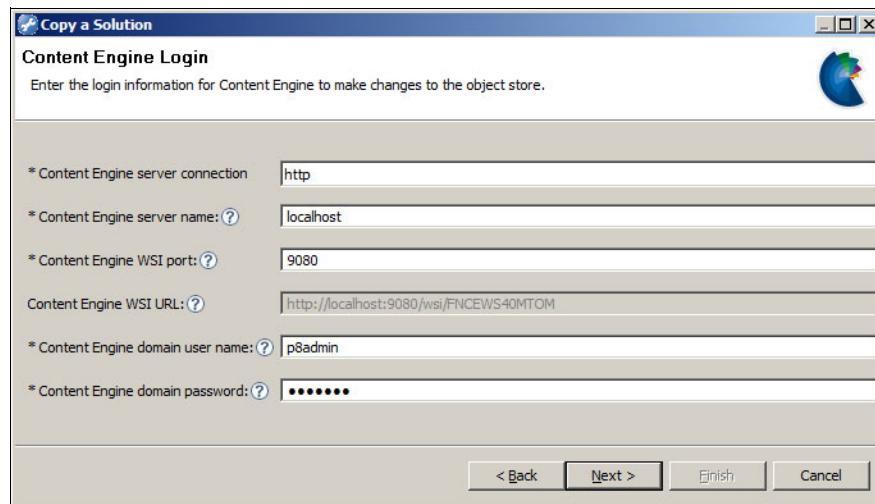


Figure 9-7 Copying a solution in CMAC

If you already have an existing profile, open it first before you select the **Tools** option. If you do so, you do not need to enter the Content Engine Login information.

5. Select the design object store and solution to be copied, and click **Next**.

In the example, **CMDOS** is selected as the design object store and **Customer Complaints** as the solution to copy. (Figure 9-8).

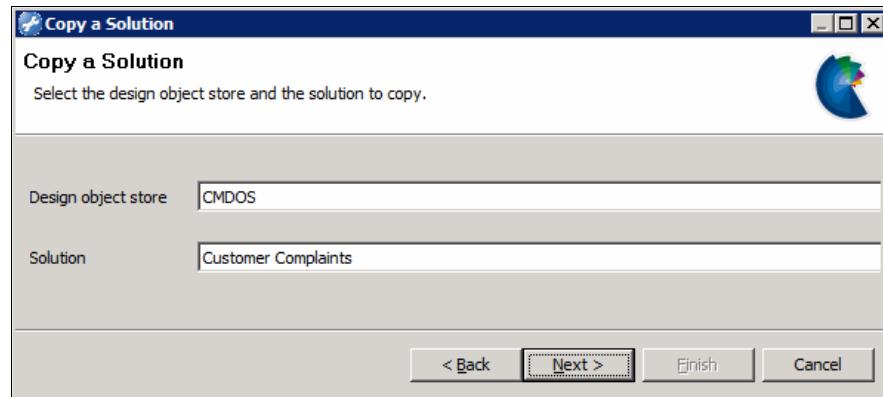


Figure 9-8 Copy a Solution: Selecting a solution to copy

6. Specify a name and prefix for the new solution, and leave the other options as default. Click **Finish**.

For more information, see 9.4.6, “Creating solution templates” on page 326.

In the example, Customer Complaints IT Duplicate is the name of the solution, and CCID is the prefix. See Figure 9-9.

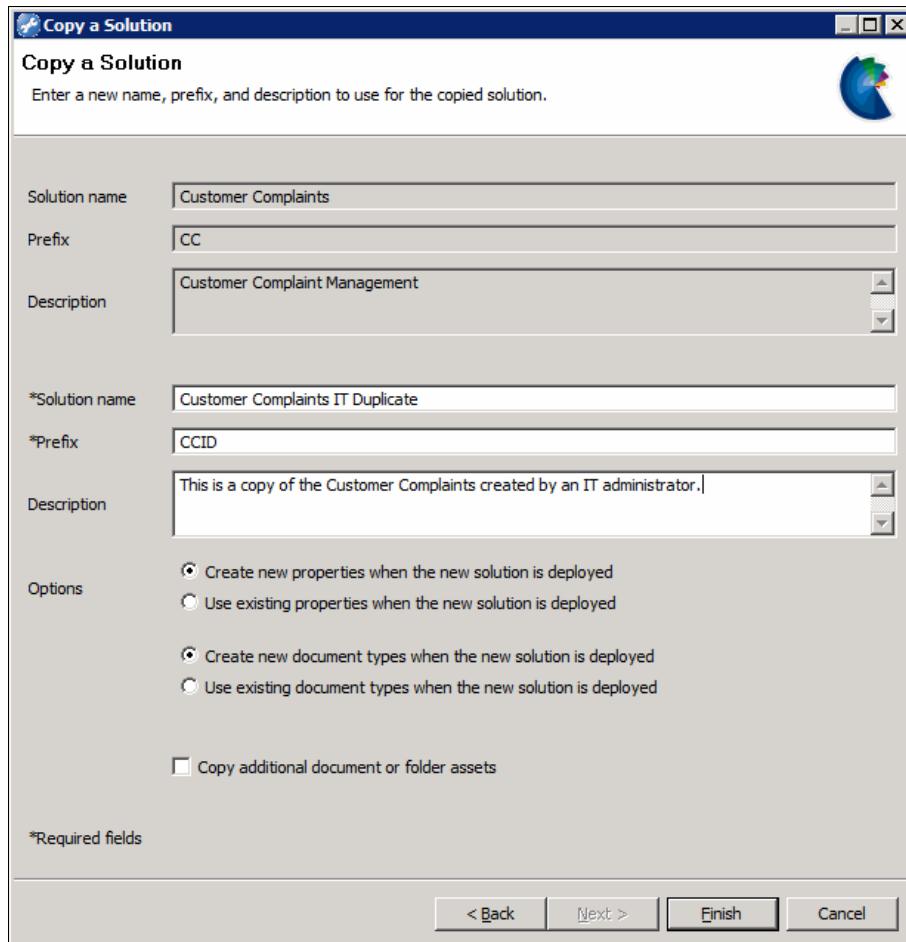


Figure 9-9 Copy a Solution: Specifying new solution details

If successful, the solution is available in Case Manager Builder for you to edit, as shown in Figure 9-10.

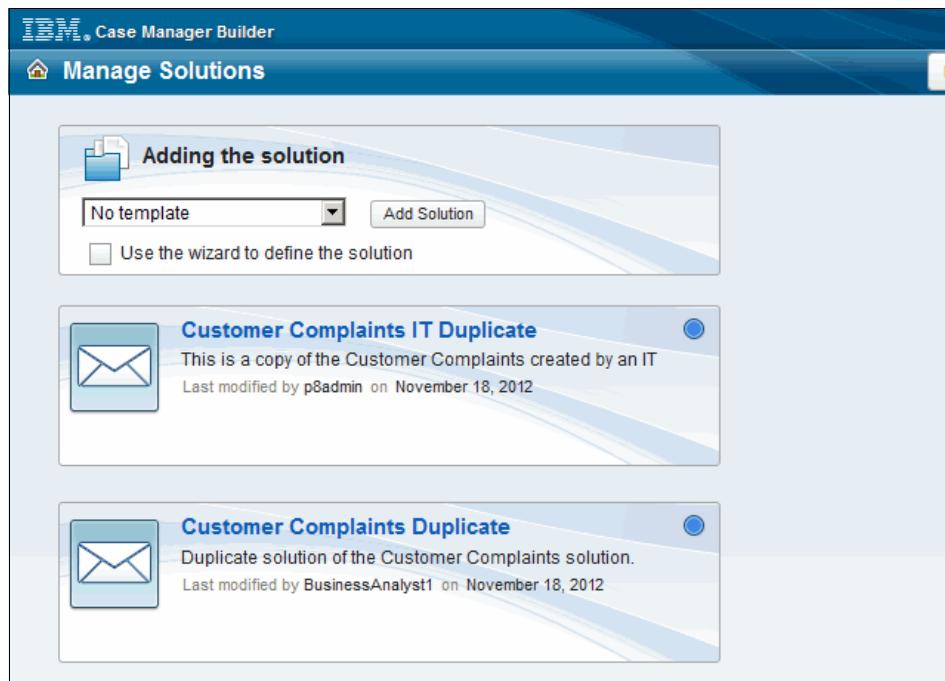


Figure 9-10 Copied solution available in Case Manager Builder

9.4.4 Extra options for an IBM Case Manager administrator

An IBM Case Manager administrator has an extra set of options that are not available through Case Manager Builder. The first set of options concern properties. The example solution has a property called Address that is used in the following examples.

- ▶ Create new properties when the new solution is deployed.

New properties are created for the new solution using the prefix that is defined for it. In the original example solution, the symbolic name of this property is CC_Address. In the new solution, with this option selected, the CC_Address property is copied into the new solution with the symbolic name of CCID_Address. The new property is controlled by the new solution, which can modify the property as needed.

- ▶ Use existing properties when the new solution is deployed.

Properties that are defined in the original solution continue to be used in the new solution. Properties that are defined in the original solution do not have

their prefix changed to match the prefix defined for the new solution. In the original solution, the symbolic name of this property is CC_Address. With this option selected, the property maintains the symbolic name CC_Address. The property is not controlled by the new solution. Only the original solution is able to modify the property.

Remember: If you use existing properties, you must track solution migration and deployments. A copied solution that reuses properties must be deployed after the original solution.

The second set of options concern document types. The example solution has a document type called **Customer Correspondence** that is used in the following examples.

- ▶ Use existing document types when the new solution is deployed

Document types that are defined in the original solution have their prefix changed to match the prefix that is defined for the new solution. In the original solution, the symbolic name of this document type is CC_CustomerCorrespondence. With this option selected, the document type takes on the symbolic name CCID_CustomerCorrespondence. The document type is then controlled by the new solution, which can modify the document type as needed.

- ▶ Use existing document types when the new solution is deployed

Document types that are defined in the original solution do *not* have their prefix changed to match the prefix that is defined for the new solution. In the original solution, the symbolic name of this document type is CC_CustomerCorrespondence. With this option selected, the document type maintains the symbolic name CC_CustomerCorrespondence. The document type is not controlled by the new solution. Only the original solution is able to modify the document type.

Remember: If you use existing document types, you must track solution migration and deployments. A copied solution that reuses document types must be deployed after the original solution.

The last option concerns assets in a solution folder.

- ▶ Copy additional document or folder assets

This option is useful if you decide to store any solution assets that are not defined in Case Manager Builder or Process Designer in the solution folder. This option duplicates those objects from the original solution folder to the new solution folder. The restriction is that only base folder and documents are copied.

9.4.5 Creating and using solution templates

A solution template is a collection of assets that are created by Case Builder that can be used as the basis of a new solution. These assets include the properties, document types, roles, tasks, and workflows for a solution.

This section describes how to convert a solution into a solution template. Creating a solution template is useful for standardizing default assets within a solution, or for creating an industry template to be deployed to external entities.

9.4.6 Creating solution templates

To promote a solution to a solution template, complete these steps:

1. Start IBM Case Manager administration client:

```
<install_directory>\CaseManagement\configure\configmgr.exe
```

2. Click **Tools** → **Convert a Solution to a Template** as shown in Figure 9-11.

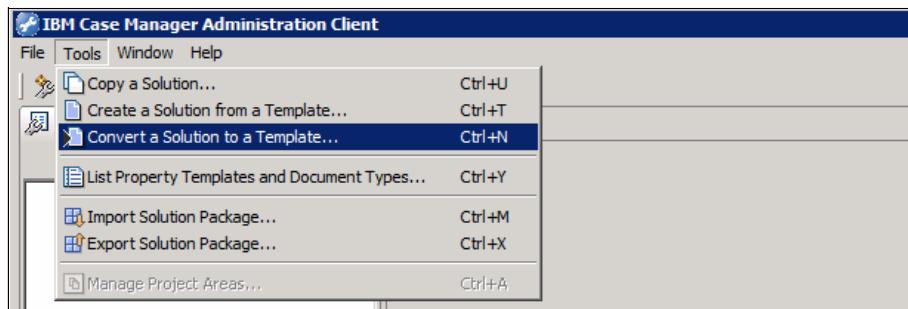


Figure 9-11 Selecting Convert Solution to a Template

3. Review the hints provided on the first window of the wizard, and click **Next**.
4. Enter the information for your environment and click **Next**.

If you already have an existing profile, you can open it first before you select the **Tools** option. If you do so, you do not need to enter login information.

For the example, icmlab1 is the Content Engine server name. The default port number is used, and p8admin is the Content Engine domain user name. See Figure 9-12.

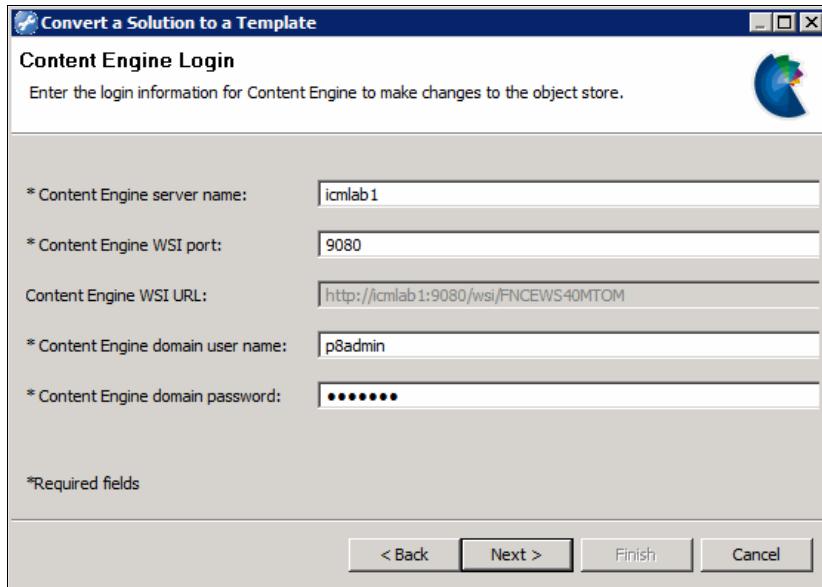


Figure 9-12 Convert a Solution to a Template: Content Engine Login

5. Select the design object store and solution to convert to template. Leave the options as default and click **Finish**. For more information, see 9.4.6, “Creating solution templates” on page 326.

For the example, **CMDOS** is the design object store and **Customer Complaints** is the solution to convert to template. See Figure 9-13.

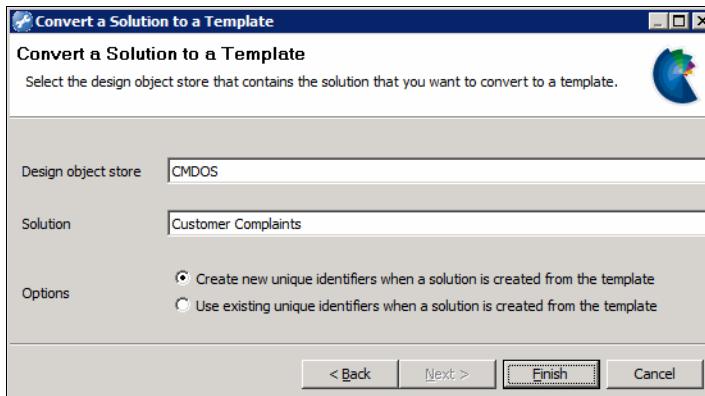


Figure 9-13 Convert a Solution to a Template: Specifying a Solution

Test your solution template after it is created and address any issues before you pass it to others to use. If the template creates new identifiers, do a run through of a new solution based on the template to verify that nothing was broken.

There are extra options when you create a solution template:

- ▶ Create new unique identifiers when a solution is created from the template
If your solution does not have any dependencies on any identifiers, select this option.
An example of using this option is if a solution was developed entirely in Case Manager Builder or Process Designer. With such a solution, no external assets rely on any symbolic names. This is the default option.
- ▶ Use existing unique identifiers when a solution is created from the template
If your solution has dependencies on any identifiers, and requires that those identifiers do not change names, select this option.
An example for using this option is if your solution contains extra assets that rely on objects in the solution. These extra assets can be an IBM Form, a stored search, or a custom application interfacing with solution data. A company-wide form might reference CC_Address from the sample solution. If so, a basic template provides the **Address** property to all new solutions so that the form does not require modification. The form continues to map to CC_Address and all new solutions that are based on the template contain CC_Address for the form.

9.4.7 Using solution templates

To create a solution from a solution template, you can use Case Manager Builder and IBM Case Manager Administration Client.

Using a template in Case Manager Builder

To create a solution using a solution template in Case Manager Builder, complete these steps.

1. Log in to Case Manager Builder:

`http://<server>:<port>/CaseBuilder`

2. Select the template that you want from the list and click **Add Solution** as shown in Figure 9-14.



Figure 9-14 Selecting a template to create a solution in Case Manager Builder

3. Complete the solution name and prefix for the new solution, and click **OK**.

In the example, Customer Complaints Newer is the solution name, and CCN2 is the solution prefix as shown in Figure 9-15.

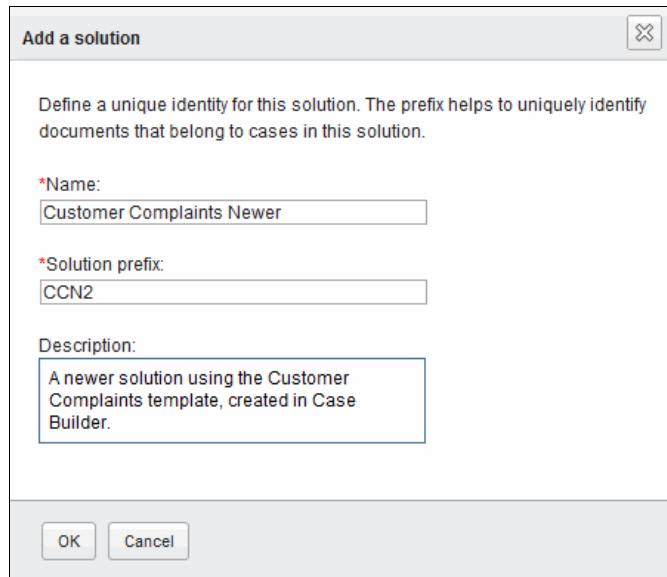


Figure 9-15 Specifying new solution details

4. The solution is created, and you are automatically taken to the edit window of the new solution.

Using a template in IBM Case Manager Administration Client

To create a solution using a solution template in IBM Case Manager Administration Client, complete these steps:

1. Start the IBM Case Manager administration client:

```
<install_directory>\CaseManagement\configure\configmgr.exe
```

2. Click **Tools** → **Create a Solution from a Template** as shown in Figure 9-16.

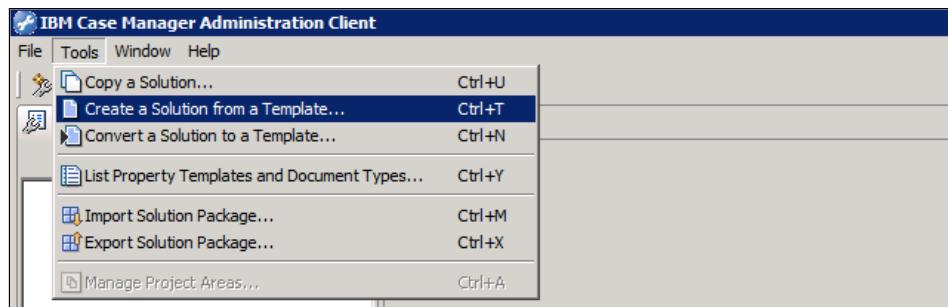


Figure 9-16 Selecting Create a Solution from a Template

3. Review the hints that are provided on the first window of the wizard, and click **Next**.
4. Enter the information for your environment and click **Next**.

For the example, `icmlab1` is the Content Engine server name. The default port number is used, and `p8admin` is the Content Engine domain user name. See Figure 9-17.

A screenshot of the 'Create a Solution from a Template' wizard. The title bar says 'Create a Solution from a Template'. The window is titled 'Content Engine Login' with the sub-instruction 'Enter the login information for Content Engine to make changes to the object store.' Below this is a logo of a globe. The form contains the following fields:

- * Content Engine server connection: http
- * Content Engine server name: localhost
- * Content Engine WSI port: 9080
- Content Engine WSI URL: `http://localhost:9080/wsi/FNCEWS40MTOM`
- * Content Engine domain user name: p8admin
- * Content Engine domain password: (redacted)

At the bottom are buttons for '< Back', 'Next >', 'Finish', and 'Cancel'.

Figure 9-17 Connection settings for creating a solution from a template

5. Select the design object store and solution template to be used, and click **Next**.

For the example, select **CMDOS** as the design object store and **Customer Complaints** as the solution template to use (Figure 9-18).

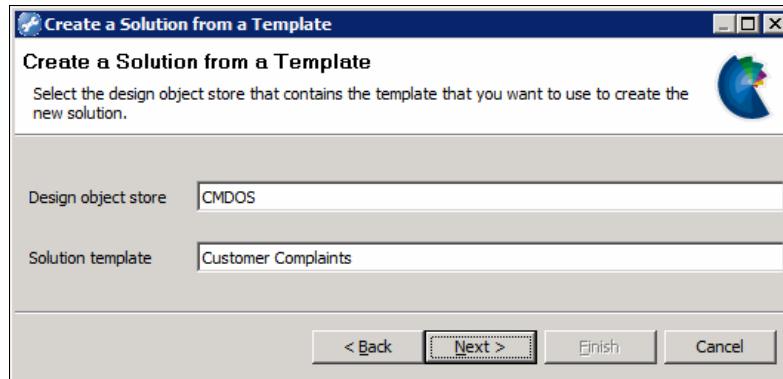


Figure 9-18 Create a Solution from a Template: Selecting solution template

6. Specify a name and prefix for the new solution and the project area. Leave the other options as default, and click **Finish**. For more information, see 9.4.6, “Creating solution templates” on page 326.

For the example, enter Customer Complaints New as the name of the new solution and CCN as the prefix. See Figure 9-19 on page 333.

If you have more than one project area, you can select another project area. For the example, use the default project area of dev_env_connection_definition.

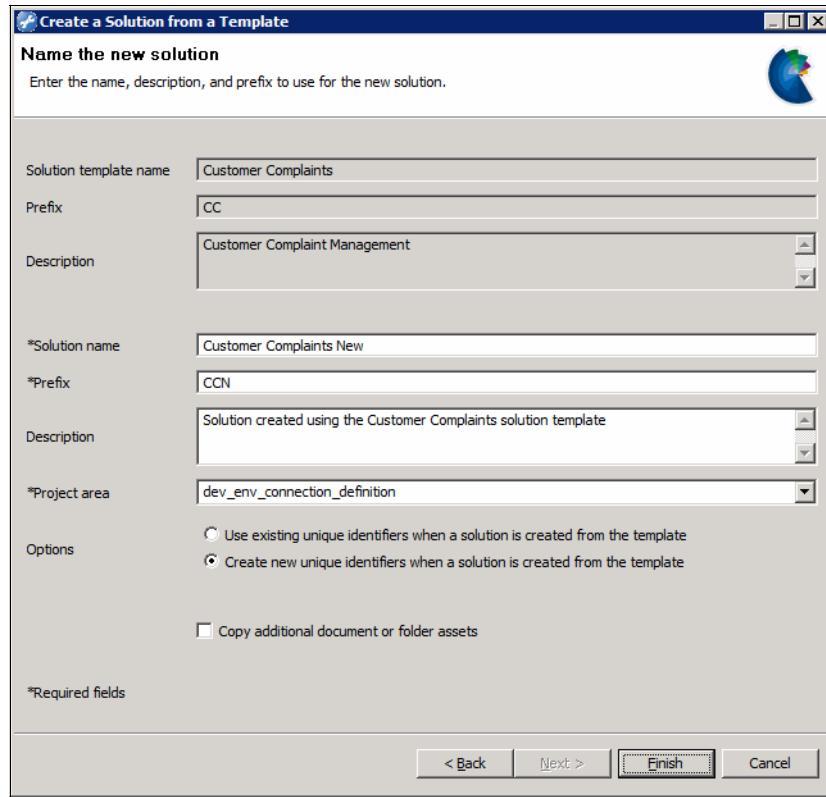


Figure 9-19 Create a Solution from a Template: Specifying new solution details

Upon completion, you are informed whether the action was successful. If so, you see your new solution in Case Manager Builder.

9.4.8 Available solution templates

IBM Case Manager comes with the following solution templates:

- ▶ Auto Claims solution template
- ▶ Credit Card Dispute Management solution template

You can use these templates as a reference guide to create your own solutions or to create other solution templates. These templates give you a jump-start in creating an IBM Case Manager solution.



User interface

This chapter addresses the user interface, which includes introducing default widgets, creating a custom iWidget, manipulating pages, controlling payloads, and customizing the runtime appearance.

This chapter includes the following sections:

- ▶ Using default widgets and their payloads
- ▶ Creating and deploying a custom iWidget
- ▶ Manipulating pages in business space
- ▶ Example of customizing the Investigator Step
- ▶ Changing the runtime appearance
- ▶ User interface navigation and case accessing
- ▶ Managing documents in context of a case
- ▶ Using the Case Form widget
- ▶ Using IBM Forms with IBM Case Manager

10.1 Using default widgets and their payloads

This section addresses default widgets that are provided by IBM Case Manager and their payloads.

10.1.1 Standard widgets provided by IBM Case Manager

Case Manager Client provides a series of widgets for case workers to manage cases and process work items. You can drag these widgets into your spaces and pages. The following are the standard widgets provided by IBM Case Manager:

- ▶ **Attachments widget**
Displays a list of the documents that are attached to a work item.
- ▶ **Case Data widget**
Used to view and edit property values of a case or work item.
- ▶ **Case Form widget**
Used to view and edit case property values or work item fields values with help of IBM Forms or FileNet eForms.
- ▶ **Case Information widget**
Displays an overview of a case. It can display up to four views: Summary, documents, tasks, and history.
- ▶ **Case List widget**
Receives search criteria from the Case Search widget or custom search widget, runs the search, and displays the search result.
- ▶ **Content List widget**
Runs a search that is based on a predefined search template, and displays the list of matching documents.
- ▶ **Case Toolbar widget**
Provides actions that case workers can take for an opened case.
- ▶ **Command widget**
This hidden widget is on each page. It controls page navigation when a case worker opens, creates, or updates a case or work item. It also handles data loading, and saving case and work item details.
- ▶ **Connector widget for IBM Business Process Manager widget**
Enables IBM Case Manager to work with the IBM Business Process Manager Inbox widget. The Inbox widget displays both IBM Case Manager work items and IBM BPM tasks.
- ▶ **Document Viewer widget**
Displays documents that are stored in FileNet P8 object stores or IBM Content Manager repositories.

- ▶ In-baskets widget
Used to view and work with work items in their personal in-basket and the in-basket that is associated with their role.
- ▶ Original Case Data widget
Displays property values for a case that is being split to create another case. A case worker can compare the values in this widget with the values that are displayed in the Split Case Data widget for the new case.
- ▶ Case Search widget
Used to define the search conditions for cases that are based on selected property values. This widget works with the Case List widget to run queries of cases.
- ▶ Select Case Documents widget
Used to select the documents from an existing case that are also to be associated with a new split case.
- ▶ Split Case Data widget
Used to create a case that reuses property values from an existing case.
- ▶ Toolbar widget
Used to open a web page, add cases, manage roles, or run a custom action.
- ▶ Work Item Toolbar widget
Used to respond to work items or to add new tasks.
- ▶ Process History widget
Shows the status of a task by showing the milestones defined in the process.

10.1.2 Widgets payload

The widget communication is established by events. Most Case Manager Client widgets publish events and handle events. The data that is published or handled by an event is called a *payload*.

A widget can publish an event in the following ways:

- ▶ The widget can broadcast the published event to all other widgets on the page. The event is processed by any widget that can handle the published event. In the mashup environment, broadcast events are not listed in the wiring dialog box.
- ▶ The widget can send the published event to specific widgets. This method of sending events is accomplished by wiring the published event to a corresponding event that is handled by the receiving widget.

You can add a wire between a source widget and a target widget. When the source widget publishes the wired event, the published event is sent to be

handled by the specified event in the target widget. The payload type must be same in both published event and handled event.

10.1.3 Wiring a widget to another widget

To wire a widget to another widget, complete these steps:

1. Click **Edit Page**.
2. On the source widget, select the Widget menu icon and select **Edit Wiring** as shown in Figure 10-1.

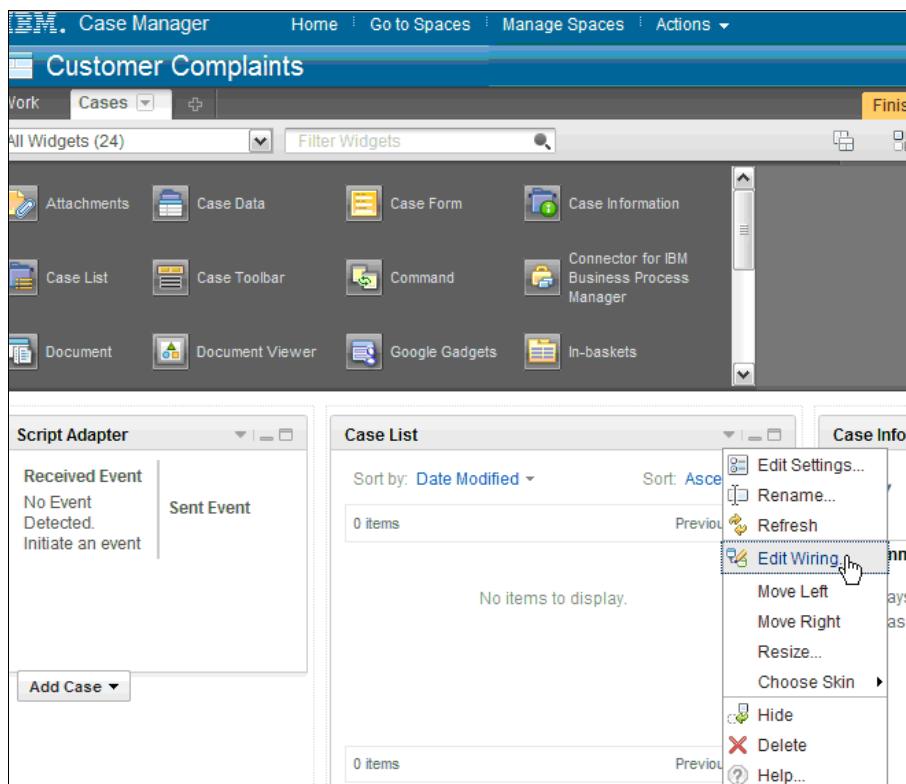


Figure 10-1 Selecting Edit Wiring from the Widget menu icon

The Widget Wiring window opens with focus on the widget and the wiring it has with other widgets on the page.

3. Click **Add outgoing wire** (Figure 10-2) or click the outgoing arrow for the widget, and select the target widget from the list. The list includes all of the widgets in the page that can handle events. The widgets that can only publish events but not handle events are not listed.

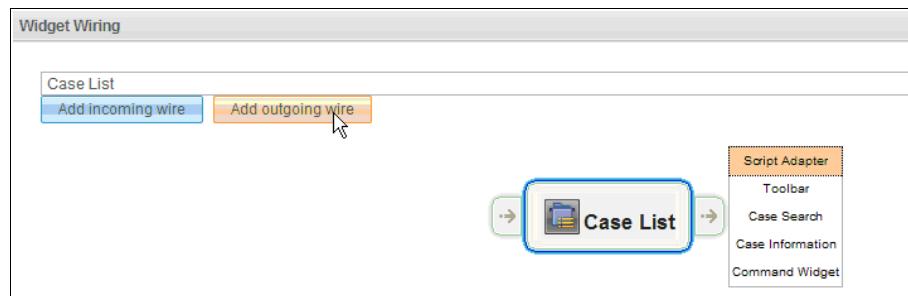


Figure 10-2 Selecting the target widget

For the example, select **Script Adapter**. The Widget Wiring window adds the target widget to its display and draws a wire between the source and target widgets. See Figure 10-3.

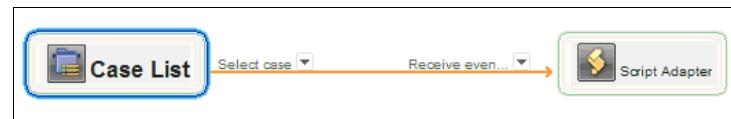


Figure 10-3 Widget wire

The source end of the wire displays the event that is sent by the source widget. The target end of the wire displays the event that the target widget receives.

4. To change the source or target event, click the wire menu icon and select an event from the list that opens.

For the example, select **Select case** for the source event as shown in Figure 10-4.

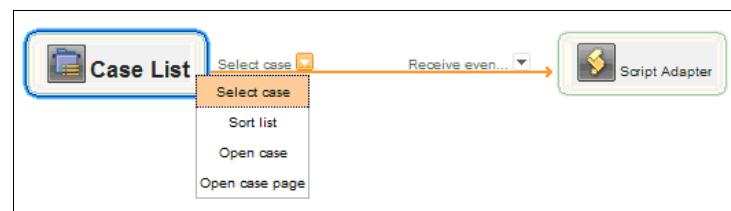


Figure 10-4 Selecting the source event

5. Select **Receive event payload** for the target event as shown in Figure 10-5

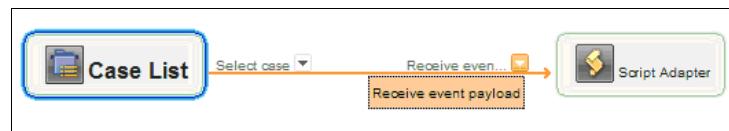


Figure 10-5 Selecting the target event

6. You can add more wires (inbound and outbound) to the widget.
7. To change the focus to another widget, select it in the diagram or from the list of widgets. Figure 10-6 shows the extra wires added between the Case List and Script Adapter for the example.

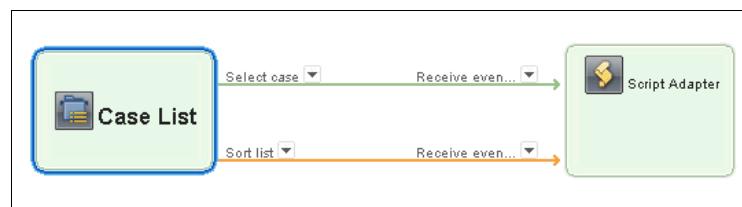


Figure 10-6 Adding a wire

8. When you complete your wiring (Figure 10-7), click **OK**.

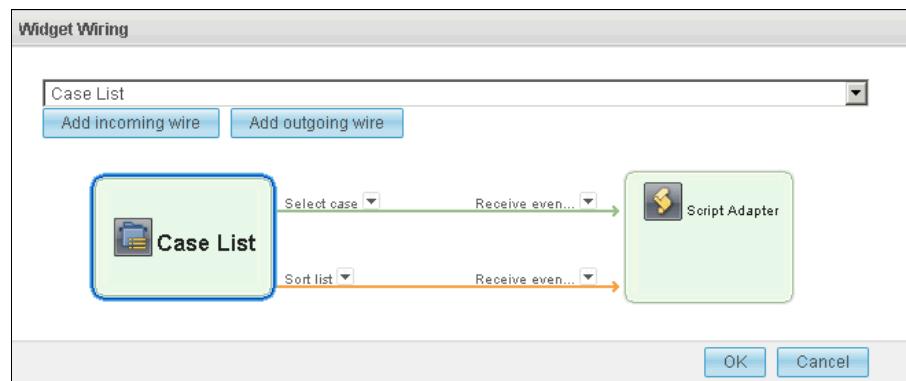


Figure 10-7 Completed widget wiring

9. Click **Save** (Figure 10-8) and **Finish Editing** to save your widget wiring.

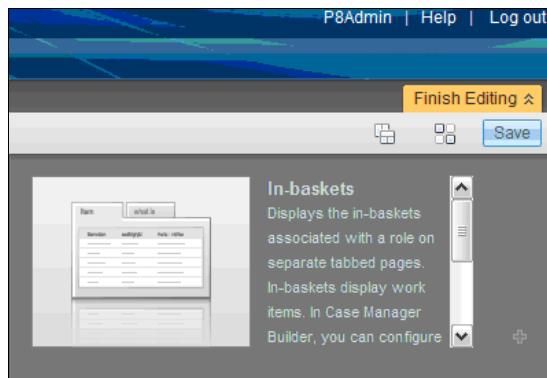


Figure 10-8 Saving the widget wiring

10. When you click one case instance in Case List widget, you can see that Script Adapter can receive the Select Case event (Figure 10-9).

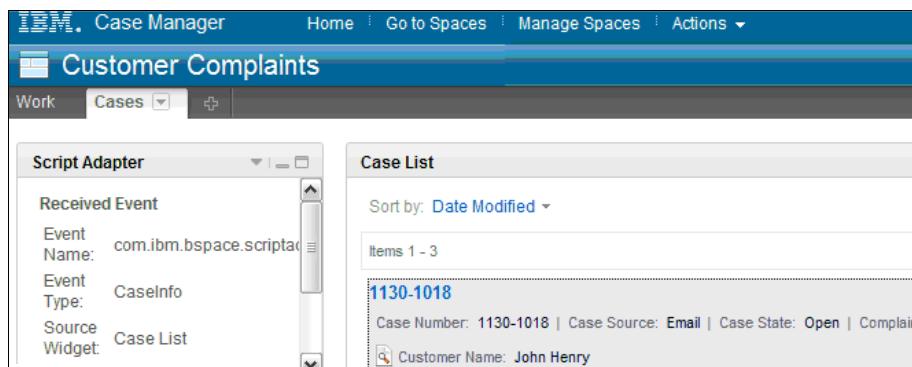


Figure 10-9 Received event in the Script Adapter widget

10.1.4 Payload types of Case Manager Client widget

Payload types define the format of data that is passed in events between widgets. If you develop custom widgets, know the payload types that are used by the events so that you can communicate with the widgets provided by Case Manager Client.

The events that are used in the Case Manager Client have many payload types. The syntax for each payload type is shown in the iWidget syntax. Example 10-1 shows the structure of a payload that is formatted as a JSON object.

Example 10-1 Structure of a payload

```
{ "ETEProperty1":  
  { "value": true,  
    "desc": "",  
    "mode": 3,  
    "modified": false,  
    "type": 4,  
    "name": "ETEProperty1",  
    "isArray": false  
  },  
}
```

10.1.5 In-basket widget event and corresponding payload

The events that are published by the In-baskets widget are displayed in Table 10-1.

Table 10-1 Events that are published by the In-baskets widget

Event title	Event ID	Payload	Description
Column name	com.ibm.ecmwidgets.acm.column.name	Value of column	The user selected a work item in an in-basket. The payload of this event contains the value of the selected column.
In-basket selected	com.ibm.ecmwidgets.acm.SelectInBasket	In-basket	This event is used to get the current in-basket tab that is run when the In-basket widget is initialized or the user clicks a tab.
Open Work Details page	com.ibm.ecmwidgets.acm.OpenWorkPage	WorkItem	The user selected a work item to open.
Row selected	com.ibm.ecmwidgets.acm.SelectRow	WorkItem	The user clicked a row or pressed Enter in the in-basket to select the work item.

Event title	Event ID	Payload	Description
Send selected rows	com.ibm.ecmwidgets.acm.SendSelectedRows	SelectedRows	This event contains information of which rows are currently selected in the In-basket widget.

The events that are handled by the In-baskets widget are displayed in Table 10-2.

Table 10-2 Events that are received by the In-baskets widget

Event title	Event ID	Payload	Description
Apply filter	com.ibm.ecmwidgets.acm.ApplyFilter	Filter	Update the work items that are listed in the in-basket based on the specified filters.
Clear content	com.ibm.ecmwidgets.acm.ClearContent	Empty	Clears the content in the In-baskets widget and displays the widget description.
Refresh	com.ibm.ecmwidgets.acm.TaskUpdated	Empty	Refresh the in-basket to update the list of work items after a user completes or closes a work item.
Request selected rows	com.ibm.ecmwidgets.acm.RequestSelectedRows	ExtraData	Send the information for the in-basket rows that are specified in the event payload.
Role Selected	com.ibm.ecmwidgets.acm.RoleChanged	Role	Updates the In-baskets widget to display the in-baskets that are associated with the specified role.

The In-basket payload type, as an example, contains information about the in-baskets that is displayed in the In-baskets widget. The in-baskets are displayed based on the role of the case worker. The payload type includes the properties as listed in Table 10-3.

Table 10-3 Payload type properties

Property	Type	Description
inbasketCount	number	The number of in-baskets that are defined for the current role of the case worker.
currentInbasket	string	The name of the selected in-basket.
queueName	string	The name of the queue with which the current in-basket is associated.
appSpaceName	string	The name of the application space in which the current role is defined.
currentRoleName	string	The name of the role that the case worker is logged in as.
filters	any	The filters that are defined for the current in-basket.
columns	any	The columns that are defined for the current in-basket.

The Script Adapter widget that is provided by business space can automatically parse the received payload from the source widget. You can wire the In-basket widget and Script adapter widget to see the payload in detail. See Figure 10-10.

```

Event Type: InBasket
Source Widget: In-baskets
Source ID: M090C05D489322D8BF289E41B5FC
Payload: {"inbasketCount":2,"currentInbasket":"Correspondence Management","currentRoleName":"Case Management","queueName":null,"columns":[]}
  
```

Figure 10-10 Payload tab in the Script Adapter widget

For more information about payloads, see the IBM Case Manager Information Center at:

<http://publib.boulder.ibm.com/infocenter/casemgmt/v5r1m0/index.jsp?topic=%2Fcom.ibm.casemgmt.install.doc%2Facmwd050.htm>

10.1.6 Case Search widget event and corresponding payload

The events that are published by Case Search widget are shown in Table 10-4.

Table 10-4 Events that are published by the Case Search widget

Event title	Event ID	Payload	Description
Search cases	com.ibm.ecmwidgets.acm.SearchCases	CaseSearch	The user starts a search for cases.

The events that are handled by Case Search widget are shown in Table 10-5.

Table 10-5 Events that are handled by the Case Search widget

Event title	Event ID	Payload	Description
Clear content	com.ibm.ecmwidgets.acm.ClearContent	Empty	Clears the content in the Case Search widget and displays the widget description.

The CaseSearch payload type contains the search criteria that a case worker enters in the Case Search widget. This payload is passed to the Case List widget that then runs the search. The CaseSearch payload type includes the properties that are shown in Table 10-6.

Table 10-6 Case Search payload type properties

Property	Type	Description
CaseType	string	The type of case for which the case worker is searching. This property is an empty string when the case worker searches across case types.
ObjectStore	string	The object store to be searched for the case.
QuerySQL	string	The SQL representation of the search query that is used to run the search.
SearchProperties	any	The properties that the case worker selected to search on. These properties are displayed for each case in the Case List widget.

Property	Type	Description
SortingProperties	any	The case properties that can be used to sort the cases that are displayed in the Case List widget.
SummaryProperties	any	The summary properties that are displayed for each case in the Case List widget. You select the summary properties in Case Manager Builder.
SystemProperties	any	The system properties that are displayed for each case in the Case List widget. This property is null when the case worker searches across case types.

10.2 Creating and deploying a custom iWidget

This section shows you how to create a widget called the Custom Search widget and deploy it in the IBM Case Manager, Version 5.1 environment. The Custom Search widget has a similar role to the Case Search widget. In simple search mode, the default Case Search widget uses only one property to search cases. The Custom Search widget allows a user to search cases with multiple properties. See Figure 10-11.

The screenshot shows the IBM Case Manager interface with the following components:

- Header:** IBM Case Manager, Home, Go to Spaces, Manage Spaces, Actions.
- Navigation Bar:** Work, Cases, Custom Search (selected), +
- Customer Complaints Dashboard:**
 - CustomSearch:** Last Modifier: P8Admin, Customer Name: John, Search button.
 - Search:** Search dropdown set to Case Number, Search input field.
- Case List:** Sort by: Last Modified Date, Sort: Ascending, Items 1 - 2.
 - 1130-1018:** Case Number: 1130-1018, Case Source: Email, Case State: Open, Last Modifier: Null, Customer Name: John Henry.
 - 1130-0001:** Case Number: 1130-0001, Case Source: , Case State: , Complaint Ca..., Last Modifier: Null, Customer Name: John Doe.

Figure 10-11 Custom Search widget

Complete these steps to create the Custom Search widget:

1. Create a project contain the files for the Custom Search widget (see Figure 10-12). You also can use Eclipse or another integrated development environment (IDE).

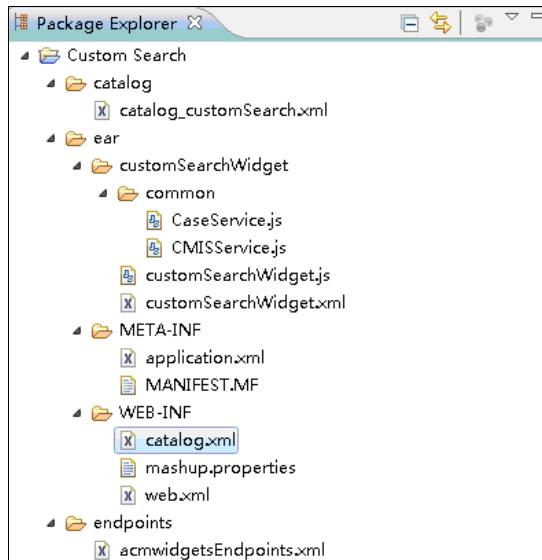


Figure 10-12 Creating a project for the Custom Search widget

2. Create the widget definition file. The file mainly defines source files, events, and attributes for the widget. A widget definition file must conform to the iWidget specification 2.0. business space loads your iWidget according to the file.

The widget name that it uses:

```
<iw:iwidget  
    id="com.ibm.im.ecmwidgets.sample.CustomSearchWidget"  
    xmlns:iw="http://www.ibm.com/xmlns/prod/iWidget"
```

Here is the iScope of the widget:

```
iScope="com.ibm.im.ecmwidgets.sample.CustomSearchWidget"
```

Only view mode can be supported in the Custom Search widget. You can also define the edit mode according to your requirement:

```
allowInstanceContent="true"  
supportedModes="view"  
mode="view"  
lang="en">
```

The services are used as resources by the Custom Search widget. CaseService.js is used to retrieve view definitions, whereas CMISService.js is used to retrieve property definitions.

```
<iw:resource uri="common/CMISService.js"/>
<iw:resource uri="common/CaseService.js"/>
```

The name of the file that contains the scope definition is:

```
<iw:resource uri="customSearchWidget.js"/>
```

The widget definition specifies which events can publish (send), and which events the widget can handle (receive). See Example 10-2.

Example 10-2 Widget event

```
<iw:event eventDescName="desc_search" published="true"
           id="com.ibm.ecmwidgets.acm.SearchCases"/>
  <iw:eventDescription id="desc_search"
                        payloadType="CaseSearch">
    <iw:alt description="The user executed a search for cases."
           lang="en"
           title="Search cases"/>
  </iw:eventDescription>
</iw:event>
```

The event definition is mapped to the widget wiring user interface as shown in Figure 10-13.

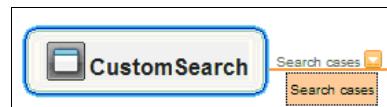


Figure 10-13 Event definition mapped to widget wiring

The View Mode markup is used to render the widget user interface (Example 10-3).

Example 10-3 View Mode markup

```
<iw:content mode="view">
  <![CDATA[
    <div id="_IWIID_viewModeContent"
         class="ecmwdgt ecmwdgtViewModeContent"/>
  ]]>
</iw:content>
<iw:content mode="edit">
  <![CDATA[
```

```

<div id="_IWID_editModeContent"
      />
    ]]>
  </iw:content>
</iw:iwidget>

```

3. Create customSearchWidget.js. The file implement onLoad and onView method. Business space calls the two methods when the iWidget is loaded. The onLoad is for initialize Case Manager REST service and CMIS service. The onView method is for rendering user interface.
4. Create CaseService.js. The service sends a REST request to retrieve the view definition when the widget is loaded. See Figure 10-14.

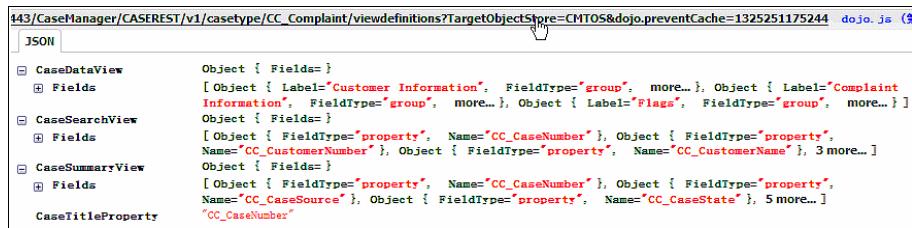


Figure 10-14 View event definition

5. Create CMISService.js. The service calls CMIS to retrieve case instances that match the search condition. See Figure 10-15.



Figure 10-15 Query to retrieve case instances

6. Create catalog.xml. This file mainly contains the entry ID, widget name, widget description, icon, and help content.
7. Create acmwidgetsEndpoints.xml. This file mainly contains the id, type, version, and URL.
8. Package the Custom Search widget:
 - a. Create CustomSearch.ear containing all files under the ear folder.
 - b. Create an archive file that contains the ear directory, catalog, and endpoints. Ensure that the structure of the archive file contains the following items:
 - ear\widgets_name.ear (one or more EAR files)
 - catalog\catalog_name.xml

- endpoints\endpoints_name.xml
9. Deploy the Custom Search widget:
- From a command line, change to the profile_root/bin directory.
 - From the command line, enter the following command (Figure 10-16):
wsadmin.bat -connType NONE

```
C:\Program Files\IBM\WebSphere\AppServer\profiles\AppSrv01\bin>wsadmin.bat -conn
type none
WASX7357I: By request, this scripting client is not connected to any server proc
ess. Certain configuration and application operations will be available in local
mode.
WASX7029I: For help, enter: "$Help help"
```

Figure 10-16 Running the wsadmin command

- c. From the command line, enter the following command (Figure 10-17):

```
$AdminTask installBusinessSpaceWidgets {-nodeName node
-serverName server -widgets fullPath}
```

```
wsadmin>$AdminTask installBusinessSpaceWidgets {-nodeName Base-Win2k8x64Node01 -
serverName server1 -widgets c:/Widgets_CustomSearch.zip}
```

Figure 10-17 Running the AdminTask command

- d. Save the configuration by entering the following command (Figure 10-18):

```
$AdminConfig save
```

```
wsadmin>$AdminConfig save
wsadmin>exit
C:\Program Files\IBM\WebSphere\AppServer\profiles\AppSrv01\bin>_
```

Figure 10-18 Running AdminConfig save

- e. Restart the WebSphere Application Server.

The custom widget is displayed in the widget pallet of business space as shown in Figure 10-19.

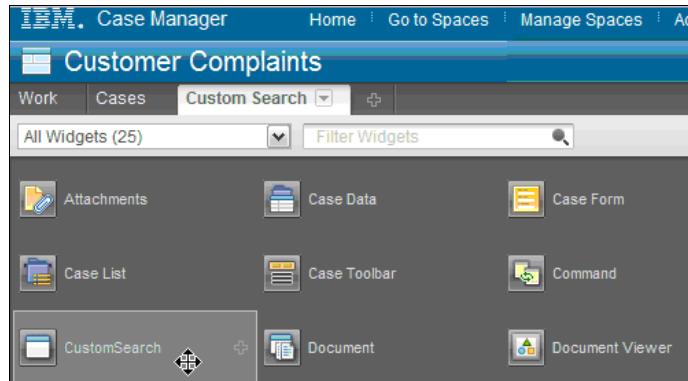


Figure 10-19 Custom Search widget displayed in the widget pallet

10. Before you use Customer Search, you need to modify some parameters to match your environment. Open customerSearchWidget in the deployed folder as shown in Figure 10-20.

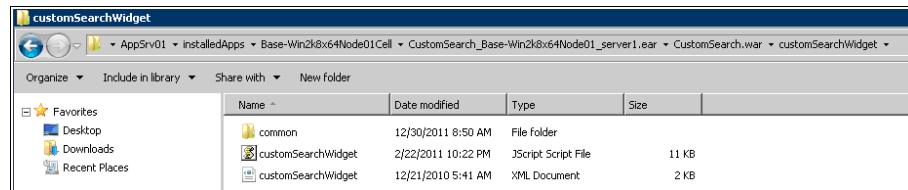


Figure 10-20 CustomSearchWidget shown in the deployed folder

In the functions `_getSolutionName`, `_getObjectStore`, and `_getCaseType`, modify the returning result for solution name, target object store, and case type to match your environment.

In the functions `_showSearchableProperties` and `_getSearchedProperties`, modify the searchable properties to match your environment.

11. Use the custom widget:

- a. Wire the widget with the Case List widget as shown in Figure 10-21.



Figure 10-21 Wiring the custom widget with the Case List widget

- b. Enter the value for searchable properties, then click **Search**. The Case List widget shows the corresponding case instance according to your search condition.

10.3 Manipulating pages in business space

This section describes how to manipulate the pages in business space. When you first deploy a solution, IBM Case Manager automatically creates a set of default pages. You can create more pages to use with your case management application. This section introduces how to add security to the business space pages and IBM Case Manager solution default pages. Additionally, it introduces how to create custom pages and cover eforms/forms pages.

This section assumes that you have already created, deployed, and tested your solution with the Case Manager Builder application in your development environment.

10.3.1 Security

Both spaces and pages are secured with View and Edit access control lists. The owner of a page or space has permission to assign access rights to groups of users.

By default, a page inherits security from the space. However, access rights can be configured on the individual pages as well. Like business spaces, pages are secured with View and Edit access control lists.

View access

Users with view access can navigate the pages in a business space and see the widgets on a page. Viewers can resize widgets and change the order of the page

tabs. Resizing widgets is specific to each user. Other users are not affected by this change.

Edit access

Users with edit access can add, configure, and delete widgets in a page. Editors can perform any action on a page that viewers can. If an editor creates a page, the editor becomes the owner of that page, even if the editor is not the owner of the space that contains the page.

By sharing action in page or space, you can assign the specified security to different user or group as shown in Figure 10-22.

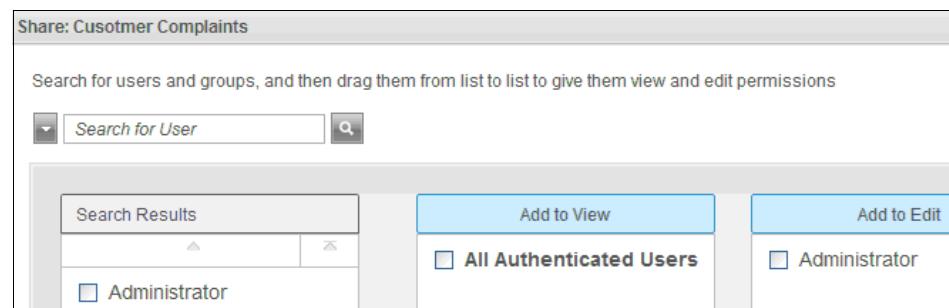


Figure 10-22 Share space

10.3.2 Default pages

Case Manager Client automatically creates three spaces for every deployed solution, Solution space, Step pages space, and Case pages space. Every space contains corresponding default pages.

Solution space

The Solution space is the main IBM Case Manager space. The space contains the following pages:

- ▶ Work page (for displaying work items)
- ▶ Case page (for displaying case instances)

Step page space

The Step page space allows a case worker to open a work item. It contains the following pages:

- ▶ Add Task page (for adding a task for the case)
- ▶ Add Task Form page (for adding a task for the case by using a form)
- ▶ Work Details page (for working on a work item)

- ▶ Work Details Form page (for working on a work item by using a form)
- ▶ Form Attachment Work Details page (for working on a form-based work item)

Case page space

The Case page space allows case workers to create or interact with specific cases. The space contains the following pages:

- ▶ Case Details page (for viewing and editing case data)
- ▶ Add Case page (for creating a case)
- ▶ Split Case page (for splitting a case thus creating a case)
- ▶ Case Details Form page (for viewing and editing case data by using a form)
- ▶ Add Case Form page.(for creating a case by using a form)

10.3.3 Custom pages

You can create and register new pages for case works in different spaces to meet specified requirements.

Customizing a page in Solution space

To customize page in Solution space, create a business space page, and configure widgets and delete widgets as needed. Remember that the command widget is required.

The custom page in Solution space can be used immediately.

Customizing a page in Case pages space

To customize page in Case pages space:

1. Create a business space page, configuring widgets and deleting widgets as needed. Remember that the command widget is required.
2. Register the custom page in the Case pages space (Figure 10-23 on page 355):
 - a. Click **Actions** → **Case Manager Page Settings**.
 - b. Select **Case page** as the page type and click **Save** to register the page.

Figure 10-23 Custom Case page registration

There are three page types: Add Case page, Case page, and Split Case page.

3. In Case Manager Builder, open a case type or solution. Then, associate the new Case page with a case type. The custom page is listed in the corresponding category according to your register page type.

For example, if you select Case page as register page type, the custom page is listed in Default layout for Case Details page (Figure 10-24).

Figure 10-24 Change Case Detail page layout

Customizing a page in Step page space

To customize page in Step page space:

1. Create a business space page, configuring widgets and deleting widgets as needed. Remember that the command widget is required.
2. Register the custom page in the Step page space (Figure 10-25):
 - a. Click **Actions** → **Case Manager Page Settings**.
 - b. Select **Step page** as the page type and click **Save** to register the page.

The screenshot shows a modal dialog box titled "Page Registration". The dialog is part of a larger interface with a header bar that includes "Work Details Form", "Custom Step Page", and a plus sign icon. The main area of the dialog has a title bar with an "X" button. Inside, there are four input fields: "Page name" containing "Custom Step Page", "Page description" (empty), and "Page type" set to "Step page". Below these is a checkbox labeled "Replace the following page with the new page:" which is unchecked. At the bottom of the dialog are two buttons: "Save" and "Cancel".

Figure 10-25 Page registration in Step page space

There are two page types: Step page and Step Launch page.

3. In the Case Manager Builder, open a task of solution, then associate the new page with one or more steps. The custom page is listed into the category that is associated with your register page type. The Step Launch page is for Launch step, and the Step page is for other steps. See Figure 10-26.



Figure 10-26 Change Step Page layout

For development environment usage, after you redeploy the solution, the custom pages are available to use. For production environment usage, you must export page in IBM Case Manager administration client first, then export the solution from the development environment. For more information, see the IBM Case Manager Information Center at:

<http://publib.boulder.ibm.com/infocenter/casemgmt/v5r1m0/index.jsp?topic=%2Fcom.ibm.casemgmt.install.doc%2Facmwd050.htm>

10.4 Example of customizing the Investigator Step

This section show a simple step by step example of how to customize the standard step page. The process uses the available settings and adds a widget to the window, in this case the Content List widget.

Take the step for the Investigator in the Investigate Product Safety Task as an example. Create a new Investigate Step page for the investigator instead of using the default step page.

The work for the investigator involves checking against the safety certification documents appropriate for the product concerned. The investigator also cross-checks whether there are any other current complaint cases that pertain to this product.

A Content List widget is added to the new page that is used to find all the documents that pertain to the safety guidelines and certification requirements for this product. A search template to find these documents using the product ID, has already been created using the WorkplaceXT Search Designer tool. Also, a Case Search widget is added to the page to find all the related cases.

The Viewer widget is removed to provide more space, and the configuration is changed so that when the investigator opens a document, it is opened in a separate window.

The original default page for the investigator is shown in Figure 10-27.

IBM Case Manager

Home | Go to Spaces | Manage Spaces | Actions ▾

p8admin |

Customer Complaints

Work Cases Work Details ▾ +

Investigate Product Safety | Investigate Product Safety

Comments

View Inst Complete | Save

Properties

- Customer Information
- Complaint Information
- Flags

Viewer

Select a document to display in this viewer.

Case Information

CC_Complaint_000000110001

Summary Documents History

Case Number: 00002

Case Source: Fax

Complaint Status: Open

Complaint Category: Billing

Complaint Received Date: Nov 20, 2012 6:00:00 AM

Customer Name: ABC

Customer Number: nnnn?

Attachment

No items to display

Figure 10-27 Original Work Page

The new page for the investigator is shown in Figure 10-28.

The screenshot shows the IBM Case Manager interface for the 'Customer Complaints' module. The top navigation bar includes links for Home, Go to Spaces, Manage Spaces, and Actions. The main title is 'Investigate Product Safety' under 'Investigate Product Safety'. A 'Comments' button is visible. On the left, there's a 'Properties' panel with sections for Customer Information (Customer Number: 34245435, Name: Ian Richards, Address: Main street, City: New York, State: New York, Email: ianrichards@someemail.com, Phone: 555-432542, Since: 2,003, Rating: Gold) and Complaint Information (Case Number:). In the center, there are two lists: 'Cases related to this customer' (sorted by Date Modified, Ascending, items 1-2, cases CC_Complaint_000000120003 and CC_Complaint_000000130001) and 'Product Reference Documents' (items 1-6, files Product Certification.pdf, Safety Check Procedures.pdf, and Product Specification.pdf, all associated with Part Number RP-123). On the right, the 'Case Information' panel shows a summary of tasks: Required (6) tasks (Send Corresponding Letter, Upsell Opportunity, Close Complaint, Investigate Product Safety, Review Product Complaint, Verify Complaint), and Optional (3) tasks (Review Non-Product Complaint). The 'History' tab is selected in the task panel.

Figure 10-28 New Work Item Page for Investigator Role

To create and customize the page, complete these steps:

1. Copy the standard work step page to a new page called Investigate Page.
See Figure 10-29.

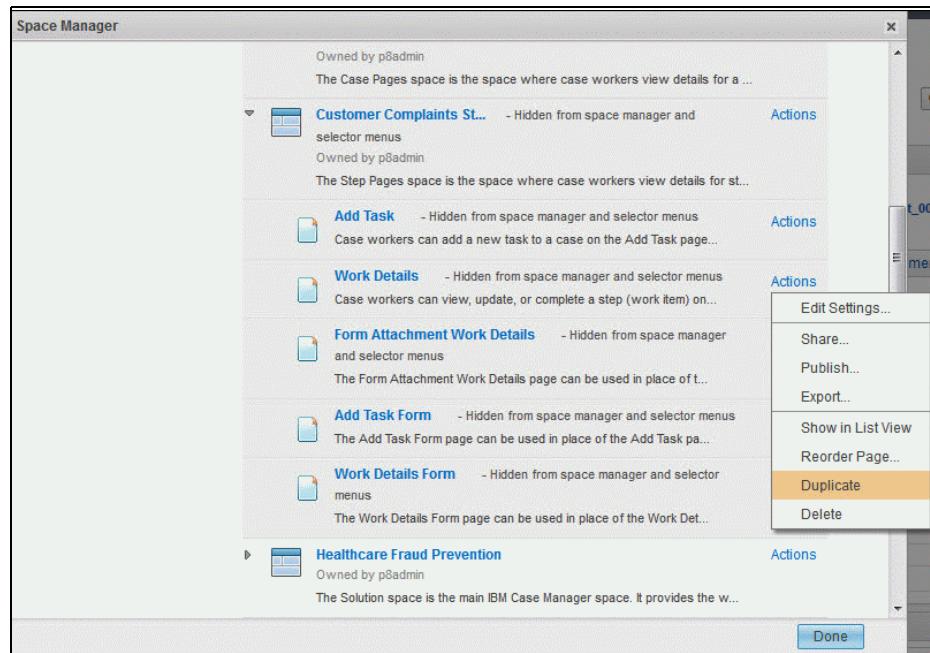


Figure 10-29 Creating the Investigate Page by duplication

2. Add the Content List widget and Case Search widget to the page as shown in Figure 10-30 on page 361.

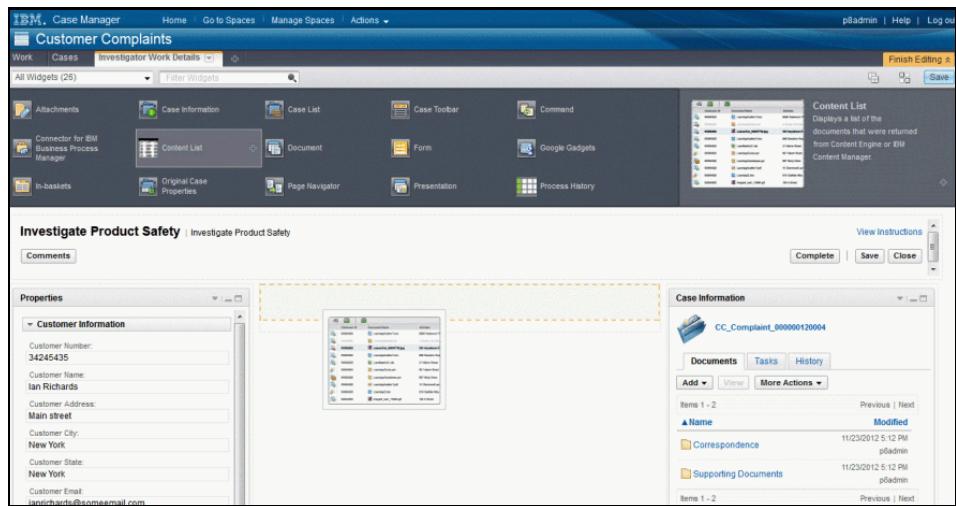


Figure 10-30 Adding widgets to the page

3. Configure the Content List widget:

- Check that there is a Search Template to find all relevant reference documents based on the Product Code. Using WorkplaceXT, test that the search is working as expected as shown in Figure 10-31.

Name	Part Number
Guide P-5246.pdf	RP-123
ISAFE Guide.pdf	RP-123
Known Hazards.pdf	RP-123
Product Certification.pdf	RP-123
Product Specification.pdf	RP-123
Safety Check Procedures.pdf	RP-123

Figure 10-31 Testing the Search Template

- b. Get the URL for the Search Template by right-clicking it, and selecting **Show Hyperlink**. See Figure 10-32.

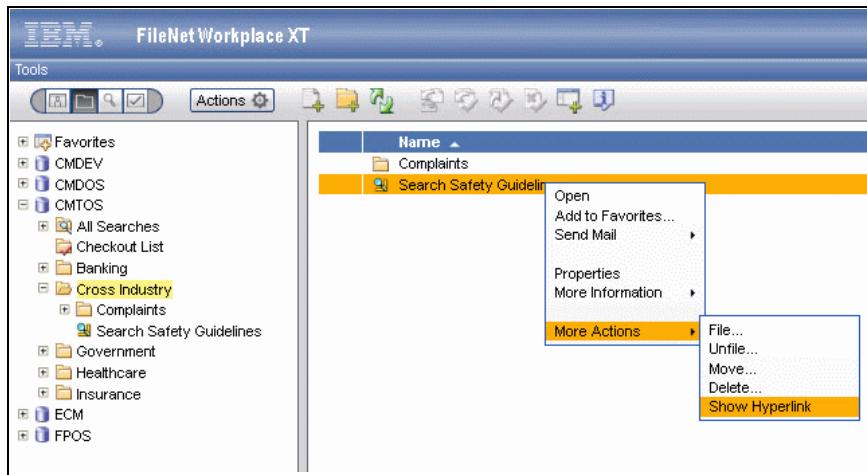


Figure 10-32 Show Hyperlink

- c. Select **Show Current**, and copy the URL to clipboard (Figure 10-33).

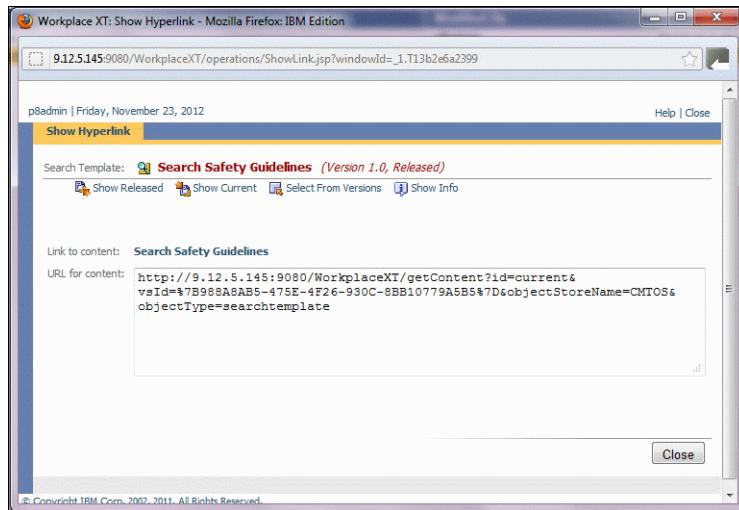


Figure 10-33 Copying the URL to the clipboard

- d. Click **Edit Settings** on the Content List widget, and enter the URL as shown in Figure 10-34.

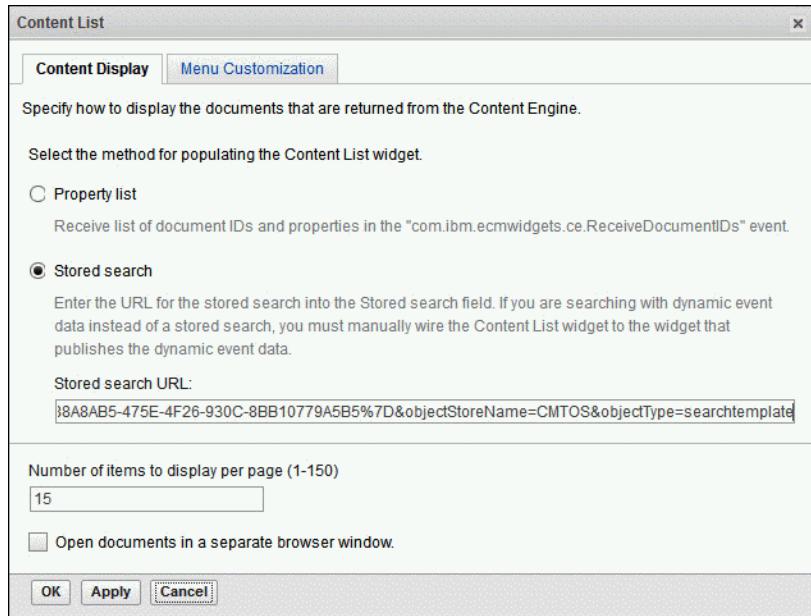


Figure 10-34 Assign Search to the Content List widget

- e. Change the name of the widget to Product Reference Documents.
f. Wire the Command widget to the Content List widget to pass the current case information to be used by the Search template (Figure 10-35).

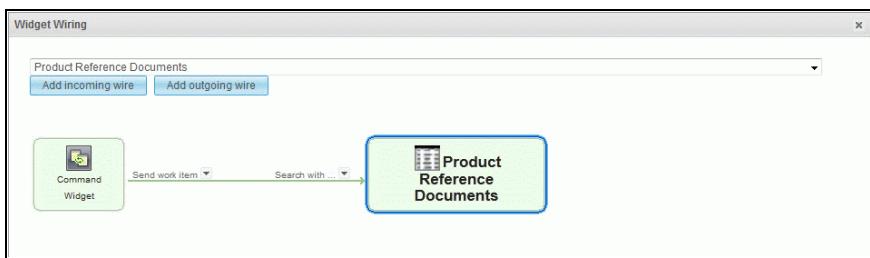


Figure 10-35 Wiring the Content List widget

4. Configure the Case Search widget by wiring it with a Java Adapter that specifies the query:
a. Add the Java Adapter and Search widgets to the page as hidden widgets. The Search widget is used as a quick way to get the payload information,

which needs to be specified for the Case Search widget. See Figure 10-36.

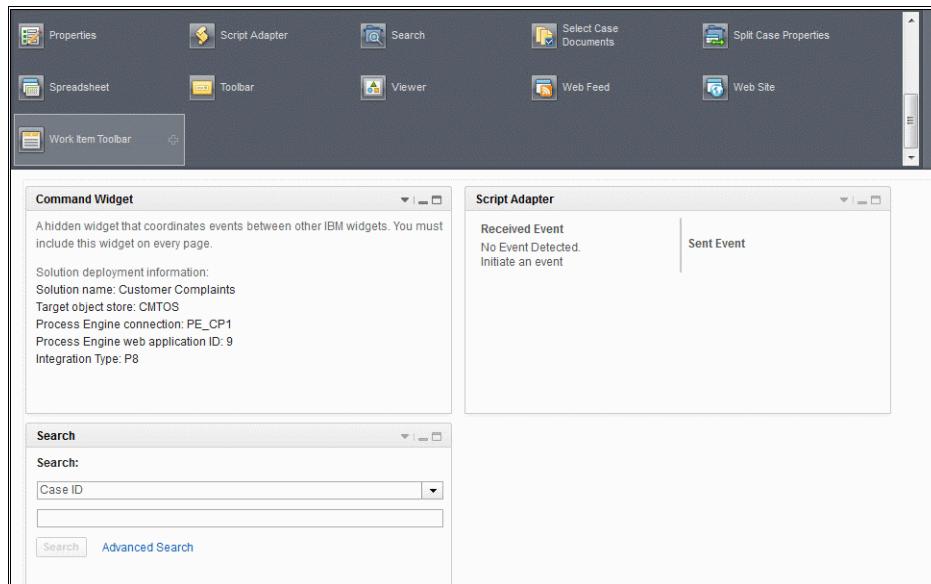


Figure 10-36 Dragging widgets to the page

- b. The search widget is only needed as a temporary device to get the payload information for the Java Adapter. Wire the Search widget to the Java Adapter as shown in Figure 10-37.



Figure 10-37 Wiring the search widget to the Java adapter widget

- c. While still in edit page mode, manually enter a search. The payload is displayed in the Java adapter. Copy the payload information to clipboard as shown in Figure 10-38.

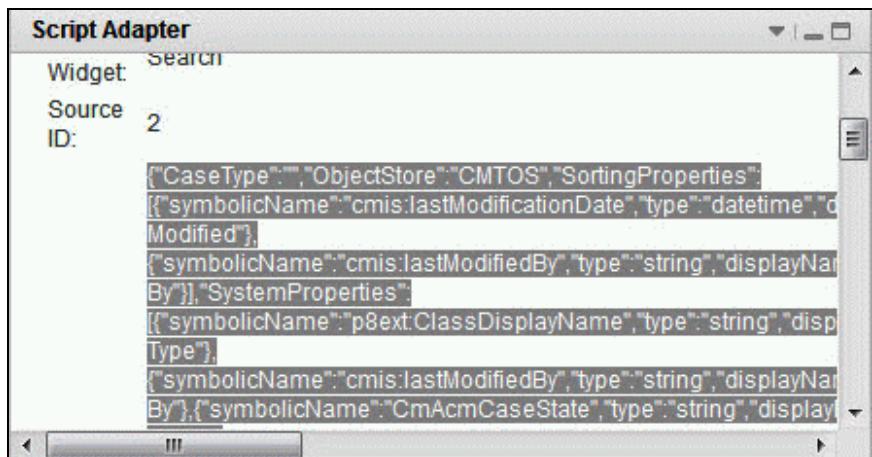


Figure 10-38 Copying the payload information

- d. Click **Edit Settings** in the Java adapter, then edit the script as shown in Example 10-4. The goal is to modify the QuerySQL at the end to find the related cases. Replace any reference to the Object Store name with the objectStore variable so that the script can work in any environment.

Example 10-4 Editing the JavaScript for Script Adapter

```

var theCustomerName = "";
var currentCaseId = "";
var prefix = "CC";
var caseTypeName = "Complaint";
var objectStore=ecmwldgt.configProvider.getTargetOS();

if (payload){
theCustomerName =
payload.caseInfo.caseInfo.caseData[prefix+"_"+"CustomerName"].val
ue;
currentCaseId = payload.caseInfo.caseId;
}
var mypayload = <paste the payload from (c) here>.....,
QuerySQL":"SELECT cmis:objectId, cmis:objectTypeId,
cmis:lastModifiedBy, CmAcMCaseState, cmis:lastModificationDate,
CmAcMCaseIdentifier, CC_CaseNumber, CC_CaseSource,
CC_ComplaintStatus, CC_ComplaintCategory,
CC_ComplaintReceivedDate, CC_CustomerName, CC_CustomerNumber,
```

```

CC_CustomerRating, CC_PartNumber, CC_UpsaleOpputruny,
CC_UpgradeCategory, CC_Address, CC_CustomerCity,
CC_CustomerState, CC_Email, CC_Telephone, CC_CustomerSince,
CC_ComplaintDescription, CC_TotalTransactionAmount, CC_Valid,
CC_SafetyCheck FROM CmAcMCaseFolder WHERE " + prefix + "_
"CustomerName = '" + theCustomerName + "'AND CmAcMCaseIdentifier
<> '" + currentCaseId + "'};

return mypayload;

```

Figure 10-39 shows the parameters of the query.

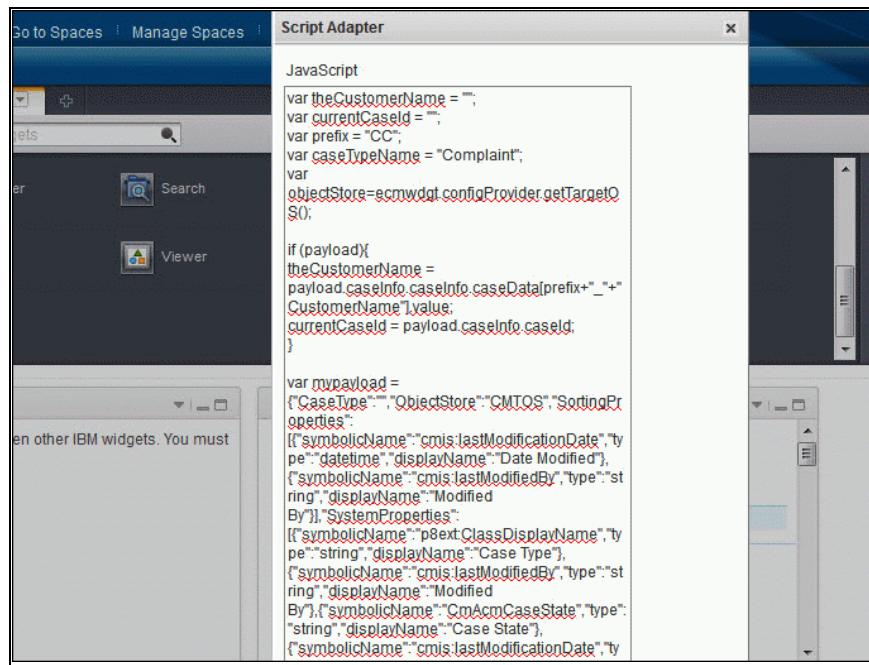


Figure 10-39 Parameters

Figure 10-40 shows the query that provides the Customer Name.

Show Script Text
 Block Outbound Event

OK Apply Restore Cancel

Figure 10-40 Query to provide the Customer Name

- e. Delete the Search widget because there is no need for it any more.
Update the wiring for the Java Adapter to link it to the Command widget and the Case List widget as shown in Figure 10-41.



Figure 10-41 Wiring the widgets

- f. The Case List widget is now configured to show all cases that have the same customer name as the current case. See Figure 10-42.

The screenshot shows a 'Case List' interface with two items listed:

- CC_Complaint_000000120003**
Case Number: | Case Source: Letter | Complaint Status: | Complaint Category: Product | Con ... More ▾
Customer Name: Ian Richards
- CC_Complaint_000000130001**
Case Number: | Case Source: | Complaint Status: | Complaint Category: | Complaint Receiv ... More ▾
Customer Name: Ian Richards

Items 1 - 2 Previous | Next

Figure 10-42 Configured Case List

5. Complete the following steps to finish customizing the page:
- Change the title of the widget to be more meaningful, by selecting the **Rename** option. See Figure 10-43.

The screenshot shows the 'Case List' interface with a 'Rename' dialog box overlaid. The dialog has the following fields:

- Widget name: Cases related to this customer
- Save and Cancel buttons

Items 1 - 2 Previous | Next

Figure 10-43 Renaming the widget

- Disable the events in the Case List widget, so as not to affect the Case Information widget when it loads. See Figure 10-44 on page 369.

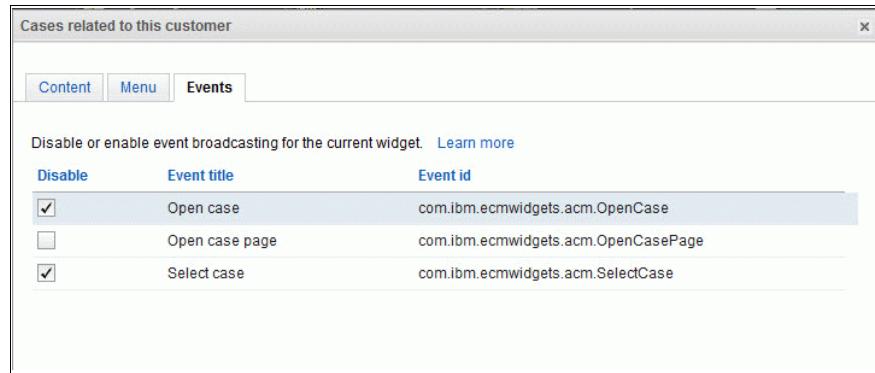


Figure 10-44 Disabling the events in the Case List widget

- c. Because the Viewer widget has been removed from the page, set the Content List widget and the Case Information widget to select the option **Open documents in separate browser window**.
6. Save the page, and click **Finish Editing**.
7. Assign the new page to the work step in the task by using Step Designer in Case Builder as shown in Figure 10-45.

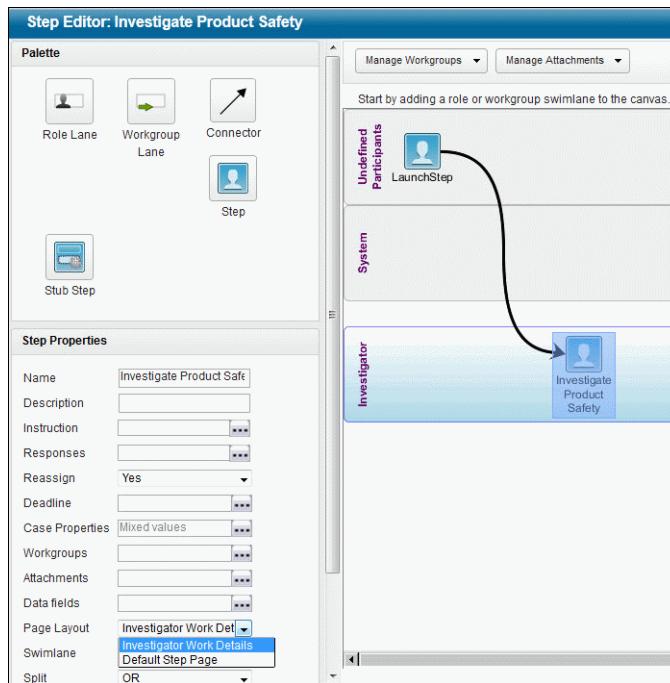


Figure 10-45 Assigning a new page to the step

10.5 Changing the runtime appearance

You can change the runtime appearance from three levels: Spaces, pages, and widgets.

10.5.1 Changing space user interface style

Business space provides several space styles. Case Management is the default style for IBM Case Manager. You can select a different style as needed.

Figure 10-46 shows where you can change the space style.

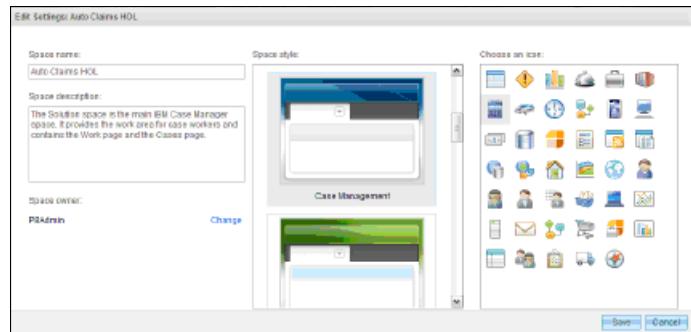


Figure 10-46 Changing space style

10.5.2 Changing page user interface style

Business space provides a series of page layouts. Figure 10-47 shows the window where you change the page style.

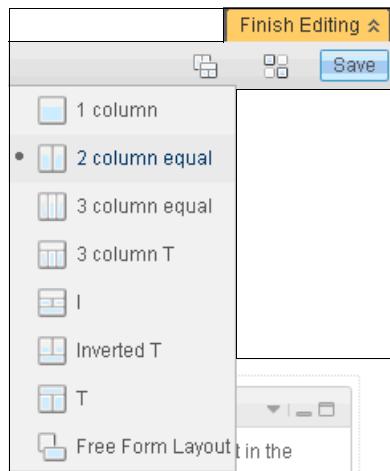


Figure 10-47 Changing page style

10.5.3 Changing widget user interface style

You can change widget skin, size, and name, or hide the widget by using the menu shown in Figure 10-48.

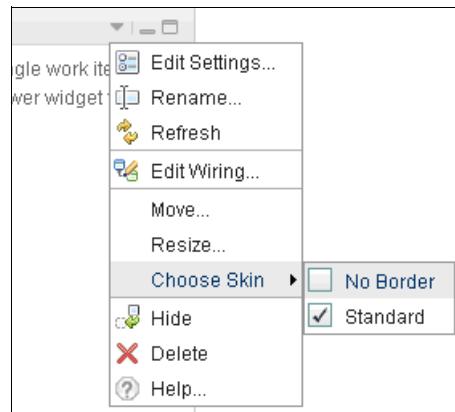


Figure 10-48 Changing widget style

You can also customize the appearance, including theme and logo. For more information, see the IBM Case Manager Information Center at:

<http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/topic/com.ibm.casemgmt.design.doc/acmwc007.htm>

10.6 User interface navigation and case accessing

You can navigate easily within IBM Case Manager spaces and pages.

10.6.1 Navigation

IBM Case Manager provides quick links to enter spaces and edit space properties. The **Go to Spaces** link changes the space you want. Use the **Manage Spaces** link to create new or edit existing spaces. To edit the current space, click **Action** → **Edit Space setting**.

Figure 10-49 shows the quick links to enter and edit the spaces.



Figure 10-49 Navigating to a space

If you want to enter a business space page, you must know the space name of the page (Figure 10-50).



Figure 10-50 Navigating to a page

After the solution is deployed in Case Manager Client, click **Test** in Case Manager Builder (Figure 10-51). The solution is opened in a new window. The Work page is automatically opened.



Figure 10-51 Testing a solution

10.6.2 Accessing cases with case link

IBM Case Manager enables users to access cases through case URLs. The URL can be sent to the user email. When the user clicks the link, IBM Case Manager automatically opens the case detail page for user handling.

In the Case List window, select **Show Link to Case** to display the link that allows direct access to the case (Figure 10-52).

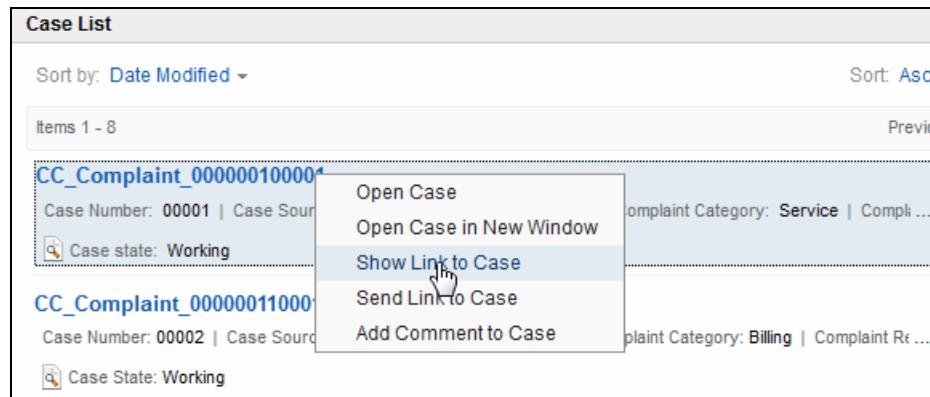


Figure 10-52 Selecting Show Link to Case

Figure 10-53 shows a sample of a case link. You can freely copy this link and use it for other purposes.



Figure 10-53 Sample link to a case

If you click **Send email** from Figure 10-53, the default mail sender is started. The subject and body of the email that includes the link to the case are automatically generated. See Figure 10-54.

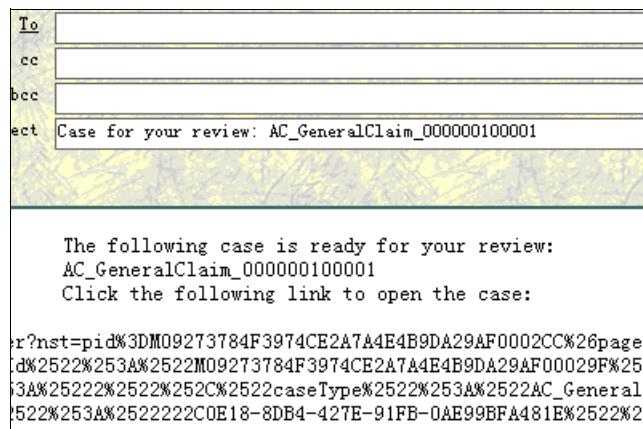


Figure 10-54 Sending an email with a link to the case

10.7 Managing documents in context of a case

When they process a case, case workers typically must manage a larger number of documents that are associated with the case. Documents are assets that contain important information required to make an informed decision or that must be kept as an evidence or for compliance reasons. In a Customer Complaints scenario, important information can be supporting documents like a contract, a billing statement, and a proof of purchasing date. It can also be correspondence like an email sent by the client or an official response letter sent out to the customer.

Therefore, a case management solution must support two main use cases related to managing documents:

- ▶ Allow case workers to easily access and view any documents that are related to the current case or task
- ▶ Provide a convenient way to add new documents to the case

IBM Case Manager release 5.1.1 significantly improves the ease of managing case documents with the new IBM Content Navigator viewer and by supporting drag and drop to add new documents to a case.

10.7.1 Integration of the IBM Content Navigator viewer

IBM Case Manager supports managing case documents in either the IBM FileNet Content Manager or the IBM Content Manager repository. The integration of the IBM Content Navigator viewer unifies the viewing experience across both repositories, and adds several benefits when viewing case documents. It also demonstrates how features provided by the IBM Content Navigator Client can be used inside an IBM Case Manager solution. It is the first step in removing the dependency of IBM Case Manager from Workplace XT.

After the IBM Content Navigator viewer integration is configured for an IBM Case Manager environment, the viewer is embedded into the Case Viewer widget. For more information about the configuration, see “Configuring the Case Manager Client Viewer” on page 377.

If the Viewer widget is embedded on the current page, it displays documents that are opened from the following widgets:

- ▶ Case Documents
- ▶ Task Attachment
- ▶ Content List

The documents are displayed unless these widgets are configured to open the document in a separate page. The IBM Content Navigator Viewer also opens in a separate page if the Viewer widget is not present on the current page, or the document is opened using the **Show Link URL**.

New features of the IBM Content Navigator viewer

Using the IBM Content Navigator viewer in IBM Case Manager delivers the following benefits:

- ▶ Rich viewer experience for documents from both IBM FileNet and IBM Content Manager
- ▶ View and annotate documents of various formats that include TIFF, JPEG, PDF, Microsoft Office, Microsoft Outlook, and IBM Lotus Notes®
- ▶ Display multiple documents at the same time in the viewer (Figure 10-55)

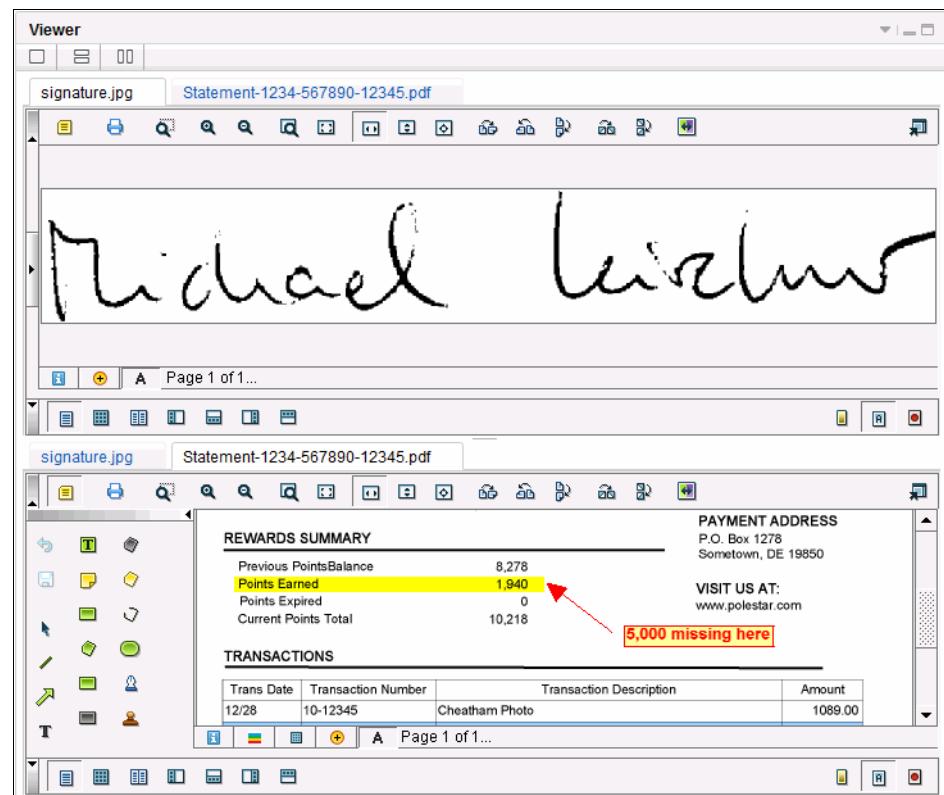


Figure 10-55 IBM Content Navigator Viewer with two documents opened

When multiple documents are opened in the Viewer widget at the same time, you can arrange documents in separate tabs, or in a split view vertically or

horizontally. You can also use this feature to display two pages side by side, for example a customer order and a specimen signature for validation purposes. In Figure 10-55 on page 376, the upper pane displays an image document, whereas the lower pane shows a PDF document with annotations applied. Annotations can be easily added to all document formats supported by the Viewer widget. The Viewer widget provisions several types of annotations that include highlighting, text boxes, and several graphics like arrows, boxes or stamps.

IBM Content Navigator viewer enables case workers to view many different types of case documents in a single viewer application. This benefits by using the same navigation controls and annotation capabilities across a variety of MIME types as opposed to using individual plug-ins for different document types.

Tip: The IBM Content Navigator viewer supports different MIME types and formats, depending on the repository used. See the IBM Case Manager Information Center for details about what formats are supported.

Configuring the Case Manager Client Viewer

With the current release of IBM Case Manager, integrating the IBM Content Navigator viewer is an optional step that is configured by using the Case Manager Administration Console. You must add the task **Configuring the Case Manager Client Viewer** to your profile and run it during the configuration process as shown in Figure 10-56.



Figure 10-56 Configuring the Case Manager Viewer with CMAC

You must provide the following information to complete the configuration process:

- ▶ Deployment type, application server node, and server name
Reuse the configuration parameters that you specified for the **Deploy Case Manager Client** task.
- ▶ Content Engine EJB URL
Reuse the configuration parameter that you specified for the **Create the Case Manager API WAR File** task.
- ▶ IBM Content Navigator URL
The URL to the IBM Content Navigator installation. It usually has the format `http(s)://servername:port/navigator`.
- ▶ IBM Content Navigator administrator user name and password
The user name and password for a user with administrative privileges on the IBM Content Navigator installation. This user is used to modify the IBM Content Navigator configuration for access by IBM Case Manager.
- ▶ Add repositories in IBM Content Navigator
If enabled, the design and target object stores defined in the IBM Case Manager deployment are added to the IBM Content Navigator desktop during the configuration process.

As a prerequisite, you must install and configure a separate Content Navigator server for each Case Manager server that is used in the Content Navigator viewer. You cannot share the Content Navigator instance among different Case Manager servers.

When you configure the URL to access Content Navigator, use fully qualified host names, domain names, and the same protocol (http or https) as for Case Manager. Doing so avoids problems in the communication between both components. If Content Navigator and Case Manager are not deployed into the same WebSphere Application Server profile, ensure that both profiles are set up to share LTPA tokens for authentication.

Important: When used with IBM Case Manager, IBM Content Navigator must be deployed into a WebSphere Application Server.

Configuring the IBM Content Navigator desktop

IBM Content Navigator uses the concept of *Desktops* to control how information managed in the Enterprise Content Management (ECM) repository is presented to the users. An IBM Content Navigator administrator uses the IBM Content Navigator Administration desktop to modify the configuration of the desktops that are defined for the users.

The desktop configuration includes, for example:

- ▶ The ECM repositories that can be accessed
- ▶ The Viewer Map
- ▶ The menus that are available

When you run the **Configuring the Case Manager Client Viewer** task, a new desktop is added to the IBM Content Navigator configuration database. The name of this desktop is **ICM** and it is used whenever IBM Case Manager calls the IBM Content Navigator viewer. This desktop uses the Default viewer map that is created as part of the standard IBM Content Navigator installation. This viewer map is configured to use the IBM Content Navigator Viewer only to view image MIME types (TIFF, JPEG, and BMP) and formats like COLD. For other MIME types like PDF, Open Office, or MS Office, it calls local applications on the client. As a result, these documents open in a separate application window as opposed to being displayed in the IBM Case Manager Viewer widget. Also, you are not able to add annotations to these documents when they open in their native applications.

To overcome this, you can create a Viewer Map to be used with the ICM desktop as described in the following steps.

Creating a Viewer Map in IBM Content Navigator

To create a Viewer Map in IBM Content Navigator, complete these steps:

1. Log on to IBM Content Navigator as an administrative user and open the administration desktop. This is usually achieved by opening the following URL:

`http(s)://servername:port/navigator/?desktop=admin`

2. From the list at the left, select **Viewer Maps**. Doing so displays the Viewer Maps configured for IBM Content Navigator.

3. Select **Default Viewer Map** and select **Copy**. Provide a name for the new Viewer Map such as ICM Viewer Map. The application automatically extracts the ID from the name (in this example, the ID is ICMViewerMap). Optionally, you can add a description for the new Viewer Map as shown in Figure 10-57.

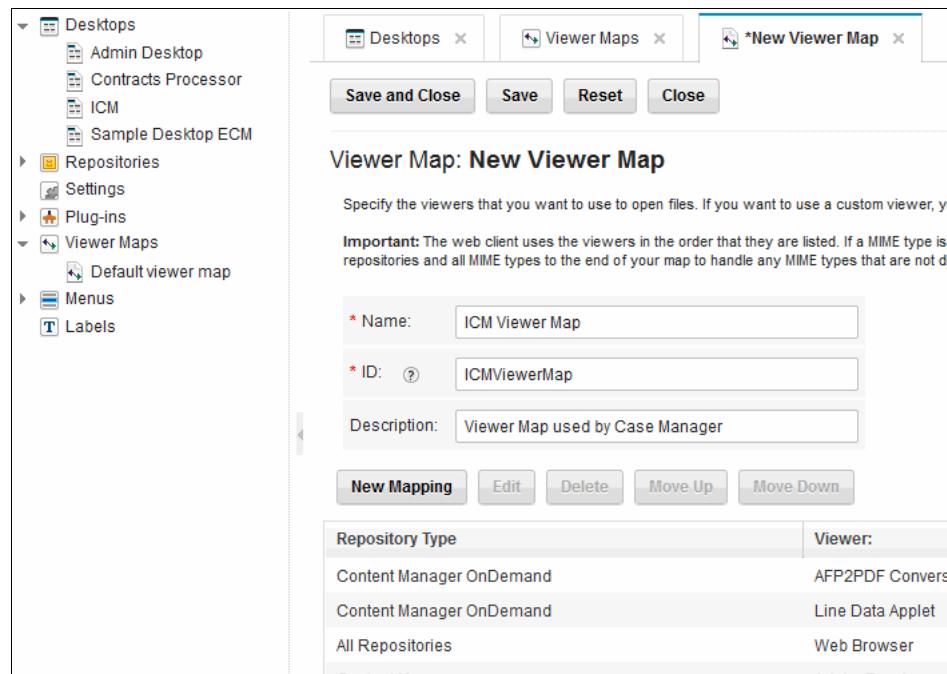


Figure 10-57 Creating a Viewer Map in IBM Content Navigator

4. In the list of repository types and viewers, select the entry where **Repository Type** is **FileNet Content Manager** and **Viewer** is **FileNet Viewer**.

5. Select **Edit** to display the mapping window that is shown in Figure 10-58. You can use this window to specify which viewer is used to for which MIME type. Ensure that **FileNet Viewer** is selected for the **Viewer** entry in the upper part of the dialog.

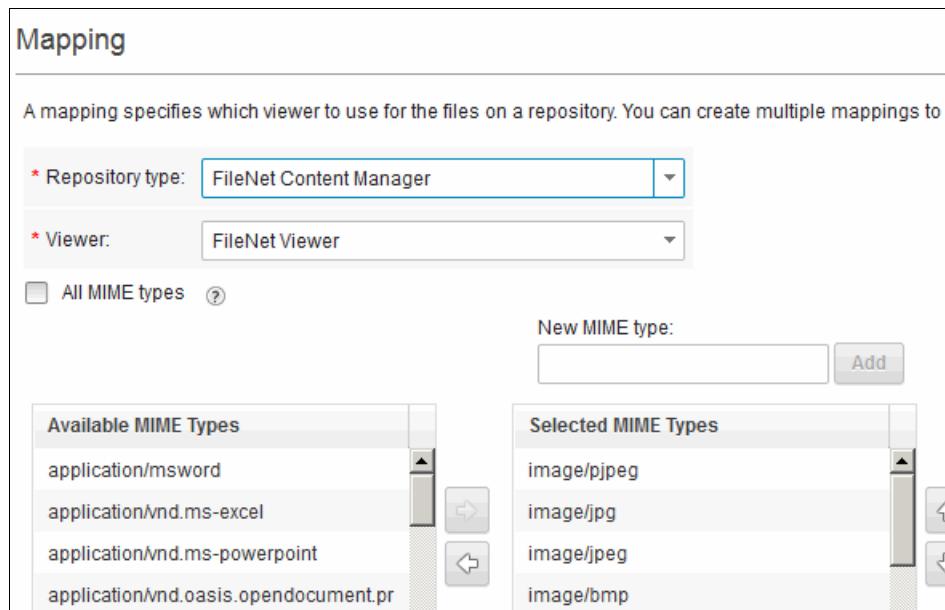


Figure 10-58 Setting up the MIME types for the Viewer Map

6. Select all of the MIME types you want to display in the FileNet Viewer by selecting them from the **Available MIME Types** list. Use the right arrow to move them to the **Selected MIME Types** list. You can use the **SHIFT** and **Ctrl** keys to multi-select entries for moving them. Click **OK** to close the window when you have assigned all MIME types that you want to display in the Viewer widget.
7. If you also want to display plain text in the FileNet Viewer, select the entry that maps All Repositories to Web Browser for the MIME Type `text/plain`. Move it beyond the one you edited in the previous steps by clicking **Move Down**. You can set up different Viewers for the same MIME type in IBM Content Navigator. They are used based on the order in the mapping list.
8. Click **Save and Close** after you are finished editing the Viewer Map.

Configuring the ICM desktop to use the new Viewer Map

Complete these steps to configure the ICM desktop to use the new map:

9. Open the list of configured desktops by either using the tab **Desktops** at the top the window or selecting **Desktops** from the list on the left side.
10. Pick the desktop named **ICM** and select **Edit**, which displays the window that is shown in Figure 10-59. Change the **Viewer map** setting from **Default viewer map** to **ICM Viewer Map** and end then click **Save and Close**.
11. Log off from IBM Content Navigator.

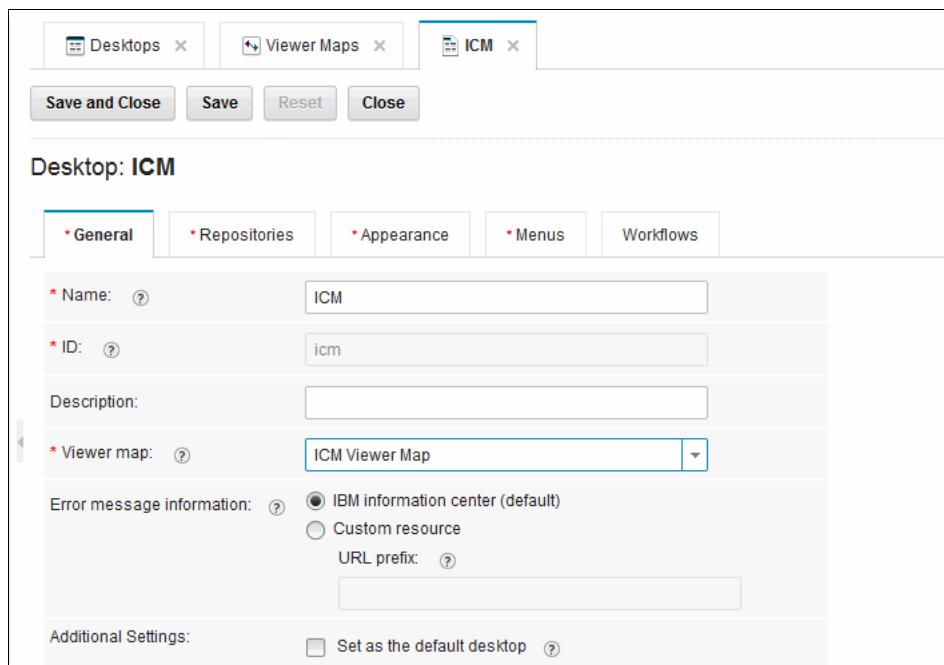


Figure 10-59 Configuring the ICM desktop to use the new Viewer Map

To validate the new configuration, log on to the Case Manager client and either search for a case or open a task. If you do not have a PDF or MS Office document stored as a case or task document, add it to the case or task and then open this document. It should display in the Viewer widget, and you should be able to add annotations to the document.

Tip: These steps refer to a configuration that uses a FileNet Content Manager repository. The process is identical for a Content Manager repository except that you choose a different repository type in step 4.

10.7.2 Manually adding documents

IBM Case Manager release 5.1.1 introduces a number of new features that benefit case workers when they work with documents.

Support for dragging

IBM Case Manager significantly simplifies manual adding documents to a case or a task by supporting dragging. Case workers can add a document by just dragging it from the file system or Lotus Notes and dropping it into the Case Management solution. IBM Case Manager supports dragging for both the document tab of the Case Information widget and the Attachment widget associated to a task. When files are dragged onto one of these widgets, a *drop zone* highlights as shown in Figure 10-60 to notify the user that the file can be released.

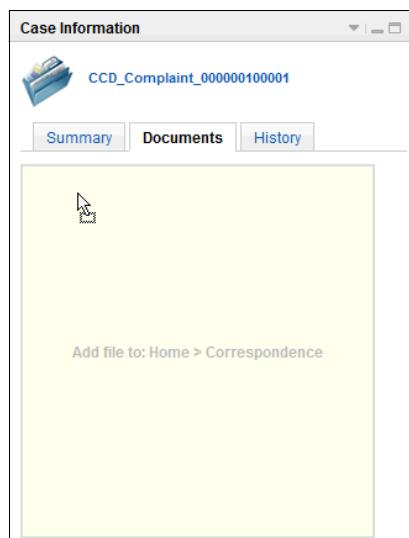


Figure 10-60 Active drop zone when manually adding a document

As described, redesigned dialogs are used to select the document type and provide document metadata for any document that is added manually, including dragging.

Restriction: The current release of IBM Case Manager supports dragging only for single files. Multiple file dragging is not supported.

New design of document-related dialogs

IBM Case Manager 5.1.1 uses a new design for windows that are related to managing documents as opposed to using IBM FileNet Workplace XT windows for this purpose. The new dialogs are available for the following actions:

- ▶ Adding a document to a case or task (including dragging)
- ▶ Checking in a document
- ▶ Modifying properties of a document

When you add a document, IBM Case Manager displays the window shown in Figure 10-61 which allows you to select the document type and provides document properties. The document title property is pre-filled with the name of the file, which you selected from the file system or dropped into the Case Manager Client.

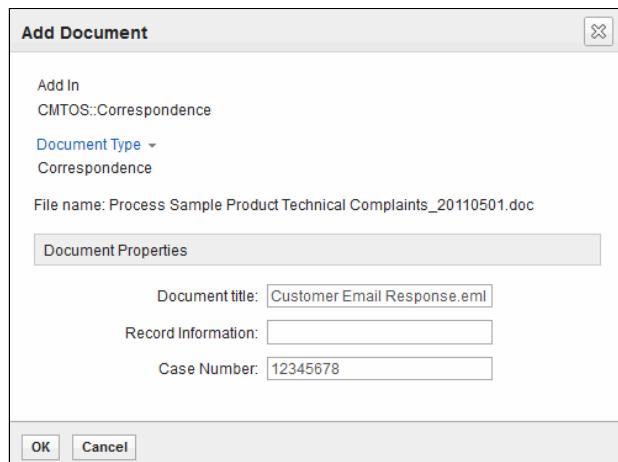


Figure 10-61 Providing document metadata when you add a document

The new dialogs are more consistent with the overall user interface design of the Case Manager Client. They also allow users to edit document properties when they are checking in a document. Additionally, there is a new configuration available as shown in Figure 10-62. With this configuration, you can restrict the selection of the document classes to the ones defined in the solution. This is as opposed to all document classes that are defined for the target object store that is used by the solution.

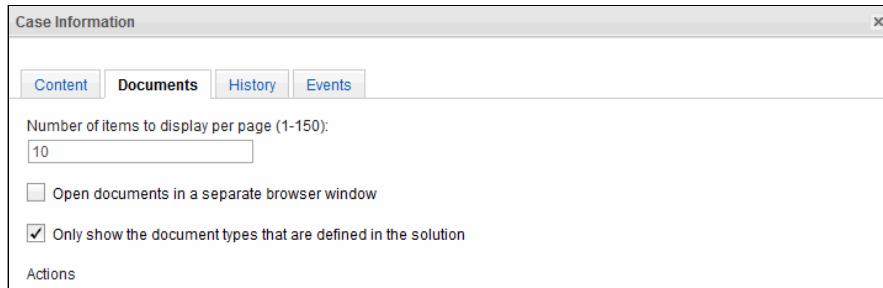


Figure 10-62 Restricting document types for manually adding documents

A document class only shows up in the selection dialog if the user currently logged on has at least viewing rights for the document class definition.

Support for IBM FileNet Workplace XT dialog

Generally, use the new dialogs that are described in the previous section for any new IBM Case Manager solution. However, there is one exception because the new dialog does not support two features that were available with the IBM FileNet Workplace XT dialogs when adding documents:

- ▶ You cannot take advantage of Entry Templates configured for a folder in IBM FileNet Workplace XT. This capability can be useful when you have many properties that are assigned to a case document type and you want to hide some of them when a user manually adds a document
- ▶ You cannot create documents of type “External File”. This feature allowed you to refer to a document based on a URL or UNC path instead of providing the actual content file. Generally, do not use external files in a case management scenario because you do not have control of the location where the document content is stored. The document might be deleted even though it is managed as a case asset.

If you must use one of these features, configure IBM Case Manager so that it uses the IBM FileNet Workplace XT dialog for adding new documents. This configuration can be done in two ways:

1. Using the CMAC when you configure the Case Manager installation.

In the Configure Bootstrap Settings task, enable the setting **Enable external files as documents** as shown in Figure 10-63.

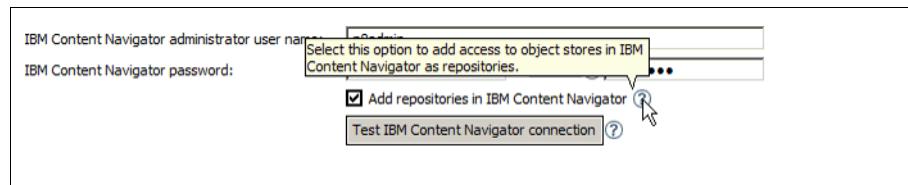


Figure 10-63 Configuring the Workplace XT dialog in CMAC

2. Updating the installation by using the WebSphere Application Server Administration Console

In the WebSphere Application Server Administration Console, change the custom property `com.ibm.acm.client.XTDialogFlag` for the Mashup Endpoint to the value true. This setting is configured by opening **Resource environment providers** → **Mashups_Endpoints** → **Custom properties** → **com.ibm.acm.client.XTDialogFlag** as shown in Figure 10-64.

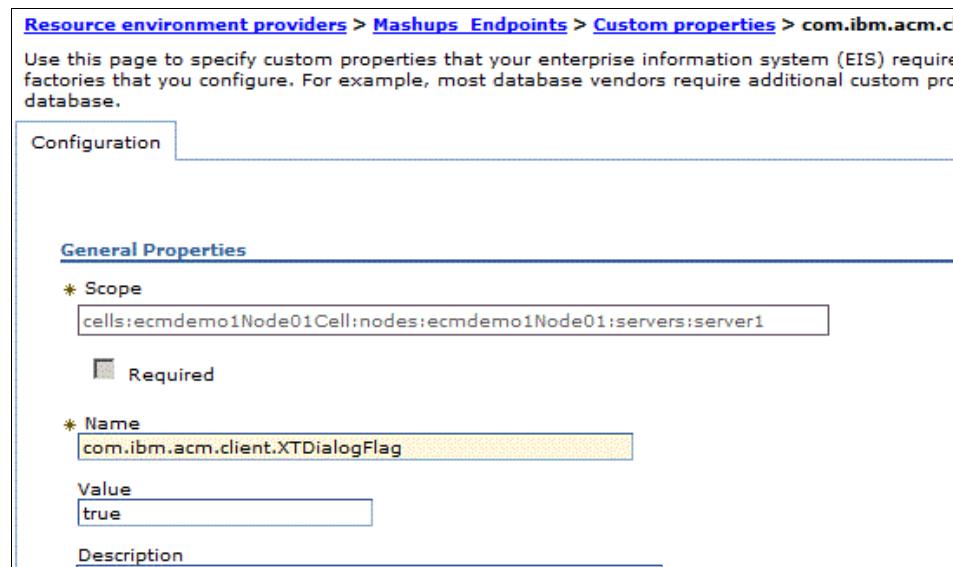


Figure 10-64 Configuring the Workplace XT dialog

Consideration: The IBM FileNet Workplace XT dialog configuration applies only for adding a document. All other document-related windows continue to use the new layout even when the dialog is enabled.

10.7.3 Adding case documents to a task

IBM Case Manager release 5.1.1 adds an addAttachment event to the Attachment widget. This event is used to add one or more documents that are managed in the ECM repository as attachments to a selected task instance. By multi-selecting documents from the Case Information Widget, you can easily configure the Case Client to enable users to select one or more case documents and add them as attachments to a task. The example assumes that your solution uses the Case Information widget with the Documents tab enabled and the Attachment widget on the step page.

Tip: In the current release, you can only add documents to a task that are managed in the Target Object Store for the solution.

The first step is to create a custom action in the Document tab of the Case Information Widget. Open the Edit Settings window of the Case Information widget and Click **Add** to display the window shown in Figure 10-65.

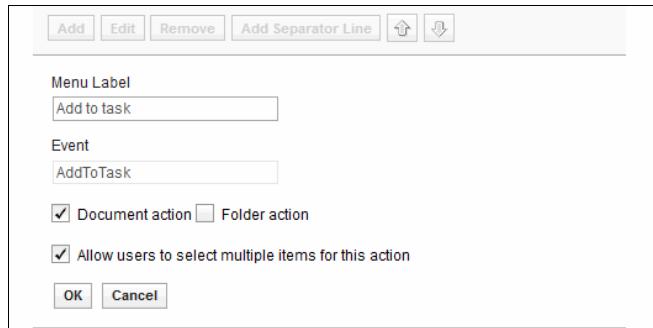


Figure 10-65 Adding a custom event for the Document tab (Case Information widget)

Provide the following information when you adding an action:

- ▶ Menu Label

The name that shows up in the menu and the action menu for the new action you are creating. The example uses “Add to task.”

- ▶ Event
Name of the event as used for wiring the widgets.
- ▶ Documents action and folder action
Determines whether this action can be applied to documents and folders.
- ▶ Allow Users to select multiple items for this action
Defines whether this action supports the selection of multiple items. If enabled, this action also shows up when more than one document or folder is selected. Otherwise, the action is grayed out when the user selects multiple items.

Then use the Script Adapter widget to match the payloads of the Case Information widget with the payload expected by the Attachment widget addDocument event. The IBM Case Manager Information explains in detail what the Document Payload structure looks like, and also shows example code for the Script Adapter to add the information that is required by the addDocument event. For more information, see Document payload type at:

<http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/topic/com.ibm.casemgmt.design.doc/acmwc009.htm>

Using the sample code that is provided in the Information Center, set the actualPayload.attachmentName property to the attachment field you use in the task workflow. In the example Customer Complaints solution, you want to store the selected case documents in the SupportingDocuments attachment array. Therefore, change the Script Adapter code as shown in Example 10-5.

Example 10-5 Sample code to add case documents to a task

```
var actualPayload = {};
var docs = [];
actualPayload.attachmentName = "SupportingDocuments";
actualPayload.overwrite = false;
actualPayload.displayConfirmation = true;
actualPayload.documents = payload.systemProperties;
return actualPayload;
```

Wire the inbound connector of the Script Adapter widget with the new action event you created and the outbound connector with the addAttachment event of the Attachment widget.

To test the changes, select a case of the Customer Complaints solution and add documents to the case if they do not already exist. Then, create a task that uses the SupportingDocuments attachment field, for example the Investigate

Employee task, and open the task from the in-basket. Select one or more case documents and run the new action **Add to task** as shown in Figure 10-66. Alternatively, you can also use the **Action** menu and run the **Add to task** operation from there.

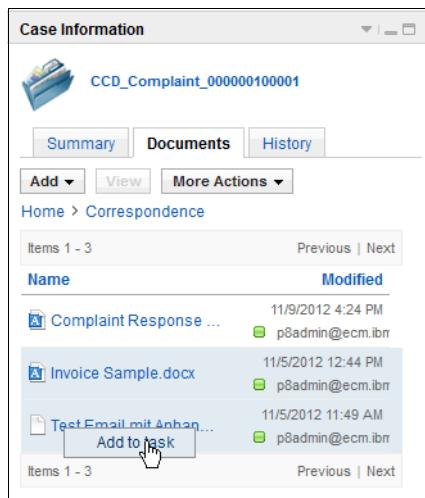


Figure 10-66 Adding two case documents to the currently opened task

As a result, the selected documents are added to the attachment field of the task you opened. Notice that the Attachment widget automatically updates the information and shows the newly attached documents in the Case Manager client.

10.8 Using the Case Form widget

When creating Customer Complaints solution, you might use the Case Form widget in place of the Case Data widget. For more information, see 2.2, “Complaints management example use case” on page 25. Use the Case Form widget if plain representation of case data is not sufficient or when case data must be supplemented with form data or form functionality.

Figure 10-67 demonstrates how a new Customer Complaints case can be rendered with help of Case Form widget.

The screenshot shows a software interface titled "Customer Complaints V2". At the top, there are tabs for "Work", "Cases", and "Add Case Form". The "Add Case Form" tab is currently selected. Below the tabs, the word "Complaint" is displayed. Underneath, a "Case Form" section is visible, containing a "Customer Complaint Form". The form has two main sections: "Customer Details" and "Complaint Details".

Customer Complaint Form		Case Number: 50730
Customer Details		
Complaint Date:	14-Dec-11	
Customer Number:	12345	
Customer Name:	John Smith	
Contact Phone Number :	444 123 4567	
Address:	1234 E 15th St Long Beach, CA	
Email:		
Complaint Details		
Category:	Billing	
Amount:	\$100.00	
Description:		

Figure 10-67 Customer Complaint form

Case Form widget can host two form products: IBM Forms and FileNet eForms. The widget handles them transparently for users.

When the widget is configured to open a form by using a form template, the Version ID or Version Series ID determines the specific product. See Figure 10-77 on page 400 for an example of configuration.

When the widget is configured to open a form when an attachment is presented, the platform started is determined by the MIME type of the attached document.

The two forms products can coexist in the same solution. You can render case pages with the help of IBM Forms while you render the step pages by using FileNet eForms. However, a page can contain only one Case Form widget.

10.8.1 View and edit modes

IBM Case Manager does not provide a case locking mechanism. This limitation might cause unexpected overwrite of case data in a multi-user environment. However, the Case Form widget provides a mechanism to lower the probability of such an event.

When a new Customer Complaints case is added, the form can be rendered in the Case Details Form page initially in a view (read-only) mode (Figure 10-68).

The screenshot shows a 'Case Form' window with a 'Customer Complaint Form' panel. The 'Customer Details' section is highlighted with a green background. The 'Case Number' field contains the value '97869'. The form fields are as follows:

Complaint Date:	14-Dec-11
Customer Number:	12345
Customer Name:	John Smith
Contact Phone Number :	444 123 4567

Figure 10-68 Customer Complaint form in view (read-only) mode

When a user decides to edit the case properties, the widget is switched to the edit (read and write) mode to allow this operation. See Figure 10-69.

The screenshot shows the same 'Case Form' window and 'Customer Complaint Form' panel as Figure 10-68, but the 'Customer Details' section is no longer highlighted. The 'Case Number' field still contains '97869'. The form fields are identical to Figure 10-68.

Figure 10-69 Customer Complaint form in edit mode

The action of switching to edit mode triggers refresh of case data. Changes that are made elsewhere become part of the data model of Case Form widget. The user can then review and edit the case properties, and save the changes.

When the Case Form widget is left in the edit mode long enough for other users to modify the case data, two outcomes are possible:

- ▶ If the user edits case properties that have not been modified by other users, the properties that are modified and saved by others remain intact. The save operation submits only the set of modified properties.
- ▶ If the user edits case properties that have been modified by other concurrent users, the latter set of properties is overwritten. The most recent changes have priority.

After the case is saved, the Case Form widget is rendered in view mode again, and user is presented with persisted case properties.

10.8.2 Overview of data merging rules

The Case Form widget maps template fields with case properties and step parameters by using field name and property or step parameter ID. When you design a template, follow this pattern:

- ▶ Assign a prefix to your solution. For example, for the Customer Complaints solution, it can be CC.
- ▶ When a new property is added, Case Manager Builder constructs the internal name as “prefix + underscore character + the property name you type in”. For example, you create a property name, CustomerName, for the Customer Complaint solution. Internally, the Case Property is CC_CustomerName. In addition, space characters are automatically removed from the resulting property ID.
- ▶ Change the template fields to match the names of Case properties. For example, change the template field to CC_CustomerName.

Although the primary purpose of a Case Form widget is to present case properties to users for viewing and editing, things usually are more complicated.

Consider part of the Customer Complaints case. The Case Form widget is on the Work Details Form page, and is passed two sets of data. One set of data originates from IBM FileNet Content Engine and the other set from Process Engine. These two data sets are received in no particular order. Each of sets is constrained by its own, disparate data schema.

The Content Engine data can be modified by external data system. This process is not apparent for Case Form widget.

In more complex cases, the Case Form widget also must deal with form data, which is not stored in the Content Engine repository.

The widget follows these steps to merge the data sets:

1. If the Case Form widget renders a form from the attached data document, the form can contain fields that are not mapped with case properties. The form is first filled with data from that data document.
If the Case Form widget renders a form from a template, the template can contain fields with default data. The form is first filled with that default data.
2. The Case Form widget merges case and step data in such a way that properties value attributes are set according to the incoming case data. This

approach is taken so that step data does not overwrite modification of case data by external data system. Another reason is that the Case Form widget presents user with a case-centric view. Tasks are treated as a means of case data modification, not as case data.

3. The result of merged case and step data is then injected into the form that was previously populated with form data or defaults.

When Case Form widget is placed on Case Details page, the process is much more simple. The widget injects case data into form that were previously populated with template default values.

The calculations that are defined in form template are applied at the end.

One merging rule worth mentioning is the enforcement of required property attribute. As mentioned earlier, Content Engine and Process Engine use different data schema. One notable difference is there is not a null constraint on all items of step data. When Case Form widget encounters a numeric mapped case property, it makes the required attribute on the resulting internal data model item true. It does so regardless of the setting that is applied by Case Manager Builder at design time.

10.8.3 Configuring Case Form widget-related pages

Case Form widget-related pages must be configured. They are not automatically registered as case or step related types by Case Manager Client. In addition, they are not set as default layout pages by Case Manager Builder.

To start rendering your case on the Case Form widget page, complete these steps:

1. Assuming that the solution is already created and deployed, create the Case Form related pages by copying the default Case Form pages. Default Case Form pages are created by Case Manager Builder. The copy operation is completed by using Case Manager Client's **Add Page** operation.
2. Customize new case pages as needed.

The previous steps are optional if custom layouts are needed to present user interface for different roles or different tasks. In simple scenarios, you can use the default Case Form pages.

3. Register the case pages as case related, and step pages as step related:
 - a. Navigate to a page through Manage Spaces, then click **Actions** → **Case Manager Page Setup** (Figure 10-70).

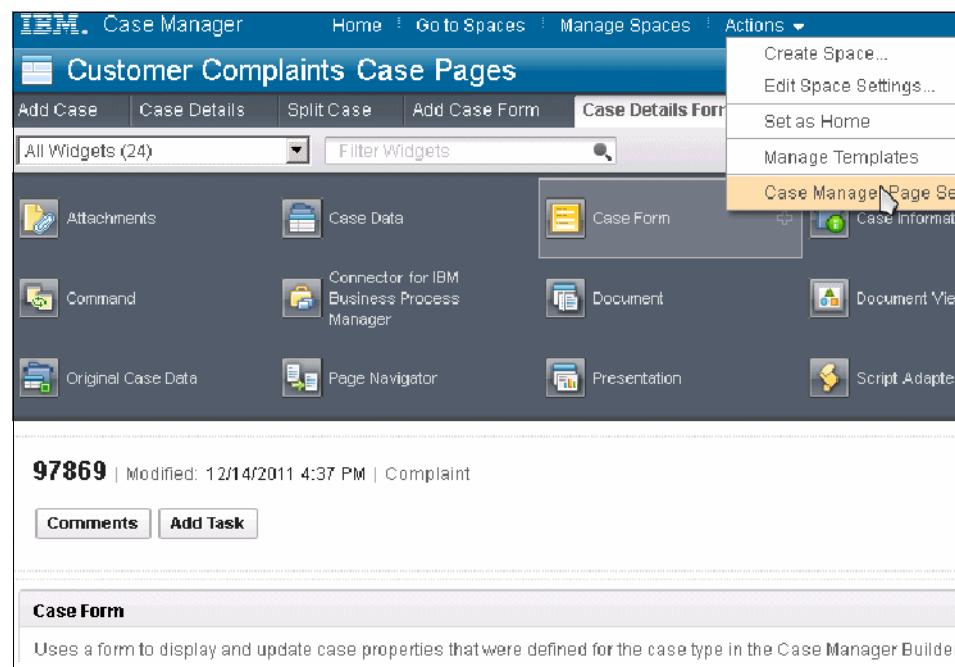


Figure 10-70 Selecting Case Manager Page Setup from the Actions menu

- b. Register all pages as corresponding page types. For example, make the Case Details Form page a page of type Case page as shown in Figure 10-71.

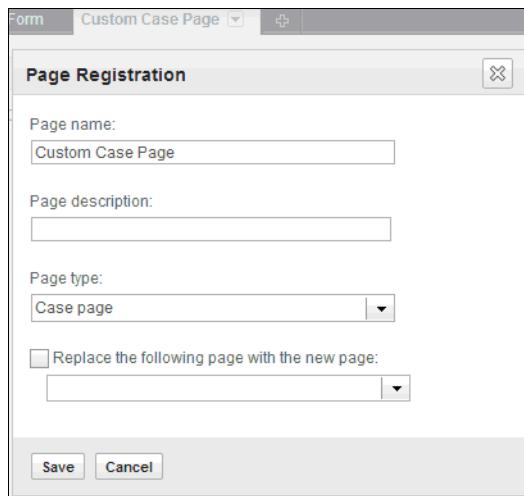


Figure 10-71 Page registration

4. Return to Case Manager Builder and associate pages with the default layouts. Associate default case layouts with the Add Case Form (Figure 10-72) and Case Details Form pages.

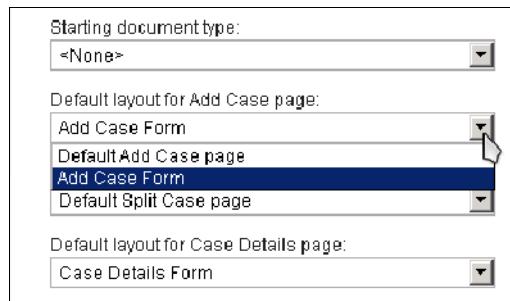


Figure 10-72 Associating the default case layout for Add Case pages

5. Open Step Editor and associate the default step layouts. Use the **Add Task Form** page for the launch step (Figure 10-73) and the **Form Attachment Work Details** page for the rest of steps.

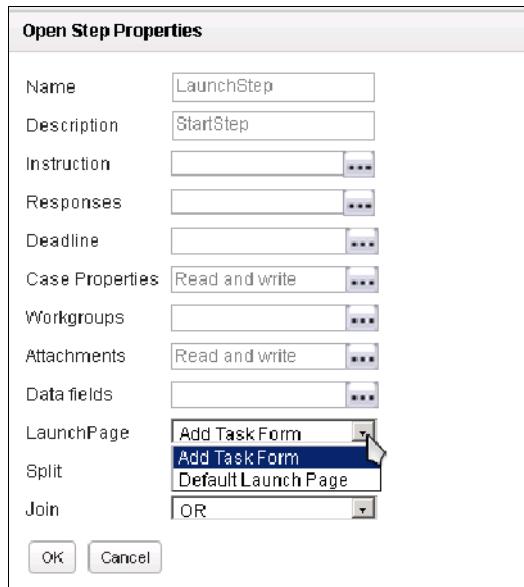


Figure 10-73 Setting the launch step to use the Add Task Form page

10.8.4 Configuring Case Form widget

Case Form widget configuration process is described in detail at:

<http://publib.boulder.ibm.com/infocenter/casemgmt/v5r1m0/index.jsp>

Click **Designing case management applications** → **Modifying the default case management client application** → **Widgets reference** → **Case Form widget**.

To use a form template, click **Configuring the Case Form widget to use a form template**.

To use a form attachment, click **Configuring the Case Form widget to use a form attachment**.

Remember: The form data can be saved only when Case Form widget is configured to open the form as an attachment.

Form data must be saved as an attachment when a form template has fields that cannot be mapped with case properties. This type of configuration is applicable only with step scenarios because attachments are not properties of a case. Instead, they are task-related objects.

When you design a solution, use the Step Editor in Case Manager Builder to specify a name for the attachment. Figure 10-74 shows the window to add attachments.

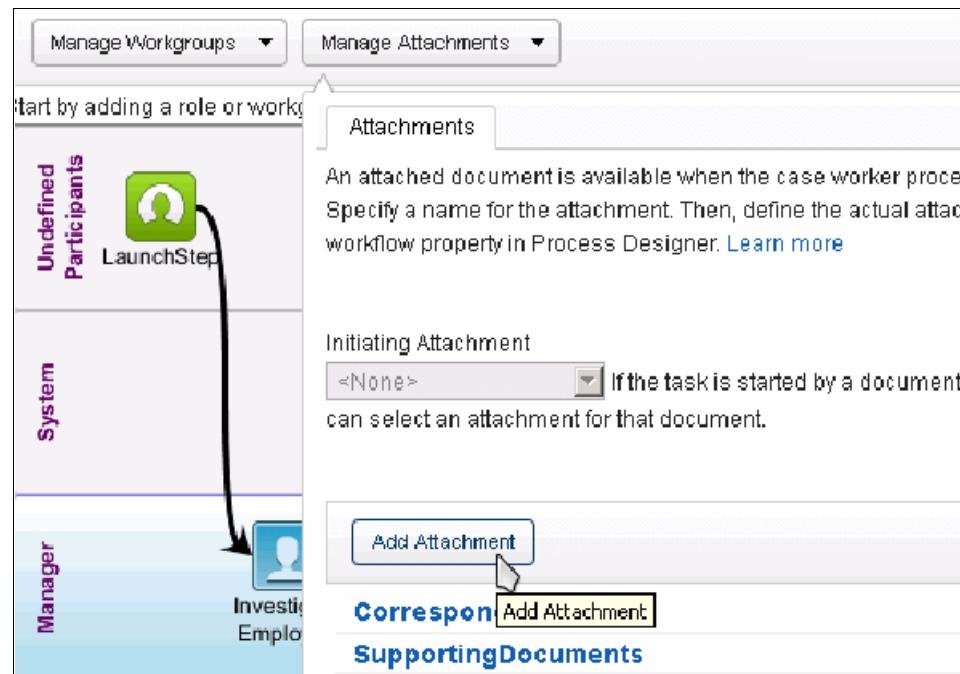


Figure 10-74 Adding an attachment

After the name of attachment is specified, this attachment is exposed to all steps as shown in Figure 10-75.

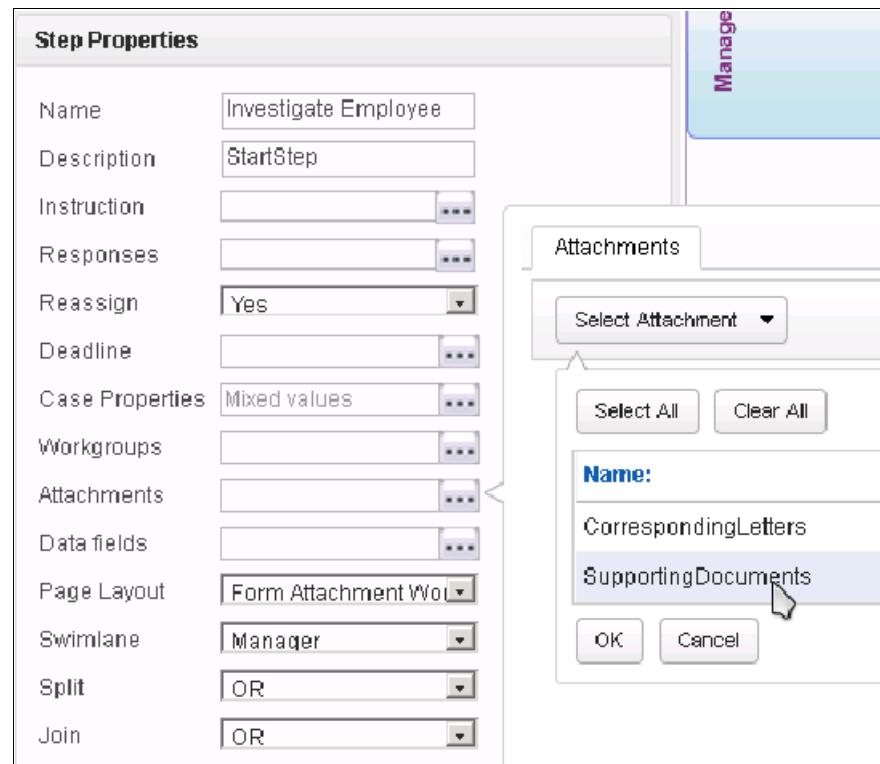


Figure 10-75 Attachment exposed to all steps

Configure Case Form widget on Form Attachment Work Details page to open a form as attachment as shown in Figure 10-76.

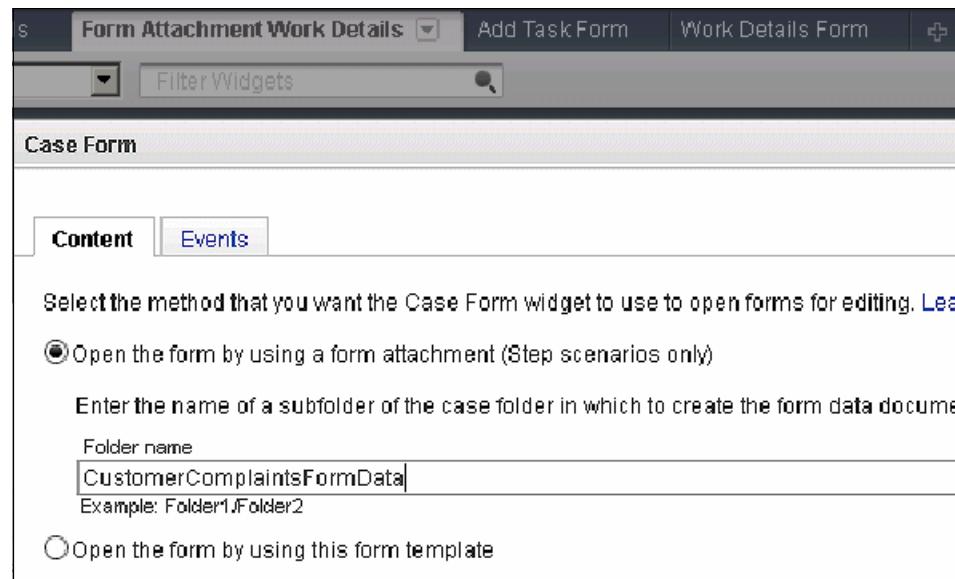


Figure 10-76 Opening a form by using a form attachment

Configure the Case Form widget on the rest of related pages to open a form by using a template as shown in Figure 10-77.

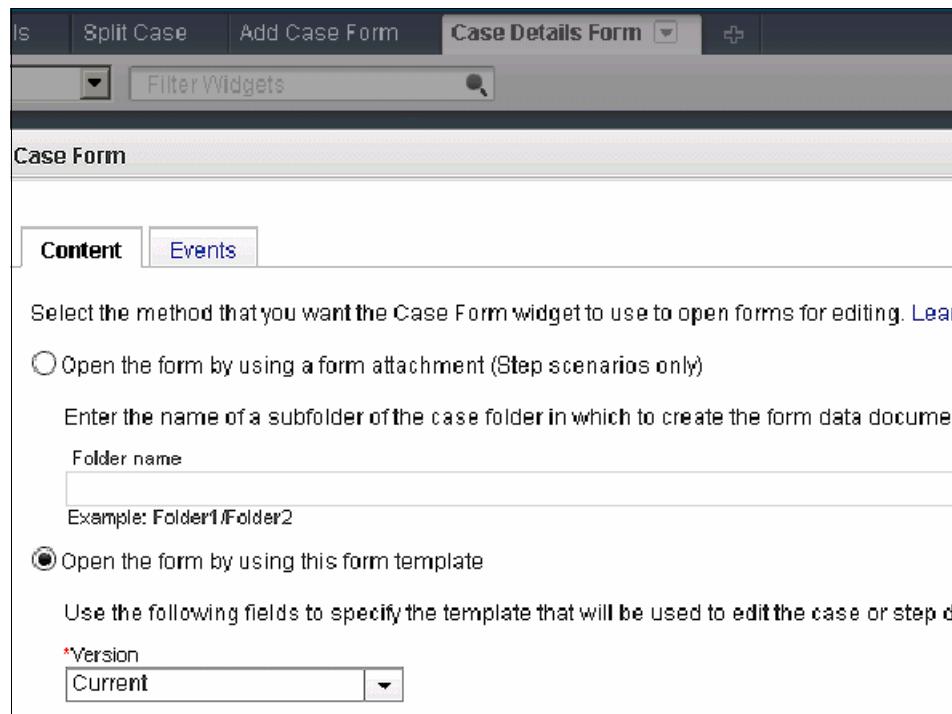


Figure 10-77 Opening a form by using form template

You attach the actual attachment when you add a task. To do so, complete these steps:

1. Add a case.
2. Render the case on the Case Details Form page and add the new task.
3. On the Add Task Form page, use the Attachments widget.

4. Search for a form template and add it under one of placeholders you created with Step Editor (Figure 10-78).

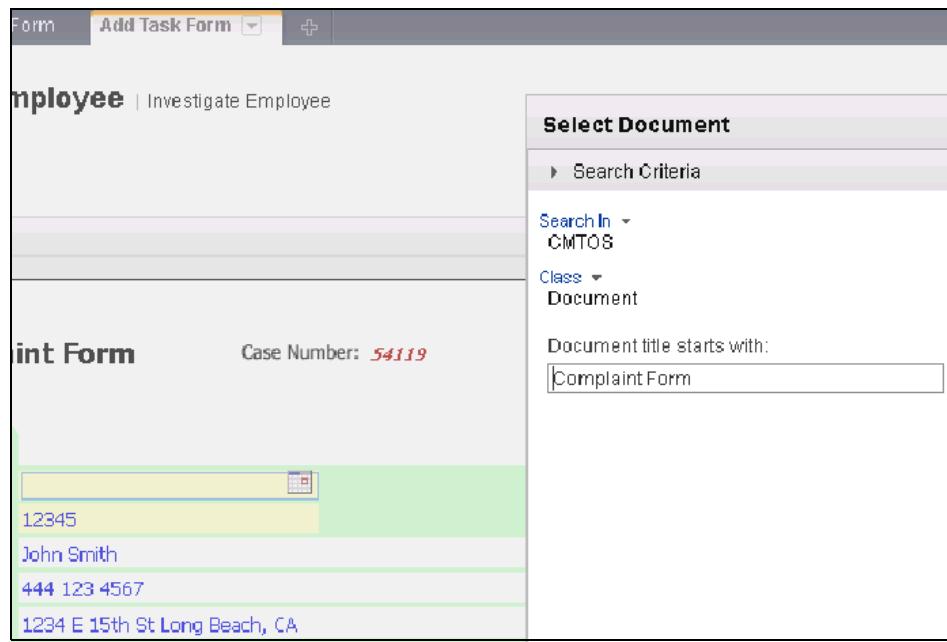


Figure 10-78 Selecting an attachment when you add a task

Tip: You can add attachments by using Workplace XT Process Designer. You can configure the Case Form widget on the Add Task Form page to open a form as attachment by using this designer.

Now that the Case Form widget-related pages are registered and set as default layout, Case Form widgets on each page are configured. The attachment is added, and the Customer Complaints case is ready for processing with forms. The Case Data widget has been replaced by the Case Form widget.

10.8.5 Using Case Form with FileNet eForm

FileNet eForms is installed by using IBM Case Manager administration client's Deploy and Configure Case Forms task.

When Case Form widget uses FileNet eForm to render a case or step data, several extra features are available:

- ▶ Rich user interface, including tabbed page view.
- ▶ Ability to embed scripts and calculations.
- ▶ Ability to produce high-resolution paper copies (form can be rendered in PDF format for printing).
- ▶ When you create a case, FileNet eForms can be configured to generate an auto increment number.
- ▶ Look up cells can be used with all types of Case Form widget configuration.

Because eForm has been in the FileNet P8 Platform for a long time, it is not covered further. For more information about eForm, see the FileNet P8 Version 5.1 Information Center at:

<http://publib.boulder.ibm.com/infocenter/p8docs/v5r1m0/index.jsp>

10.9 Using IBM Forms with IBM Case Manager

Although IBM Case Manager provides a default view for working with data, you might want to use a form to create the user interface for your applications. Forms deliver an intelligent user interface that enhances and simplifies the user experience for case-based solutions.

As mentioned earlier, you can design forms by using either IBM Forms or FileNet eForms in IBM Case Manager, Version 5.1. These options offer similar functions, but there are key differences.

This section provides an overview of IBM Forms, including its installation, configuration, and integration with IBM Case Manager. It also addresses using IBM Forms to create an intelligent user interface.

10.9.1 IBM Forms overview

The following options for capturing case data are available in IBM Case Manager:

- ▶ Default Case Data widget

IBM Case Manager provides a default widget for gathering data. This widget requires no configuration or extra tools, but also does not provide the rich experience of an e-forms interface.

- ▶ FileNet eForms

FileNet eForms are one alternative to the default case widget. These eForms are integrated with IBM Case Manager, and an eForm designer is available for creating them.

- ▶ IBM Forms

IBM Forms are another alternative to the default case widget. Similar to FileNet eForms, IBM Forms are integrated with IBM Case Manager, and a form designer is available.

Of these options, IBM Forms delivers best-in-class features with these advantages:

- ▶ Dynamic experience: Offers a rich user interface and rules driven interaction that increases case management productivity. This feature includes the ability to create multi-step wizard forms, dynamic choice lists, and embedded HTML form components.
- ▶ Data validation: Embedded constraints and customized business rules that help eliminate data capture errors.
- ▶ Simple integration: Web-based architecture makes setting up enterprise integration and service extensions easy. This feature includes the ability to design reusable form components that are easy to update.
- ▶ Digital signatures: A comprehensive range of signing methods are available to protect transactions.
- ▶ Platforms: Timely OS and web browser support keeps you competitive.
- ▶ Languages: Multilingual support extends the reach of your solution.
- ▶ Accessibility: Full, standards-based accessibility support.

IBM Forms helps speed automation by providing flexible, scalable electronic forms that can be embedded in web pages. The IBM Forms product has several components, including a form Designer and both Server and Viewer components.

The **Viewer** is a rich client that enables users to open, fill, and submit forms. It supports both online and offline processing, and works as either a browser plug-in or a stand-alone application.

The **Server** is a collection of components for constructing applications that use forms. The Server includes these components:

- ▶ API: A collection of specialized functions that reads, writes, and manipulates forms.
- ▶ Webform Server: Converts XML-based IBM Forms into HTML and JavaScript, allowing users to view, fill, sign, and submit forms in a browser. This application eliminates the need to install the Viewer on client computers.
- ▶ IBM Case Manager Integrator: Extends the Viewer and Webform Server to allow them to interact directly with IBM Case Manager.
- ▶ FileNet Integrator: Extends the Viewer and Webform Server to allow them to interact directly with FileNet Content Manager.

The **Designer** provides a complete what-you-see-is-what-you-get (WYSIWYG) environment for creating IBM Forms on Windows computers.

When you use IBM Forms with IBM Case Manager, you can use either the Viewer or Webform Server to render forms for the client. The Viewer must be installed on all client systems, but might provide better performance because the form internal logic and data validation are run locally. Webform Server allows users to process everything through a browser, but requires a server that processes forms for the user.

Use the IBM Case Manager administration client tool to configure which component you want to use to render forms.

10.9.2 Installing the IBM Forms Server

To use IBM Forms, you must first install IBM Forms Server, Version 4.0.0.2 in the IBM Case Manager environment. For detailed instructions, click **Extending your case management system → Installing IBM Forms** in IBM Case Manager, Version 5.1 Information Center.

Important: When you install IBM Forms Server, ensure that you select **IBM Case Manager Integrator** as one of the components to install. See Figure 10-79.

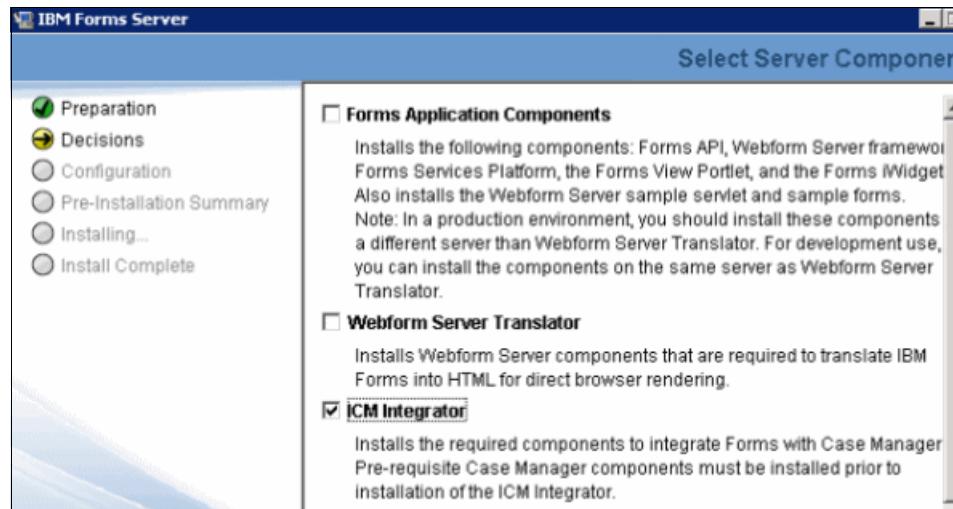


Figure 10-79 IBM Forms Server 4.0.0.2: Server Components

Webform Server can be installed on the same server as IBM Case Manager, or on a remote server. In either case, remember the URL for the Webform Server Translator. You must enter this URL into the IBM Case Manager administration client Deploy Forms Application task.

The example assumes that IBM Form Server 4.0.0.2 is installed with IBM Case Manager integrator and the Webform Server Translator is installed on icmlab1.

10.9.3 Configuring IBM Forms with IBM Case Manager

To configure IBM Forms with IBM Case Manager, run the **Deploy Forms Application** task from the IBM Case Manager administration client Task View.

Enter information for the Environment, Application server node, Application server name, and Application server virtual host fields. In addition, configure the following fields, which this section addresses:

- ▶ IBM Forms rendering application
- ▶ Webform Server Translator URL

IBM Forms rendering application

This field specifies whether forms are rendered by using the Viewer or Webform Server. Figure 10-80 shows different options available when rendering applications.

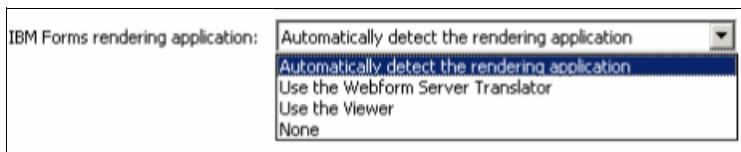


Figure 10-80 Deploy Forms Application task: IBM Forms rendering application options

You have these rendering options:

- ▶ Automatically detect the rendering application.

The IBM Case Manager environment detects whether the Viewer is available, and uses it if it is available. Otherwise, it uses Webform Server. This option is useful if some of your users might have the Viewer and others do not. The Viewer is used in preference to Webform Server because of performance considerations.
- ▶ Use the Webform Server Translator

The IBM Case Manager environment uses Webform Server to render forms.
- ▶ Use the Viewer

The IBM Case Manager environment uses the Viewer to render forms.
- ▶ None

Disables IBM Forms and set defaults to use FileNet eForms instead.

The example uses the **Automatically detect the rendering application** option.

Webform Server Translator URL

This field sets where the IBM Case Manager environment looks for the Webform Server Translator.

In the example, Webform Server is installed on the IBM Case Manager server. Therefore, it uses this URL (Figure 10-81):

`http://localhost:8085/translator`

The screenshot shows a dialog box titled "Deploy Forms Application task". Inside, there is a field labeled "Webform Server Translator URL:" containing the value "http://localhost:8085/translator". Below this field is a checkbox labeled "Restart WebSphere Application Server" which is checked. The entire dialog is enclosed in a light gray border.

Figure 10-81 Deploy Forms Application task: Webform Server Translator URL

If you use a remote server instead, replace localhost with the appropriate host name or IP address.

Tip: After you run this task, restart the WebSphere Application Server.

If you select the Restart WebSphere Application Server check box, the server is restarted automatically after you complete these settings. If you do not select the check box, the administrator must restart the WebSphere Application Server manually.

10.9.4 Installing IBM Forms Designer

You must use IBM Forms Designer to develop IBM Forms. The IBM Forms Designer tool can be installed on systems that are running Windows.

For more information about the supported platforms, see the detailed system requirements for IBM Forms Designer 4.0.0.2 at:

https://www.ibm.com/support/docview.wss?uid=swg27022316#IBM_Forms_Designer_4_0_0_2_Operating_Systems

To install the Designer, run the Windows installation package on the appropriate system.

10.9.5 Integrating IBM Forms with IBM Case Manager

Integrating IBM Forms with IBM Case Manager follows the same general steps as integrating FileNet eForms. This section highlights only those steps that are required for an IBM Forms integration. After you complete these steps, you can

follow the steps that are provided in 10.8, “Using the Case Form widget” on page 389.

Designing IBM Forms

As mentioned earlier, you use IBM Forms Designer to design the forms that are required for your solution. The IBM Forms Designer is a what-you-see-is-what-you-get (WYSIWYG) design tool that allows the business analysts to build forms by dragging and dropping items from a palette of choices.

The Designer provides rich user interface features so business analysts can easily re-create their existing paper forms. However, redesigning the forms for a web-oriented presentation has advantages, such as wizard-style forms.

For more information about designing forms, see the IBM Forms Designer 4.0 product wiki at:

<http://www-10.lotus.com/ldd/lfwiki.nsf/xpViewCategories.xsp?lookupName=Product%20documentation>

For the Customer Complaints example, the form is designed as shown in Figure 10-82 using IBM Forms Designer.

The screenshot shows a software interface titled "Customer Complaints V2". The top menu bar includes "Work", "Cases", "Add Case Form", and a "+" button. Below the menu is a section titled "Complaint" with a sub-section "Case Form". The main form is titled "Customer Complaint Form" and displays the following data:

Customer Complaint Form		Case Number: 50230
Customer Details		
Complaint Date:	14-Dec-11	
Customer Number:	12345	
Customer Name:	John Smith	
Contact Phone Number :	444 123 4567	
Address:	1234 E 15th St Long Beach, CA	
Email:		
Complaint Details		
Category:	Billing	
Amount:	\$100.00	
Description:		

Figure 10-82 IBM Forms Designer: Customer Complaints form

To integrate this form with the case properties, make sure the item types used to create the form match the case properties defined for the Customer Complaints solution.

Mapping case properties to IBM Forms

To integrate the form with IBM Case Manager, mark each item that shares data as a *Public Data Element* and add the case property's Unique Identifier. This maps each form item to a data element in the IBM Case Manager as shown in Figure 10-83.

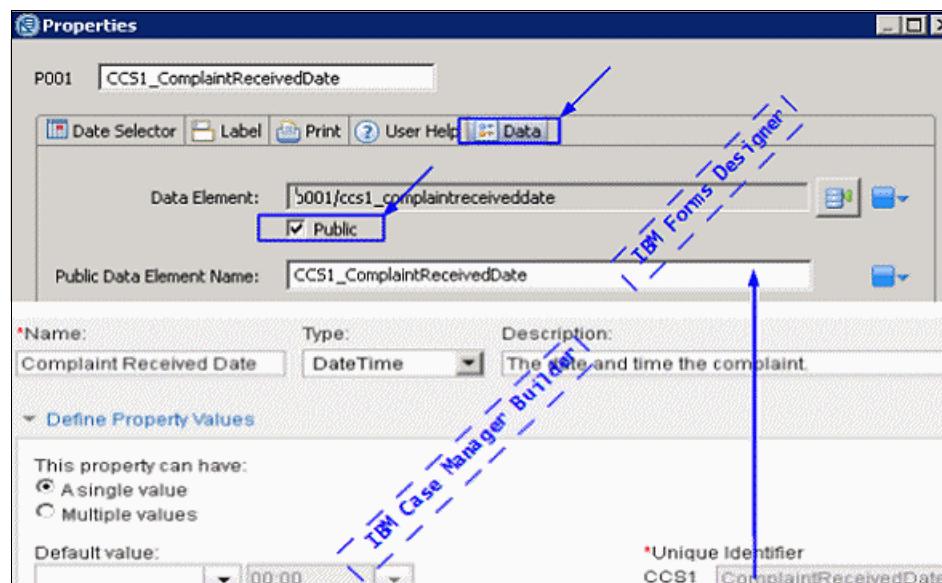


Figure 10-83 IBM Case Manager: Case Properties mapping with IBM Forms

To create this mapping, complete these steps:

1. Right-click an item and select **Properties**.
In the example, right-click **Complaint Date** and select **Properties**.
2. Click the **Data** tab.
3. Select **Public**.
4. Update the Public Data Element Name with the unique identifier of the case property.

In the example as shown in Figure 10-83, all the form items are mapped with the case properties Unique Identifier and marked as public. For example, the form item Complaint Date is mapped to case property Complaint Received Date by updating the **P001** property value with the unique identifier **CCS1_ComplaintReceivedDate**.

Consideration: By default, IBM Form items are not public. When mapping form items to case properties, be sure to select **Public**.

5. After the form design is complete, save the form in the XFDD file format.

Important: By default, the Designer save the files in the XFDL format, which is not supported by IBM Case Manager, Version 5.1. Be sure to use **Save as** to override the default file format.

Adding the form to the target object store

To add the form to the target object store of IBM Case Manager, use the document class ITX Form Template.

Tip: You can add the form template to the target object store with the subclass of the document class ITX Form Template.

In the example, CustomerComplaint.XFDD is added into target object store.

Customizing the existing Case Form widget

After you develop the form with IBM Forms and add it to the object store, use the Case Form widget to open it with the form template. For more information, see 10.8, “Using the Case Form widget” on page 389.



Development topics

This chapter provides a detailed description of a number of important or advanced topics that you might require to learn when you develop a case management solution.

This chapter includes the following sections:

- ▶ Using project areas when developing solutions
- ▶ Automated handling of ingested documents
- ▶ Splitting a case
- ▶ Reuse of an existing FileNet workflow
- ▶ Multilingual support

11.1 Using project areas when developing solutions

Companies require case-based solutions for various different requirements. Creating these solutions requires the involvement of various teams, business analysts, and the departments of the organization.

In the Customer Complaints use case example, the company has case-based solutions for account opening and maintenance, and payroll processing. The account opening cases are not relevant to the business analysts who handle the internal payroll processing system. These two solutions are not related to each other, and the payroll processing should not be available to all the users. The payroll processing solution should not be visible to business analysts other than those who belong to payroll department.

The case-based solutions can be grouped by using project areas so that only the relevant business analysts are given access privileges to the solution.

In addition, project areas help to limit the reset of the test environment. Resetting the test environment removes deployed case solution artifacts from the target object store and the process configuration information. A reset also reinitializes the target object store and the connection point that is associated with the project area.

When business analysts modify a deployed solution, the test environment provides them with an environment in which they can improve the solution iteratively. Business analysts might need to enhance, deploy, test, enhance, reset the test environment, redeploy, and test the solution again.

Instead of resetting everything in a test environment, you can use project areas to reset the test environment only of a particular project area. This isolation prevents the reset from disrupting the work of other business analysts who are working on different project areas.

Users and solutions can easily be moved from one project area to another.

Tip: Moving users and solutions are independent operations. Moving a user to another project area does not move that user's solutions as well.

Project areas are only available in the development environment. You must have at least one default project area. You can then create new project areas according to your business needs.

The IBM Case Manager example development environment has two project areas created. These areas are Customer, to maintain account related solutions, and Payroll, to group all the internal payroll-related system solutions.

Project areas can be created and modified by using either the graphical user interface of IBM Case Manager administration client tool or the command-line tool.

Remember: In this section, only the equivalent command-line commands are given for the graphical user interface operations in the examples. See the IBM Case Manager, Version 5.1 Information Center at:

<http://publib.boulder.ibm.com/infocenter/casemgmt/v5r1m0/index.jsp>

For the purposes of illustration, the artifacts that are shown in Table 11-1 are already available in the example environment.

Table 11-1 Available artifacts

Artifact	Name
Process Engine Connection Point	NewCP, NewCP2
Connection point object store	TOS01, TOS02
User	psmall, p8admin

You create the artifacts that are shown in Table 11-2 as part of this exercise.

Table 11-2 Artifacts to be created

Artifact	Name
Default project area	dev_env_connection_definition
Non-default project area	Payroll Process
Solutions	Payroll Process Sample solution

11.1.1 Creating the default project area

Each development environment requires a default project area that is named dev_env_connection_definition. To create the default project area, complete these steps:

1. Using the IBM Case Manager administration client application, open the required development profile.

For more information, see the IBM Case Manager, Version 5.1 Information Center at:

<http://publib.boulder.ibm.com/infocenter/casemgmt/v5r1m0/index.jsp?topic=%2Fcom.ibm.casemgmttoc.doc%2Fic-homepage.html>

Click **Installing and configuring IBM Case Manager** → **Installing and configuring on a stand-alone server** → **Configuring IBM Case Manager** → **Configuring the development environment**.

2. Open the **Create Default Project Area** task by double-clicking the task.
3. Enter a description for the **Project area description** field.
For the example, enter This is default Project Area.
4. Select the **Process Engine connection point**.
For the sample default project area, select **NewCP** from the menu.
5. The Process Engine connection point can be initialized and associated by selecting **Initialize the Process Engine connection point** if not already done by Workplace XT.
6. Select the **Connection point object store**. Each project area requires one object store.
For the example, select **TOS01**.
7. Administrative users can be assigned to the selected object store by selecting **Add Case Management object store administrators as users for the default project area**.
8. Save and run the task.

Restriction: The default project area name dev_env_connection_definition cannot be changed.

Figure 11-1 shows the default project area setup for our sample.

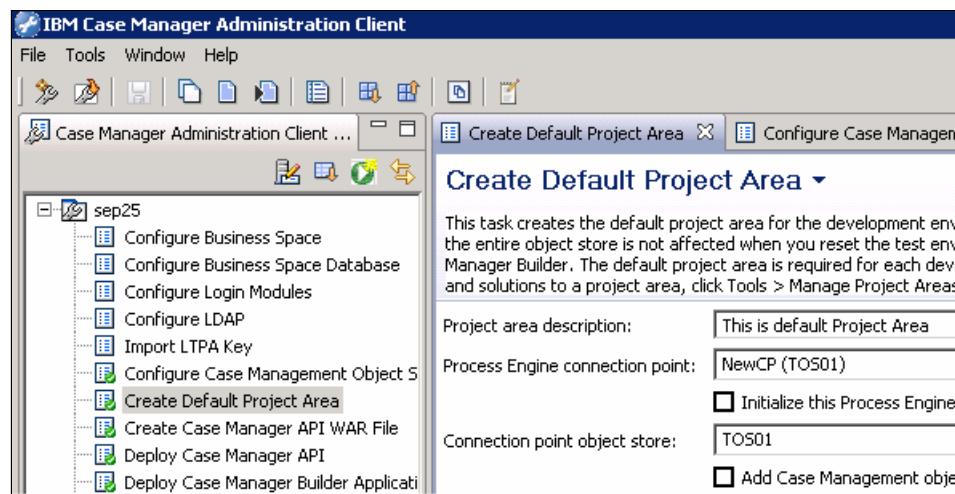


Figure 11-1 Creating the default project area from IBM Case Manager

You can also create the default project area by using the command line.

Figure 11-2 shows the command that is used for the example. This section uses Windows commands. For platform-specific commands, refer IBM Case Manager, Version 5.1 Information Center.

```
C:\Program Files (x86)\ibm\CaseManagement\configure>configmgr_c1  
createProjectArea -profile devprofile -projectAreaName  
dev_env_connection_definition -projectAreaDesc "Default project area"  
-peConnPt NewCP -initPEConnPtOS TOS01
```

Figure 11-2 Creating a default project area from the command line

11.1.2 Creating a project area

To create another project area, complete these steps:

1. Using the IBM Case Manager administration client tool, open the required development profile.
2. Click **Tools** → **Manage Project Areas**.
3. Click **Create** to start the Create Project Area wizard and click **Next**.

Figure 11-3 shows the Payroll project area setup.

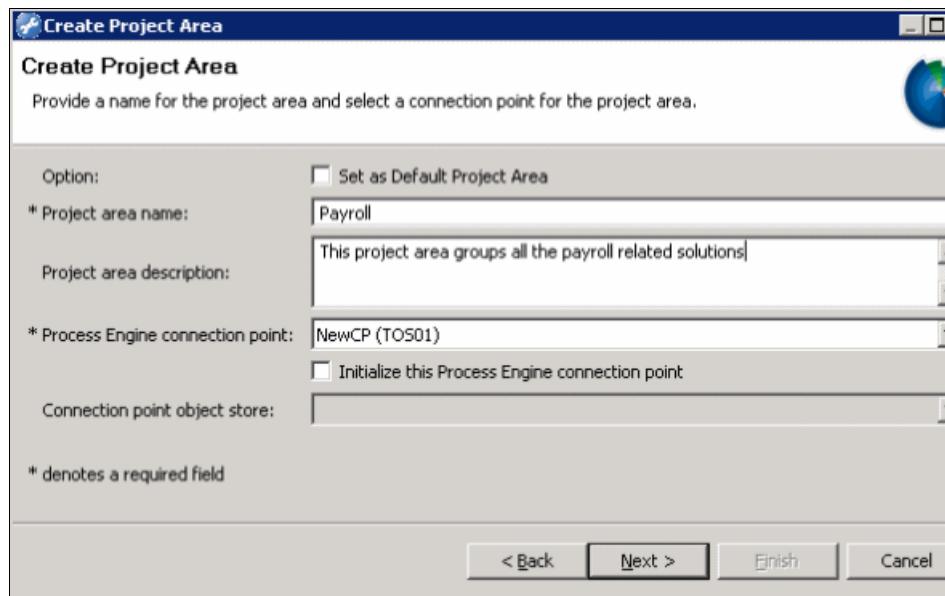


Figure 11-3 Creating the Payroll project area

4. Enter the Project area name and Project area description.

For the example, enter Payroll in the **Project area name** field. For **Project area description** field, enter This project area groups all the payroll related solutions.

5. Select the **Process Engine connection point**.

For the example, select **NewCP2**.

6. Select the **Connection point object store**.

For the example, select **TOS02**.

7. The project area can be set as default project area by selecting **Set as Default Project Area**.

8. Click **Finish** to complete the project area creation.

In this example, the project area called Payroll gets created to group all payroll-related case management solutions and assign the required users.

You can also create the project area by using the command line as shown for the example in Figure 11-4.

```
C:\Program Files (x86)\ibm\CaseManagement\configure>configmgr_c1  
createProjectArea -profile devprofile -projectAreaName payroll  
-projectAreaDesc "Payroll related project area" -peConnPt NewCP2  
-initPEConnPtOS TOS02
```

Figure 11-4 Creating a project area for the sample

11.1.3 Assigning solutions to a project area

Users and solutions can be assigned to or removed from the default and non-default project areas. You can assign them by using IBM Case Manager administration client's Manage Project Areas wizard.

Perform these steps to assign solutions to the project area:

1. Using IBM Case Manager administration client, open the required development profile.
2. Click **Tools → Manage Project Areas**.
3. Select the required project area. If it is the default project area, select **dev_env_connection_definition**. Click **Modify**.
4. Click **Next** twice to open the Assign Solutions window. Solutions that are assigned to this project area are listed.
5. To add more solutions, click **Add**.

For the example, add the solution Payroll Process to the project area **dev_env_connection_definition** as shown in Figure 11-5.

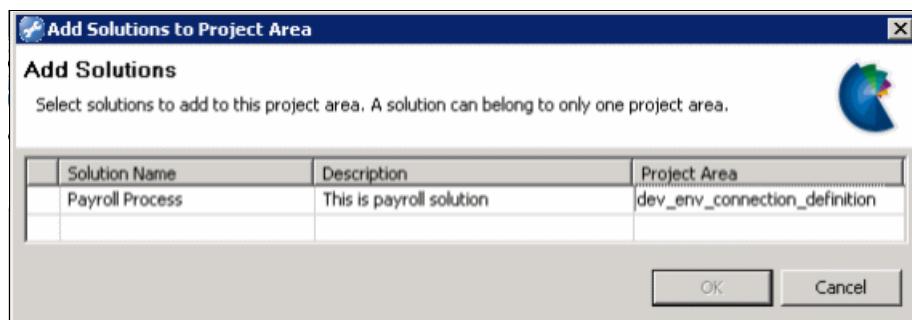


Figure 11-5 Adding solutions to a project area

6. Select the required solutions to be added to the new project area from the solutions that are listed in the Add Solutions to Project Area.
For the example, select **Payroll Process** from the list.
7. Click **OK**.
8. Click **Finish** and then **Close** to complete the solution assignment.

In this example, the Payroll Process solution is created in the default project area `dev_env_connection_definition` and is moved to the project area Payroll.

You can also add the solutions to a project area by using the command line. For the sample, use the command that is shown in Figure 11-6.

```
C:\Program Files (x86)\ibm\CaseManagement\configure>configmgr_c1  
addSolutions -projectAreaName payroll -profile devprofile -solutions  
"payroll process"
```

Figure 11-6 Command to add the Payroll Process solution to the Payroll project area

11.1.4 Removing solutions from a project area

The solutions can be removed from a project area. Perform these steps to remove the solution from a project area:

1. Using IBM Case Manager administration client, open the required development profile.
2. Click **Tools** → **Manage Project Areas**.
3. Select the required project area. If it is default project area, select `dev_env_connection_definition`. Click **Modify**.
4. Click **Next** twice to open the Assign Solutions window. Solutions that are assigned to this project area are listed.
5. Select the solution that you want to remove from the project area, and click **Remove** and then **Finish**.

For the example, select **Sample Solution**.

6. Click **Close**.

In this example, the Sample Solution is removed from the Payroll project area.

When the solution is removed from a non-default project area, it is automatically moved to default project area `dev_env_connection_definition`.

Figure 11-7 shows selecting the Sample Solution to be removed from the project area.

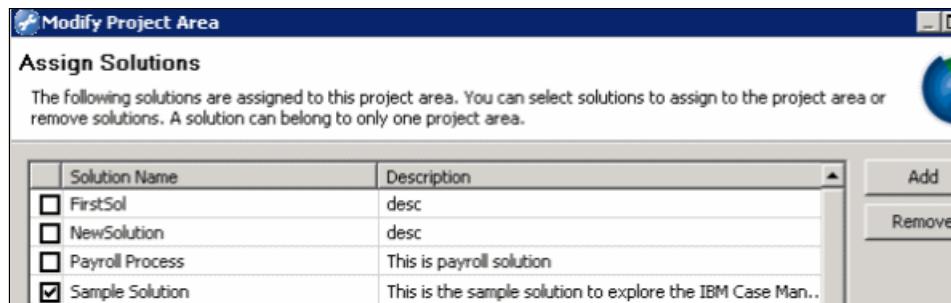


Figure 11-7 Removing solutions from a project area

You can remove the solutions from a project area by using the command line. Figure 11-8 shows the command used from the command line to remove “sample solution” from the payroll project area.

```
C:\Program Files (x86)\ibm\CaseManagement\configure>configmgr_c1  
removeSolutions -projectAreaName payroll -profile devprofile  
-solutions "sample solution"
```

Figure 11-8 Removing the “sample solution” from the payroll project area

11.1.5 Assigning users or groups to a project area

The users or groups can be assigned to a project area.

Important: For the default project area, both users and groups can be assigned. For non-default project areas, you can assign only users.

Perform these steps to assign users or groups to a project area:

1. Using IBM Case Manager administration client, open the required development profile.
2. Click **Tools** → **Manage Project Areas**.
3. Select the required project area. If it is the default project area, select **dev_env_connection_definition**. Click **Modify**.
For the example, select **Payroll** as the project area.
4. Click **Next** three times to open the Select Users and Groups window. Users and groups that are assigned to this project area get listed.

- To assign users or groups, click **Add**.
- Enter the user or group name and click **Search** as shown in Figure 11-9.

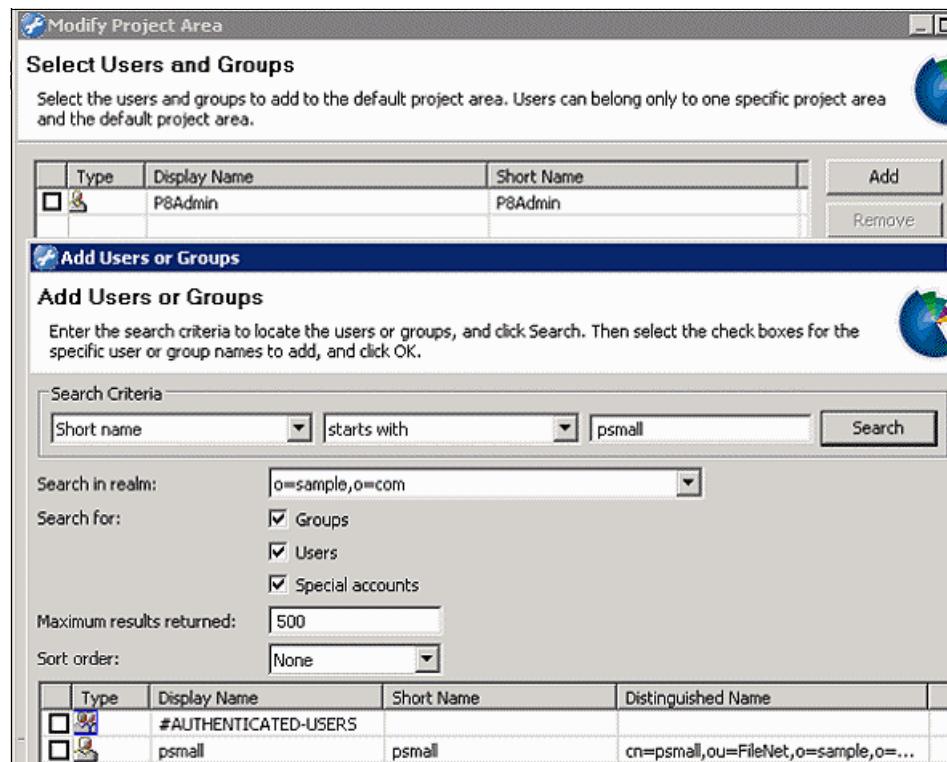


Figure 11-9 Adding users or groups to a project area

- Select the user or group from the result list and click **OK**.
For the example, select the user called **psmall**.
- Click **Finish** and **Close** to complete.

You can add users or groups to a project area through the command line. Figure 11-10 shows the command that is used to add the users for the example solution.

```
C:\Program Files (x86)\ibm\CaseManagement\configure>configmgr_c1
addPrincipals -projectAreaName dev_env_connection_definition -profile
devprofile -users psmall
```

Figure 11-10 Command to add the user psmall to the default project area

11.1.6 Removing users or groups from a project area

To remove users or groups from a project area, complete these steps:

1. Using IBM Case Manager administration client, open the required development profile.
2. Click **Tools** → **Manage Project Areas**.
3. Select the required project area. If it is the default project area, select **dev_env_connection_definition**. Click **Modify**.
For the example, select the project area **Payroll**.
4. Click **Next** three times to open the Select Users and Groups window. Users and groups that are assigned to this project area are listed.
5. Check the user or group that you want to no longer access the selected project area.
For the example, select the user called **p8admin**.
6. Click **Remove** and click **Finish**.
7. Click **Close** to complete.

In this example, the user called P8Admin is removed from the Payroll project area as shown in Figure 11-11.

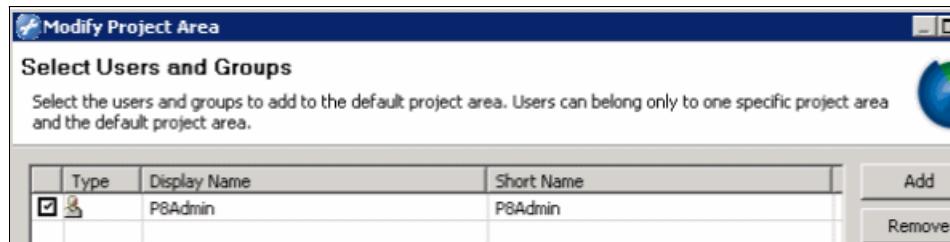


Figure 11-11 Removing the user P8Admin from a project area

You can remove users or groups from a project area by using the command line. Figure 11-12 shows the command that is used for the example.

```
C:\Program Files (x86)\ibm\CaseManagement\configure>configmgr_c1  
removePrincipals -projectAreaName payroll -profile devprofile -users  
psmall
```

Figure 11-12 Command to remove the user psmall from the payroll project area

Figure 11-13 shows the command line to remove the group called clerks.

```
C:\Program Files (x86)\ibm\CaseManagement\configure>configmgr_c1  
removePrincipals -projectAreaName payroll -profile devprofile  
-groups clerks
```

Figure 11-13 Command to remove clerks group from the payroll project area

11.1.7 Removing a project area

When a project area is no longer required, the project area can be deleted by using IBM Case Manager administration client or from the command-line command.

Exception: Default project areas cannot be deleted.

To delete a project area from IBM Case Manager administration client, complete these steps:

1. Open the required development profile.
2. Click **Tools** → **Manage Project Areas**.
3. Select the project area to be deleted and click **Delete**.
For the example, select **Payroll Process**.
4. Click **Yes** to confirm the deletion and then click **Close**.

Figure 11-14 shows the confirmation window for deleting the project area.

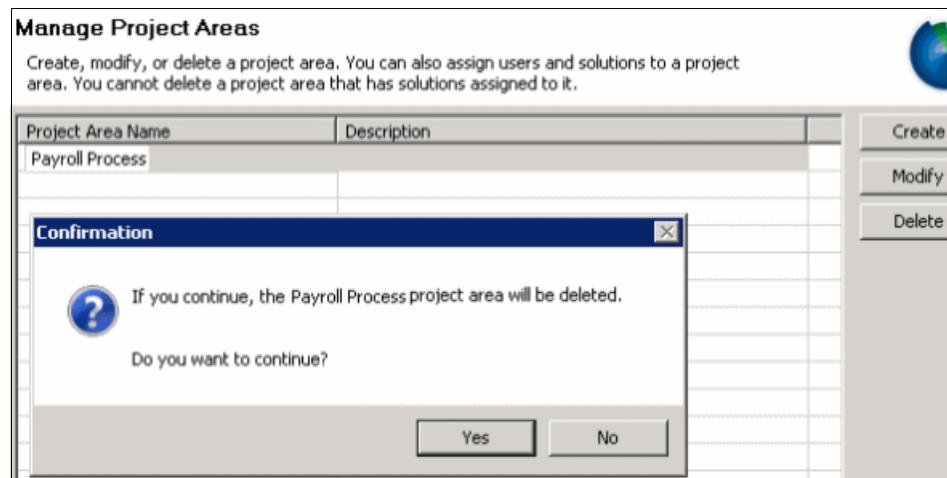


Figure 11-14 Deleting the Payroll Process project area

You can also delete the project area by using the command line. Figure 11-15 shows the command that is used for the example.

```
C:\Program Files (x86)\ibm\CaseManagement\configure>configmgr_c1  
deleteProjectArea -profile devprofile -projectAreaName payroll
```

Figure 11-15 Deleting a project area by using the command line

11.1.8 Determining the project area that you are in

When you log in to Case Manager Builder to develop a case-based solution, you are taken to the assigned project area based on the user ID provided.

If you have access to only the default project area in the example, you cannot see the solutions that are created under other project areas. These restricted areas can include Payroll Processing and Customer.

To determine what project area the user must log in to, click **About** from Case Manager Builder as shown in Figure 11-16.

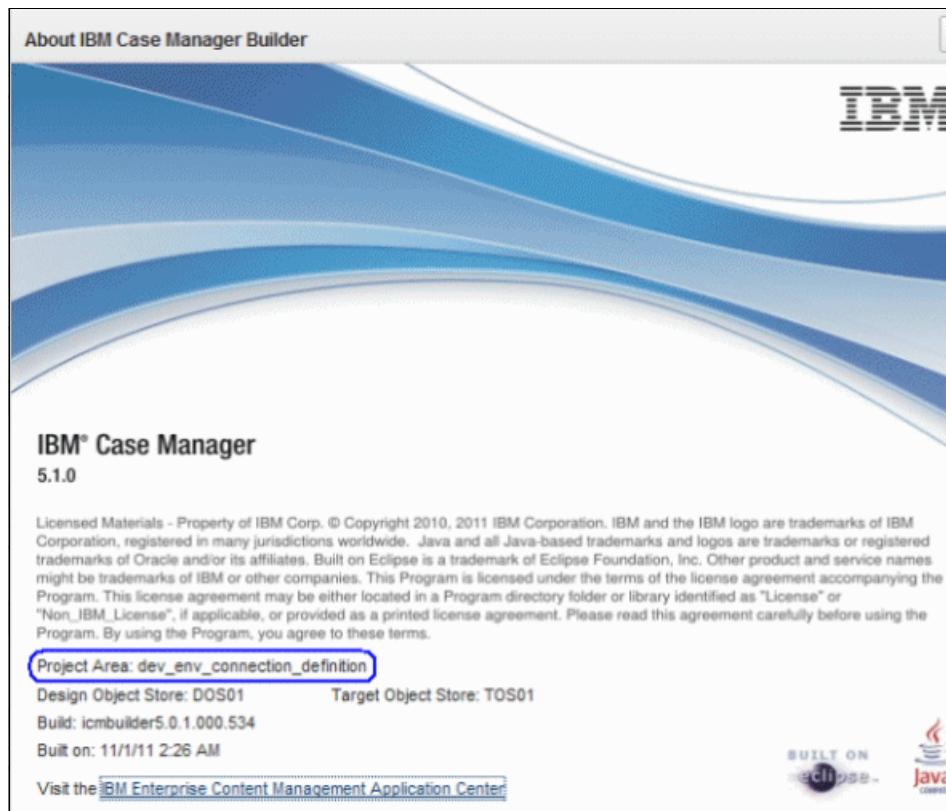


Figure 11-16 About in the Case Manager Builder showing the current project area

11.1.9 Summary

This section described the benefits of having multiple project areas and how to create and delete them. In addition, it described assigning solutions, users, and groups to a project area and removing them from the project area.

When the business analyst resets the test environment, only the solutions and artifacts in the project area that the analyst is logged in to are removed.

Project Area adds strength to the IBM Case Manager iterative model of solution development.

11.2 Automated handling of ingested documents

Often documents are ingested from other systems that must find their way into a corresponding case. These sources of documents are commonly used together with IBM Case Manager:

1. Scanned documents:
 - Datacap Taskmaster and other capture systems
 - Desktop scanning software
2. Email messages and document attachments:
 - IBM Content Collector for Email (Microsoft Exchange / IBM Lotus Domino® / SMTP)
 - FileNet Integration for Microsoft Office (Microsoft Outlook)
 - FileNet Services for IBM Lotus Quickr (Microsoft Office / IBM Lotus Notes)
3. Microsoft Office documents:
 - FileNet Integration for Microsoft Office
 - FileNet Services for Lotus Quickr
4. Sharepoint documents:
 - IBM Content Collector for Sharepoint
5. File server documents:
 - IBM Content Collector for Files
6. Desktop documents:
 - Documents that are uploaded into the system with Workplace or WorkplaceXT
 - FileNet Services for Lotus Quickr
7. Other systems:
 - Third-party software that interfaces with FileNet APIs to ingest documents

11.2.1 Generalized approach for handling ingested documents

To handle documents that come into the system from other systems and automatically file them into the correct case folder has these considerations:

- ▶ The document ingestion software classifies the document according to the document types defined in the solution.
- ▶ The case type and document type have a property (or set of properties) that serves as the *key* to the case. That is, each case has a unique, business-level

identifier that is associated with every case. Sample keys include Claim Number, Loan Number, Investigation ID, Case Number, Quote Number, and Request Number.

- ▶ Every document from another system that is filed into a case folder is indexed with this key.
- ▶ An event action can be triggered by document class (type) if there is an event subscription to the creation event.
- ▶ The event action locates the corresponding case folder by using the case identification key from the document and files the document into the case folder.

11.2.2 Implementation methods

There are several methods that can be deployed to automate the implementation as outlined in Table 11-3.

Table 11-3 Implementation methods for automated case document filing

Location	Method	Description
Client-side	Custom code that runs on client application	The client application runs the case folder search and stores the document in the matching case folder upon ingestion.
Server-side	JavaScript Event Action	A JavaScript-based event action is created to handle the matching and filing of the case document. This method is appropriate for low-volume and non-production implementations.
Server-side	Java Event Action	A Java-based event action is created to handle the matching and filing of the case document. This method is the best method for production implementations.

11.2.3 Sample Javascript-based implementation procedure

An example Javascript-based implementation procedure has these steps:

- ▶ Using Case Manager Builder:
 - a. Define the case property to be used as the key.
The example uses Case Number as the key.

Remember: The key that is used is not necessarily the same as the Case Title designated property, although this is a common practice.

- b. Add the property to the document type that will be ingested.
 - c. Deploy the solution.
- Using FileNet Enterprise Manager:
- a. Identify the document class (type) used in your solution.
The example uses the *Correspondence* document type.
 - b. Add a subscription (Figure 11-17) to this document class.

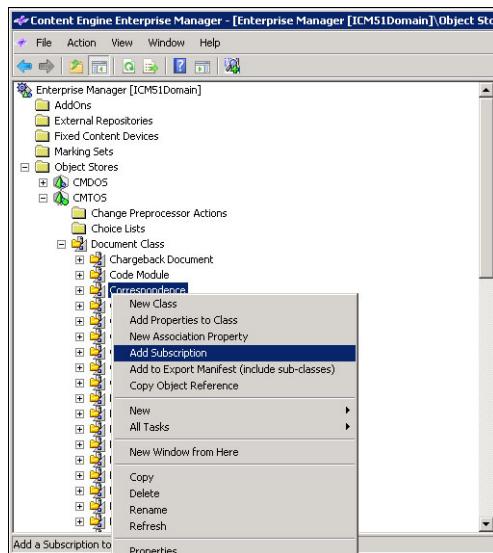


Figure 11-17 Adding a subscription

Figure 11-18 on page 430 shows the subscription information.

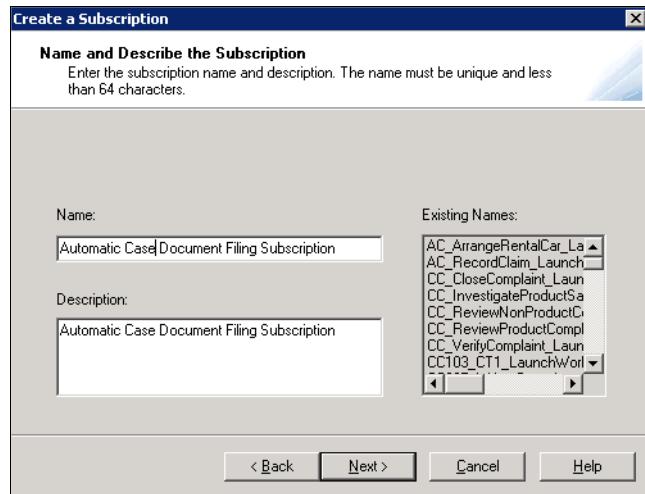


Figure 11-18 Create a Subscription wizard

Figure 11-19 shows the subscribed events.

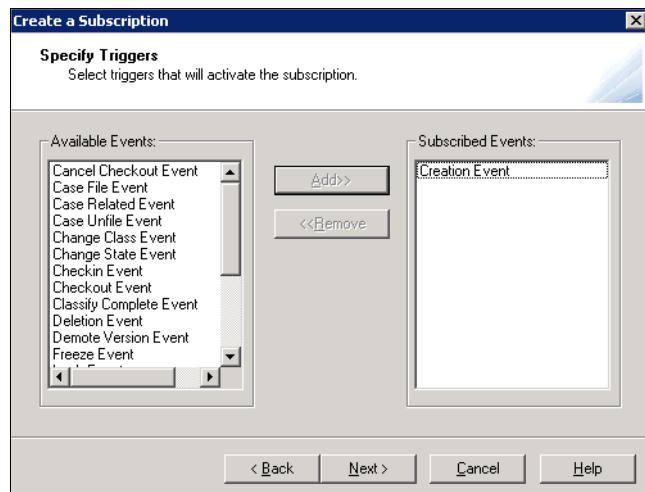


Figure 11-19 Specify Trigger window

- c. Using the wizard, add an event action. In the example, it is called Automated Case Document Filing Action as shown in Figure 11-20.

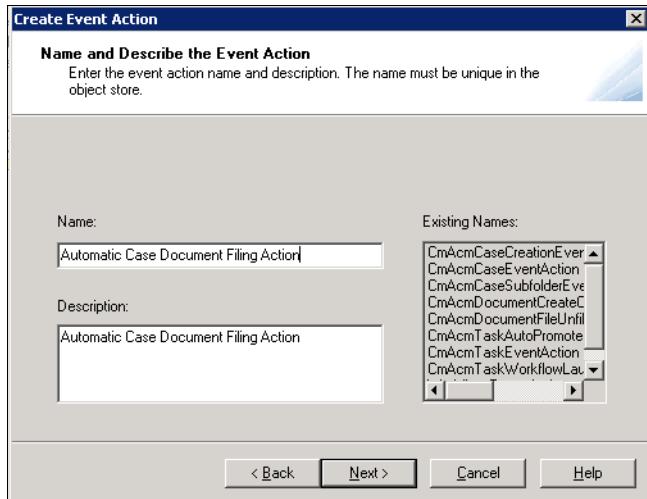


Figure 11-20 Create Event Action wizard

- d. Select JavaScript as the event action type as shown in Figure 11-21.

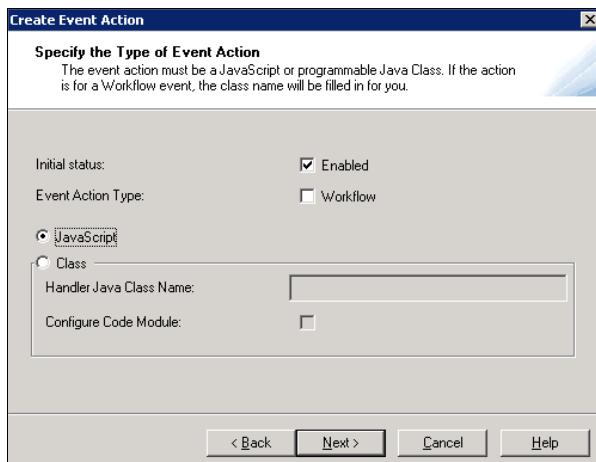


Figure 11-21 JavaScript Event Action Type

- e. Paste the JavaScript code that is provided as shown in Figure 11-22.

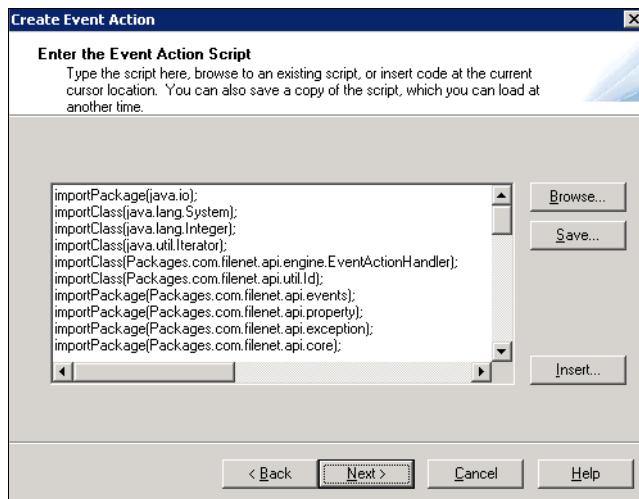


Figure 11-22 JavaScript for Event Action

Example 11-1 shows the sample JavaScript code.

Example 11-1 Sample JavaScript code

```
importPackage(java.io);
importClass(java.lang.System);
importClass(java.lang.Integer);
importClass(java.util.Iterator);
importClass(Packages.com.filenet.api.engine.EventActionHandler);
importClass(Packages.com.filenet.api.util.Id);
importPackage(Packages.com.filenet.api.events);
importPackage(Packages.com.filenet.api.property);
importPackage(Packages.com.filenet.api.exception);
importPackage(Packages.com.filenet.api.core);
importPackage(Packages.com.filenet.api.constants);
importClass(Packages.com.filenet.api.query.SearchSQL);
importClass(Packages.com.filenet.api.query.SearchScope);
importClass(Packages.com.filenet.api.collection.IndependentObjectSet
);

function onEvent(event, subscription) {
// customize to some other string property symbolic name as needed
var keyIdentifierName = "CC_CaseNumber";
var os = event.getObjectStore();
```

```

var id = event.get_SourceObjectId();

// get document's key identifier value
var fe = new FilterElement(null, null, null, "Name " +
keyIdentifierName, null);
var pf = new PropertyFilter();
pf.addIncludeProperty(fe);
var doc = Factory.Document.fetchInstance(os, id, pf);
var identifier =
doc.getProperties().get(keyIdentifierName).getStringValue();

// find case folder of same key identifier value
var sqlStr = "SELECT [" + keyIdentifierName + "], [FolderName],
[PathName], [Id] FROM [CmAcmCaseFolder] WHERE (" +
keyIdentifierName + "] = '" + identifier + "')";
var sql = new SearchSQL(sqlStr);
var ss = new SearchScope(os);
var objectSet = ss.fetchObjects(sql, new Integer(1), null, null);

var iter = objectSet.iterator();
if (iter.hasNext())
{
    var folder = iter.next();

    // file the doc and save
    var drcr = folder.file(doc, AutoUniqueName.AUTO_UNIQUE,
doc.getProperties().getStringValue("Name"),
DefineSecurityParentage.DO_NOT_DEFINE_SECURITY_PARENTAGE);

    drcr.save(RefreshMode.NO_REFRESH);
}
}

```

Test your implementation by creating a document of the specified class. Make sure that the document is filed into the appropriate case folder.

11.3 Splitting a case

Sometimes you need to split a single case into two separate cases. For example, in the complaints management solution, a customer might lodge a complaint for both a product defect and a service failure on the same complaint form or email. However, based on company guidelines, these are two different types of

complaints that must be resolved through separate review processes. In this situation, the case worker must split the original case into two separate cases. Many of the details such as customer name and date when the complaint was lodged are the same for both cases.

The *Split Case page* is specially designed for splitting cases. The Split Case page is started when the user clicks **Split Case** on the Case toolbar.

When a case is split, a new case is created based on information from the original case. Documents in the original case can be selectively moved to the new case. The new case can be of the same case type or a different type within the same solution. After the split, the two cases continue independently. The case history shows the split action and the other related case.

This section describes how to split a case by using the Split Case page. It also outlines the additional steps to incorporate the Split Case page when you upgrading an IBM Case Manager system from version 5.0 to version 5.1.

11.3.1 Configuring the case toolbar to show the Split Case button

The Split Case page is available in the Case windows of the solution space from IBM Case Manager, Version 5.1 onwards.

To find the Split Case window, log in to Case Manager Client, select **Manage Spaces**, and locate and open the solution that you are working on.

Important: For IBM Case Manager installations that are upgraded from older versions, manual steps are required to add the page. For more information, see 11.3.3, “Registering the Split Case page for upgraded systems” on page 441.

Figure 11-23 shows the **Split Case** button.

The screenshot shows the IBM Case Manager Space Manager interface. It lists several spaces under the 'Customer Complaints' category. The 'Customer Complaints Ca...' space is expanded, showing its description and two actions: 'Add Case' and 'Case Details'. Below these is the 'Split Case' action, which is highlighted with a red rectangular box. At the bottom of the list is the 'Add Case Form' action. Each action row includes a small icon, a label, a brief description, and an 'Actions' column.

Action	Description	Actions
Add Case	- Hidden from space manager and selector menus Case workers can add a new case of a selected type on the Add C...	Actions
Case Details	- Hidden from space manager and selector menus Case workers view and update a case on the Case Details page. ...	Actions
Split Case	- Hidden from space manager and selector menus Case workers can split a new case from a current case on the Spl...	Actions
Add Case Form	- Hidden from space manager and selector menus The Add Case Form page can be used in place of the Add Case p...	Actions

Figure 11-23 Split Case link

The split case action is initiated through **Split Case**. Place this button in the Case Toolbar widget on the Case Details window. A manual configuration is required to place this button on the Case Toolbar. Perform these steps once per solution:

1. Open your solution.
2. Go to the Case Details window by selecting a case from the case search page.

You can also get to the Case Details window by clicking the case name on the Case Information widget (Figure 11-24). If no cases are created yet, you must create one.

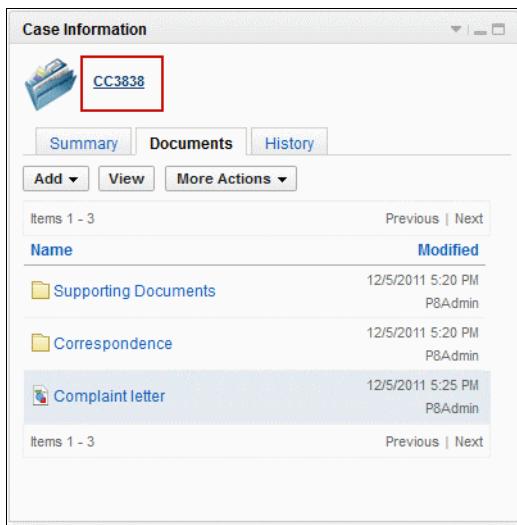


Figure 11-24 Case name from Case Information widget

3. Edit the settings of the Case Toolbar widget, which is usually at the top of the Case Details page. The frame of the Case Toolbar is usually configured to be invisible, so click **Edit Page** to make the frame visible. Click the small triangle on the upper right corner of the widget to display the menu options, and select **Edit Settings**. See Figure 11-25.

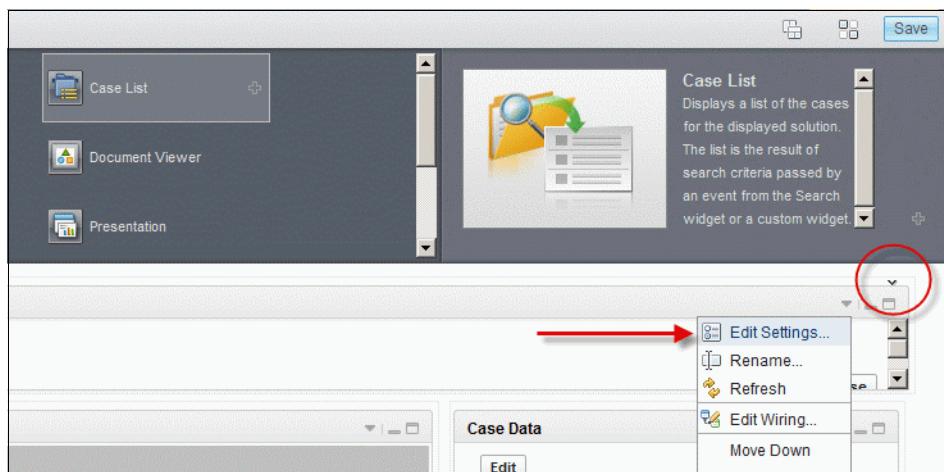


Figure 11-25 Edit Settings on the Case Toolbar

4. Add the **Split Case** button to the toolbar:

- Click the **Add** tab (Figure 11-26).

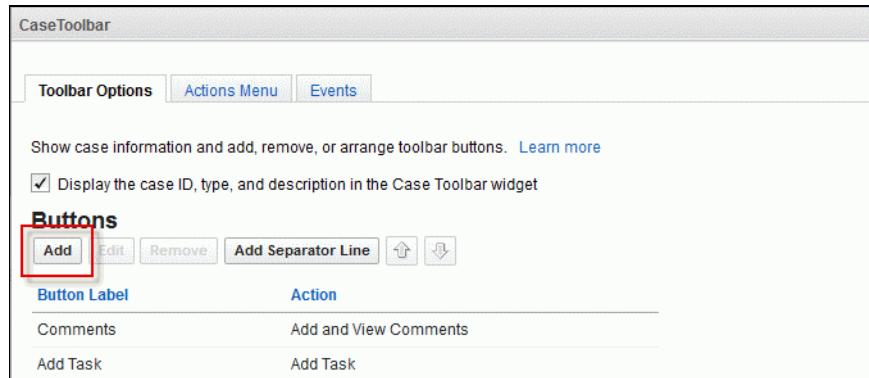


Figure 11-26 Add button on Case toolbar

- Assign a label to the button as shown in Figure 11-27, and select the **Split Case** action.

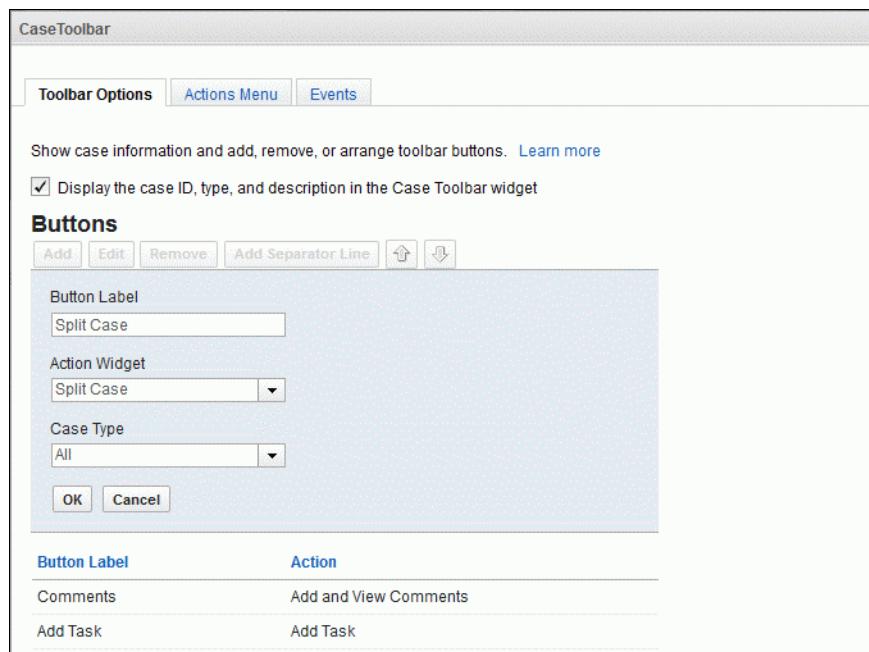


Figure 11-27 Adding the Split Case button

5. To allow the user to select the case type for the new case created by the split at run time, select **All** for case types. Alternatively, select a specific case type to be used for the new case created by the split. Click **OK**.
6. Click **Save** below the **Finish Editing** button.
7. Click **Finish Editing**.
8. Refresh the case toolbar.

The **Split Case** button is now available on the Case toolbar (Figure 11-28).

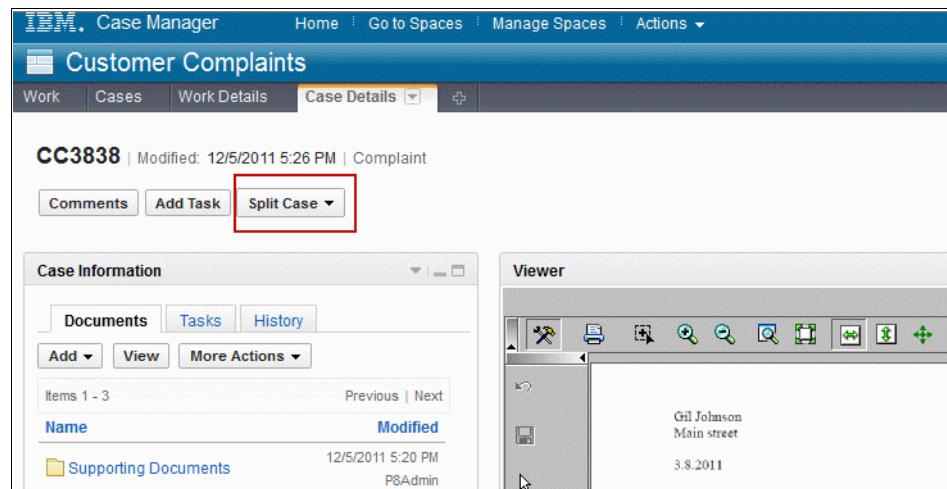


Figure 11-28 Split Case button on the Case toolbar

11.3.2 Splitting a case into two cases

The Split Case window is opened by clicking **Split Case** in the Case Details page (Figure 11-29). If there are multiple case types in the solution, the user can select the case type of choice for the new case created by the split.

The screenshot shows the 'Customer Complaints' application interface. The title bar says 'Customer Complaints'. The menu bar includes 'Work', 'Cases', 'Work Details', 'Case Details', 'Split Case' (which is currently selected), and a '+' button. The main content area has a header 'Split From: CC3838 | Complaint'. Below this are two main sections: 'Split Case Data' and 'Select Case Documents'. The 'Split Case Data' section contains fields for 'Customer Information' (disabled) and 'Complaint Information'. Under 'Complaint Information', fields include 'Case Number' (CC3842), 'Complaint Received Date' (10/16/2011, 12:00 AM), 'Complaint Category' (Service), 'Complaint Description' (Withheld important information), 'Case Source' (Letter), and 'Case State' (Open). The 'Select Case Documents' section allows users to choose documents to be removed from the original case. It lists 'Name' (unchecked), 'Supporting Documents' (unchecked), 'Correspondence' (unchecked), and 'Complaint letter' (checked). A note says 'Select the documents that you want removed from the original case.'

Figure 11-29 Split Case window

The Split Case page contains five widgets:

- ▶ Case toolbar: Displays header information.
- ▶ Split Case Data: Shows the case properties that can be edited for the new case. The properties fields are pre-filled with matching values from the original case.
- ▶ Select Case Documents: Allows user to select documents from original case, to be added to the new case.
- ▶ Original Case Data: Displays read-only property values from the original case.
- ▶ Command widget: Provides buttons to add the new case or cancel.

Tip: Normally the property fields in the new case are pre-filled with matching values from the original case. However, if the case has a property that has a default value, and the current value of the property in the original case is NULL, the property is set to the default.

Review the data that are presented for the new case, edit them if necessary, and select the case documents to be added to the new case. After user confirmation, the split operation is carried out, which copies the information to a new case instance. The new case and original case continue independently, and complete separately.

The tasks from the original case continue to run, and new tasks for the new case are initiated in the normal fashion. They function the same as though they had been created through the Add Case function. If the user chooses to do so, selected documents are filed into new copied case at the case root level. The history of both the original and the new case shows the split action and a link to the other case.

Figure 11-30 shows the history of the split case activity for the example use case.

The screenshot shows the 'Customer Complaints' application interface. The top navigation bar includes 'Work', 'Cases', 'Work Details', 'Case Details' (selected), and a '+' button. Below the navigation is a case identifier 'CC3838' with a modification timestamp 'Modified: 12/5/2011 5:21 PM | Complaint'. A toolbar below the identifier contains 'Comments', 'Add Task', and 'Split Case' buttons. The main area is divided into two panels: 'Case Information' on the left and 'Viewer' on the right. The 'Case Information' panel has tabs for 'Documents', 'Tasks', and 'History' (selected). It displays a summary for all items, showing 'Items 1 - 10' and a date range 'Today' (12/5/2011). A list of activities is shown, with the first item, 'Case split into a new case' (05:26 PM, CC3842), highlighted with a red border. The 'Viewer' panel on the right shows a document preview for 'Gil Johnson Main street 3.8.2011 Dear Sir: Re: 1269-5347'.

Figure 11-30 History of the case

11.3.3 Registering the Split Case page for upgraded systems

Because of compatibility considerations, the Split Case page is not automatically available for solutions that were created in IBM Case Manager Version 5.0. However, you can enable this feature manually for those solutions that require it.

Repeat the following steps for each upgraded solution:

1. In the Case pages space for the solution, create a Split Case page by using the available widgets. This page is available for new installations of IBM Case Manager by default.
2. After the Split Case page is created, go to the Case Manager Builder to edit your solution. For each case type in the solution, register the created Split Case page as the default Split Case page, and then redeploy the solution.

The detailed steps by step instructions are described in the IBM Case Manager Information Center at:

<http://publib.boulder.ibm.com/infocenter/casemgmt/v5r1m0/index.jsp?topic=%2Fcom.ibm.casemgmttoc.doc%2Fic-homepage.html>

Click **Installing and configuring IBM Case Manager → Upgrading and configuring IBM Case Manager → Upgrading solutions → Enabling the case split feature.**

Important: If you update an upgraded solution to add a case type, the Split Case page must to be available in the Case pages space. Therefore, this page must be created manually as described here. Otherwise, the solution fails to deploy.

11.4 Reuse of an existing FileNet workflow

IBM Case Manager supports the reuse of existing FileNet Process Engine workflow definitions as tasks in IBM Case Manager solutions.

The mechanism of reusing an existing Process Engine workflow process into an IBM Case Manager solution is provided by the Solution Workflow Collection feature.

Each IBM Case Manager solution has a Solution Workflow Collection, which is empty when a new solution is created. Existing Process Engine workflows can be selectively imported into the Collection, making them available to the IBM Case Manager solution. These imported processes are available to all the case types within a solution, and can then be linked to individual tasks.

Figure 11-31 shows a sample Solution Workflow Collection with three workflow definitions. Tasks 1 and 3 in Case Type 1 solution reuse Workflows 1 and 3 from the collection. Task 2 of Case Type 2 solution reuses Workflow 2 from the collection.

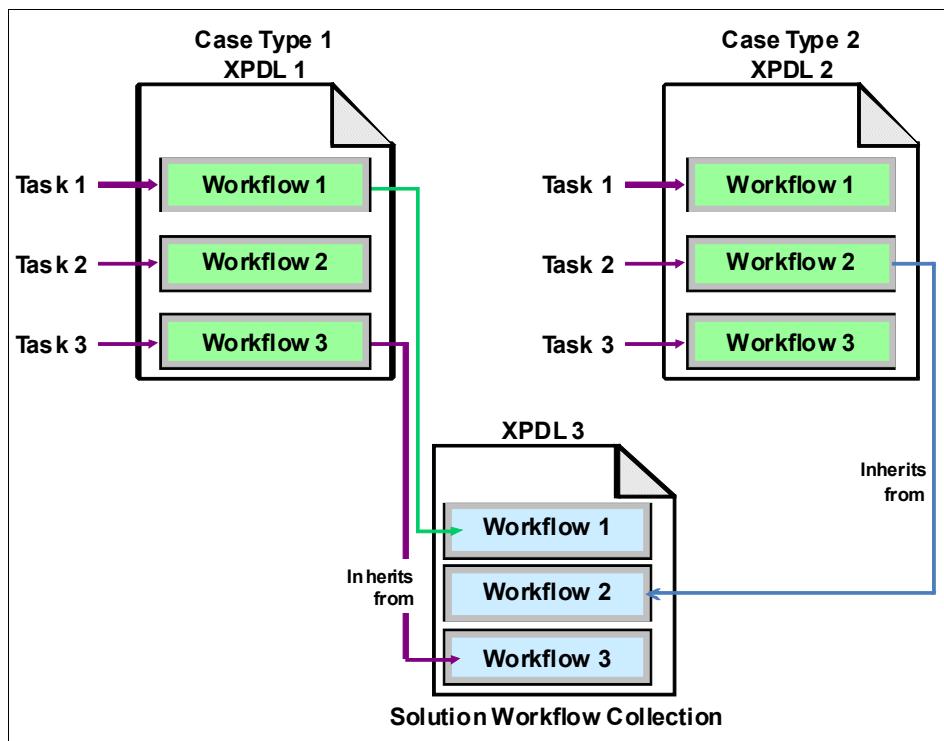


Figure 11-31 Linking to the Solution Workflow Collection

The Case Manager Builder is used to associate case tasks with the individual FileNet workflow processes inside the collection. With Case Manager Builder, you can define the mapping between the data fields of reused FileNet workflow process and the case data. For tasks that are created from scratch in the Case Manager Builder, this mapping is done transparently by default.

When a FileNet process from the Solution Workflow Collection is assigned to an IBM Case Manager task, the Case Manager Builder runs a number of steps:

1. Creates an IBM Case Manager process that inherits from the reused process.
2. Creates a submap that is called UpdateCase, which implements the mapping of the workflow data fields to the case data. You configure these data fields when you associate the FileNet process to the case task.

3. Modifies the Terminate system submap of the IBM Case Manager process to run the UpdateCase submap.

These steps by the Case Manager Builder ensure that the case data is properly updated at the end of the process.

11.4.1 Reusing an existing FileNet process as a case task

To reuse an existing FileNet process, complete these steps:

1. Import an existing FileNet process into the Solution Workflow Collection:
 - a. Using Process Designer, open the solution definition file for editing by clicking **File** → **Solution** → **Edit**.
 - b. Select the **Solution Definition** file for editing.
 - c. When presented with the case type window, select **Solution Workflow Collection**. See Figure 11-32.

Important: Do not try to select “Solution Workflow Collection” in the Open Solution window. Doing so results in an error message.

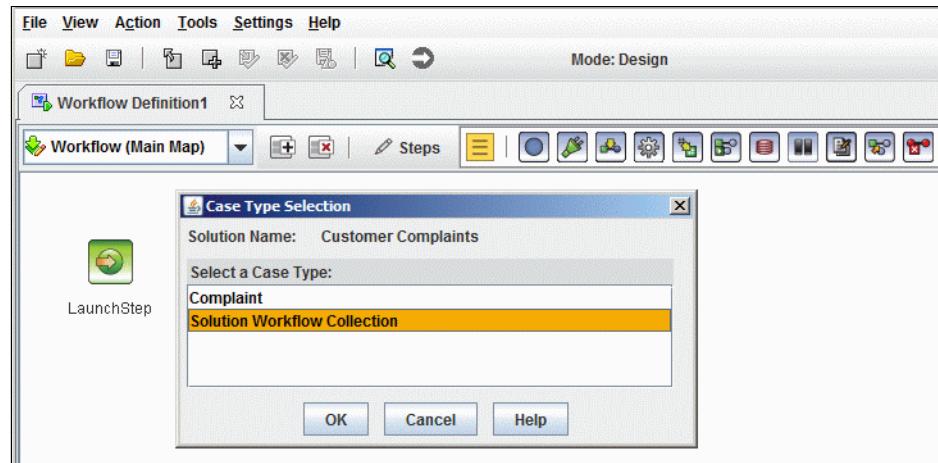


Figure 11-32 Editing the Solution Workflow Collection

- d. Click **File** → **Insert** to import the existing FileNet BPM process either from the repository or from local drive.
- e. Select your workflow process definition.

In the complaint management example, the company already has an existing workflow process definition for reviewing the safety of the products against a strict set of standards. Therefore, select **Workflow from Repository** as shown in Figure 11-33.

Tip: The Scenarios options in the menu (Figure 11-33) are for use with Process Simulator.

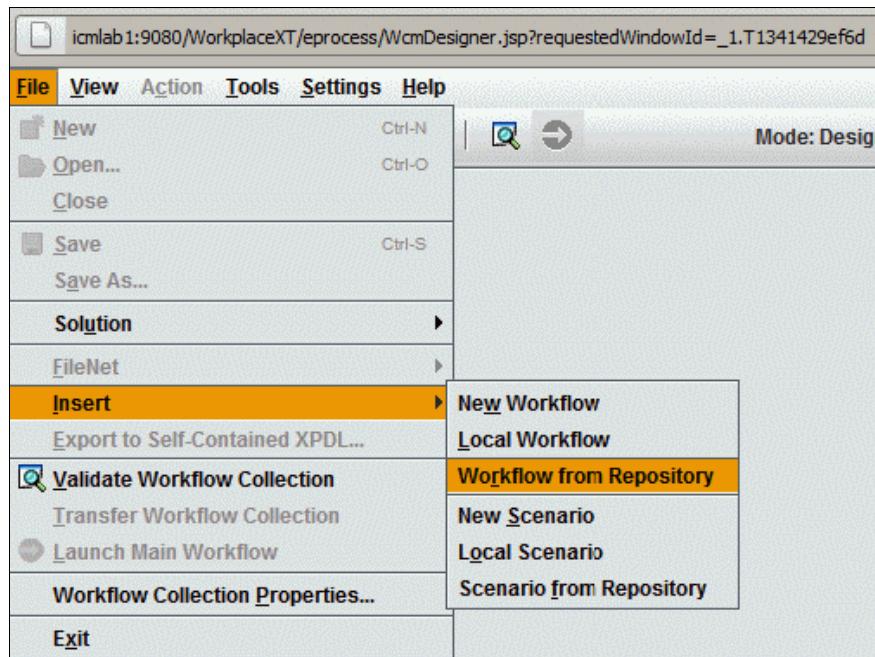


Figure 11-33 Inserting an existing workflow for reuse

- f. Import this workflow process definition into the complaints management solution, and use it to create a manual task inside the solution. This task is started at the user's discretion for complaints that are related to product safety. The review is carried out by an investigator who specializes in product safety.

- g. Upon import, the system prompts you about updating the base class. Confirm the update by clicking **Yes**. This process changes the base class of this workflow process to CaseWorkObject (Figure 11-34). Only then can this workflow process definition be used as a task in the Case Manager Builder.

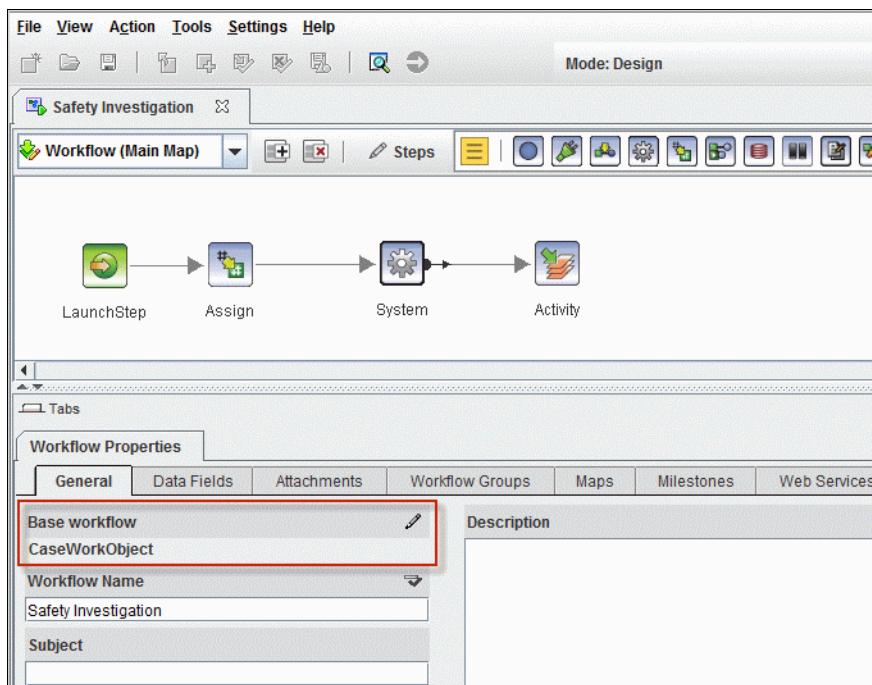


Figure 11-34 Base Workflow class must be CaseWorkObject

This class conversion affects only the imported workflow process definition. The original definition is not changed.

Consideration: If you do not intend to use the imported workflow process directly as a case task, you do not need to change the base class. You can instead refer to it in alternative ways such as in a **Create** or **WaitForCondition** system step.

- h. If there are user interactions in the imported workflow process definition, change the Step Processor to use the appropriate page in the IBM Case Manager solution.

This page is something like XX_CmAcmSTEP_DEFAULT_PAGE, where XX is the solution prefix. See Figure 11-35.

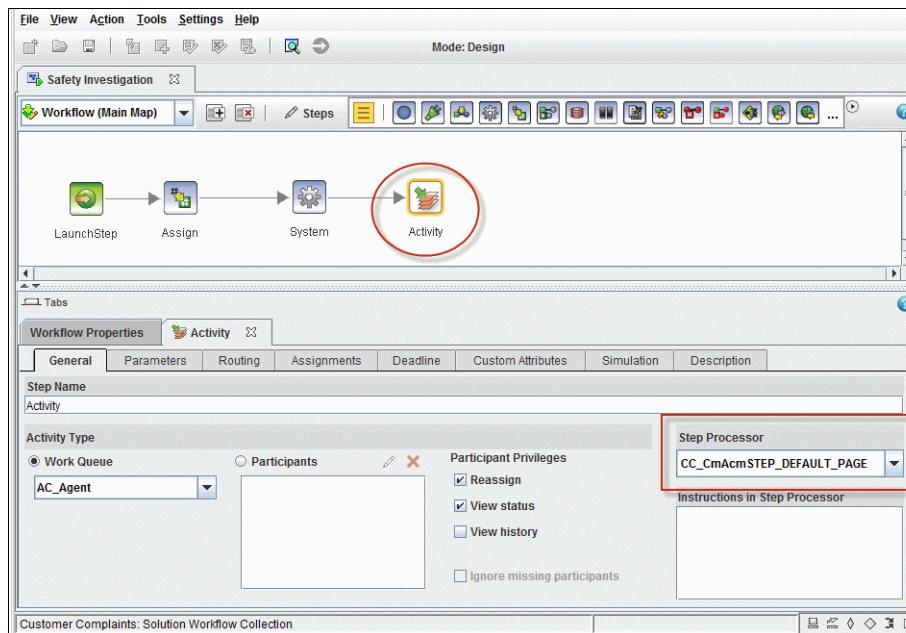


Figure 11-35 Updating the step processor

The start step is automatically changed to use the IBM Case Manager default page, in order that the start information can be seen in the Case Manager Client.

2. Depending on your solution you might need to change some configuration in the imported workflow process definition.

As shown in Figure 11-36, workflow configuration is automatically imported from the solution. The import includes queue definitions, in-basket definitions, roster definitions, and event logs. The imported workflow process definition is brought in the AC_agent queue, whereas all the CC_ queues were defined in the Case Manager Builder for the Customer Complaint solution.

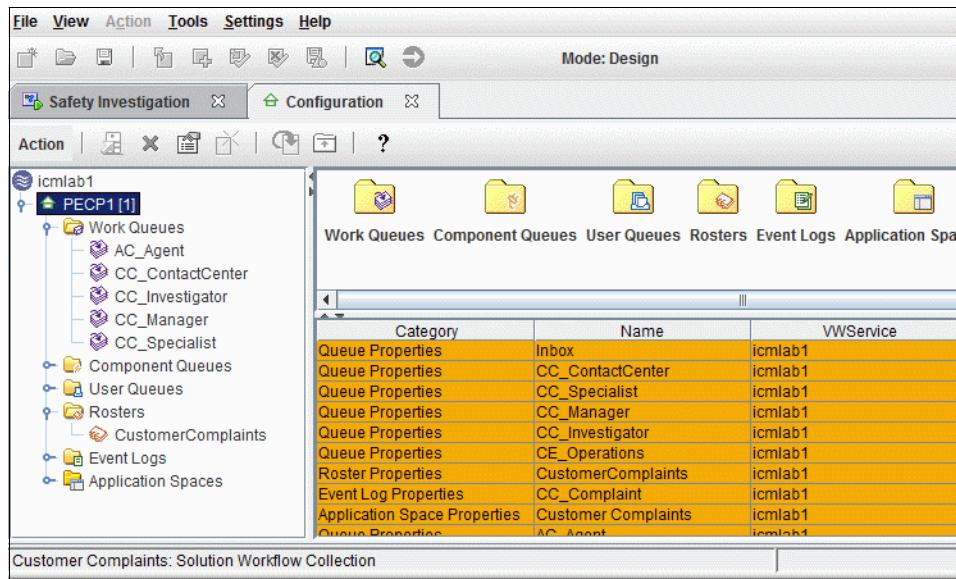


Figure 11-36 The IBM Case Manager solution configuration is imported

The example includes a work queue for the investigators who participate in the complaints reviews, who are labeled CC_Investigator (Figure 11-37). Therefore, change the configuration of the imported workflow process definition to use this work queue, and ensure the in-baskets are set up for the investigator role. This configuration allows the investigators to use the Case Manager Client to process their in-baskets.

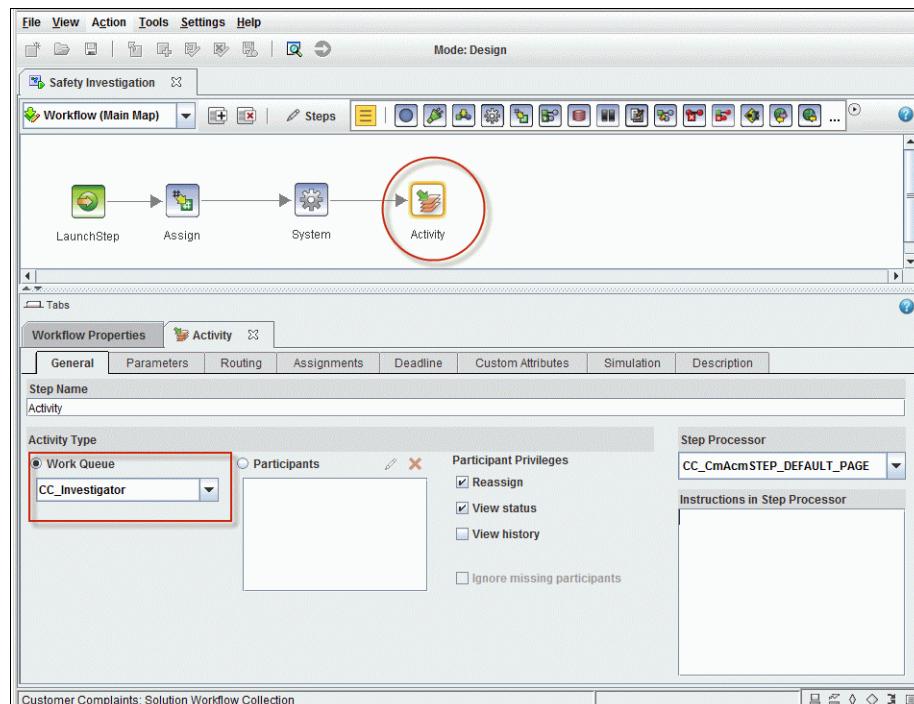


Figure 11-37 Changing the work queue for the user

The process definition is now ready to be used as a task in the Case Manager Builder.

Validate the changes, then save and close the solution by clicking **File → Solution → Save and Close**.

3. Create an IBM Case Manager task with the imported BPM workflow:
 - a. Open Case Manager Builder.
 - b. Edit the solution and go to the task page for Case Type. Select **Add task**, with the **Add FileNet BPM Process** option (Figure 11-38).

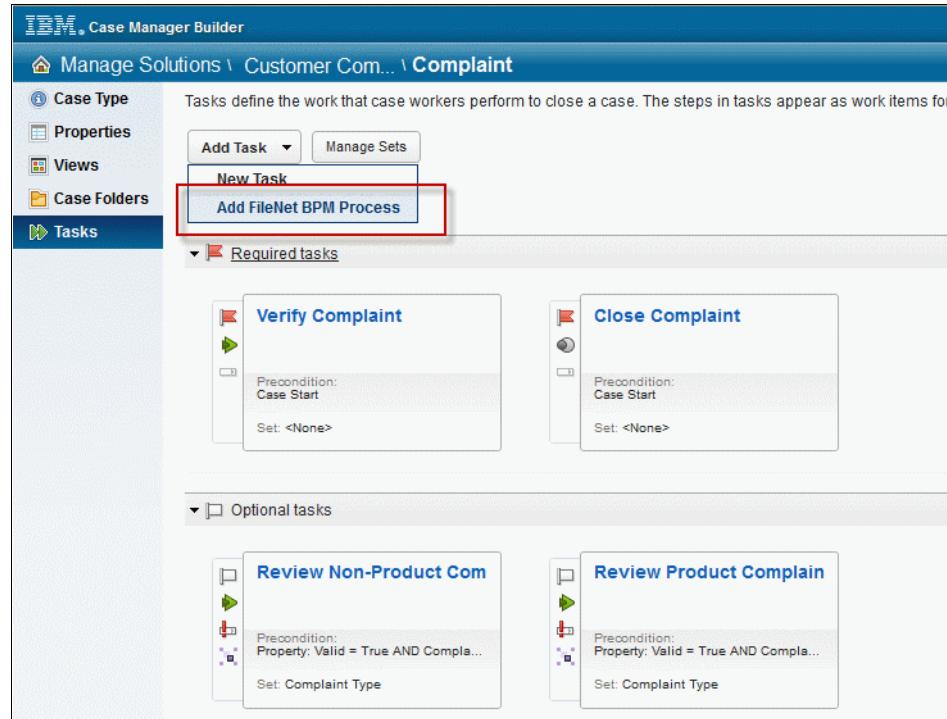


Figure 11-38 Creating the IBM Case Manager task by reusing BPM workflow

- c. Configure the new task.

For the example, the task is to investigate product safety, and it must be available to the case worker only if the complaint is valid. The task is started manually based on the judgment of the case worker.

Figure 11-39 shows the task configuration:

- Name: Investigate Product Safety
- Task starts: **Manually**
- Required: Not selected

The screenshot shows the 'Add Task' dialog box with the following fields filled in:

- Name:** Investigate Product Safety
- Unique Identifier:** CC_InvestigateProductSafety
- Description:** (empty text area)
- Task starts:** Manually
- Required:**
- Assign to set:** <None>

At the bottom right are buttons for Back, Cancel, and Next.

Figure 11-39 Adding task dialog as usual

- d. Select the appropriate workflow from the Solution Workflow Collection. In the example, select **Safety Investigation** as shown in Figure 11-40.

The screenshot shows the 'Select Process' dialog box with the following interface:

- Select Process**: A section for selecting a process to reuse.
- Process Name**: A dropdown menu showing "Safety Investigation".
- Description**: A table showing the selected process details:

Process Name	Description
Safety Investigation	(empty)

- At the bottom right are buttons for Back, Cancel, and Next.

Figure 11-40 Selecting from the Solution Workflow Collection

4. Configure mapping between workflow data fields and case data. Based on this information, the workflow fields are initialized with the values from the case data at the start of the process. These values are mapped back to case data at the end of the execution of the process. This process is done in the UpdateCase submap. Not all workflow data fields must be mapped to case data.

For the example, map the process data fields with case type properties as follows (see Figure 11-41):

- CaseNumber with Case Number
- Name with Customer Name
- StartDate with Complaint Receive Date

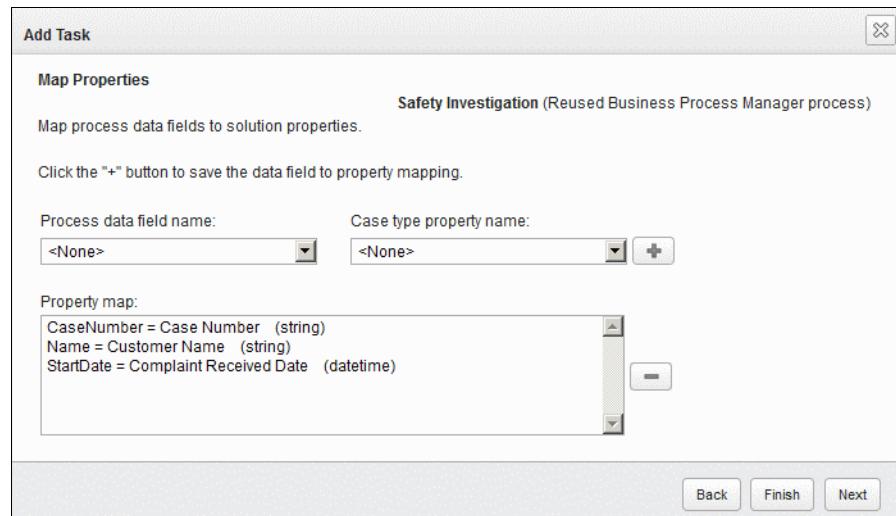


Figure 11-41 Mapping the workflow data fields with case data

Click **Finish** and then the reuse configuration is completed.

Because this is a reused FileNet workflow definition, the task cannot be edited by using the Step Editor. Therefore, the Editor icon is disabled. See Figure 11-42.



Figure 11-42 New IBM Case Manager task with the Step Editor icon disabled

5. Validate, save, and redeploy the solution.
6. Test the new task in the Case Manager Client by creating a product complaint, making sure that, after it is validated, it arrives at the product specialist in-basket.

Figure 11-43 shows the new task, Investigate Product Safety, is available and ready to start.

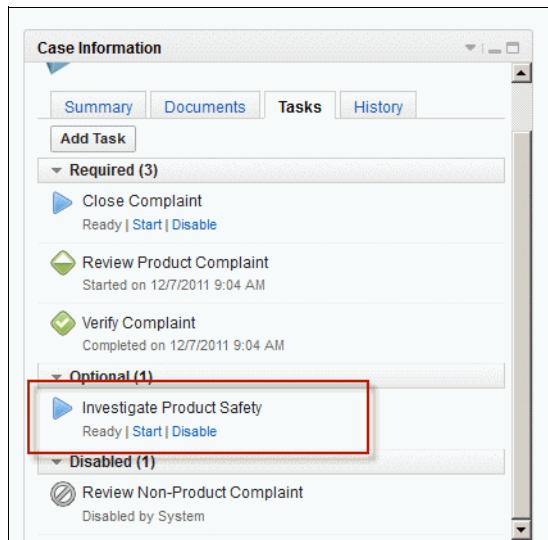


Figure 11-43 New task ready to start

After the case worker starts the task, the investigator runs the task according to the defined process until it completes. The process can be seen in the History tab of the case as shown in Figure 11-44.

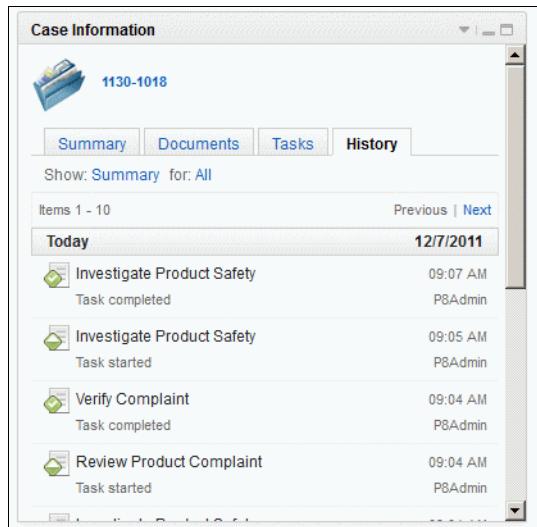


Figure 11-44 Task completed

Tip: An alternative way to use an existing FileNet process with IBM Case Manager tasks is to start the FileNet process as a separate process from within the IBM Case Manager task. Use this method for situations where the existing workflow process does not involve any user interactions.

This method can be done by using the Process Designer to edit the solution task. Add a System Step with a Create Workflow function to start the existing FileNet workflow. If required, use the WaitForCondition function to wait until that workflow reaches the point where the condition is met.

11.4.2 Native tasks and tasks by using reused processes comparison

There are a few differences between the native tasks that are created by using Case Manager Builder, and the ones that reuse a FileNet process.

The reuse mechanism is not a way of converting a FileNet BPM process into an IBM Case Manager process. These differences exist between the native IBM Case Manager tasks and those from reusing BPM processes:

- ▶ An IBM Case Manager task with a native IBM Case Manager process can be edited in the Case Manager Builder Step Editor. An IBM Case Manager task with a reused FileNet process cannot.
- ▶ Case data is updated at each step of a task with a native IBM Case Manager process. However, case data is updated only at the end of the process for tasks with reused FileNet processes.
- ▶ They have different base classes. When a new task is created with a reused BPM process, the task process inherits the logic from the reuse process as dictated by the FileNet BPM inheritance feature. Hence the base class of this task is the reused process, whereas the base class of a native IBM Case Manager task is CaseWorkObject. Also, the main map of the task gets replaced by the main map of the reuse process, and is read-only.

Figure 11-45 shows the IBM Case Manager task, which inherits from the reused BPM process in the Solution Workflow Collection.

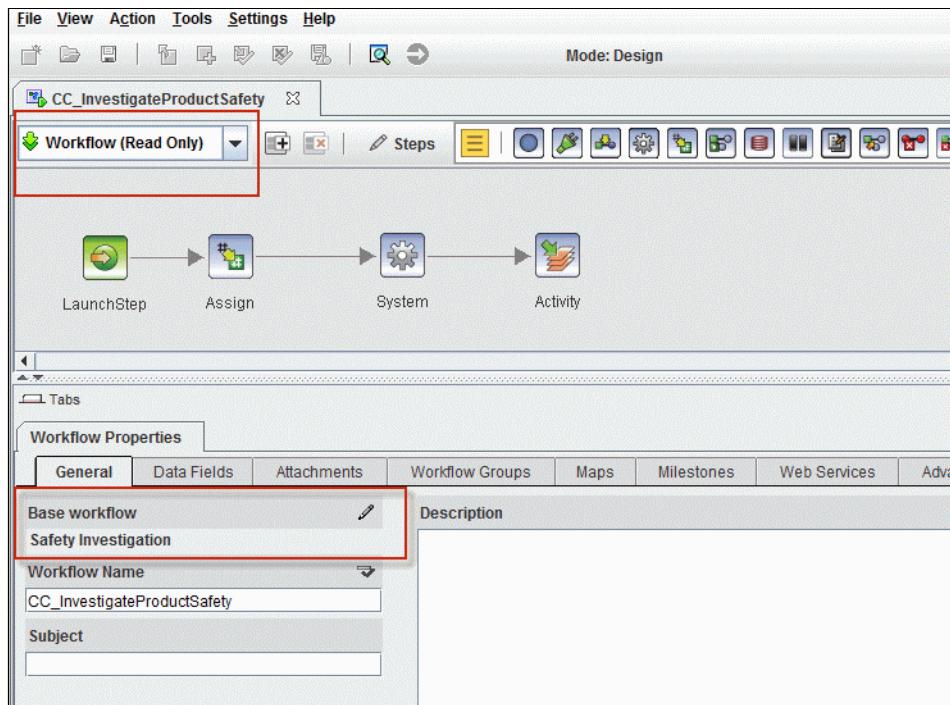


Figure 11-45 IBM Case Manager task inherits from reused BPM process

11.5 Multilingual support

Companies need to develop solutions that can be used in different geographic regions with support for multiple languages. IBM Case Manager has multilingual support throughout the solution. A single solution can be translated to more than one language and locale.

IBM Case Manager 5.1.1 supports 23 languages, including right-to-left languages such as Hebrew and Arabic

This section addresses what can be localized, and how to configure individual components for multilingual support in IBM Case Manager.

11.5.1 Use cases for Case Builder designer

A case solution designer can localize the following items:

- ▶ Rename a space in a specific language
- ▶ Rename a page in a specific language
- ▶ Rename a widget title in a specific language
- ▶ Rename a button in a work item toolbar in a specific language
- ▶ Redeploy a solution and still see previously localized content

11.5.2 Use cases for Case Client user

A case user can see the following items:

- ▶ See inbox column names in a specific language
- ▶ See the inbox roles and tabs in a specific language
- ▶ See the case type name in a specific language
- ▶ See the document and folder classes in a specific language
- ▶ See the document properties names in a specific language
- ▶ Edit and see FileNet eForms forms in a specific language
- ▶ Edit and see IBM Forms forms in a specific language

11.5.3 Business Space localization

The following objects can be localized in Business Space:

- ▶ Space name
- ▶ Page name
- ▶ Widget name of both IBM Case Manager built-in and custom widgets

When you create a space, page, or widget, the item gets a language-specific name. The name can come from a definition that is contained within the system, or it can be a name that you provide. If you provide a name in one language, that name serves as a default name for the other languages. The default name is just a placeholder until a language-appropriate name replaces it. When you rename the spaces, pages, and widgets in Case Manager Client, Business Space overwrites the default name with the new name, but only for that specific language. The other languages are not affected.

For example, if your browser is set to English and you create a space, the name that you give it applies to all languages. If you change your browser language to Chinese and rename the space (Figure 11-46), all users who set their browser language to Chinese and open that space see the Chinese name. All users who have their browser language set to any other language see the original English name because it is the default. To support multiple languages, you must change browser languages and rename the space in each language.

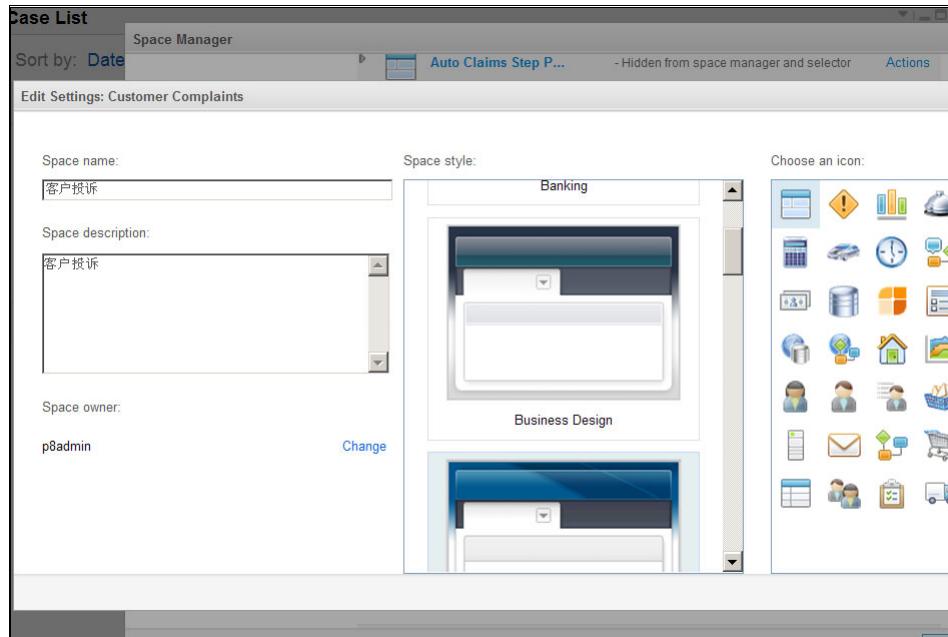


Figure 11-46 Setting the space name in Chinese

For more information about how to configure Business Space localization, see the following links:

- ▶ http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r5mx/index.jsp?topic=/com.ibm.wbpm.bspace.dev.doc/doc/developing_widgets/localization.html
- ▶ <http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/index.jsp?topic=%2Fcom.ibm.casemgmt.design.doc%2Facmdc034.htm>

11.5.4 Process Engine element localization

The following objects can be localized in Process Engine:

- ▶ Workflow names (work class name)
- ▶ Roster names

- ▶ Queue names
- ▶ Milestone names
- ▶ Step name, step instruction, step parameters, and responses
- ▶ Field names (user-defined data fields, system fields, attachments, workflow groups)

You can enable users to view user-authored items, such as step responses, field names, and milestone names, in the language of their current locale. You can translate your solution artifacts that are stored in Process Engine by exporting and translating an XLIFF file by using Process Configuration Console. This file contains the strings for the Process Engine artifacts. This feature is available for FileNet P8 applications only.

XLIFF is an XML-based file format that provides a standard format for data to be localized into a target language. Using a translation application that supports XLIFF 1.1, localization engineers can translate the items in each file for the specific language. They can then deploy the localized files on the Application Engine server.

Tip: To reduce the incidence of syntax errors in the translated file, use an XLIFF translation application to translate the file, rather than a text editor. After you translate the file, use the Verify XLIFF file option in Process Configuration Console to check the file for errors. When you edit a PCC/XLIFF file, do not use Windows Notepad or other tools that add a byte order mark (BOM) in front of the PCC/XLIFF file

Fig 10-47 shows creation of XLIFF file to translate Process Engine elements into Japanese.

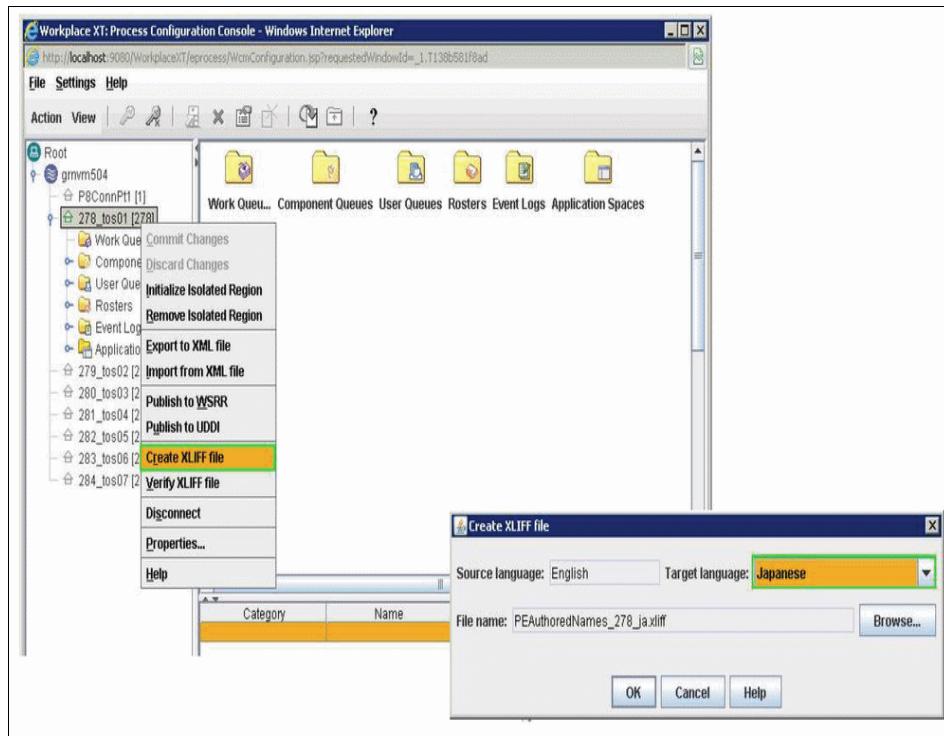


Figure 11-47 Creating an XLIFF file for translation into Japanese

For more information about how to configure Process Engine localization, see:

<http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/index.jsp?topic=%2Fcom.ibm.casemgmt.design.doc%2Facmdc036.htm>

11.5.5 Content Engine element localization

The following objects can be localized in Content Engine:

- ▶ Document and folder classes
- ▶ Document properties
- ▶ Choice lists

All system classes and properties are localized when you install a localized IBM FileNet Content Engine language pack. You can add multiple language packs, containing all released languages, or a language pack that contains only a single

language. Language packs contain Content Engine DLL and XML files that contain translated display names of document classes and properties of add-on features. You can then import these XML files into an object store.

After you install a language pack, localize new objects that you create in different locales by using FileNet Enterprise Manager (FEM).

You can translate the solution artifacts that are stored in Content Engine by translating the display name of case types, document types, tasks, case properties, and choice list items by using FEM. Figure 11-48 shows the Complain Category choice list in Spanish

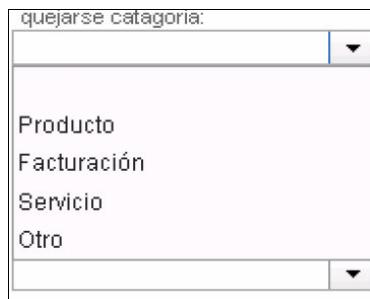


Figure 11-48 Complain Category Choice list translated into Spanish

For more information about how to configure Content Engine localization, see:

<http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/index.jsp?topic=%2Fcom.ibm.casemgmt.design.doc%2Facmdc038.htm>

11.5.6 IBM Content Manager element localization

The following objects can be localized in IBM Content Manager:

- ▶ Item types
- ▶ Document attributes

You can provide translations for numerous Display Name fields in IBM Content Manager if you first create a language definition. A language definition consists of a language code and a language name.

Translate the item type, attribute, and attribute group names in IBM Content Manager so your solutions can display these objects in the appropriate language for the worker.

Figure 11-49 shows the **First Name** attribute name is translated in French.

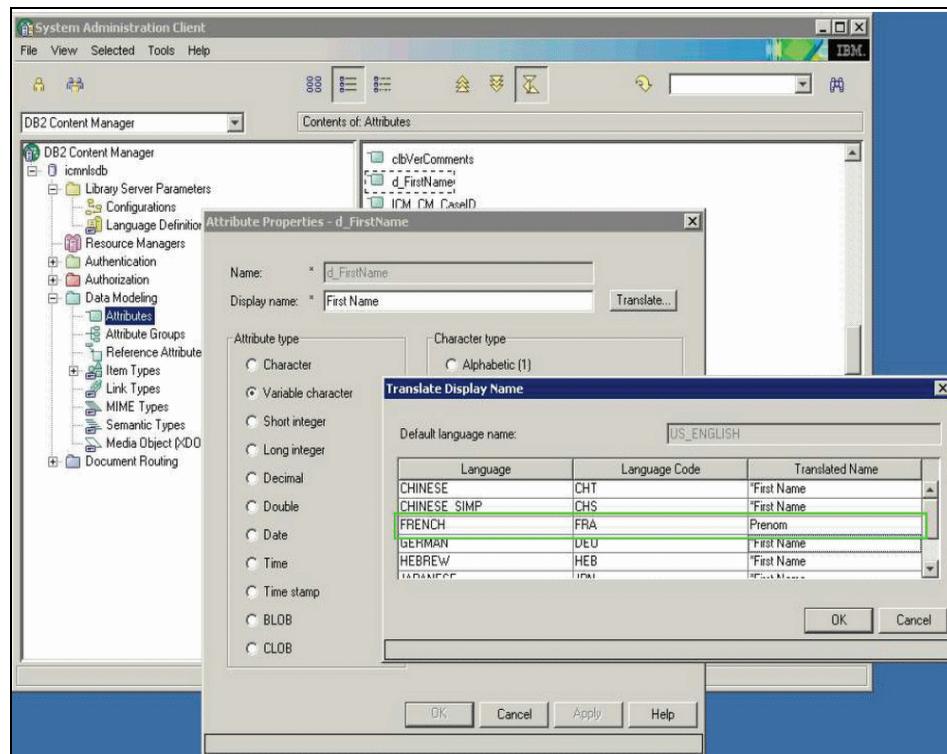


Figure 11-49 Translating the First Name attribute name into French

For more information about how to configure IBM Content Manager localization, see these links:

- ▶ <http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/index.jsp?topic=%2Fcom.ibm.casemgmt.design.doc%2Facmdt033.htm>
- ▶ <http://pic.dhe.ibm.com/infocenter/cmgt/v8r4m0/index.jsp?topic=%2Fcom.ibm.administeringcm.doc%2Fc1sh0010.htm>

11.5.7 FileNet eForms element localization

The following objects can be localized in FileNet eForms:

- ▶ Labels, help text, text objects
- ▶ Regional settings (number formats, date formats, currency, and so on) and bidirectional error messages

Keep in mind the following considerations when you configure localization in FileNet eForms:

- ▶ The interactive user interface automatically adapts to browser/client locale.
- ▶ Labels, help text, and text objects are statically defined in the template.
- ▶ Regional and bidirectional settings are controlled by the form's statically defined locale.
- ▶ Business Analysts can now configure the Form widget to use a localization proxy document.
- ▶ The form is now rendered by using the localization proxy document that allows you to select the template corresponding to a specific locale.
- ▶ The data document is statically bound to the specific template used to create it. The browser locale does not affect the localized content of the data document.

Figure 11-50 shows localization proxy document with a French form template reference.

```
1  <?xml version="1.0"?>
2  <localizationProxy>
3      <default>
4          <version>released</version>
5          <id>(785747A3-1F92-4E4D-B4DF-F7001B297A9A)</id>
6      </default>
7      <mappings>
8          <mapping locale="en">
9              <version>released</version>
10             <id>(785747A3-1F92-4E4D-B4DF-F7001B297A9A)</id>
11         </mapping>
12         <mapping locale="fr-CA">
13             <version>released</version>
14             <id>(3F44D448-2B59-457F-8B3A-264761B29E62)</id>
15         </mapping>
16     </mappings>
17 </localizationProxy>
```

The XML code defines a localization proxy document. It starts with a default template entry for the English locale ('en'), which has a specific version and ID. Following this, there is a mapping for the French Canadian locale ('fr-CA'), also with a specific version and ID. Both entries are enclosed in a 'mappings' block, which is itself within a 'localizationProxy' block. Two sections of the code are highlighted with green boxes: the entire 'default' block and the entire 'mappings' block, including its contents.

Figure 11-50 Form localization proxy document

Figure 11-51 shows configuring the Form widget to use a localization proxy document.

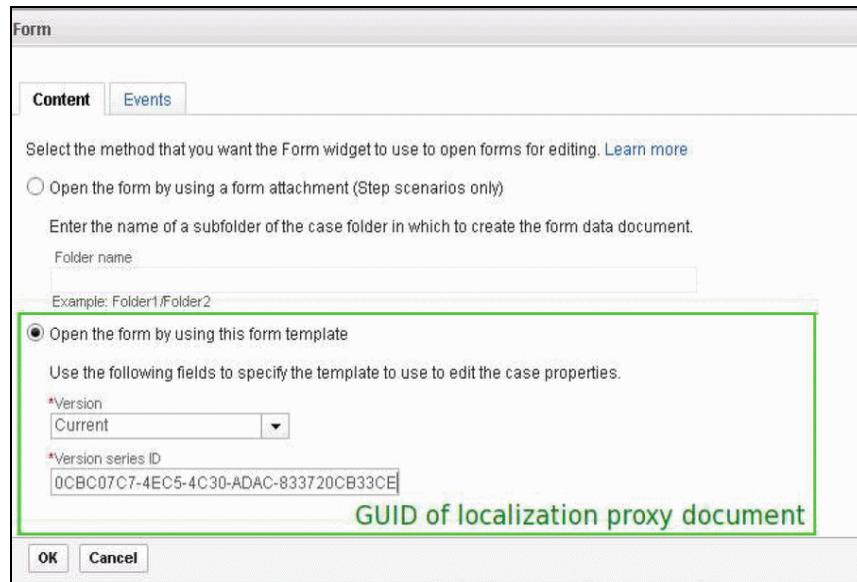


Figure 11-51 Specifying the localization proxy document

For more information about how to configure FileNet eForms localization, see:

<http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/index.jsp?topic=%2Fcom.ibm.casemgmt.design.doc%2Facmdt032.htm>

11.5.8 IBM Forms element localization

The following objects can be localized in FileNet eForms:

- ▶ Labels, help text, text objects
- ▶ Regional settings (such as number formats, date formats, currency) and bidirectional error messages

Keep in mind the following considerations when you configure localization in IBM Forms:

- ▶ The interactive user interface automatically adapts to browser/client locale.
- ▶ Labels, help text, and text objects are statically defined in the template.
- ▶ Regional and bidirectional settings are controlled by the form's statically defined locale.

- ▶ Business Analysts can now configure the Form widget to use a localization proxy document.
- ▶ The form is now rendered by using the localization proxy document that allows you to select the template corresponding to a specific locale.
- ▶ The data document is statically bound to the specific template used to create it. Browser locale does affect the localized content of the data document.

For example, you can create English and Chinese versions of the form templates and store these templates in your target object store. You can then map the translated form templates to different locales by using a localization proxy document as shown in Figure 11-50 on page 462. The localization proxy document contains the version series IDs of the translated form templates. Configure your case management application to use the localization proxy document shown in Figure 11-51 on page 463. This configuration allows your case management application users to view and work with the forms in the language of their choice.

After you complete this task, your users can set their browser locale to Chinese or English, and open the form in the language in which their browser locale is set.

For more information about how to configure IBM Forms localization, see these links:

- ▶ http://www.lotus.com/1dd/1fwiki.nsf/xpDocViewer.xsp?lookupName=Extensible+Forms+Description+Language+XFDL+8.0#action=openDocument&res_title=Locale_information_XFDL_8&content=pdcontent
- ▶ <http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/index.jsp?topic=%2Fcom.ibm.casemgmt.design.doc%2Facmdt032.htm>

11.5.9 Toolbar, Case toolbar, and Work Item toolbar localization

The following labels can be localized in FileNet eForms:

- ▶ Button labels
- ▶ Menu items labels

All labels are created in the locale that is used when creating the page. Renaming the button/menu label in a different locale affects only the label under that particular locale. Regional and bidirectional settings are controlled by the form's statically defined locale. This configuration is consistent with Business Space mechanism for space/page/widget names.

For example, if a button is created in French locale, the default is the French label for all language locales:

- ▶ When the label is updated/edited in the Italian locale, it displays the Italian label for Italian locale, and the French label for English and all other languages.
- ▶ When the label is update/edited in the Chinese locale, it displays the Italian label for the Italian locale, the Chinese label for the Chinese locale, and the French label for English and all other languages.

Figure 11-52 shows configuring the toolbar button in German.

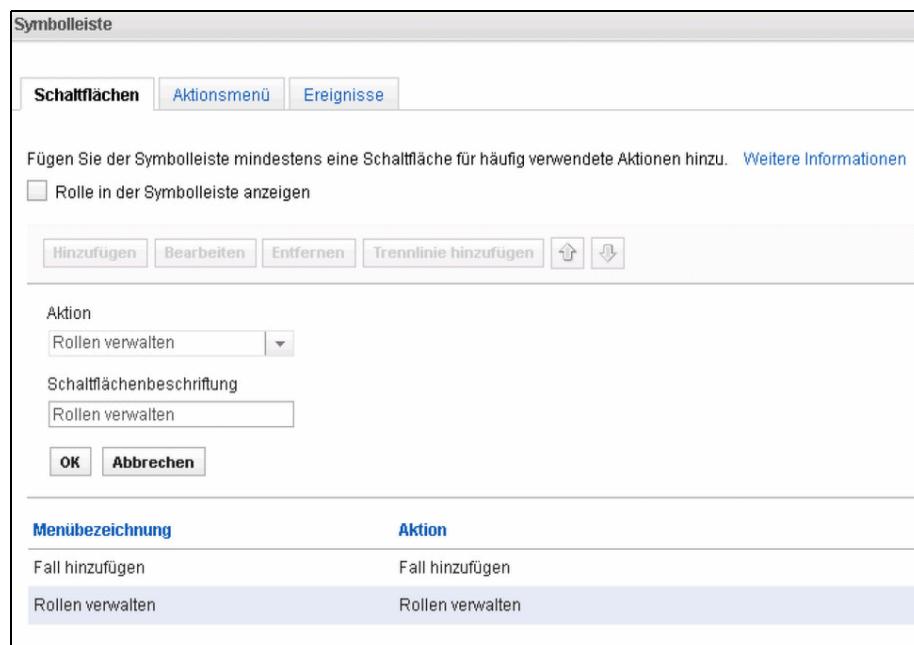


Figure 11-52 Setting the toolbar button in German

Tip: You cannot translate the **Add Case** button and the **Manage Roles** label in the Toolbar widget. However, you can remove the default **Add Case** button, create the button again, and enter a name of your choice. You can also remove the **Manage Roles** button, create the button again, and rename it.

You also cannot translate the Add Task button and the **Comments** button in the Case Toolbar widget. However, you can remove the **Add Task** button, create the button again, and give it a different name. You cannot replace the **Close** button and create the button again

For more information about translating button and menu item labels in the Toolbar and the Case Toolbar widgets by using Case Manager Client, see:

<http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/index.jsp?topic=%2Fcom.ibm.casemgmt.design.doc%2Facmdt034.htm>

11.5.10 Multilingual limitations

Multilingual support has these limitations:

- ▶ Unified Inbox column names are displayed in a single locale.
- ▶ Solution names cannot be localized. However, Space names, which are what case workers see, can be localized to multiple languages/locales.



Round-tripping workflow editing

This chapter provides details about how to perform some of the typical round-tripping actions.

This chapter includes the following sections:

- ▶ Introducing round-tripping
- ▶ Adding CE_Operations in a round-tripping edit
- ▶ Creating more in-baskets
- ▶ Exposing error handling in the Case Manager Client

12.1 Introducing round-tripping

Round-tripping in IBM Case Manager is the process of developing parts of the solution in Case Manager Builder, then in Process Designer, and then back to Case Manager Builder.

For some solutions, you can model an entire solution in Case Manager Builder and not need round-tripping for further development. For other solutions, some level of system processing is necessary and requires at least one round-tripping action.

In IBM Case Manager 5.1.1, Process Designer is integrated with Case Manager Builder. You now can access Process Designer within Case Manager Builder at the Solution and Task levels. Table 12-1 shows the differences between the two integration levels.

Table 12-1 Differences between Solution level and Task level integration

Solution Level Integration	Task Level Integration
Process Designer handles the checkout and check in of the solution artifacts	Case Manager Builder handles the checkout and check in of the solution artifacts. Process Designer operates only on the reservation objects of the solution artifacts
User can switch between the case type workflow collections and solution workflow collection	User can only switch between process maps that belong to the same case type workflow collection
User can edit the solution configuration objects such as Views → Configuration, Views → Roles, and Views → In-baskets are enabled	User cannot edit the solution configuration objects such as Views → Configuration, Views → Roles, and Views → In-baskets are disabled

12.2 Adding CE_Operations in a round-tripping edit

IBM Case Manager is built on the core IBM FileNet P8 Platform. This platform offers various enterprise level business processing capabilities by using the Process Engine. This section addresses how to extend a solution to include some system level enhancements.

The component integrator is part of the toolset from Process Engine. It offers the capability to use a Java class with public methods to act as a component that processes work items in a queue. CE_Operations is an example of such a

component and is available with the IBM Case Manager product. CE_Operations refers to IBM FileNet Content Engine extended operations.

The following examples use the simple solution that was developed in Chapter 6, “Building a simple solution: Part 1” on page 159. It is extended to verify values from a document object. The use case is intended to ensure that a supporting document added to a case is correctly titled and has the correct case number. If the supporting document is invalid, it is removed from the case and placed in a specific folder for review. CE_Operations is used to access the document object, and demonstrate the round-tripping process.

12.2.1 Modeling in Case Manager Builder

For the use case, use these steps to model in Case Manager Builder as a business analyst:

1. Log in to Case Manager Builder:

`http://<server>:<port>/CaseBuilder`

2. Edit the **Customer Complaints** solution by pointing to the solution and clicking **Edit**.
3. Click **Case Types** and select the **Complaint** case type.
4. Click **Tasks**.
5. Create a task by selecting **Add Task → New Task**.

6. In the example, enter Verify Document for the name of the task. This is an automatic task and not required, so do not change the defaults as shown in Figure 12-1.

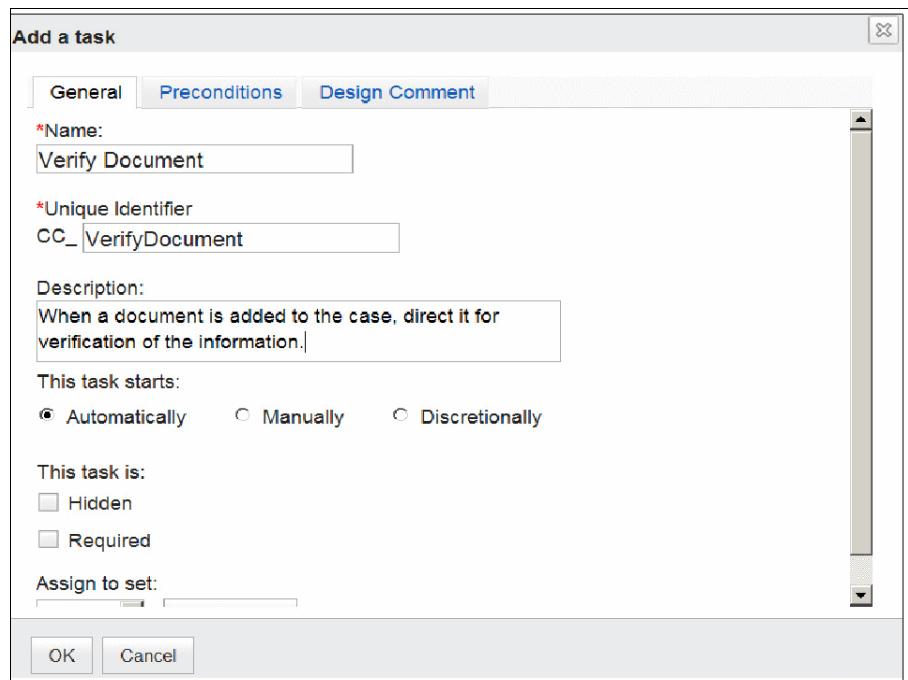


Figure 12-1 Creating a Verify Document task

7. Click the **Preconditions** tab.
8. Select the following check boxes and click **OK**:
 - A document is added to the case
 - Document of a type that is defined for this case
 - **Supporting Document** for the document type
 - Task is repeatable

These settings allow this task to run multiple times when a document of the Supporting Document type is added to the case (Figure 12-2).

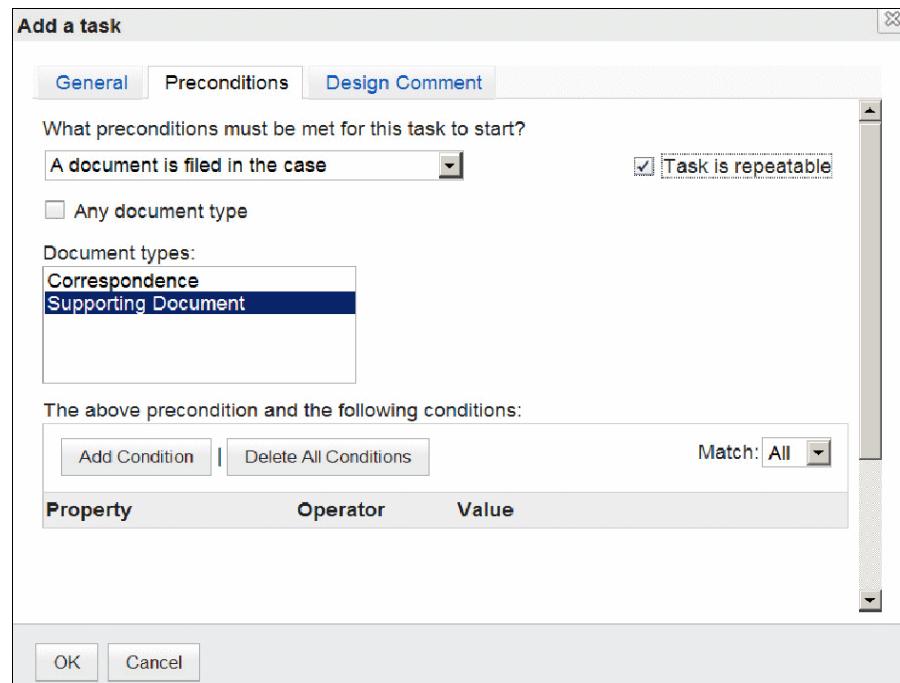


Figure 12-2 Applying preconditions to a task

9. Point to the created task as seen in Figure 12-3 and select the Step Editor icon (the icon just below the Pencil icon).



Figure 12-3 Created task with mouse over

10. Create a basic workflow. If you are not familiar with Step Editor, see 7.1, “Configuring workflow diagrams by using Step Editor” on page 214.

- a. Add the **Contact Center** role swimlane.
- b. Add a step. In this example, apply these step properties:
 - Name: Verify Document
 - Description: Determine applicability of document
 - Instruction: Observe the attached document to see if it is applicable to the case or if it should be removed.
 - Responses: Validated document, Remove document

You establish the basic elements as seen in Figure 12-4.

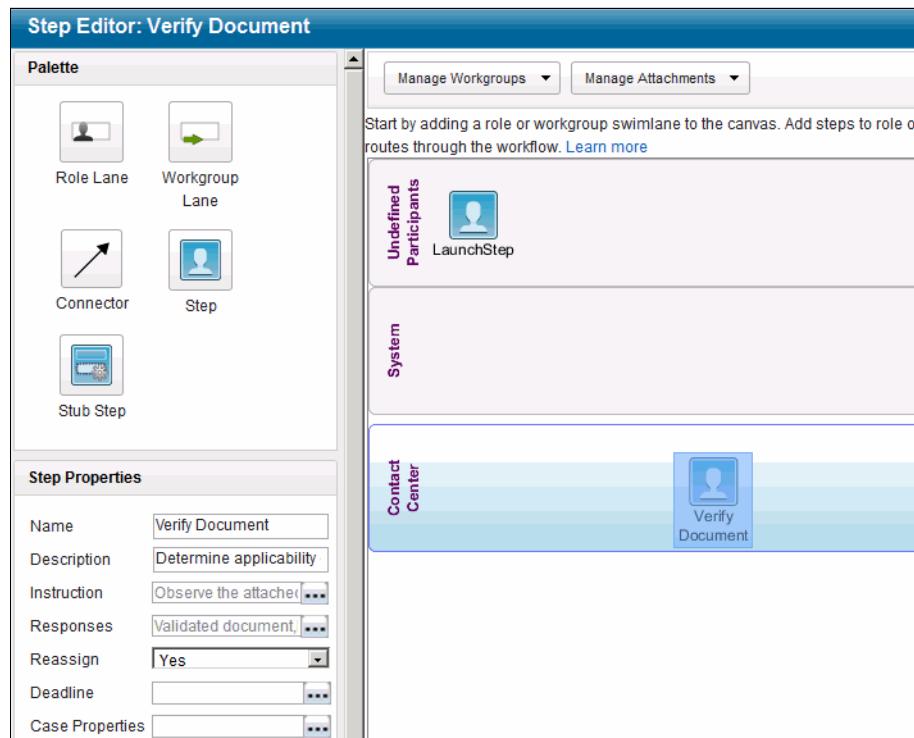


Figure 12-4 Start of a workflow

11. Add three **Stub Steps** to the workflow and place them on the **System** swimlane as shown in Figure 12-5. A Stub Step is a representation of a system step to be implemented by an IT Administrator.

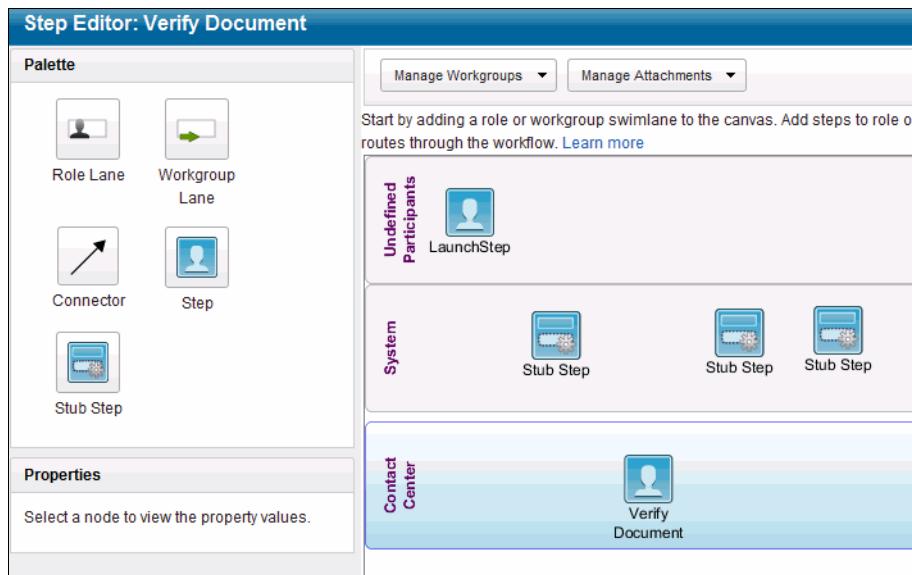


Figure 12-5 Workflow with Stub Steps added

12. Apply names to the Stub Steps to indicate actions to the IT Administrator.

The example uses Get Doc Properties, Unfile Document, and Set Doc Properties as seen in Figure 12-6.

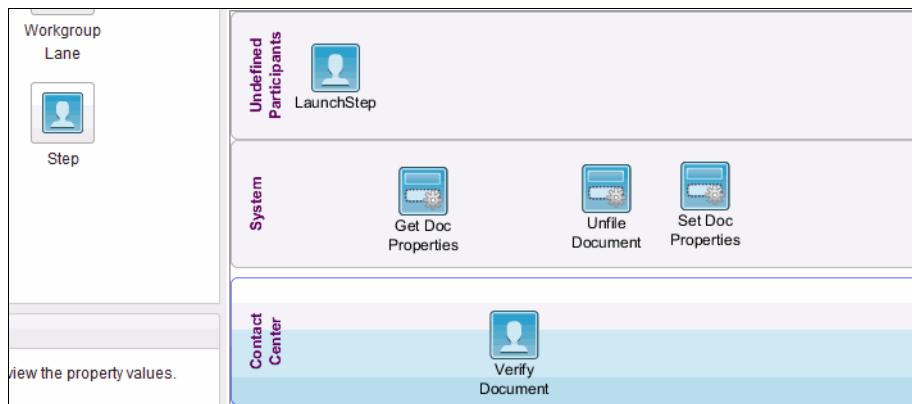


Figure 12-6 Names applied to Stub Steps

13. Apply connectors to all steps. In the example, For the route from **Verify Document** to **Unfile Document**, apply the response condition **Remove Document**.

For the route from **Verify Document** to **Set Doc Properties**, apply the response condition **Validated Document**.

The results are displayed in Figure 12-7.

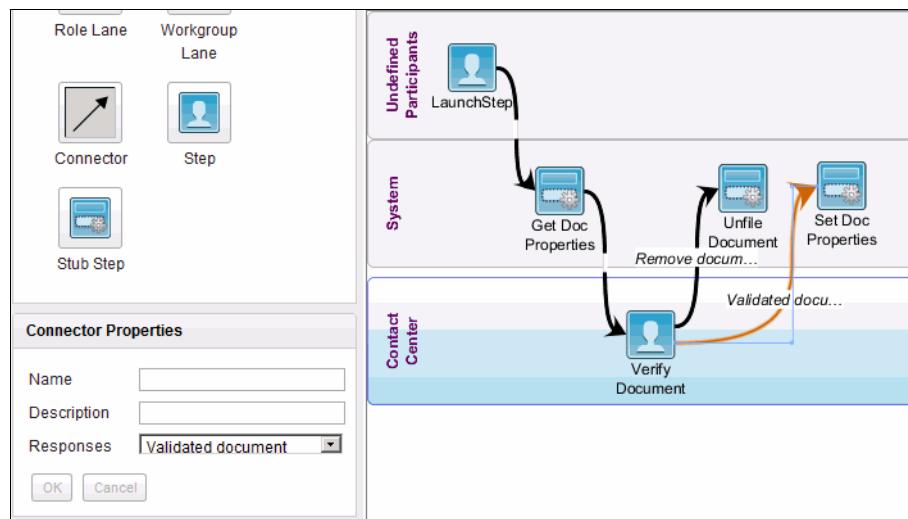


Figure 12-7 Applied connectors and conditional connectors

14. Add an attachment to the task workflow by clicking **Manage Attachments** and then **Add Attachment**. Enter Document for both Attachment Name and Prompt as seen in Figure 12-8, and click **OK**.

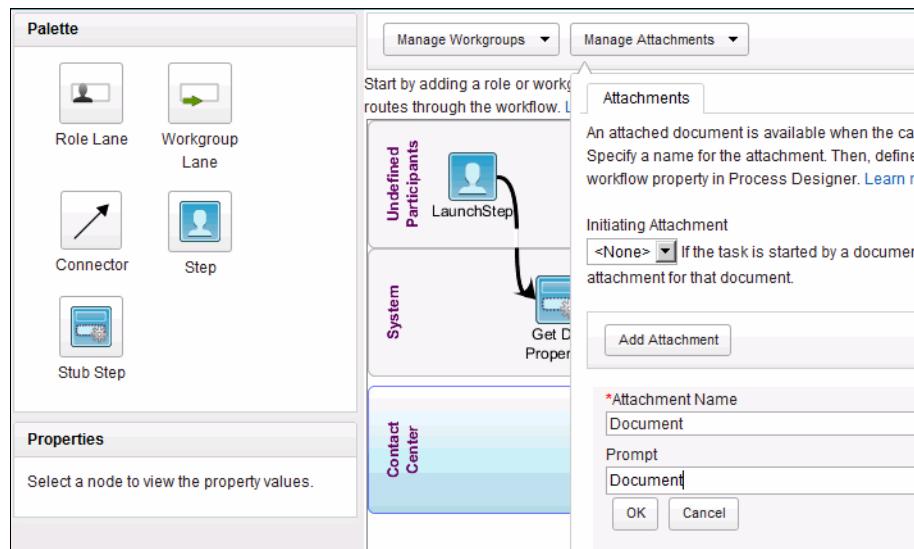


Figure 12-8 Adding an attachment to the workflow

15. Specify **Document**, the attachment previously added, as the Initiating Attachment as seen in Figure 12-9, and click **Close**.

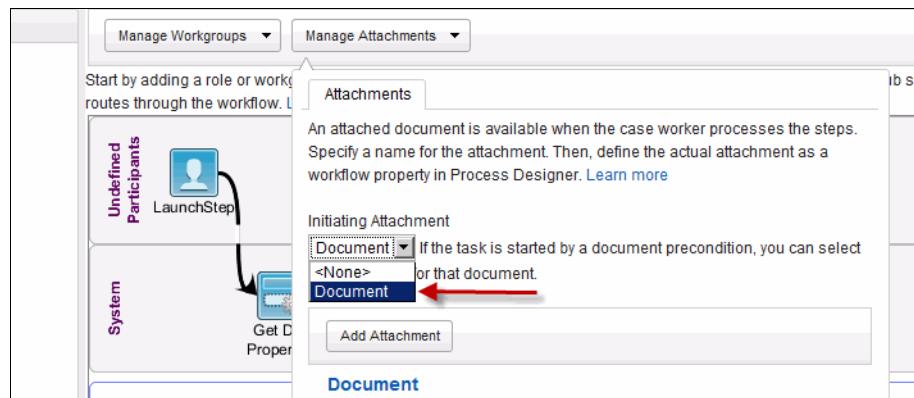


Figure 12-9 Applying the Document attachment as the initiating attachment

16. Return to the Verify Document step. In the Step Properties window, add **Document** as an attachment by expanding Attachments, clicking **Select Attachments**, and selecting the attachment called **Document**. The result is shown in Figure 12-10.

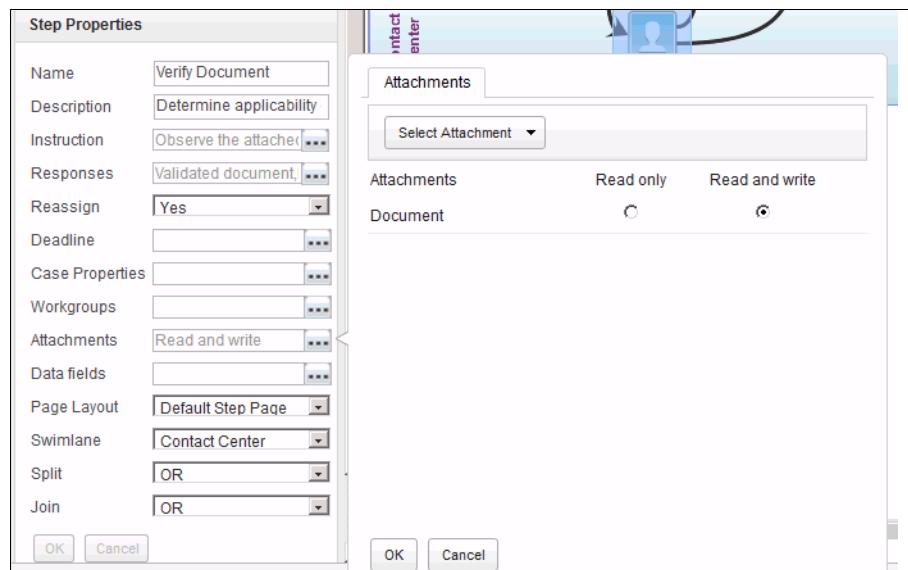


Figure 12-10 Applied Document as an attachment for the Verify Document step

17. Validate, save, and close the solution.

12.2.2 Round-tripping to Process Designer

Perform a round-tripping by editing a solution as an IT Administrator:

1. Access WorkplaceXT:
<http://<server>:<port>/WorkplaceXT>
2. Create a folder for later use. Go to your object store (**CMTOS** is used in this example) and click the icon to create a folder.
3. Enter the folder name **Dispose**, and click **Add** and then **OK**.

Your folder is added to the repository as shown in Figure 12-11.

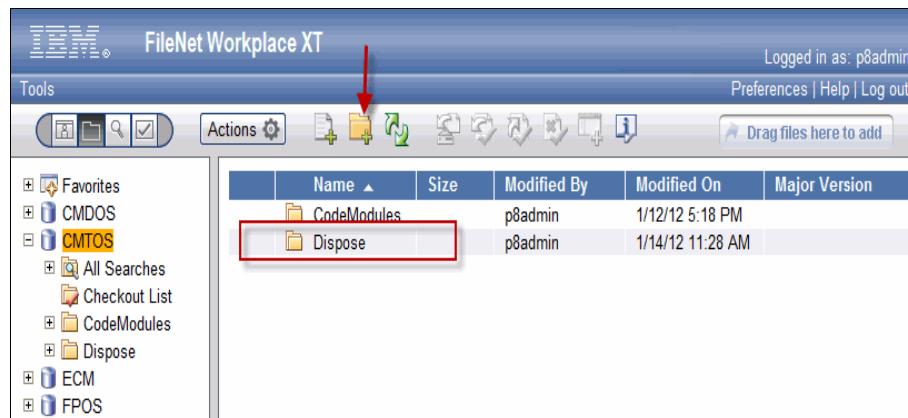


Figure 12-11 The folder added to the repository

4. Point to the Verify Document task as seen in Figure 12-12 and select the Process Designer icon (the second icon just below the Pencil icon).

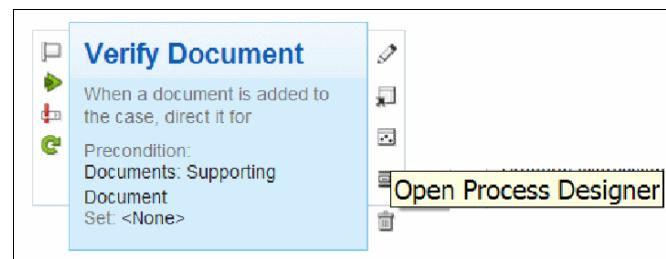


Figure 12-12 Selecting the Process Designer icon

5. Edit the workflow of the **Verify Document** Task created in Case Manager Builder.
6. Click **Workflow Properties → Data Fields** and add the following data fields:
 - Workflow_DocTitle (String)
 - Workflow_DocCaseNumber (String)
 - Workflow_DocCustomerNumber (String)
 - Workflow_DocCustomerName (String)

The result looks like Figure 12-13.

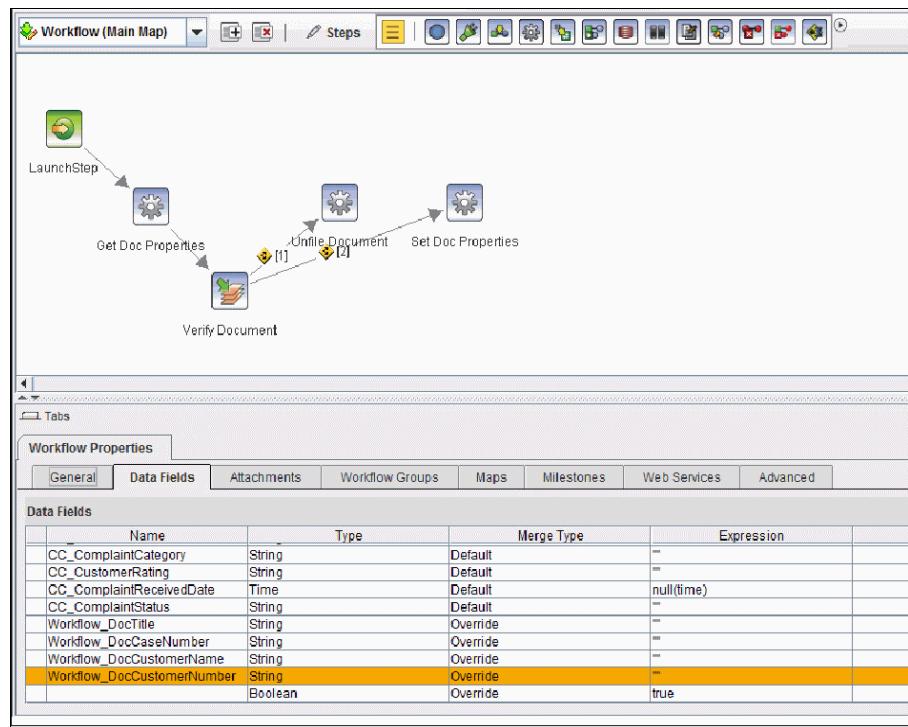


Figure 12-13 Setting the local data fields in the workflow

7. While still in Workflow Properties, click the **Attachments** tab.
8. Click to clear the **Document** attachment that was added in Case Manager Builder by step 14 on page 475. See Figure 12-14.

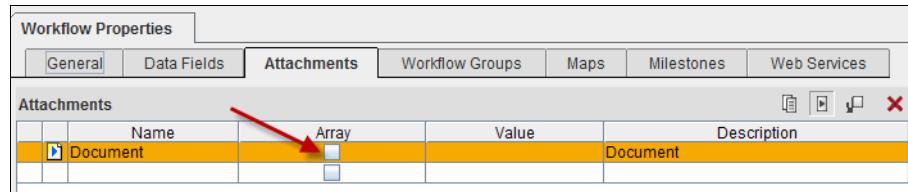


Figure 12-14 Clicking to clear Array for the attachment

9. Add an attachment by double-clicking the blank cell under **Name** and entering **Dispose**. This attachment is used in a later step.
10. Set the value of the attachment by double-clicking the blank cell under **Value** for the **Dispose** attachment.

11. Select the folder **CMTOS** → **Dispose**, which was created in step 2 on page 476.

There are two attachments as indicated in Figure 12-15.

Workflow Properties				
General		Data Fields	Attachments	Workflow Groups
			Workflow Groups	Maps
Attachments			Milestones	Web Services
Name	Array	Value	Description	
Document			Document	
Dispose		Dispose		

Figure 12-15 Added attachment **Dispose** pointing to a folder

12. Right-click the **Get Doc Properties** step and select **Change Step Type** → **Component Step**. You should see the system icon change to a component icon as shown in Figure 12-16.

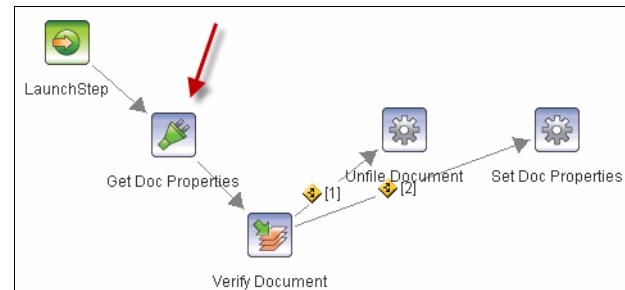


Figure 12-16 Changing the step type from system to component

13. To extract the properties from the document that initiated this task, click the component step.

14. Add four operations by completing these steps four times:

- a. Click the **Add** icon.
- b. Select **CE_Operations** for the component
- c. Select **getStringProperty**
- d. Click **OK**

The operations are added as seen in Figure 12-17.

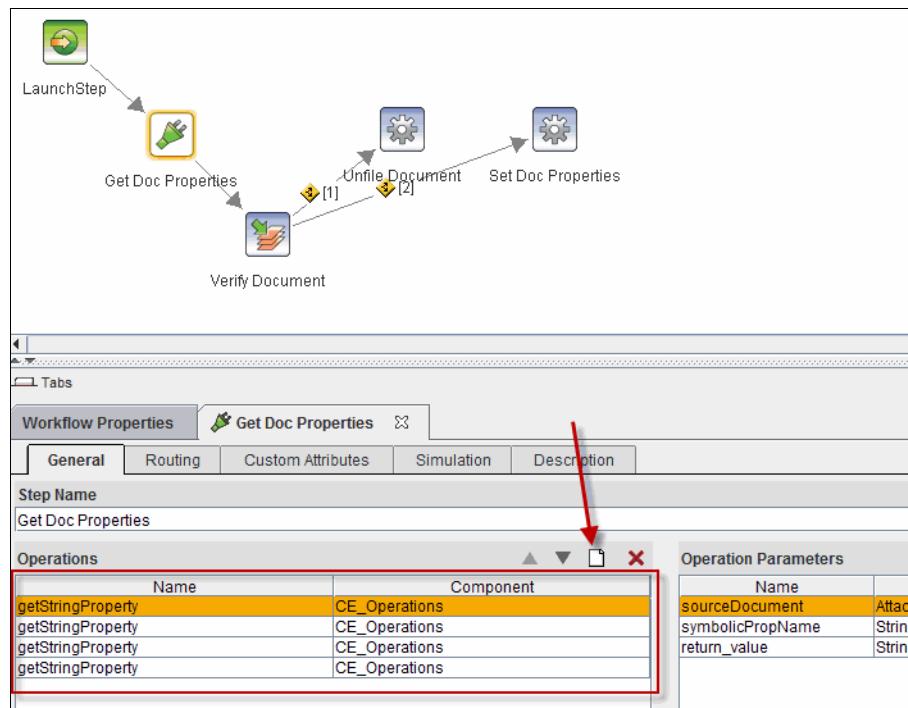


Figure 12-17 Four getStringProperty operations added to the component step

15. Select the first **getStringProperty** and enter the following values:

- sourceDocument: Document
- symbolicPropName: “CC_CaseNumber”
- return_value: Workflow_DocCaseNumber

The result is seen in Figure 12-18. Be sure to include the quotation marks for **symbolicPropName**.

Consideration: The difference between items with quotation marks and no double quotation marks in Process Designer is that quotation marks are treated as a string. Items with no quotation marks are treated as a variable, whether it is a data field, attachment, or other. You can use the expression builder by selecting **Build Expression** to assist with supplying values for a field.

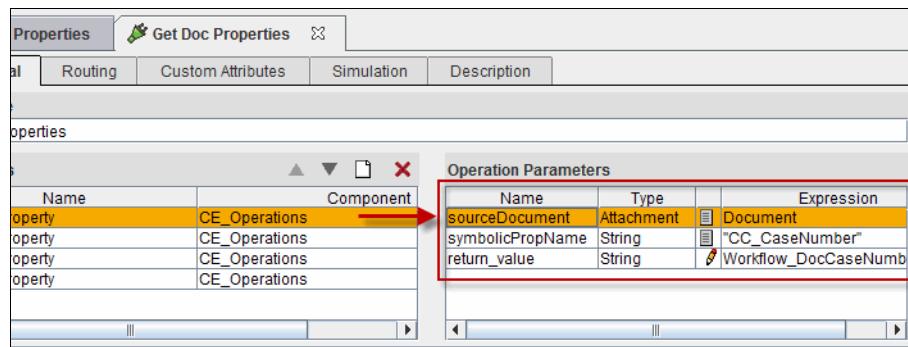


Figure 12-18 Applying expressions to the component operation *getStringProperty*

Repeat this step for the other three operations using the values in Table 12-2.

Table 12-2 Parameters for each *getStringProperty*

sourceDocument	symbolicPropName	return_value
Document	"CC_CustomerName"	Workflow_DocCustomerName
Document	"CC_CustomerNumber"	Workflow_DocCustomerNumber
Document	"DocumentTitle"	Workflow_DocTitle

16. In the example, clearly display the extracted document property values to the user. Click the **Verify Document** step and then the **Parameters** tab.
17. You already exposed the attachment called **Document** as a parameter in Step Editor. Add the additional workflow-specific data fields, which are not available in Step Editor. Click **Workflow_DocCaseNumber**, **Workflow_DocCustomerName**, **Workflow_DocCustomerNumber**, and **Workflow_DocTitle**. Add them to the **Selected Parameters** column by clicking the green arrow.

The result is shown in Figure 12-19.

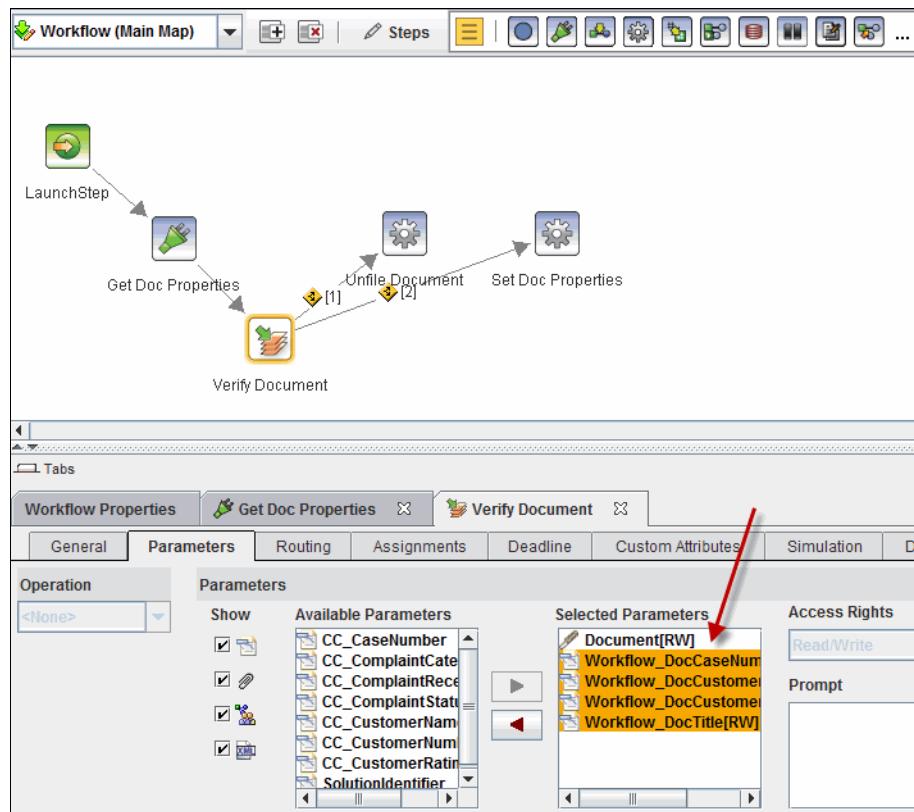


Figure 12-19 Parameters to display for the Verify Document step

18. Convert the **Unfile Document** step to a component step, similar to step 12 on page 479. Additionally, the business analyst confirms that the intention of this step is to delete the document.

With this new information, rename the **Unfile Document** step to Delete Document as shown in Figure 12-20.

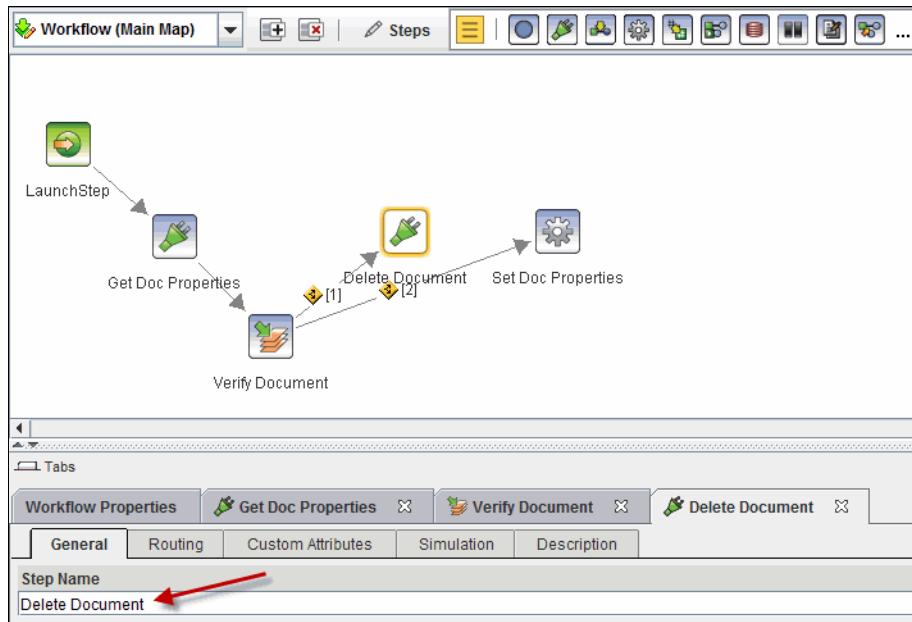


Figure 12-20 Renaming of component step to Delete Document

19. Following the same concept in step 13 on page 479, add a **move** operation to this component. For the **move** operation, select the attachment called **Document** as the **att** parameter and select the folder called **Dispose** as the **destFolder** parameter.

The results are shown in Figure 12-21.

The document is moved to a folder where you can later delete the document in a controlled manner, using company policies on retention. Deleting files based on retention policies is not covered in this book.

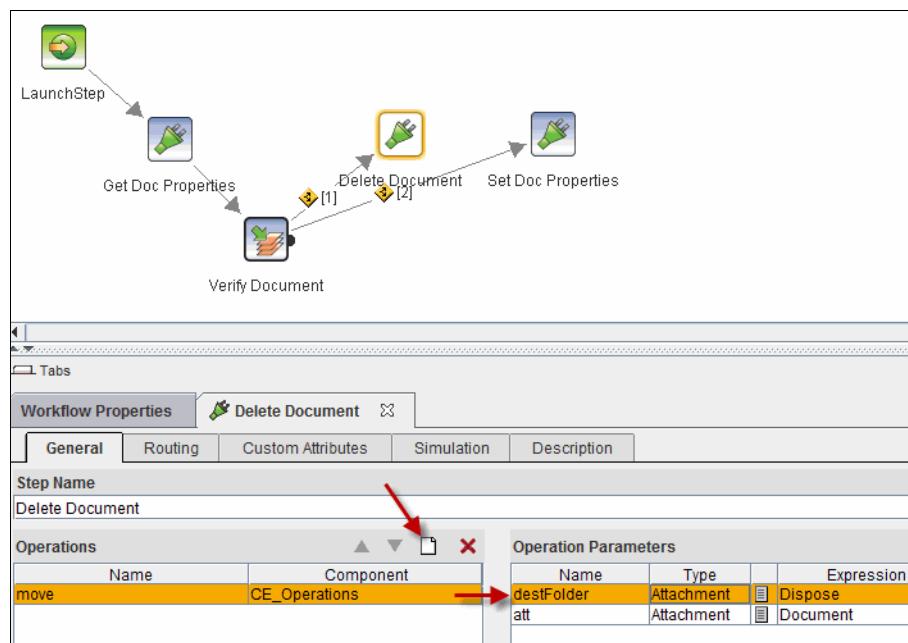


Figure 12-21 Adding the move operation and specifying parameters

20. Convert the **Set Doc Properties** system step to a component step as seen in step 12 on page 479.
21. Similar to step 13 on page 479, add four operations for **setStringProperty** to assign the local data fields to the document properties.
22. Similar to step 15 on page 480, set the parameters for all four operations according to Table 12-3. Be sure to include the quotation marks.

Table 12-3 Parameters for each *setProperty*

destDocument	symbolicPropName	value
Document	"CC_CaseNumber"	Workflow_DocCaseNumber
Document	"CC_CustomerName"	Workflow_DocCustomerName
Document	"CC_CustomerNumber"	Workflow_DocCustomerNumber
Document	"DocumentTitle"	Workflow_DocTitle

An example of the first operation is displayed in Figure 12-22.

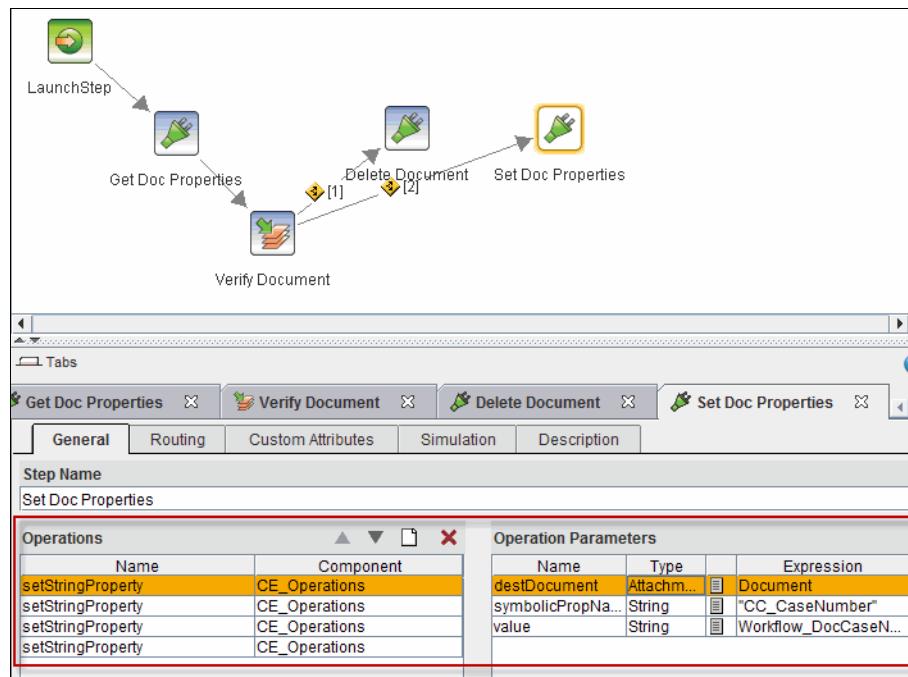


Figure 12-22 Using the four operations to set properties on a document

23. Before you return to Case Manager Builder, validate the workflow collection by clicking **File** → **Validate Workflow Collection** and completing the steps.

Important: Always validate your workflow collection before you save and close your solution in Process Designer. Make sure that there are no errors before you return to Case Manager Builder.

24. Close the solution by clicking **File** → **Solution** → **Save and Close**.

12.2.3 Round-tripping back to Case Manager Builder

After you modify the solution with Process Designer, look at the changes in Case Manager Builder:

1. Log in to Case Manager Builder:

<http://<server>:<port>/CaseBuilder>

2. Edit the **Customer Complaints** solution by pointing to the solution and clicking **Edit**.

3. Go to **Case Types** and select the **Complaint** case type.
4. Go to **Tasks**.
5. Open the Step Editor for the **Verify Document** task.
6. Notice that the system steps look unchanged (Figure 12-23). An indication that the round-tripping made changes is the **Unfile Document** step is renamed to **Delete Document**.

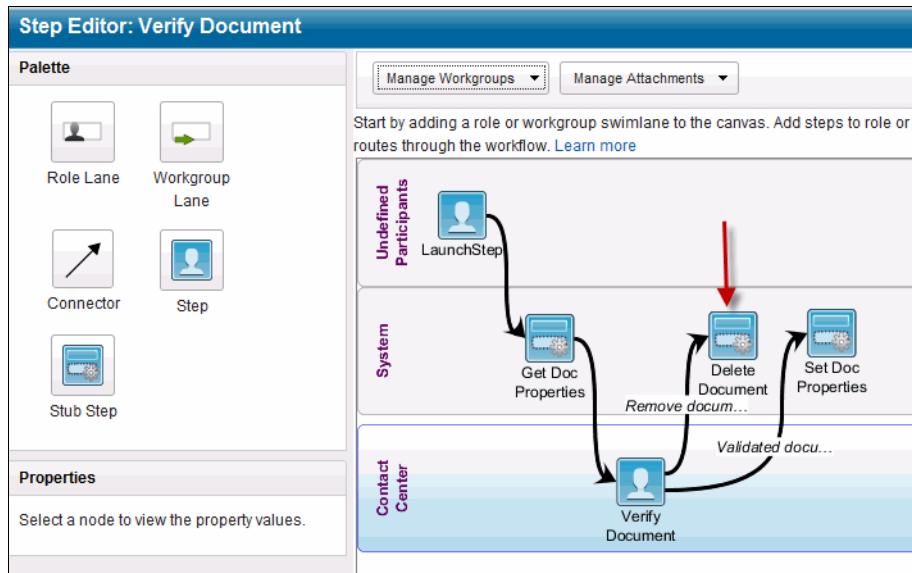


Figure 12-23 Displaying a workflow after you modify it in Process Designer

12.3 Creating more in-baskets

Case Manager Builder allows the business analyst to specify roles. A single in-basket is defined for each role. An IT administrator can add more in-baskets for extra server-side filtering or organization. This section covers some use cases for adding in-baskets.

There are various reasons to create more in-baskets for a role. Extra in-baskets can help with server-side filtering on geographic regions or customer status, just to name two examples. For the example customer complaints scenario, create an in-basket for complaint categories. Creating a choice list makes it easy to designate in-baskets.

The following choice list is for the complaint categories as defined in Table 6-6 on page 166:

- ▶ Product
- ▶ Service
- ▶ Billing
- ▶ Other

Tip: As you consider names, keep in mind the special characters that are supported in role names and in-basket names. They are hyphen (-), underscore (_), and period (.).

The following steps create an in-basket for each category. Apply the in-baskets to the Contact Center role.

1. In Case Manager Builder, select Customer Complaints solution, and click **More Actions → Open Process Designer**

Figure 12-24 shows the Access Process Designer in Case Manager Builder at the solution level.

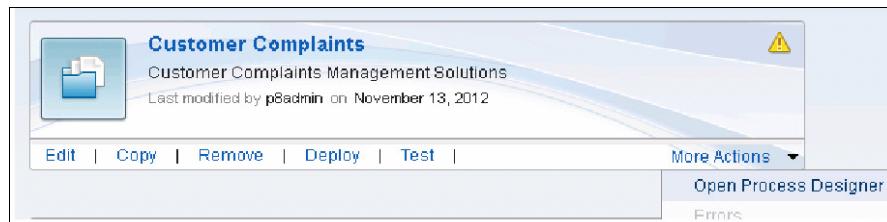


Figure 12-24 Access Process Designer in Case Manager Builder

2. When asked to select a case type, select **Complaint** as shown in Figure 12-25.

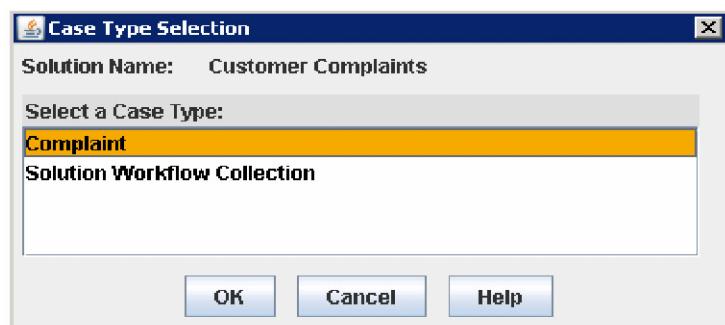


Figure 12-25 Selecting Complaint Case Type

3. Click **OK**.
4. Click **View → In-baskets** as shown in Figure 12-26.

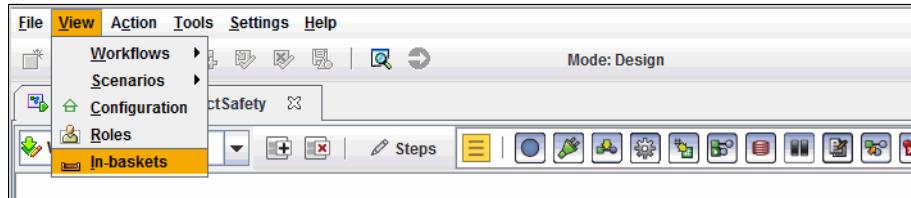


Figure 12-26 Selecting In-baskets for a solution

5. Check that you are modifying the **CC_ContactCenter** queue under **Queue for in-baskets**.
6. Select the Add icon as seen in Figure 12-27 to add four new in-baskets.

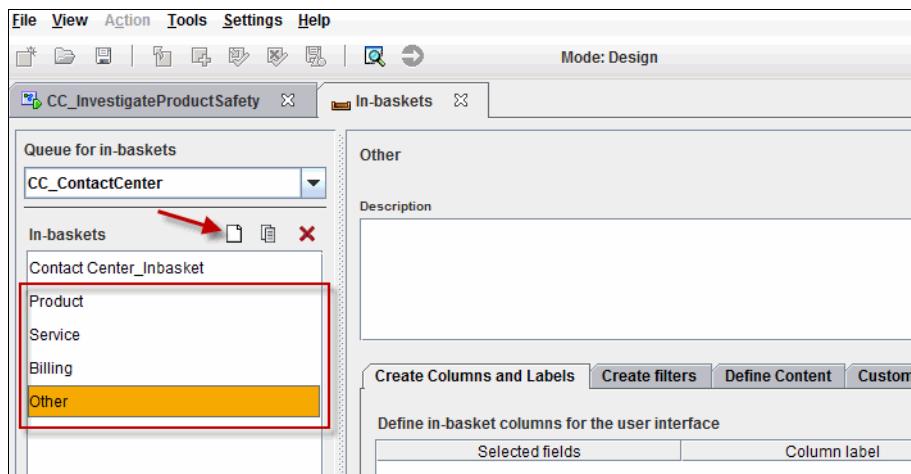


Figure 12-27 Added four new in-baskets to the Contact Center queue

7. Double-click each in-basket and rename them to Product, Service, Billing, and Other.
8. Select the first custom in-basket, **Product**.
9. On the **Create Columns and Labels** tab, click the Add icon.

Expose all fields on the initial role in-basket. Repeat this step for each in-basket. Select all the boxes and click **OK** (Figure 12-28).

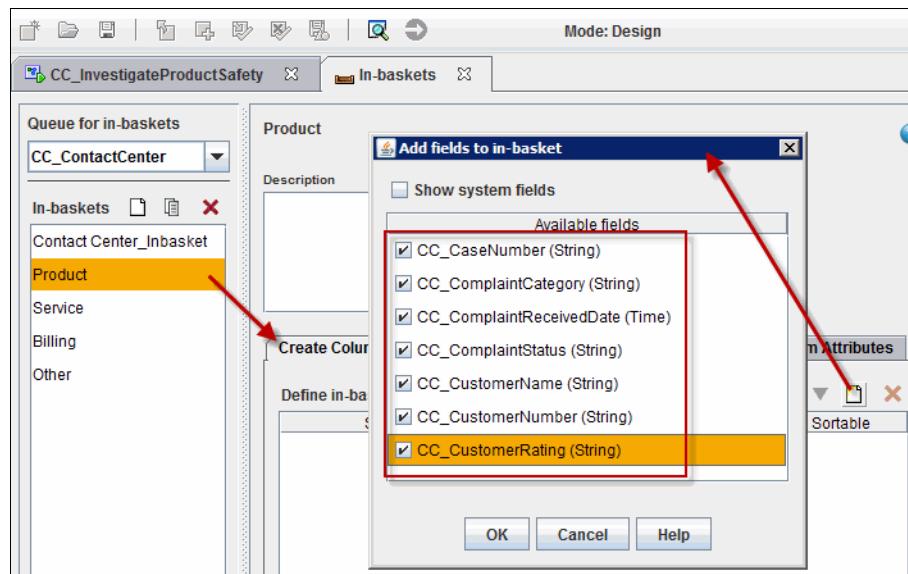


Figure 12-28 Adding fields to an in-basket

10. The column labels for each field are not clear to a user. Double-click each column label and rename them to something clearer. Renaming the labels to remove the prefix, CC_, and add spaces are shown in Figure 12-29.

Selected fields	Column label	Sortable
CC_CustomerNumber (String)	Customer Number	<input type="checkbox"/>
CC_CustomerName (String)	Customer Name	<input type="checkbox"/>
CC_ComplaintStatus (String)	Complaint Status	<input type="checkbox"/>
CC_ComplaintReceivedDate (Time)	Complaint Received Date	<input type="checkbox"/>
CC_CaseNumber (String)	CC_Case Number	<input checked="" type="checkbox"/>
CC_ComplaintCategory (String)	CC_ComplaintCategory	<input type="checkbox"/>
CC_CustomerRating (String)	CC_CustomerRating	<input type="checkbox"/>

Figure 12-29 Renaming the column label for each field

11. Repeat steps 8 on page 488 through step 10 for the other three custom in-baskets.
12. Return to the **Product** in-basket and click the **Define Content** tab.
13. Create a filter:
- Select **Create a filter to define in-basket content**.
 - In the first section, **Select attributes**, select **CC_ComplaintCategory**.
 - Leave **is equal** selected.
 - Enter ‘Product’.
 - Click **ADD** to add the filter you specified.

Figure 12-30 shows a configured filter for the Product in-basket.

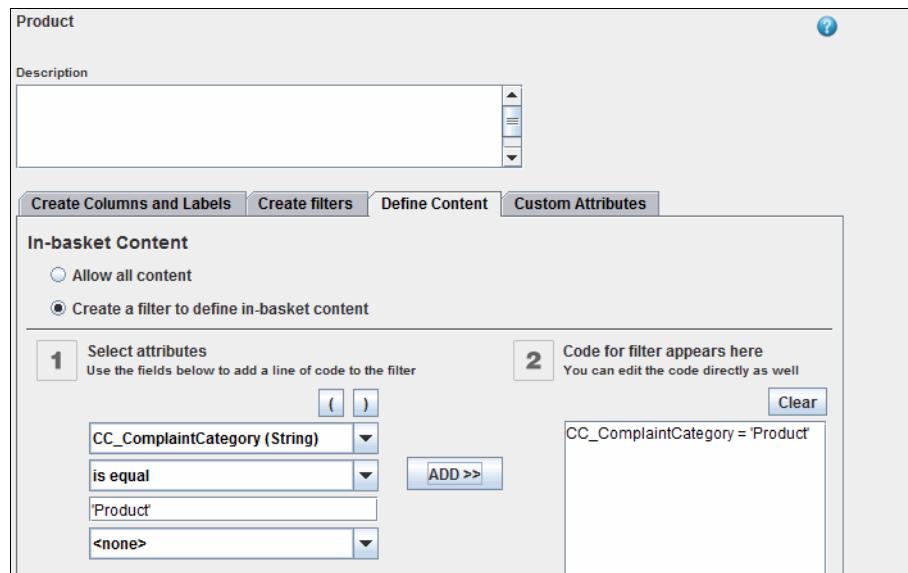


Figure 12-30 Defining a filter for the Product in-basket

14. Repeat step 13 on page 490 for each in-basket, defining the filter appropriately:

- For the **Service** in-basket, specify CC_ComplaintCategory = ‘Product’
- For the **Billing** in-basket, specify CC_ComplaintCategory = ‘Billing’
- For the **Other** in-basket, specify CC_ComplaintCategory = ‘Other’

15. To associate these in-baskets with a role, click **View** → **Roles** as shown in Figure 12-31.

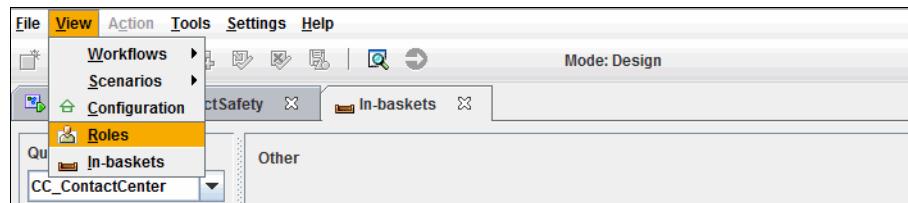


Figure 12-31 Accessing roles in a solution in Process Designer

16. Select the **Contact Center** role.

17. Click the Edit icon next to **Select in-baskets for this role**.

18. Select the **Product**, **Service**, **Billing**, and **Other** in-baskets that were created earlier, and click **OK**. See Figure 12-32.

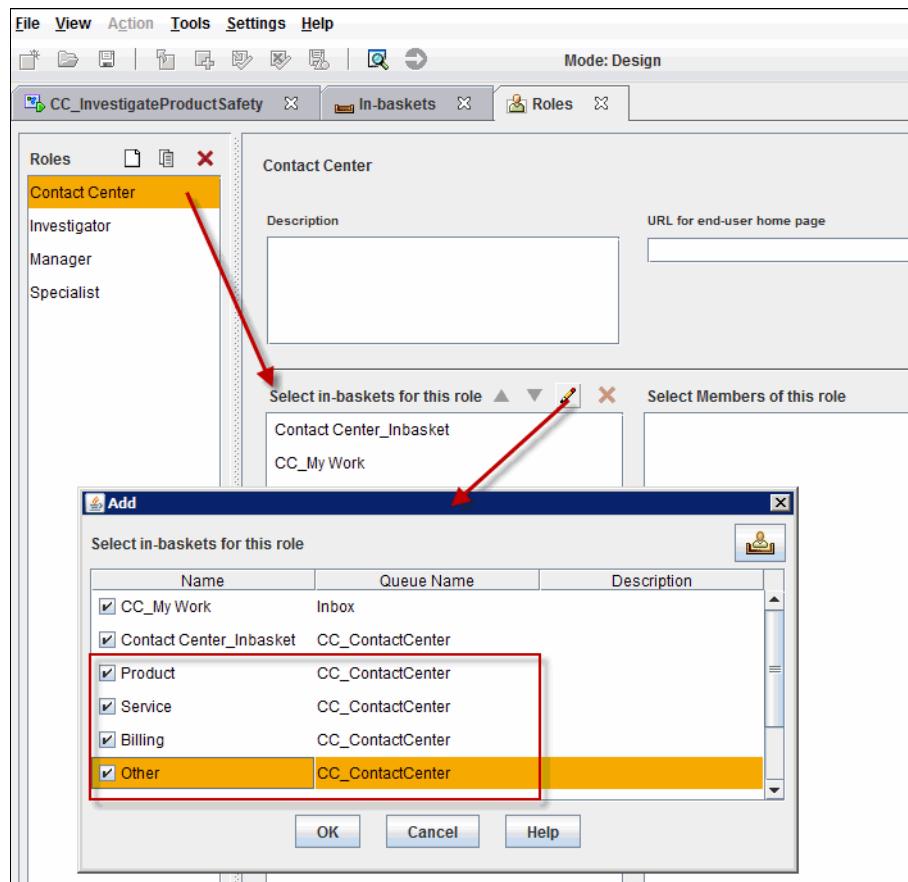


Figure 12-32 Adding in-baskets to a role

19. Save and deploy the solution.

Test the changes as shown in Figure 12-33. All work items are showing up in the Contact Center in-basket.

The screenshot shows the IBM Case Manager interface with the following details:

- Header:** IBM Case Manager, Home, Go to Spaces, Manage Spaces, Actions.
- Section:** Customer Complaints
- Toolbar:** Work (selected), Cases, Add Case, Manage Roles.
- In-basket:** Contact Center(5) (highlighted by a red arrow).
- Filter:** No filter is applied.
- Table:** Displays 5 items (1 - 5) with columns: Complaint Received Date, Customer Rating, Complaint Category, Case Number, and Status. The Complaint Category column is highlighted with a red border.

Complaint Received Date	Customer Rating	Complaint Category	Case Number	Status
10/18/2011 12:00 AM	Good	Product	C0063	Open
10/19/2011 12:00 AM	Excellent	Billing	C1333	Pro
10/21/2011 12:00 AM	Good	Product	C0128	Pro
10/28/2011 12:00 AM	Excellent	Service	C1233	Clos
10/31/2011 12:00 AM	Fair	Billing	C1083	Pen

Figure 12-33 Viewing the main in-basket of the Contact Center role

Figure 12-34 shows that only the work items of the Product category are showing up in the Product in-basket.

The screenshot shows the IBM Case Manager interface with the title 'Customer Complaints'. The top navigation bar includes 'Home', 'Go to Spaces', 'Manage Spaces', and 'Actions'. Below the title, there are tabs for 'Work' (selected), 'Cases', and a '+' button. Under 'Work', there are 'Add Case' and 'Manage Roles' buttons. The main area is titled 'In-basket' with tabs for 'Contact Center', 'My Work', 'Product(2)' (highlighted with a red arrow), 'Service', 'Billing', and 'Other'. A table below shows 'Items 1 - 2' with columns: 'Complaint Received Date', 'Customer Rating', 'Complaint Category' (which is highlighted with a red box), 'Case Number', and 'Cor'. The data rows are: 10/18/2011 12:00 AM, Good, Product, C0063, Ope; and 10/21/2011 12:00 AM, Good, Product, C0128, Pro.

Figure 12-34 Viewing the custom in-basket of the Contact Center role

You can create multiple in-baskets for a queue that can be added to a role. However, you cannot assign multiple roles to the same queue as illustrated in Figure 12-35.

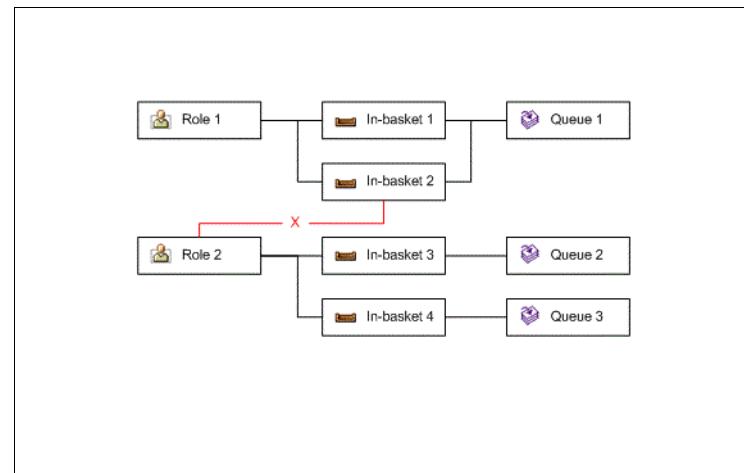


Figure 12-35 Restriction on assigning roles to queues

12.4 Exposing error handling in the Case Manager Client

The default error handling procedure for Process Engine processes requires an IT administrator to view malfunction steps in the Process Administrator tool. However, it might be more convenient to view the errors in the Case Manager Client. Some errors might not require the deep technical capabilities available in the Process Administrator tool. This section describes how to expose errors in the Case Manager Client to facilitate error handling.

A default Malfunction process map is associated with every workflow. This default process map can be overridden so that errors are exposed in the Case Manager Client. To expose process errors to a role in the Case Manager Client, complete these steps:

1. In Case Manager Builder, select Customer Complaints solution, and select **More Actions → Open Process Designer**

Figure 12-36 shows the Access Process Designer in Case Manager Builder at the solution level.

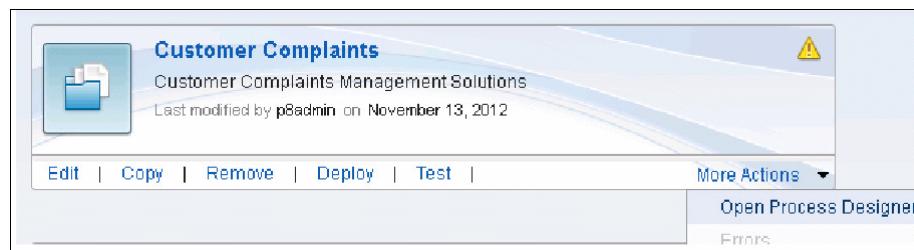


Figure 12-36 Access Process Designer in Case Manager Builder

2. Select the case type **Complaint** as shown in Figure 12-37.

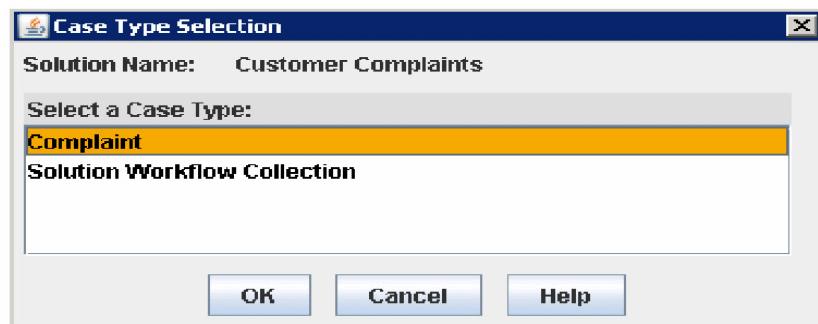


Figure 12-37 Selecting the Complaint Case Type

3. Click **OK**.
4. Click **View → Configuration** as shown in Figure 12-38.



Figure 12-38 Selecting Configuration for a solution

5. Right-click **Work queues** and select **New**.
6. For the queue name, enter **Administration** and click **OK**.

This queue, as seen in Figure 12-39, is the queue that receives work items to process errors.

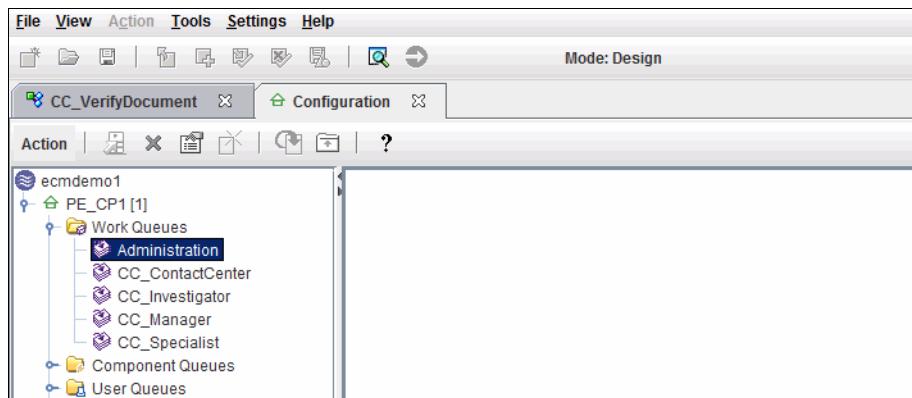


Figure 12-39 Creating the Administration queue

7. To configure the queue, right-click **Administration** and select **Properties**.
8. To add system fields that can be exposed with an in-basket, click the **System Fields** tab.

9. Select **F_StepName** under **Available Items** and use the green arrow to add it to **Selected Items**. The results are shown in Figure 12-40. For the example, add **F_StepName**, but any others can be added as needed.

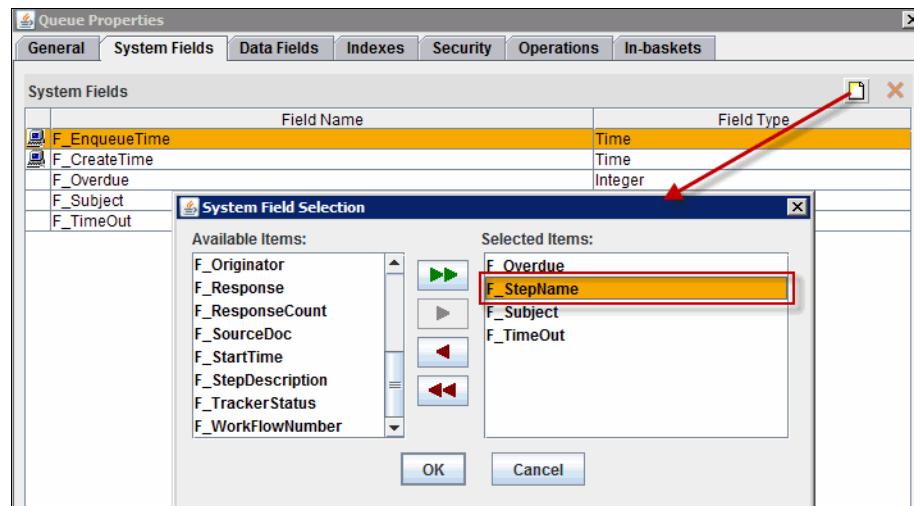


Figure 12-40 Adding a system property to the system fields on the new queue

10. Click **OK**.
11. To add data fields that can be exposed with an in-basket, click the Data Fields tab.
12. Click the Exposed Data Fields icon in the upper right.

13. Add the properties of the solution to **Selected Properties** as shown in Figure 12-41.

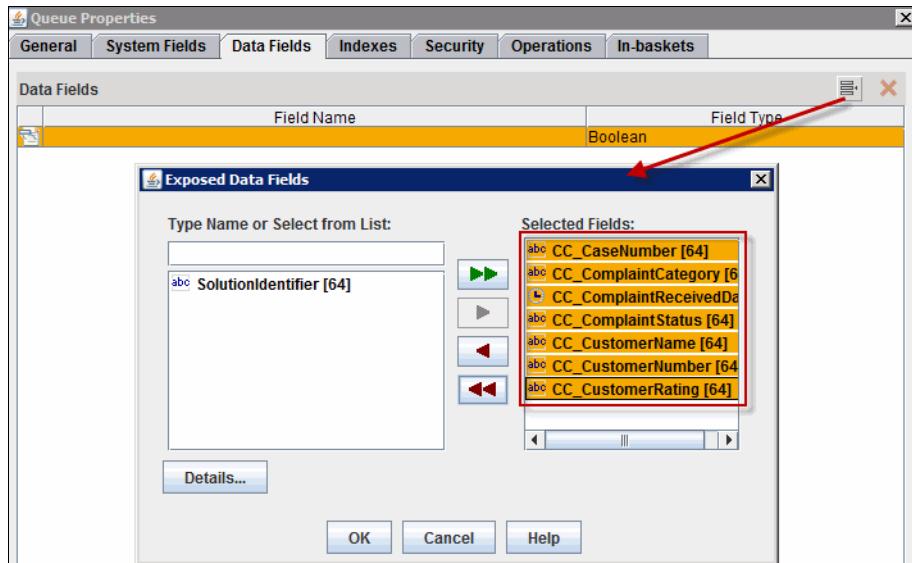


Figure 12-41 Adding solution properties to data fields on the new queue

14. Click **OK**.

15. You can manually add more data fields. In step 6 on page 477, you created extra workflow data fields for the **Verify Document** task. Add those fields to the queue as well. Double-click the cell with a blank **Field Name**, and enter the data fields that you defined. The result should look like Figure 12-42.

Field Name	Field Type
CC_CaseNumber	String
CC_ComplaintCategory	String
CC_ComplaintReceivedDate	Time
CC_ComplaintStatus	String
CC_CustomerName	String
CC_CustomerNumber	String
CC_CustomerRating	String
Workflow_DocTitle	String
Workflow_DocCaseNumber	String
Workflow_DocCustomerNumber	String
Workflow_DocCustomerName	String
	Boolean

Figure 12-42 Adding the additional data fields to the new queue

16. To create an in-basket on this queue, click the **In-baskets** tab.

17.Click the Add icon.

18.Double-click the new queue name to rename it, and Enter Malfunction.

The new in-basket is added as seen in Figure 12-43.

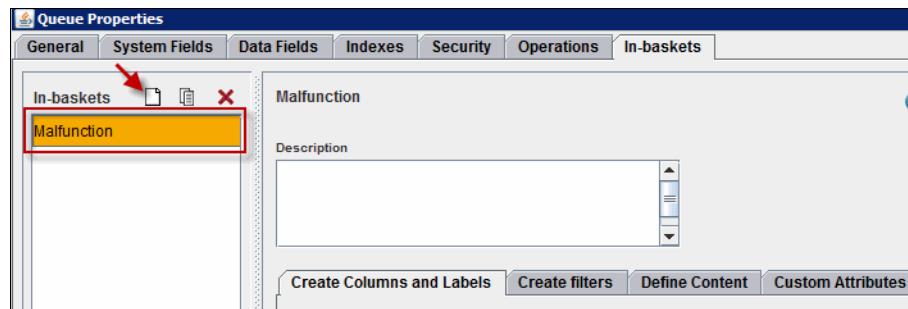


Figure 12-43 Adding a Malfunction in-basket

19.For the **Malfunction** in-basket, under the **Create Columns and Labels** tab, click the Add icon. In the example, select **Show system fields**.

20.Select the following data fields:

- CC_CaseNumber
- CC_CustomerName
- CC_CustomerNumber
- F_StepName
- F_Subject
- Workflow_DocCaseNumber
- Workflow_DocCustomerName
- Workflow_DocCustomerNumber
- Workflow_DocTitle

Rename the column label for each data field to be more readable. Then, use the up and down arrow icons to sort the columns in a logical sequence. The result is shown in Figure 12-44.

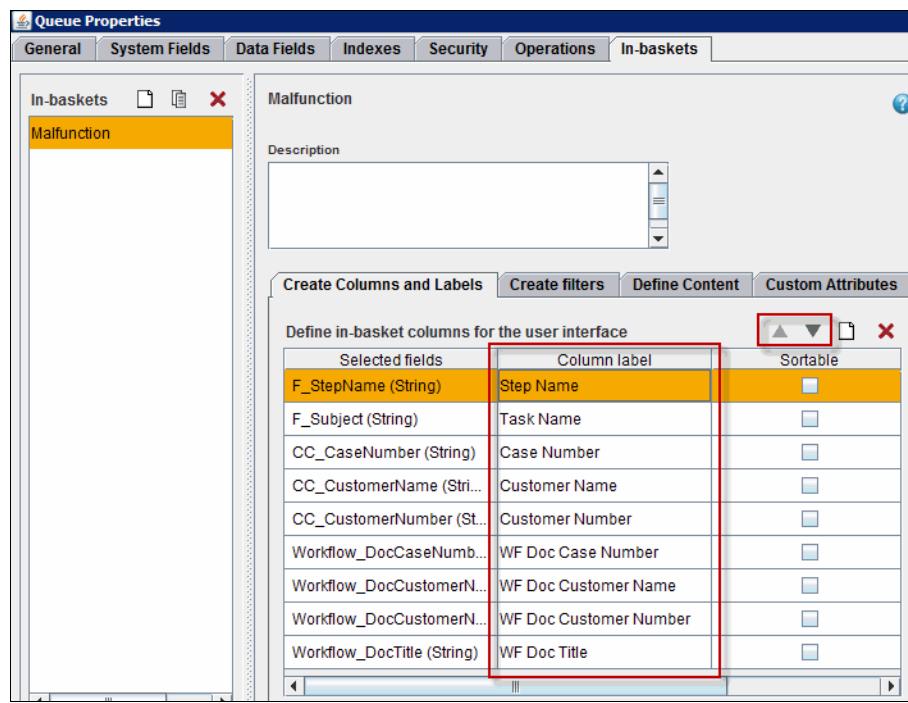


Figure 12-44 Configuring the Malfunction in-basket on the Administration queue

21. Click **OK** on the queue properties window to close it and return to the solution configuration.
22. Open the **Validate Document** workflow.
23. Navigate through submaps to see the **Malfunction** submap. Notice that it is read only.
24. To override the submap, click the Create Map icon and select **Override Inherited Map**.

The **Malfunction** map is selected by default as shown in Figure 12-45.

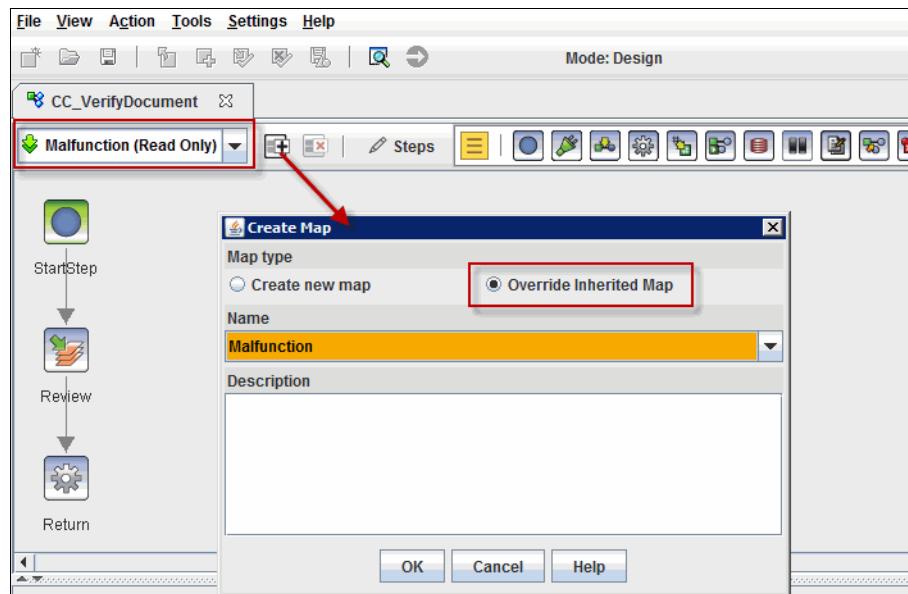


Figure 12-45 Creating a map to override the Malfunction map

25. Click **OK**.

26. Modify the process of the Malfunction map. Start by selecting the **Review** step.

27. For the **Step Name**, enter Review_Error.

28. For the **Activity Type**, select the **Administration** work queue as shown in Figure 12-46.

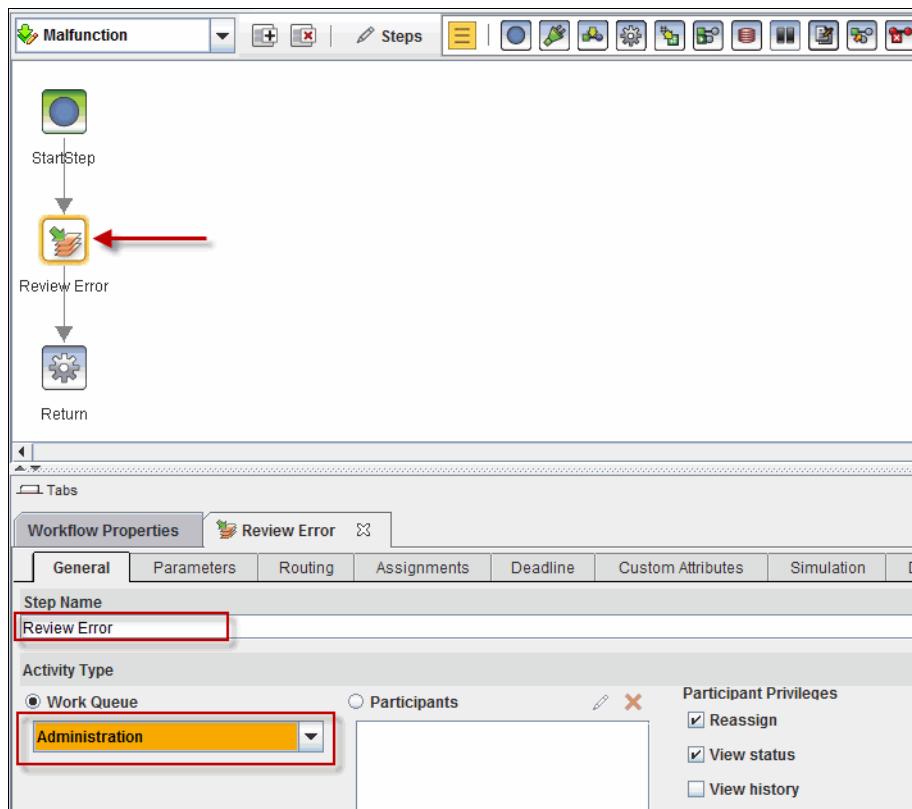


Figure 12-46 Configuring a step in the Malfunction map

29. This step must function within the Case Manager Client user interface so you must use an available Step page as your step processor. Click the **General** tab and select the step processor.

The example uses **CC_CmAcmSTEP_DEFAULT_PAGE** as shown in Figure 12-47.

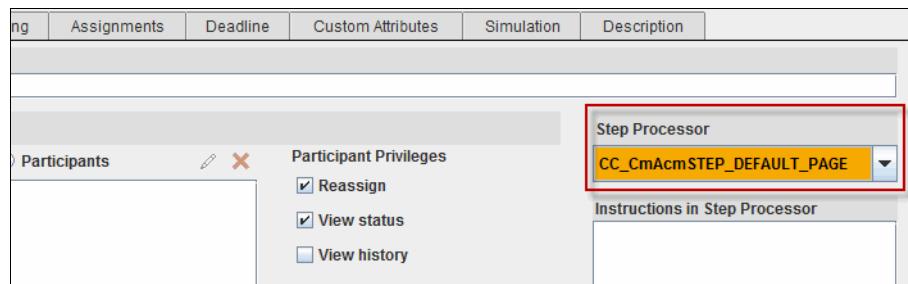


Figure 12-47 Selecting the default Step page

Tip: You might get an error on validation or deployment that says:

A step with a queue assigned to it should have a valid requested interface.

This error means that you have not set the Step Processor to a valid Step page. For every activity step in a solution, the Step Processor must be set to a registered Step page for the solution. For more information about registering custom pages, see “Customizing a page in Step page space” on page 356.

30. While in the **Review Error** step, select the **Parameters** tab.

Move everything from **Available Parameters** other than **SolutionIdentifier**, as seen in Figure 12-48.

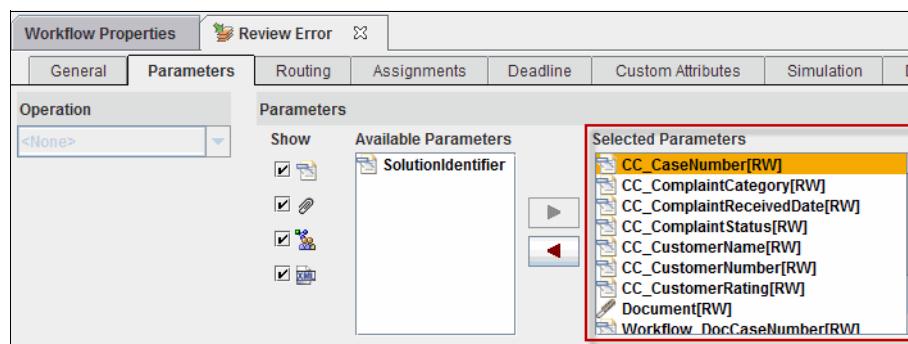


Figure 12-48 Exposed parameters of the Review Error step

31. Click the **Routing** tab.

32. Add the two responses by double-clicking the empty cell under **Name** to edit a response and enter Skip. Enter Repeat for the second response.

The configured responses are shown in Figure 12-49.

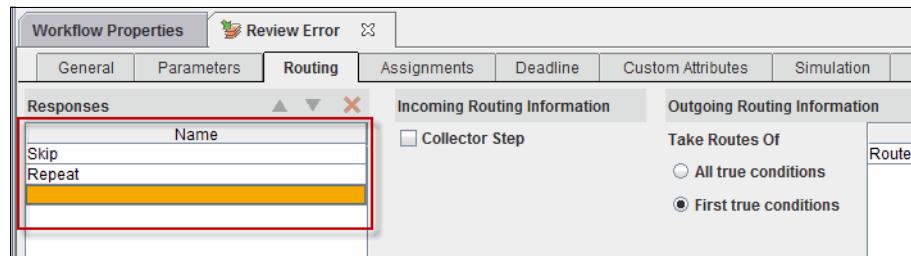


Figure 12-49 Configured responses for the Review Error step

33. In the workflow map, create two return steps to replace the default system step. Right-click the existing **Return** system step and select **Delete**.

34. Make sure that the General System Palette is available as shown in Figure 12-50. The palette menu is opened by clicking the arrow icon just to the right of the available palette.

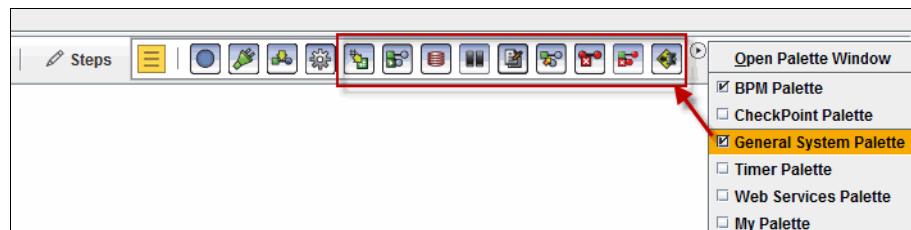


Figure 12-50 Available system palette

Drag the two return steps into the workflow as shown in Figure 12-51.

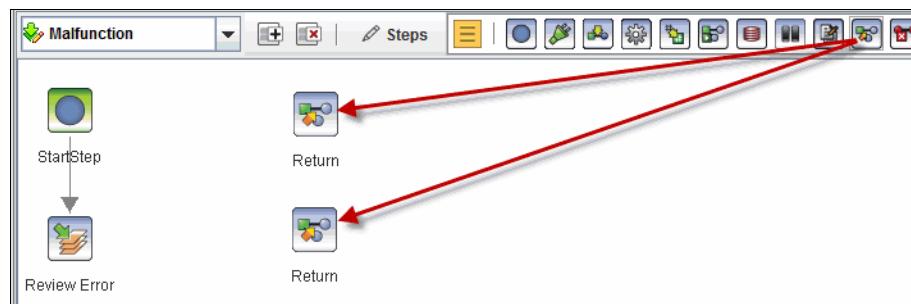


Figure 12-51 Adding two return steps after you remove the system step

35. Rename each return step and set its expression:

- a. Click the first **Return** step.
- b. Under the **General** tab, enter Return True for the **Step Name**.
- c. Enter true for the **Return Expression**.
- d. Click the second Return step.
- e. Under the **General** tab, enter Return False for the **Step Name**.
- f. Enter false for the **Return Expression** as shown in Figure 12-52.

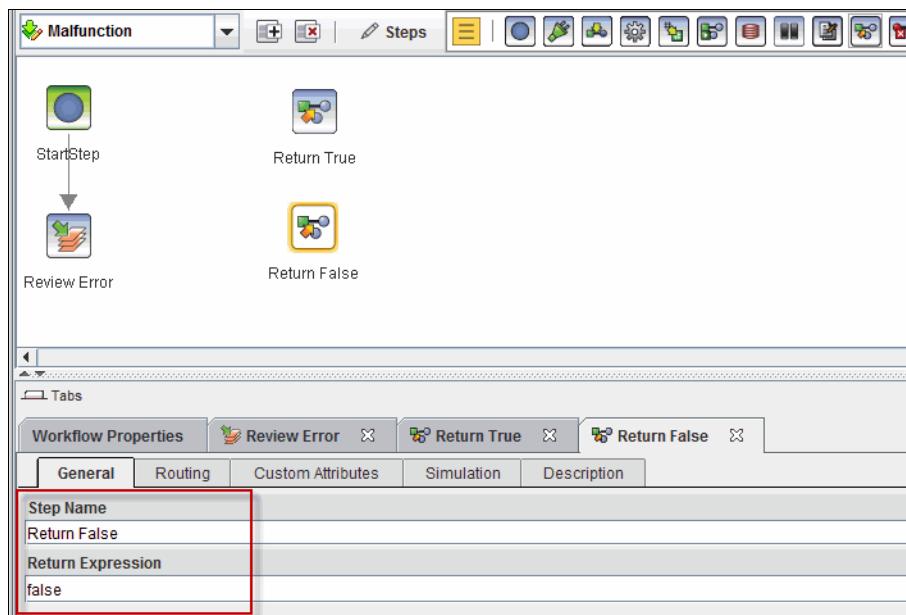


Figure 12-52 Configured the return steps

For more information about how return values are used in a submap, see the IBM FileNet Information Center at:

<http://publib.boulder.ibm.com/infocenter/p8docs/v5r1m0/topic/com.ibm.p8.pe.designerui.doc/bpfhd003.htm>

36. Create a route from the **Review Error** step to the **Return True** step.

37. On the route properties, enter Repeat for the **Route**.

38. Select **Conditional Route** under **Routing**.

39. Select **ALL** for the **Condition**.

40. Select **Repeat** for the **Response**.

41. Click **ADD** to create the condition.

The result should look like Figure 12-53.

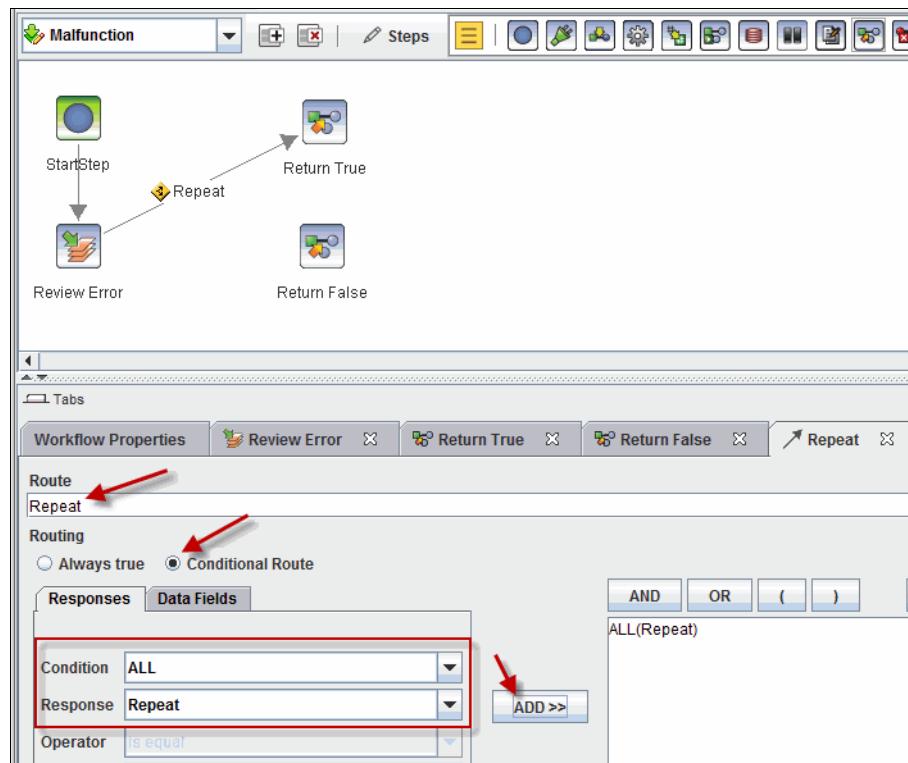


Figure 12-53 Adding a conditional route that is based on a Repeat response

42. Create a route between the **Review Error** step and the **Return False** step. Enter Skip as the route name and set the response as **Skip**. The route configuration is shown in Figure 12-54.

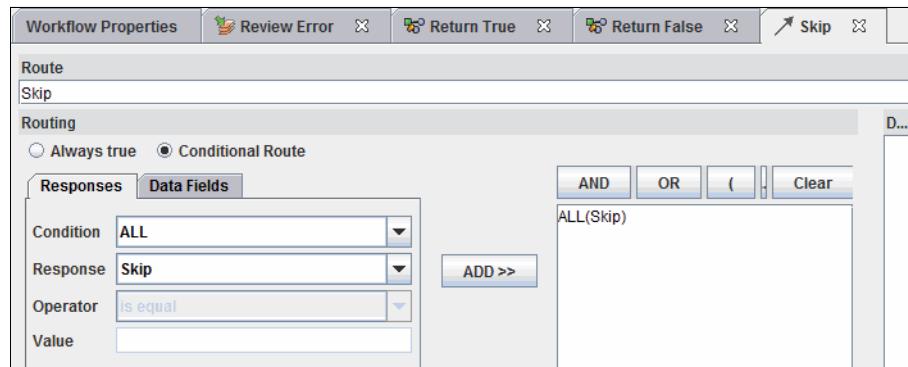


Figure 12-54 Adding a conditional route that is based on a Skip response

43. Validate the workflow collection by clicking **File** → **Validate Workflow Collection**. The example workflow validation is successful as seen in Figure 12-55.

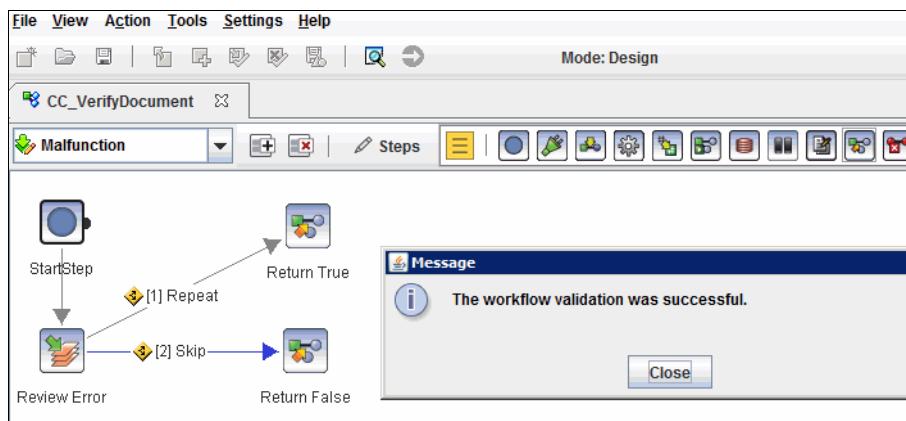


Figure 12-55 Validated workflow collection after you configure the Malfunction map

44. Save and deploy the solution.

Malfunctions now insert a work item into the Administrators queue. You can assign the Malfunction in-basket to any role. For more information, see 12.3, “Creating more in-baskets” on page 486. It can be an IT role or the Contact Center role in the example solution. This user can look at values and determine whether the step must be repeated, possibly with new values, or end and the workflow process continue.

As a brief example of this error handling, the following process might occur with the Verify Document task:

1. Add the Malfunction in-basket to the Contact Center role.
2. A document is added to a case as a Supporting Document with the following properties:
 - Title: Bob's Attachment
 - Case Number: C1600
 - Customer Name: Bob
 - Customer Number: 88489299
3. The Verify Document task is started.
4. A user in the Contact Center role sees the Verify Document work item.
5. The user determines that the content is not relevant.
6. For some reason, the user removes the document from the attachments by using the attachments widget. The user then clicks Remove Document.

7. An error occurs because the system has a null value for the Document attachment. The CE_Operations cannot move a null object.
8. An advanced user in the Contact Center role sees a work item in the Malfunction in-basket as shown in Figure 12-56.

In-basket					
Contact Center	Product	Service	Billing	Other	Malfunction(1)
Items 1 - 1					
Step Name	Task Name	Case Number	Customer Name	Customer Number	Work Item ID
Review Error	Verify Document	C1600	Bob	88489299	C1600-1
Items 1 - 1					

Figure 12-56 Work item in an in-basket because of error

9. This advanced user notices that an attachment is missing. The user adds the document in the case, Bob's Attachment, back to the attachment as shown in Figure 12-57.

Attachment	
View More Actions	
Document Add	
No items to display.	

Attachment
View More Actions
Document Add
Bob's Attachment

Figure 12-57 Adding a document from the case as a workflow attachment

10. The advanced user believes that this action will fix the problem and selects the **Repeat** response so that the CE_Operations can try again.
11. The document is confirmed as moved out of the case.



Part 3

Integration

This part addresses IBM Case Manager integration with external products and components.

This part includes the following chapters:

- ▶ Integration points
- ▶ Integration with WebSphere ILOG JRules
- ▶ Integration with IBM Content Manager
- ▶ External Data Service Framework
- ▶ Integration with IBM Business Process Manager
- ▶ Integration with IBM Content Analytics



Integration points

Typically case management systems are not stand-alone systems. Depending on the industry and the type of application, a case management system can be integrated with back-end systems, databases, and other business applications. Integration with the external systems can be achieved in many ways. Often, it depends on the integration points, and tools available for the systems involved.

This chapter describes at a high level the different application programming interfaces (APIs) and other integration points that are supported by IBM Case Manager. It is intended as a quick reference guide on considering the best approach for integration.

This chapter includes the following sections:

- ▶ REST APIs
- ▶ Using Java APIs
- ▶ Using Case Manager Client side integration
- ▶ Using Process Engine workflow integration
- ▶ External data service

13.1 REST APIs

There are three Representational State Transfer (REST) APIs that are provided with IBM Case Manager. Each one serves a specific purpose. Keep in mind that these APIs are Hypertext Transfer Protocol (HTTP) calls that offer an abstraction from lower-level coding languages.

13.1.1 Case Manager REST API

The Case Manager REST API is used for case-related actions. As of IBM Case Manager 5.1.1, these same actions are now available in the Case Manager Java API described in 13.2.3, “Integrating by using the Case Java API” on page 517.

This API returns responses in the JavaScript Object Notation (JSON) format.

You use this API for case-specific information:

- ▶ Solutions available and their information
- ▶ Case types available in a solution
- ▶ Task types available for case types
- ▶ Tasks available in a case and actions that can be performed on them
- ▶ Comments for a case or for a task or document within a case
- ▶ History of a case

Your application can request information about case objects and manipulate those case objects by using this API. The API can be used in the following examples:

- ▶ A web front end on the internet retrieves case history based on a case record locator and presents it to the external user. Use the case history resource for a case.
- ▶ An external application must read the status of all tasks in a case. Use the list of task instances resource for a case.
- ▶ An external application processes fees associated with a case and adds a comment to the case. Use the case comments resource for a case.
- ▶ An external application must create a case for processing extra information. Use the cases resource.
- ▶ A custom widget must create a task without leaving the mashup page. Use the discretionary task types resource to list available tasks to create, and use the create new task resource to create the task.

For more information about this API, see IBM Case Manager, Version 5.1.1 Information Center at:

<http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/topic/com.ibm.casemgmt.development.doc/acmdv014.htm>

13.1.2 Process Engine REST API

The Process Engine REST API is used for business process-related information. All the methods in this API are either immediately available or can be built by using the Process Engine Java API.

This API returns responses in JSON format.

Use this API for role and work item-related information:

- ▶ Roles
- ▶ Work basket information and filters
- ▶ Work items that are assigned to a role or user
- ▶ Work item information
- ▶ Saving and completing work items
- ▶ Current user and search for users

Your application can request information about business process objects and act on those objects by using this API. It can be used in the following examples:

- ▶ A web front end on the internet must display actions for external users as part of a process in a case. Use the queue elements resource on a designated role in-basket with query parameters as necessary.
- ▶ A web front end on the internet must complete an action, or work item, as part of a process in a case. Use the step element resource to update the work item information and dispatch it, which completes the step.
- ▶ An external application that employees use daily must display work item count for each of their role work baskets. Use the role resource to get work baskets. Then, use queue elements count resource on those work baskets.
- ▶ A custom widget must show all work items across multiple roles for a manager to view in one list. Use the role names resource on the solution application space to gather work baskets, and then queue elements.

For information about this API, see IBM FileNet P8, Version 5.1 Information Center at:

http://publib.boulder.ibm.com/infocenter/p8docs/v5r1m0/index.jsp?topic=/com.ibm.p8.pe.dev.doc/rest/rest_ref.htm

13.1.3 Content Management Interoperability Services API

The Content Management Interoperability Services (CMIS) API is used for content-related information. All the methods in this API are either immediately available or can be built by using the IBM FileNet Content Engine Java API.

This API returns responses in XML format.

Use this API for content-related information:

- ▶ Get and update properties on content objects
- ▶ Create content objects
- ▶ Query for content objects
- ▶ Upload or download document content
- ▶ Check in or check out document objects
- ▶ Work with versions of document objects

Your application can request information about content objects and act on those objects by using this API. It can be used in the following examples:

- ▶ A web front end on the internet must add form details from an external user as a document in a case. Use the create document resource to provide the form details as a content stream.
- ▶ An external application must capture the state of a case at the time of an event in the external system. Use the content resource for the case folder to retrieve object properties.
- ▶ A custom widget is working with a specific document and must present a toolbar that is based on user security for that document. Use the content resource for the document object to retrieve allowable actions.

For more information about this API, see the OASIS website about Content Management Interoperability Services Version 1.0 at:

<http://docs.oasis-open.org/cmis/CMIS/v1.0/os/cmis-spec-v1.0.html>

13.2 Using Java APIs

There are two Java APIs that are provided as standard IBM FileNet APIs. IBM Case Manager is built on FileNet P8, so these APIs are readily available with IBM Case Manager. These APIs offer much more flexibility and granular control than the REST APIs. These APIs are based on Java, and require Java programming.

Added in Case Manager 5.1.1 is the new Case Java API.

For more information, see "Java application development at:

<http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/topic/com.ibm.casemgmt.development.doc/acmj001.htm>

In addition to the two Java APIs, FileNet P8 supports other languages through a web services abstraction for Content Engine and Process Engine. For brevity, topics are addressed by referencing the Java APIs that map to web services.

13.2.1 Integrating by using Content Engine Java API

The Content Engine API is used to implement full enterprise content management capabilities. The Content Engine Java API can access all objects in the Content Engine repository. These objects include but are not limited to documents, compound documents, folders, custom objects, document types, subscriptions, and security.

You use this API to harness the full capabilities of Content Engine.

In comparison to CMIS, the Content Engine Java API offers finer control of all objects in the repository. CMIS is a specification still under development and is designed for compatibility with various repository types. As a result, only a subset of Content Engine functions are exposed through CMIS.

For more information about this API, see IBM FileNet P8, Version 5.1 Information Center at:

<http://publib.boulder.ibm.com/infocenter/p8docs/v5r1m0/topic/com.ibm.p8.ce.dev.java.doc/overview-summary.html>

Your application can access and manipulate content objects in a FileNet P8 environment by using this API. It can be used in the following examples:

- ▶ An external application must run actions in a batch. Use the batch utilities to group the operations together.
- ▶ An external application must configure complex access control lists (ACLs) on objects. Use the security utilities to configure ACLs.
- ▶ An external application must not abstract data for performance reasons. Use the communication utilities to avoid the performance effect of abstracting data to XML or other format for communication.
- ▶ An external application must take advantage of core FileNet P8 capabilities. Use the extensive range of utilities that are provided by the Content Engine Client.

- ▶ A custom REST API must be developed to meet a specific requirement that is not available in CMIS. Use the extensive range of utilities that are provided by the Content Engine Client and expose them by using a custom REST API.

13.2.2 Integrating by using Process Engine Java API

The Process Engine API is used to implement full FileNet P8 Business Process Management capabilities. The Process Engine Java API can access all information that is related to workflows. This information includes but is not limited to application spaces, roles, work items, exceptions, queues, rosters, security, and system details.

You use this API to harness the full capabilities of a FileNet P8 Business Process Manager system.

In comparison to Process Engine REST API, the Process Engine Java API offers finer control of all information that is related to workflows. Process Engine REST API has been around for a few years, whereas the Process Engine Java API has been available for many years. As a result, only a subset of Process Engine functions is exposed through Process Engine REST API.

Your application can access and manipulate workflow-related objects in a FileNet P8 environment by using this API. It can be used in the following examples:

- ▶ An external application must generate a report on the history of work items. Use the utilities to provide the step information.
- ▶ An external application must analyze a process. Use the process utilities to analyze a running workflow.
- ▶ An external application must initiate a workflow. Use the workflow utilities to start a workflow.
- ▶ An external application must be able to configure a Process Engine system. Use the configuration utilities to work with isolated regions or the entire system.
- ▶ An external application must take advantage of an action not available in the Process Engine REST API. Use the appropriate utilities that are provided by the Process Engine Client.

For more information about this API, see the IBM FileNet P8, Version 5.1 Information Center at:

<http://publib.boulder.ibm.com/infocenter/p8docs/v5r1m0/topic/com.ibm.p8.pe.dev.java.doc/filenet/vw/api/package-summary.html>

13.2.3 Integrating by using the Case Java API

New in 5.1.1, IBM Case Manager provides a Java API for handling common case-based operations. These operations include methods and objects for gathering information, creating cases and tasks and to update existing case information.

Similar to the Case Rest API, described in Section 13.1.1, “Case Manager REST API” on page 512, the new Case Java API can be used to retrieve and manipulate case-specific information:

- ▶ Deployed solutions and Case Manager configuration information
- ▶ Case types and document types that are included in a deployed solution
- ▶ Available task types
- ▶ Create, update, and split cases
- ▶ Create relationships between cases
- ▶ Create and retrieve case comments
- ▶ Retrieve case history
- ▶ Start manual and discretionary tasks

Your application can request information about case objects and manipulate those case objects by using this API. The API can be used in the following examples:

- ▶ An external, Java-based application must read the status of all tasks in a case. The com.ibm.casemgmt.Case class has methods for retrieving the tasks of a case, represented as com.ibm.casemgmt.tasks.Task objects.
- ▶ An external enterprise Java application processes fees associated with a case and adds a case comment. The com.ibm.casemgmt.Case class can be used to add and retrieve comments, represented as com.ibm.casemgmt.Comment objects.
- ▶ An external application must create a case for processing extra information. The com.ibm.casemgmt.Case class has factory methods that can be used to create a case.

For more information about this API, see the IBM Case Manager, Version 5.1.1 Information Center at:

<http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/topic/com.ibm.casemgmt.development.doc/acmjd001.htm>

13.3 Using Case Manager Client side integration

The IBM Case Manager interface is designed by using a mashups container. Mashup containers by design are a combination of various widgets that can run actions, display information, or both. Case Manager Client side integration means that the user's browser is requesting or presenting data from a resource. This resource is not necessarily an IBM Case Manager resource.

13.3.1 Web Site widget

Simple integration is available by using the Web Site widget. This widget is provided with Business Space.

The basic function of the widget is to display a website within an iframe. A default website can be configured for when the widget initially loads. This first page can be your home page or a launch page for work in another system. Because the website is displayed in an iframe, browsing in this widget enables the user to navigate around and still reference the IBM Case Manager user interface.

An extended function is the ability to show websites dynamically. The widget can receive a URL. If a URL payload is received, the widget renders that URL in the iframe. The URL can contain a query string so you can access any web resources, even with query parameters. You can develop a custom widget to send a URL or you can use the Script Adapter to integrate other widgets with the website display.

13.3.2 Script Adapter widget

For complex integration between widgets and retrieving external data, use the Script Adapter widget. This widget is provided with Business Space.

The basic function of this widget is to receive a payload from an event and send that payload as an event. Both the payload and the event can be anything. This feature is most useful when you translate payloads between existing widgets on a page. One widget might send a payload with "ContactName" whereas another widget requires just "Name." In this instance, the Script Adapter can transform the payload as needed.

An extended function of the Script Adapter widget is to run actions that JavaScript can perform. For example, "ContactName" is not allowed to be empty as it is passed to another widget. The Script Adapter can either throw a JavaScript alert or open a dojo.dialog to alert the user that "ContactName" is empty. Additionally, the Script Adapter does not necessarily have to send a

payload. It can react to receiving a payload and run an xhr.post to save payload information to a server-side API.

13.3.3 Custom widget

For complete control of a widget, from what it does to what it presents to which payloads are available, you can create a custom widget.

When developing your own widget, you have few limitations. You can harness the full capabilities of JavaScript and the Dojo libraries. You can customize payloads and create useful events either on receiving or sending. You can interact with existing systems that provide a REST interface, or you can create your own iframe as an advanced Web Site widget. You can render data from other widgets or back-end systems as required. You can also create a settings interface to configure the widget differently on different pages.

The following is a list of things to consider when you create a custom widget:

- ▶ Events to send and receive
- ▶ Payloads to send and receive
- ▶ Calling the back-end services by using HTTP requests
- ▶ Authentication that might happen when calling the back-end services
- ▶ Capabilities of JavaScript
- ▶ Capabilities of Dojo
- ▶ Widget definition XML
- ▶ Widget settings and configuration
- ▶ Presentation of the widget on a page

13.4 Using Process Engine workflow integration

Tasks within a solution include workflows that are processed by IBM FileNet Process Engine. As a full enterprise business process management engine, Process Engine provides many useful integration capabilities. These integration capabilities include the customer component, web services, and database interaction.

13.4.1 Custom components

A custom component is a Java object that can be used to run any action in the process of a workflow. A Component Integrator makes a custom component available within the process of a workflow by monitoring a queue and passing information along to the custom component.

The custom component offers nearly limitless processing capabilities within a workflow. If you need to react to data in a workflow in a way not available in the Process Designer palette, use a custom component. The full capabilities of Java are available when you develop a component.

The following are some of the ways a custom component is useful:

- ▶ Converting a deadline date to an integer that represents minutes for use in a step deadline. Input: Time (date). Output: Integer (minutes).
- ▶ Calculating total compound interest on a monthly deposit over time. Input: Integer (months), Float (deposit), Float (rate). Output: Float (interest).
- ▶ Evaluating if two sets of integer arrays are equal, element by element. Input: Integer[] (first array), Integer[] (second array). Output: Boolean (evaluation).
- ▶ You have existing Java code that runs database lookups or interfaces with other systems, and would be useful in the process of a workflow. The input and output values vary, but you might need to consider a harness Java class as an interface to reuse your code.

13.4.2 Web services

Web services is a standard method of communication between two systems, typically over HTTP. Process Engine offers an interface to use external web services by providing or receiving data from within a workflow.

Web services have been a standard practice for many years. Therefore, you might find that you are integrating IBM Case Manager with an external system that provides a web service. If so, web services integration by using Process Designer and other documented Process Engine configuration is often the quickest method of integration.

The following are some of the reasons why web services might be useful:

- ▶ To have an external system that provides a web service to process information in the flow of a task.
- ▶ To interface with WebSphere Service Registry and Repository or Universal Description Discovery and Integration (UDDI) registries.
- ▶ To interface with WebSphere ILOG business rules.

13.4.3 Database interaction

The DbExecute system step in Process Designer allows a flexible method to run a stored procedure in a remote database.

If you store procedures created in your database, it is easy to implement the DbExecute system step as a step in your task workflow.

13.5 External data service

An extra point of integration for a Case Manager solution is an external data service. This service allows Case Manager to provide users with contextual property information when they are working with cases and case documents.

The following are common use cases for an external data service:

- ▶ Creating conditional choice lists where the values available in a property are dependent upon the value of another property. A common example is city and state where which cities are available in the city drop-down depends on the selected state.
- ▶ Populating fields based on a lookup such as account number.
- ▶ Complex data validation.

An external data service (EDS) is implemented as a RESTful service that handles JSON-based property update requests from Case Client. The EDS is called from the Case Manager API application that is deployed on the Case Manager server.

For more information about the external data services, see:

<http://pic.dhe.ibm.com/infocenter/casemgmt/v5r1m1/topic/com.ibm.casemgmt.development.doc/acmdv087.htm>



Integration with WebSphere ILOG JRules

This chapter describes the integration between IBM Case Manager and IBM WebSphere ILOG JRules. It outlines the benefits of using JRules to manage business rules in a case management system, and uses a simple example to illustrate how this works.

This chapter includes the following sections:

- ▶ Benefits of using rules engine software
- ▶ Incorporating a business rule in the case solution

This chapter does not cover the IBM WebSphere ILOG JRules product in detail. For in-depth information about ILOG JRules, see the Redbooks publication: *Patterns: Integrating WebSphere ILOG JRules with IBM Software*, SG24-7881.

Licensing information: The IBM Case Manager license program includes a support program for IBM WebSphere ILOG JRules. This is a limited use license bundle. For more information about the scope and limitations, see the software license agreements website at:

<http://www.ibm.com/software/sla/sladb.nsf>

14.1 Benefits of using rules engine software

There are many rules for business operations, government procedures, pricing calculations, eligibility checks, validations, underwriting, and fraud detection.

Often application software has some of these business rules embedded in the code or in the process definitions. This practice means that any changes to the business rules require the involvement of an IT engineer or software developer.

Rules engine software is designed to separate business rules from application software. Rules engine software provides a set of tools for defining, managing, testing, and running business rules. It also supports the integration of the rules into all kinds of applications. Using rules engine software, the business analysts can concentrate on defining and managing of the business rules. Likewise, the IT engineers and developers can focus on the application software.

For case management systems, human judgment and the business rules are key factors in deciding the outcome of cases. Using a rules engine to guide the case workers, and support the business processes, helps to get better outcomes.

Also, business rules typically change more frequently than the workflow processes. Separating the definition and deployment lifecycle levels of the workflow system and the rule logic results in more adaptive system overall.

Incorporating a rules engine into a case management system is useful because a change in the business rules affects the behavior of ongoing tasks and new ones.

The following are a number of ways in which rules engines can be used to enhance case management systems:

- ▶ Intelligently assign priority to cases.
- ▶ Intelligently assign case workers to tasks.
- ▶ Business rules within the case can be used to automatically create and assign tasks or trigger fully automated actions. The rules do so based on either external events, completion of other case tasks, or expiration of task deadlines.
- ▶ Apply rules to the key facts and information, and guide the responses based on that information.
- ▶ Simplify certain tasks through automating the decision logic.
- ▶ Increase consistency by using decision rules across similar cases.

IBM WebSphere ILOG JRules product family includes the Rule Execution Server (RES), which provides robust, scalable, and secure rules engine software for building and deploying rules-based applications. It also includes a powerful

development environment that is based on Eclipse for rules definition, called Rules Studio. The example in the next section uses Rules Studio.

The WebSphere ILOG JRules product family has a number of other powerful extension components to support web-based rules management, desktop rules authoring, testing, and simulation for non-technical users. An example of this capability is the ability to have complete control over the vocabulary that is presented to non-technical users. These users can then edit the business rules. At the time of publication, these components are not part of the license support bundle.

14.2 Incorporating a business rule in the case solution

This section shows a simple example of creating a rule by using the IBM WebSphere ILOG JRules Rules Studio.

Using the complaints management example, create a rule to return the customer rating, either “DIAMOND”, “GOLD”, or “SILVER”. This rating depends on two factors:

- ▶ The total transaction amount the customer has had with the company
- ▶ How long the person has been a customer with the company.

There are five basic stages in adding rules to an IBM Case Manager solution:

1. Design: Create the JRules project and design the business language and vocabulary.
2. Orchestrate: Define the flow of parameters through logic defined in rules.
3. Author: Create the business rules.
4. Deploy: Deploy the rule to the rules execution engine.
5. Integrate: Integrate the rule into an IBM Case Manager solution.

This integration uses the JRules Hosted Transparent Decision Service (HTDS), which is provided by JRules RES as a web services interface.

The rules are defined in a JRules Rule project, and are then deployed to the JRules RES. The rule projects are accessed from the FileNet Process Engine by defining an external source. This external source defines the URL for the location of HTDS Web Services Description Language (WSDL) on RES.

At run time, an IBM Case Manager task includes a step to run the business rules in the JRules project. The rules are run through the web service interface by

using the WSDL. Data fields are mapped and exchanged with the business object model (BOM) that are defined in the JRules project.

14.2.1 Getting started: Creating a JRules project

To create a JRules project, complete these steps:

1. Start the **Rules Studio** to create the business rules (Figure 14-1).

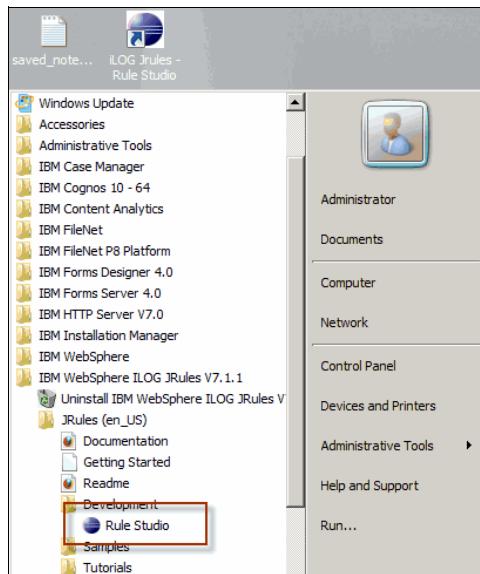


Figure 14-1 Starting Rule Studio

2. Create a JRules project for the business rules (Figure 14-2).

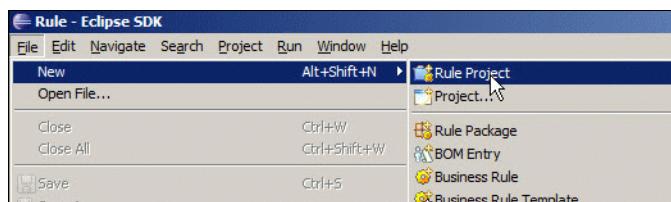


Figure 14-2 Creating a JRules project

3. Click **Next** with the default settings until you create a blank project as shown in Figure 14-3.

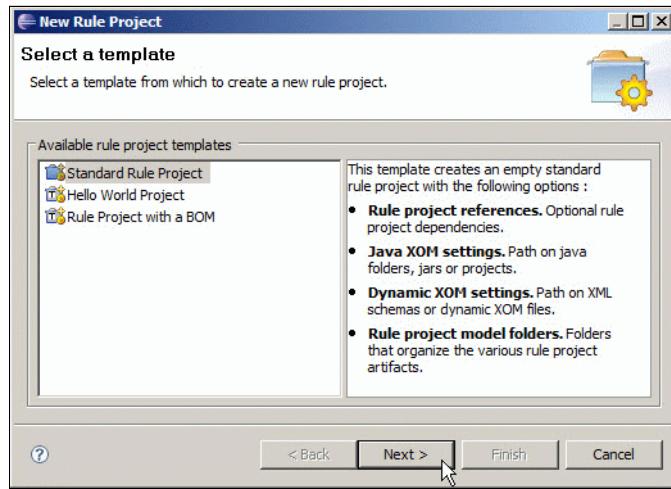


Figure 14-3 Creating a Standard project

14.2.2 Design: Defining the business model for your rules (XOM)

WebSphere ILOG JRules defines business terms for values that are used in the business rules. JRules makes the business rule definition more readable and easier to understand for non-technical users. Terms and phrases are stored in the BOM. The underlying definitions and technical data of the JRules are stored in the execution object model (XOM). This separation enables JRules to provide the natural language approach to rules authoring.

For example, the data structures defined in the complaint process can be imported into a JRules project. By default, every attribute of the model has an English phrase that is associated with it. So if the Complaint object has a submitDate field, JRules automatically creates the phrase "the submit date of the compliant" in the BOM. If you do not like the default mapping, replace it with your own, for example, "the complaint date".

For the simple rules in the complaint case, define a data model that includes the name of a customer, total transaction amount, and the year that the customer joined:

1. Create and import a sample XOM file:

- a. To create the simple XOM file, enter the content that is shown in Example 14-1 into a file, `complaintdata.xsd`.

Example 14-1 XOM file content

```
<?xml version="1.0" encoding="UTF-8"?>
<xss:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="http://ibm.com/example"
  targetNamespace="http://ibm.com/example"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xss:element name="Customer">
    <xss:complexType>
      <xss:sequence>
        <xss:element name="name" type="xs:string" nullable="true"
          minOccurs="0"/>
        <xss:element name="joiningYear" type="xs:int"
          nullable="true" minOccurs="0"/>
        <xss:element name="totalTransactions" type="xs:float"
          nullable="true" minOccurs="0"/>
      </xss:sequence>
    </xss:complexType>
  </xss:element>
</xss:schema>
```

You can use phrases like “the joining year of ‘the customer’” in the definition of the business rules with this XOM.

This example illustrates the way to set up such phrases with the use of a simple structure for customer. Alternatively, you can use a set of simple parameters rather than the complex type.

After the project is created, the Rule Project Map wizard shows all the steps that are needed to develop the rules and deploy them to the RES.

- b. Click **Import XOM** to import the file that you created into the project.

- c. Select **Dynamic Execution Object Model**, and click **Add External XSD** as shown in Figure 14-4.

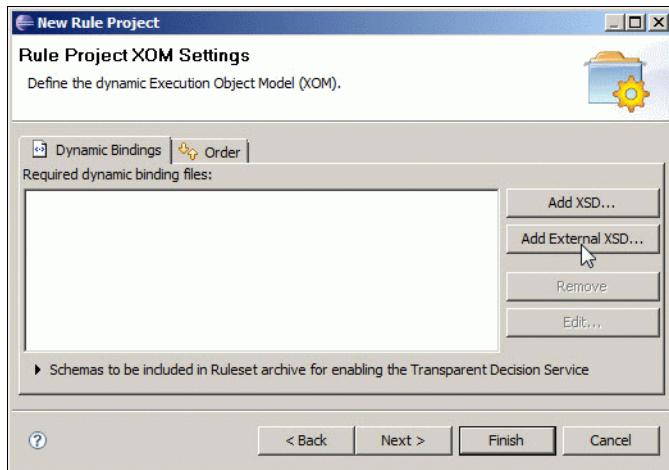


Figure 14-4 Loading the external XOM

One XOM was created, as shown by the (1) in the map in Figure 14-5. The next step is to create the BOM.

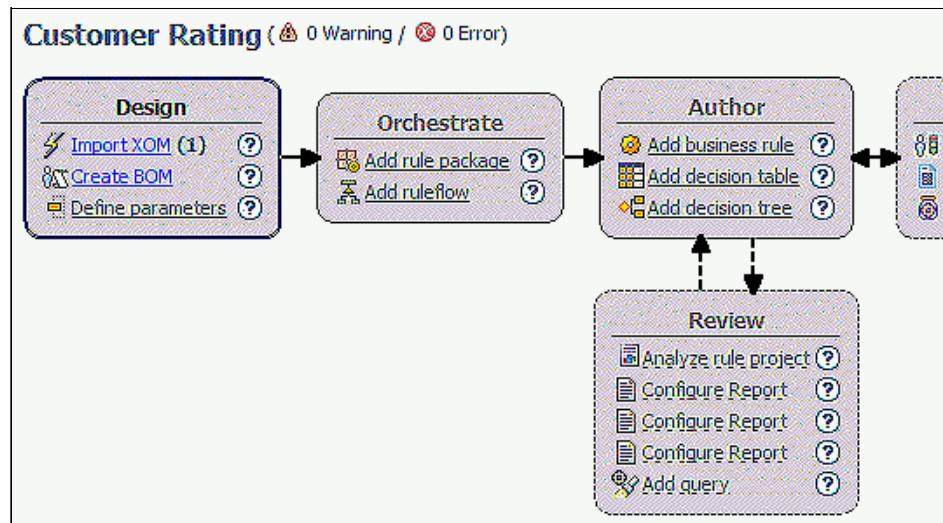


Figure 14-5 Rules map

2. Create the Business Object Model (BOM):
 - a. Click **Create BOM**, and use all defaults as shown in Figure 14-6.

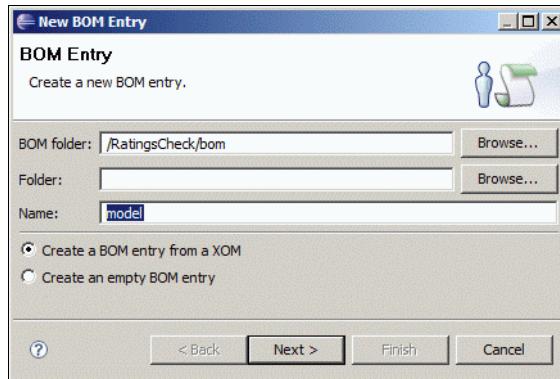


Figure 14-6 Creating a BOM

- b. Select the complaintdata.xsd file to be loaded. Be sure to select the check box for the customer data model (Figure 14-7) and finish the BOM creation process.

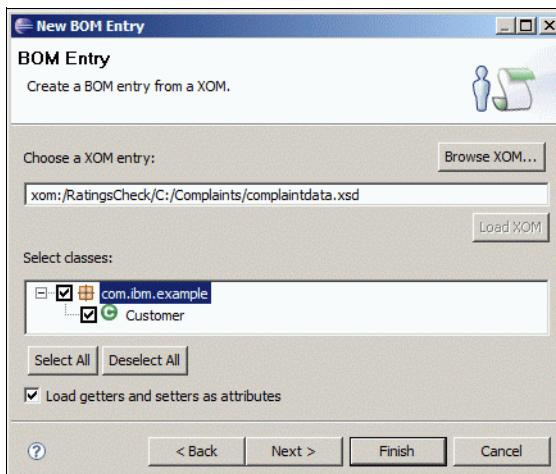


Figure 14-7 Completing a BOM entry

Remember: The BOM does not need to be based solely on process data. It can access data from other sources such as directly from databases, other systems, and even web services. For example, a dynamic lookup can check how busy the call center is to determine routing decisions.

- Following the wizard map, define all the parameters that are needed for the business rules. There are divided into three types (IN, IN_OUT, OUT) depending on whether they are input or output parameters, or both.

The example uses these types as shown in Figure 14-8:

- OUT - Rating
 - IN - Customer

Figure 14-8 Defining business rule parameters

In the example, customer is a complex type. For the type field, click the ... button and select **Customer** as shown in Figure 14-9.

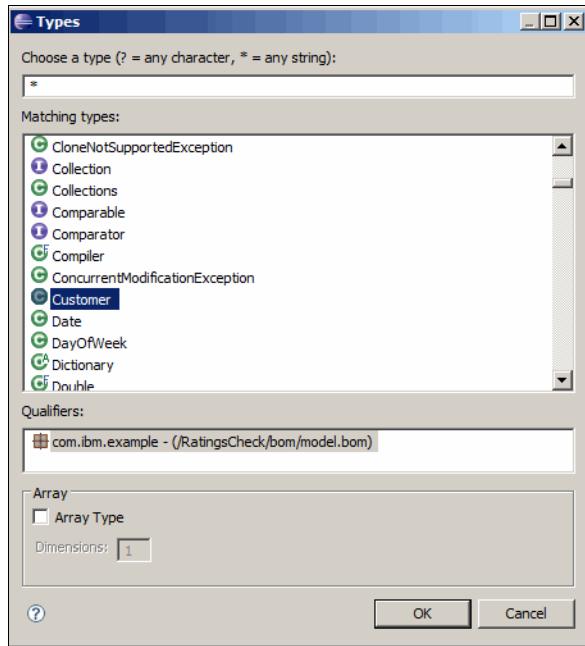


Figure 14-9 Parameter types

If the customer type is missing, go back and check the previous steps, especially the check box shown in Figure 14-7 on page 530.

For the example, the default value for customer rating is SILVER.

The **verbalization** value refers to the natural language term used to reference this parameter in the rules.

Restriction: At run time, the input parameters must never have null values.

14.2.3 Orchestrate: Creating a rule package and rule flow

Rule packages are used to group related rules together, and the rule flow defines the high-level flow of execution of the rules. To create them, complete these steps:

1. Following the map wizard, click to create a rule package. Enter a package name as shown in Figure 14-10.

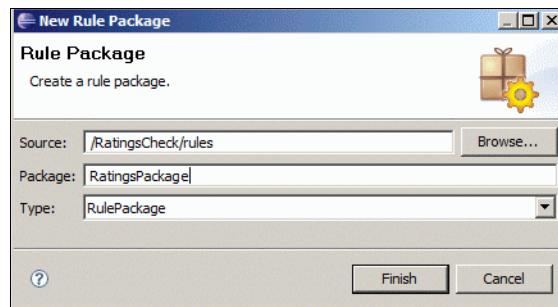


Figure 14-10 Creating a rule package

2. Click to create a rule flow that is evaluated by the rule web service as shown in Figure 14-11.

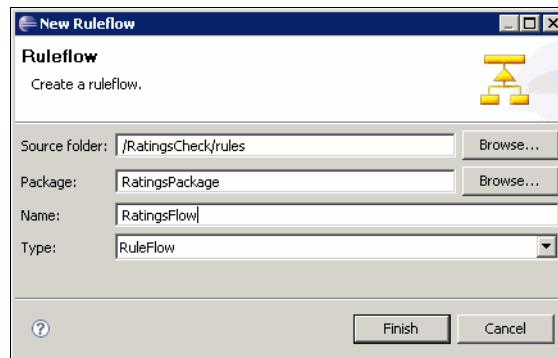


Figure 14-11 Creating a rule flow

3. Perform these steps to create the rule flow:
 - a. In the rule flow canvas, define a flow by dragging from the palette.
 - b. Insert Start (green) and Stop (red) markers to the canvas.
 - c. Drag your rule package from the Rule Explorer to the rules flow canvas, between the start and end markers. Use the connector to define the flow.

- d. Click the **Organization** icon as circled in Figure 14-12 and the system lines them up automatically.

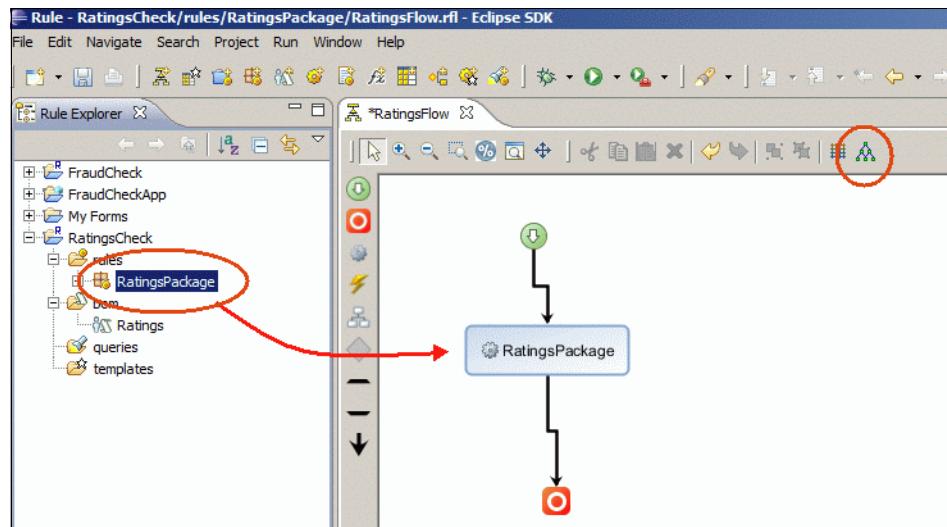


Figure 14-12 Lining up the rule flow

14.2.4 Author: Creating the business rules

Rules Studio allows the JRules developer to write rules, design rule templates, and organize the folders that are used to manage the rules.

The web and desktop authoring extension components are designed to allow non-technical users to author, manage, and deploy business rules.

Click **Add business rule** to create the business rules in the rules editor. Figure 14-13 shows the DiamondRatingRule created for the example.

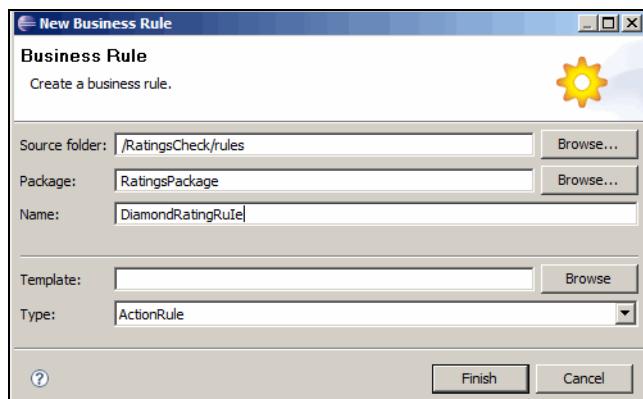


Figure 14-13 Creating a business rule

Figure 14-14 shows the code setup for the DIAMOND customer rating in the DiamondRatingRule.



Figure 14-14 Diamond rule

Figure 14-15 shows the code setup for the GOLD customer rating in the GoldRatingRule.

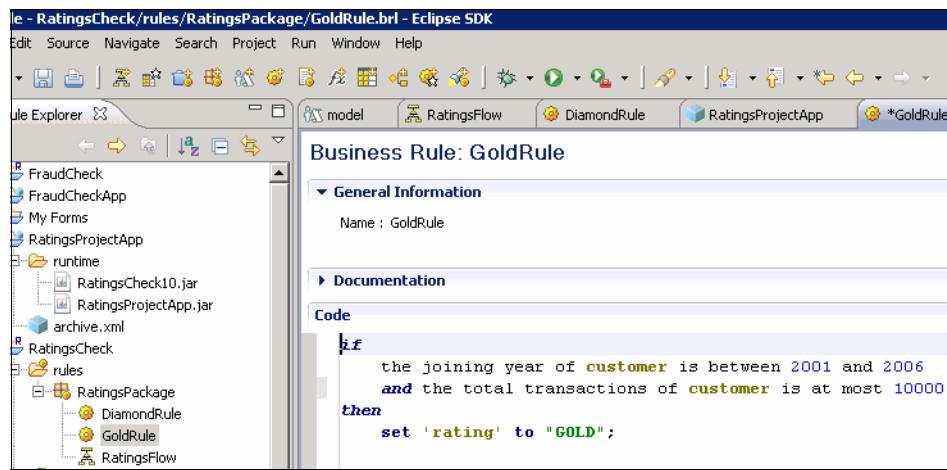


Figure 14-15 Gold rule

Besides the textual form (the code) for the rules, WebSphere ILOG JRules also supports several other formats, including decision tables and decision trees. Decision tables are used to manage large sets of symmetric rules, and decision trees graphically represent rules that have many asymmetric branches.

For more information about the different forms that are used for defining business rules, see the WebSphere ILOG JRules Information Center at:

<http://publib.boulder.ibm.com/infocenter/brjrules/v7r1>

14.2.5 Deploy: Creating the rule application project

The complete development cycle for a JRules project involves rules authoring, reviewing, and testing.

This simple example shows the deployment procedures:

1. Create a rule application project. The name must not be the same as the Rule Project created in step 14.2.1, “Getting started: Creating a JRules project” on page 526. The RuleApp properties page is then shown (Figure 14-16).

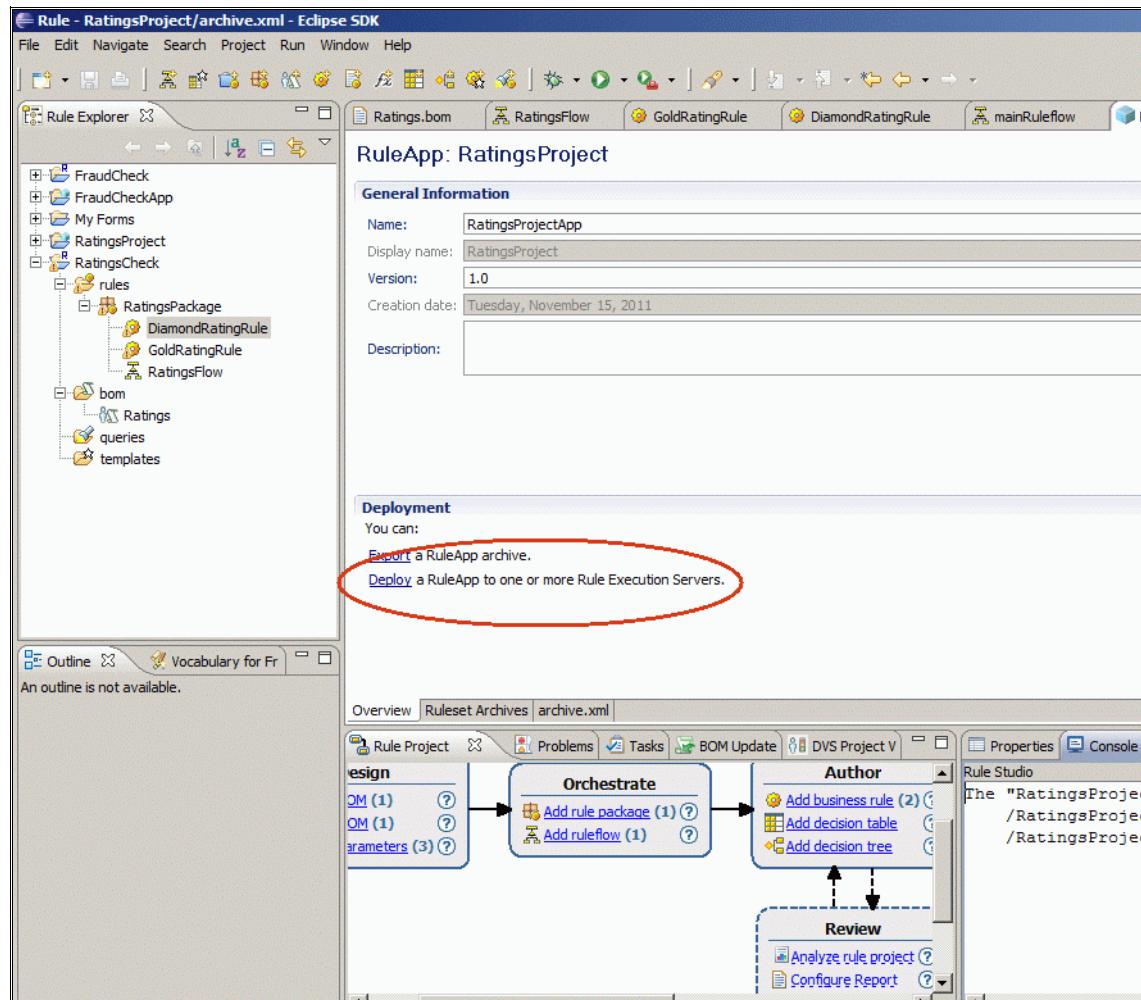


Figure 14-16 Deploying the rules application

2. Click **Deploy** to deploy the application.

3. Select the deployment type as shown in Figure 14-17

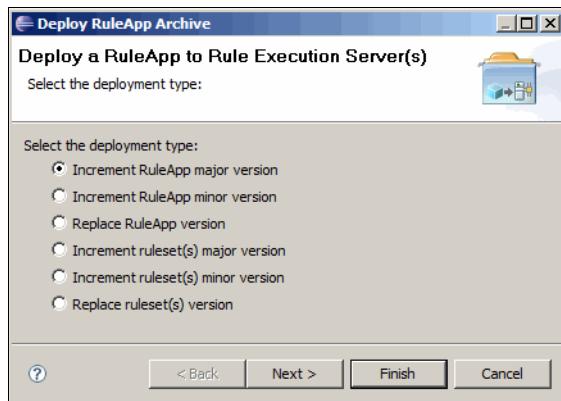


Figure 14-17 RuleApp versions

4. Select the other options as shown in Figure 14-18.

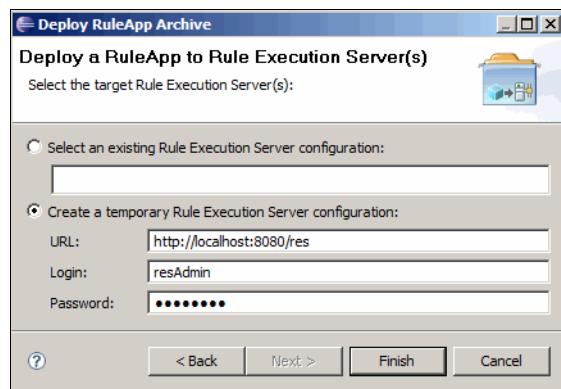


Figure 14-18 RES Server URL

The Console window on the lower right side shows the result of the deployment.

Remember: Ensure that Rules Execution Server (RES) is running before deploying, and make sure that the port details for the target deployment server is correct.

To configure the integration to the Rules project, you need the location of HTDS WSDL.

5. Log in to the Rules Execution Server console:
`http://localhost:8080/res`, username/password = resAdmin/resAdmin
6. Click the Explorer tab.
7. Expand **RuleApps** in the Explorer tree to see the deployed versions of your rule application as shown in Figure 14-19.



The screenshot shows the WebSphere Rule Execution Server (RES) console interface. The title bar reads "WebSphere. Rule Execution Server Console". The top navigation bar includes tabs for Home, Explorer (which is selected), Decision Warehouse, Diagnostics, and Server Info. Below the navigation bar, the breadcrumb path "Explorer > RuleApps" is displayed. On the left, a "Navigator" panel shows a tree structure with "RuleApps (2)" and "Decision Services" expanded. The main content area is titled "RuleApps View" and contains three buttons: "Add RuleApp", "Deploy RuleApp Archive", and "Update RuleApps". A section titled "RuleApps" displays the total number of RuleApps as 2. A table lists the two RuleApps: "FraudCheckApp" and "RatingsProjectApp", both version 1.0. The table includes columns for "Select All", "Name", "Version", "Creation Date", and "Number of ru". A "Name Filter" input field and a "View c" button are also present. At the bottom, a footer indicates "RuleApp 1 - 2 of 2" and provides navigation links for "prev 10 next 10".

Figure 14-19 RES console

- Click the link for the latest ruleset WSDL as shown in Figure 14-20. You need the WSDL to configure in FileNet Process Designer to start this rule as a web service.

The screenshot shows the 'Ruleset View' interface. On the left, there's a 'Navigator' pane with a tree view containing 'RuleApps' (2), including '/FraudCheckApp/1.0 (1)' and '/RatingsProjectApp/1.0 (1)', and 'Decision Services'. The main right pane is titled '/RatingsProjectApp/1.0/RatingsCheck/1.0'. It displays the following details:

Name	RatingsCheck
Version	1.0
Creation Date	Nov 15, 2011 10:05:24 PM GMT-08:00
Display Name	RatingsCheck
Description	
Status	✓ enabled
Debug	✗ disabled
WSDL	Get HTDS WSDL for this ruleset version Get HTDS WSDL for the latest ruleset version

Below this is a 'Ruleset Parameters' section with a single entry:

Direction	Name	Kind
	totalTransactionAmount	nativ

Figure 14-20 Getting the WSDL

- This action brings up the WSDL in the browser (Figure 14-21). Copy the URL for the location of the WSDL for use later.

The screenshot shows a web browser window with the address bar containing 'localhost:8080/DecisionService/ws/RatingsProjectApp/2.1/RatingsCheck?WSDL'. The page content is an XML document:

```

<definitions targetNamespace="http://www.ilog.com/rules/DecisionService">
  <types>
    <xsd:schema attributeFormDefault="unqualified" elementFormDefault="qualified" targetNamespace="http://www.ilog.com/rules/DecisionService">
      <xsd:import namespace="http://www.ilog.com/rules/param"/>
      <xsd:element name="DecisionServiceRequest">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element maxOccurs="1" minOccurs="0" name="DecisionID" type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:schema>
  </types>
</definitions>

```

Figure 14-21 WSDL for Rules application project

- Use the WSDL obtained for your rule application from the Rules Execution server to configure a partner link in FileNet Process Designer.

Tip: This URL contains a specific rule application version number. However, removing the RuleApp version automatically references the latest version of the RuleApp. The altered URL looks like:

```
localhost:8080/DecisionService/ws/RatingsProjectApp/RatingsCheck?WSDL
```

14.2.6 Integrate: Configuring web services to start the rule

The next step is to configure the step in the workflow task to start the web service for the rule.

Ensure that the FileNet Component Manager and the ILog JRules Execution Server (RES) are running before you configure the web service to start the rule.

Using the Complaints Management example, the customer's rating is based on the total transaction amount and how long the customer has been with the company. The rating in turn determines whether an outreach to the customer by a client manager is necessary.

The calculation of the rating can be done at various points in the case. It can also be configured to run as an individual task. For this example, the rating value is determined at the beginning of the case so that this information is available from the start.

A task to verify the complaint information at the start of the case already exists. Extend that task to include a step to get the customer rating from WebSphere ILOG JRules by using these steps:

1. Using Case Manager Builder, configure the Verify Case task, adding a Stub step to the system swimlane. Extend this stub in the next step by using the Process Designer to start the web services call to WebSphere ILOG JRules.

Figure 14-22 shows the Stub step, which is named CalculateRating.

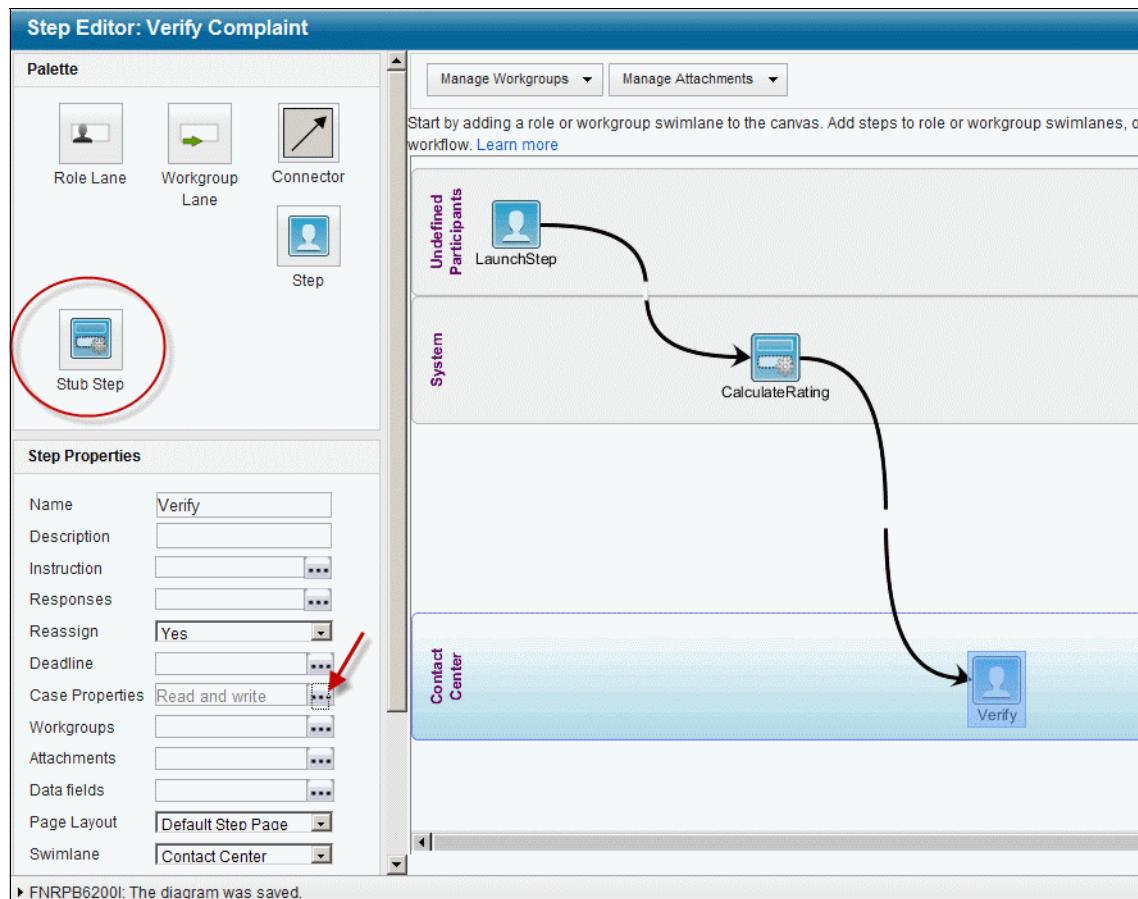


Figure 14-22 CalculateRating stub step

Ensure that all the necessary case properties are assigned to the steps, and then save and redeploy the solution.

2. Set up the partner link:
 - a. Log in to the Workplace XT and open the Process Designer.
 - b. Select the solution option from the menu, and navigate through the design object store to find and edit the **Customer Complaints** solution.
 - c. Select the **Verify complaint** workflow definition from the workflow list in the **View** menu.
 - d. Click the Web Services tab.
 - e. Select Partner Link, and configure it by clicking **Invoke**, and entering the WSDL URL as shown in Figure 14-23. This example uses a Partner Link called “Rules_TDS”.

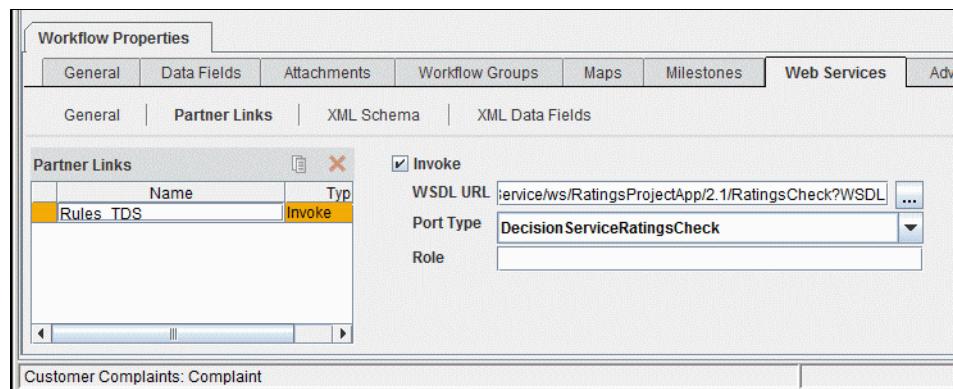


Figure 14-23 Setting up the partner link

3. Set up the parameters.

The main parameters needed to calculate the ratings should be there already, because they are part of the case data created by the Case Manager Builder:

- CustomerName - string
- Total transaction amount - float
- CustomerSince - integer
- CustomerRating - string

You need to add a number of workflow data field for this workflow. These fields are required for calling the Rules web services. They are not part of the case data, and therefore these data fields do not exist yet.

- DecisionID - string
- ILOGMessage - string
- count - integer

Figure 14-24 shows the data fields for the workflow. The default values ensure that none of the input parameters to the web service are null. The detail usage of each one is explained in the next step.

Data Fields					
Name	Type	Merge Type	Expression		
CC_ComplaintDescription	String	Default	""		
CC_CaseNumber	String	Default	""		
CC_ComplainReceivedDate	Time	Default	null(time)		
CC_CaseState	String	Default	""		
CC_Valid	Boolean	Default	true		
CC_SafetyCheck	Boolean	Default	true		
CC_CaseSource	String	Default	""		
CC_ComplaintCategory	String	Default	""		
CC_CustomerSince	Integer	Default	2011		
CC_TotalTransactionAmount	Integer	Default	100		
DecisionID	String	Override	"not null"		
ILOGMessage	String	Override	""		
count	Integer	Override	0		

Figure 14-24 Data fields for the workflow

Requirement: Ensure that none of the parameters that are used to start the web service have a null value at run time.

- Configure the web services Invoke step by clicking the **System** step, called CalculateRating, that was created in Case Manager Builder. Select the function **Invoke**, followed by **Assign** as shown in Figure 14-25.

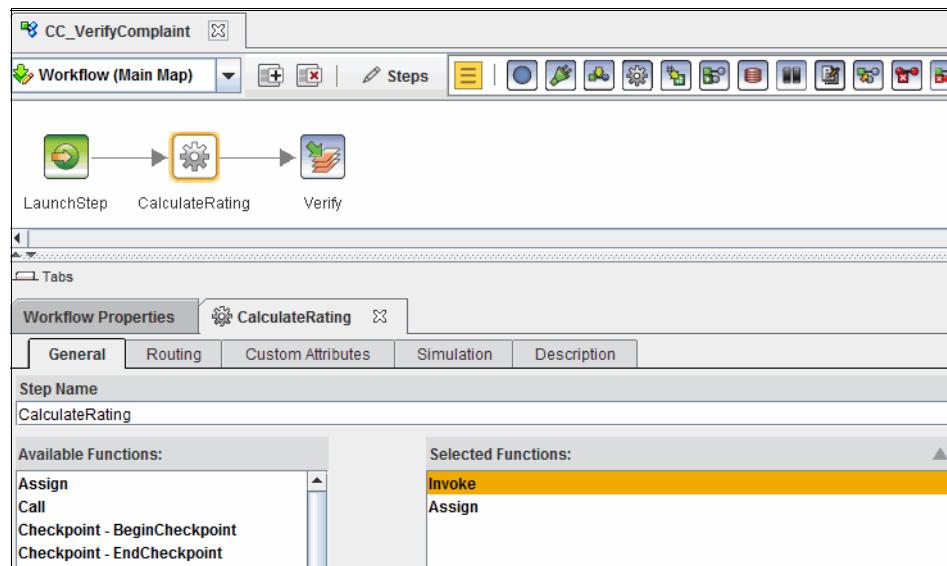


Figure 14-25 System step

- Configure the **Invoke** function.

The **Invoke** function starts the WebSphere ILOG JRules web service. Configure the parameters as needed for the rules application. These parameters must match the parameter definitions that were made earlier in the Rules Studio.

Process Designer does not support data fields that are complex structures. However, the **Invoke** function automatically maps to the data elements up to one level down.

For the example, the definition of the parameters in Rules Studio includes a structural object for Customer. It is not possible to use the customer structure directly for the **Invoke** function. Instead, the **Invoke** function automatically initializes to show the data elements next level down such as name, joiningYear, and totalTransactions.

Further information: The **Invoke** function can be configured in two different ways. The usual parameter mapping method is shown here. If you require more complex structures, use the XML message method.

For more information, see the FileNet P8 Information Center at:

<http://publib.boulder.ibm.com/infocenter/p8docs/v5r1m0/index.jsp>

Click **Integrating workflow into document management** → **Process Designer** → **About steps** → **Using Web Services in FileNet workflows** → **Web Services - XML message example**.

Figure 14-26 shows the **Invoke** function configuration.

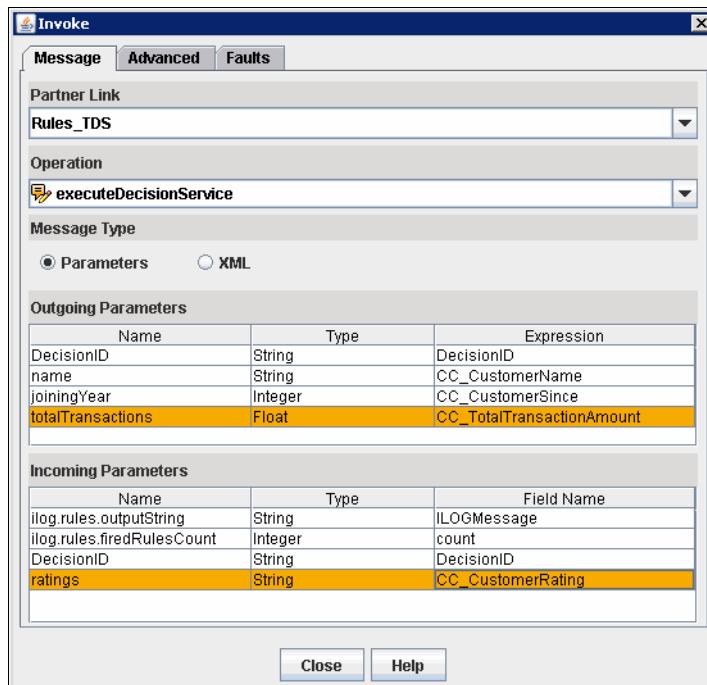


Figure 14-26 Configuring the *Invoke* function

Apart from the workflow data fields, the other parameters are as follow:

- DecisionID: This is an optional parameter on input for JRules, but is available by default when configuring the Invoke step for the integration.
- ILOGMessage: This returns messages from the rules engine.
- count: returns the count of rules fired.

Restriction: At run time, none of the outgoing parameters can be null. Null values can cause errors on the web service invocation. This restriction includes all the internal parameters, such as DecisionID.

6. Configure the **Assign** function.

The returned value for the customer rating is saved into the parameter CC_CustomerRating. This is a workflow parameter in this task. There is also a Case Data value, CC_CustomerRating, that holds the corresponding information for the case. By default, when a task is created in Case Manager Builder, the case data is mapped automatically to workflow parameters.

After you extend this task, map the value back into the Case Data so that it is shown on the Case Manager Client user interface. Use the assign function and the F_CaseFolder value, which identifies the case folder instance.

CustomerRating is the only parameter that changed in this example.

Set Assign as follows (Figure 14-27):

```
F_CaseFolder.CC_CustomerRating (case data) = CC_CustomerRating  
(workflow data)
```



Figure 14-27 Assigning rating to case data

7. Verify and save the solution, and exit Process Designer. Use Case Manager Builder to deploy and test the updated task (Figure 14-28).

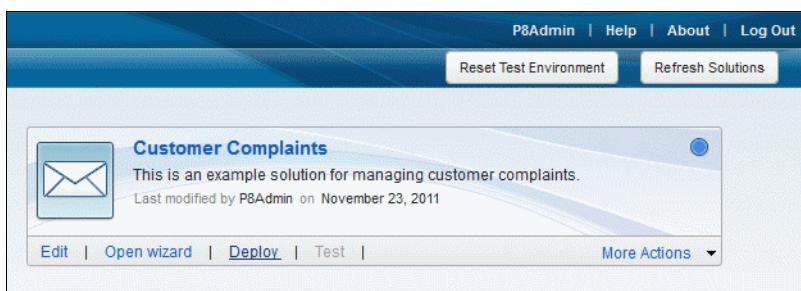


Figure 14-28 Deploying the updated task

Tip: To check whether a rule was run, use **View Statistics** (from the Ruleset view) from the JRules RES console.

Alternatively, use Process Administrator Tracker to make sure that WS_queue is not accumulating WS invocation requests.

8. Test the business rule by creating a complaint case in the Case Manager Client. Make sure that when the complaint case reaches the Contact Center representative, the customer rating value is calculated and displayed in the Case Data widget (Figure 14-29).

The screenshot shows the 'Case Data' interface with the 'Customer Information' section expanded. The fields and their values are:

- Customer Number: 1247093
- Customer Name: Fiona Dexter
- Address: Main Street
- Email: FDexter@email.com
- Telephone: (empty)
- Total Transaction Amount: 10,000
- Customer Since: 2,004
- Customer Rating: GOLD (This field is circled in red)

Below the main section are collapsed sections for 'Complaint Information' and 'Flags'.

Figure 14-29 Customer rating that is shown in the Case Manager Client



Integration with IBM Content Manager

The integration between IBM Case Manager and IBM Content Manager allows users to directly access documents that are stored in Content Manager more easily from case solutions. Companies that use Content Manager content repositories can use the integration to seamlessly use Content Manager documents in case solutions.

This chapter includes the following sections:

- ▶ Benefits of Content Manager integration
- ▶ Architecture of the integration
- ▶ Integration configuration steps
- ▶ Using Content Manager documents in case solutions

Important: This chapter refers to the product IBM Content Manager as Content Manager. This reference is not to be confused with the IBM FileNet Content Manager product.

15.1 Benefits of Content Manager integration

Before this integration, companies with content repositories in Content Manager were not able to use their Content Manager documents directly in case solutions. They had to depend on federation of their Content Manager content with IBM FileNet P8 system by using FileNet Content Federation Services.

With this integration, you can address and access the Content Manager documents in case solutions directly without moving them out of their repositories. Content Manager documents can remain in their own repositories, and do not need to be moved or federated with the FileNet P8 system.

With the integration, the Content Manager documents have the same set of features as FileNet P8 documents. Content Manager documents can be used for the following functions:

- ▶ Initiate cases
- ▶ Launch tasks
- ▶ Access natively in tasks
- ▶ Search and view within Case Manager Client
- ▶ Add to Case folders

Content Manager customers can add case management capabilities without impacting core content management infrastructure.

15.2 Architecture of the integration

This integration allows Content Manager documents to be accessed from IBM Case Manager server. Workers can use them in case solutions just like FileNet P8 documents are used.

For IBM Case Manager to access document content in Content Manager, it uses a case event handler that runs on the Content Manager server. The case event handler listens to the events that come from its own content manager server and from the Content Engine event handler. It uses this information to service various requests.

Figure 15-1 shows the architecture diagram of IBM Case Manager and Content Manager integration.

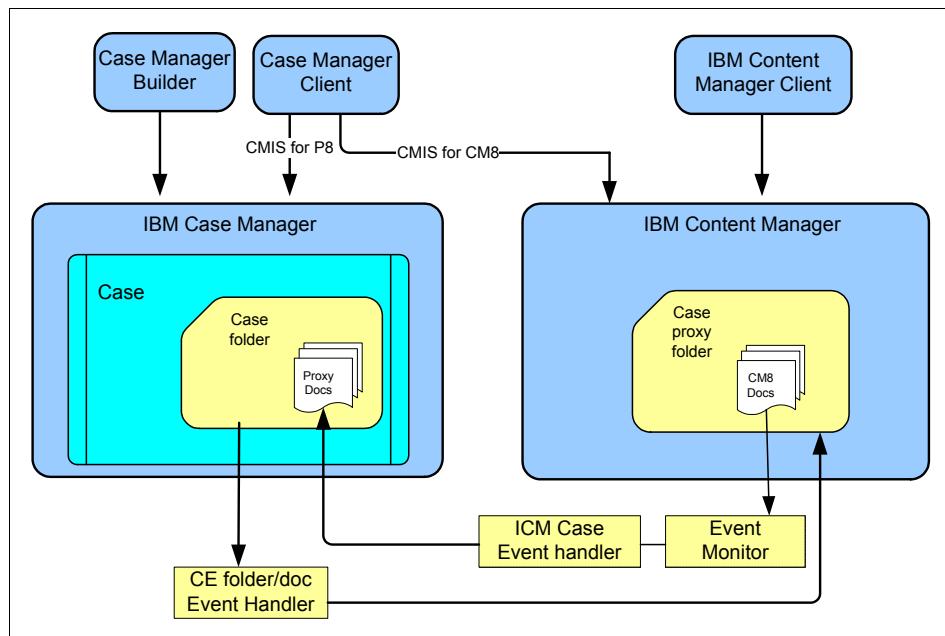


Figure 15-1 IBM Case Manager integrating with IBM Content Manager

15.2.1 Overview of the architecture

In the integration, the direct server-to-server integration is made possible by the following event handlers:

- ▶ Case event handler on Content Manager server
- ▶ Content Engine event handler on IBM Case Manager side

The Content Manager documents remain in their repository. Each document has a corresponding proxy object.

For a document in Content Manager, there is a *proxy document* in IBM Case Manager that contains the external reference back to Content Manager. The proxy document maintains only the external reference to Content Manager document, and has no user-defined properties.

For a case folder in IBM Case Manager, there is a *proxy case folder* in Content Manager. This folder contains various identifying properties such as the case ID and the object ID of the IBM Case Manager case folder, and has no user-defined properties.

The proxy case folder on IBM Content Manager maintains the following information:

- ▶ Case ID
- ▶ Case folder GUID
- ▶ Initiating document ID
- ▶ Object store symbolic name
- ▶ Solution name
- ▶ Case type
- ▶ Case folder name
- ▶ Case Year
- ▶ Case Month
- ▶ Case Day
- ▶ Case Hour
- ▶ Case Timestamp

Content Manager document (known as item type instance) browsing in IBM Case Manager is provided by the CM-CMIS API and IBM Content Manager OEM Toolkit API, which communicates with Content Manager

15.2.2 Interaction between IBM Case Manager and Content Manager

When a document is created in Content Manager, it can start a new case on the IBM Case Manager. This process typically involves these steps:

1. Content Manager Client creates a document (DocA) in Content Manager.
2. Case event handler on Content Manager receives the ItemCreate event.
3. Case event handler creates the proxy document (proxy DocA) in IBM Case Manager.
4. The creation of proxy DocA causes a new case (CaseClaimA) to be created in IBM Case Manager.
5. The Content Engine folder event handler listens to case folder property change turned to working.
6. The Content Engine folder event handler calls Content Manager to create a proxy folder (ACM_ClaimA) with the CaseID, GUID, and PidString of the originating DocA in Content Manager.
7. Content Engine folder event handler calls Content Manager to file DocA into the ACM_ClaimA folder.
8. Case event handler receives the FolderAdd event and checks the ACM_ClaimA folder. It finds that DocA is the originator of the case instance, so no further action is taken.

Figure 15-2 depicts the interaction steps between IBM Case Manager and Content Manager of starting a case when a document is added into Content Manager.

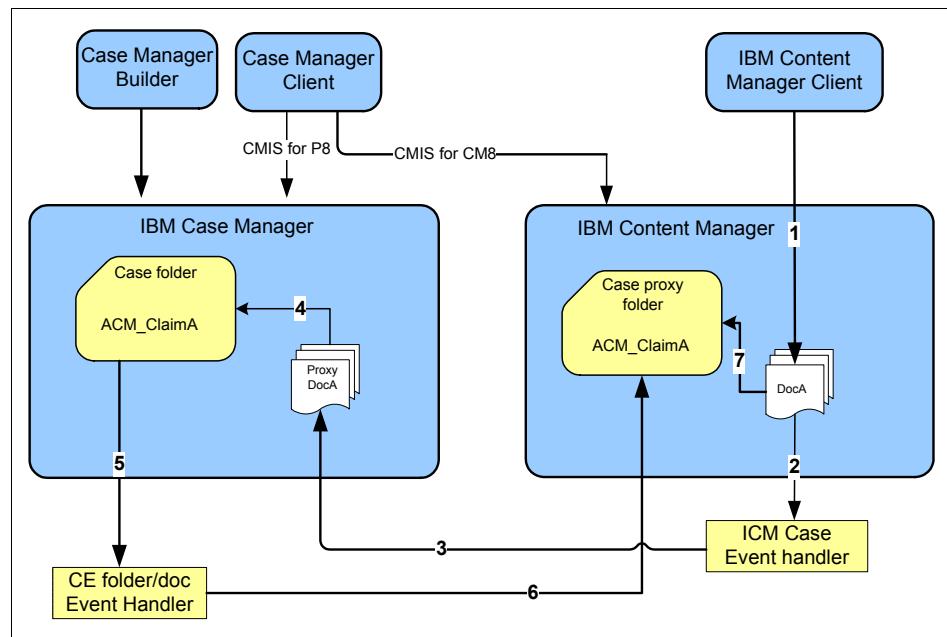


Figure 15-2 Starting a case when a document is added into Content Manager

A similar interaction between IBM Case Manager and Content Manager takes place for other actions such as:

- ▶ Manual case creation
- ▶ Adding a document to a case
- ▶ Removing a document from a case
- ▶ Deleting a case document
- ▶ Splitting a case

15.3 Integration configuration steps

This section describes the configuration steps for integrating IBM Case Manager with Content Manager.

Remember: The following steps are for IBM Case Manager Version 5.1 and IBM Content Manager Version 8.4.3 FixPack 1.

15.3.1 Prerequisites for integration

Before you configure the Content Manager and IBM Case Manager integration, the following prerequisites must be fulfilled:

- ▶ IBM Case Manager, Version 5.1 is installed and is working properly.
- ▶ IBM Content Manager Version 8.4.3 Fixpack 1 is installed or upgraded to this version, and it is working properly.
- ▶ Both IBM Case Manager and Content Manager share the LDAP server.
- ▶ Both the design and target object stores are created.
- ▶ Case Manager Builder and Case Manager Client are configured on the IBM Case Manager server.

If you want to maintain a separate project area for Content Manager configuration, create and configure separate project area and target object stores.

For more information about the integration steps, see the online Information Center documentation:

- ▶ <http://publib.boulder.ibm.com/infocenter/casemgmt/v5r1m0/topic/com.ibm.casemgmt.install.doc/acmcp021.htm>
- ▶ <http://publib.boulder.ibm.com/infocenter/cmgmt/v8r4m0/topic/com.ibm.administeringcm.doc/dcmce001.htm>

15.3.2 Overview of the configuration steps

The configuration is divided mainly into the Content Manager server steps and the IBM Case Manager server steps.

From a high level, the following configuration takes place:

1. On the Content Manager Server:
 - a. Run the Content Engine Client to copy the Content Engine files to the installed location of Content Manager.
 - b. Set up the user accounts and permissions as necessary.
 - c. Configure single sign-on (SSO) for Case Manager Client, CM-CMIS, and Content Manager.
2. On the IBM Case Manager server:
 - a. Start IBM Case Manager administration client and complete these tasks:
 - Configure design and target object store.
 - Configure Content Manager host properties.

- Create CM-CMIS WAR file.
 - Deploy CM-CMIS WAR file.
3. Run the Configuration Manager for Content Engine to deploy the FileNetEngine.ear file.
 4. Start the Event monitor and Case Event Handler on Content Manager server.

15.3.3 Detailed configuration steps on Content Manager server

Perform these steps to configure from the Content Manager server side for the integration:

1. Install the IBM FileNet Content Engine Client on the Content Manager server where the system administration client is located and where the case event handler runs. Make sure that the version of the Content Engine Client is the same as that is used by IBM Case Manager.

The Content Engine Client is required by the case event handler that runs on the Content Manager server. It is used to communicate with the IBM Case Manager server.

2. Set up the user IDs and permissions on the Content Manager server.

Integration requires three user IDs on the Content Manager side for various interactions from IBM Case Manager. Make sure that the following user IDs are created in the Content Manager server:

- icmadmin: The Content Manager administrator user ID needed for Content Manager event monitor. This ID is a default user ID created in Content Manager system. Verify that it is already there.
- icmconct: The shared database connection ID on Content Manager. This ID is a default user ID created in Content Manager system. Verify that it is already there.
- icmintid: The integration user ID that IBM Case Manager uses for Content Engine event handler and Case Manager Representational State Transfer (REST) API. This user is mandatory for IBM Case Manager integration, and must be created at the operating system level.

The Content Engine Event Handler and Case Manager REST API are authenticated with a common icmintid user. This user has its credentials encrypted into a Custom Object in the development environment (project area, target object store). Therefore, the user account must be configured on the Content Manager server.

Use the Content Manager System Administration tool (SysAdmin tool) to configure the user:

- a. If icmintid does not exist, complete the following steps:
 - i. Start the Content Manager System Administration tool.
 - ii. Click **Authorization** → **Privilege Sets node**.
 - iii. Copy the ClientUserAllPrivils privilege set into a new one named ICMIntPrivSet.
 - iv. Add the SystemSuperDomainAdmin privilege to this privilege set and save it.
 - v. Click **Authentication** → **Users node**.
 - vi. Right-click **Users node** and select **New**.
 - vii. Enter the user name, icmintid, as shown in Figure 15-3. This is the integration user ID.

The screenshot shows the 'User Properties ICMINTID' dialog box. It has tabs for 'Define Users', 'Assign to Groups', and 'Set Defaults'. The 'Define Users' tab is selected. The fields are as follows:

- User name: ICMINTID
- User description: ICMINTID
- Additional information: (empty)
- Password: ****
- Confirm password: ****
- Check box: Obtain from LDAP (unchecked)
- Check box: Use system password (unchecked)
- Password expiration:
 - At next logon (radio button)
 - After 307 Days (radio button)
 - Use system default (radio button)
 - Never expires (radio button)
- Maximum privilege set: IcmIntPrivSet
- Create Privilege Set... button

Figure 15-3 Creating icmintid, the integration user ID in Content Manager

- viii. Enter its password.

- ix. Select **ICMIntPrivSet** as the Maximum privilege set.
 - x. Select the appropriate password expiration option.
 - xi. Click the Set Defaults tab and select **PublicReadACL** for the Default item access control list.
 - xii. Click **OK** to save the user.
- b. Ensure that the Content Manager administration user (icmadmin) is added to the user list for the operating system. This user must also be added to the Database Administrators group (or the primary group of the DB2/Oracle instance owner). It is needed to configure the Content Manager host properties.
- c. Make sure that the same administration user (icmadmin) who performs the Content Manager bootstrap process has the appropriate permissions in the Content Manager System administration client:
- i. Under the authentication node of the server, select **Users**.
 - ii. Select the Content Manager administration user, **icmadmin**.
 - iii. Set the Maximum privilege set field to be **AllPrivs**.
 - iv. Click the Set Defaults tab.
 - v. Leave the default item access control list, **PublicReadACL**.
 - vi. Click **OK** to save.
3. Configure SSO for Case Manager Client, CM-CMIS, and Content Manager from the Content Manager System administration client:
- a. Change the Database Shared Connection ID to the appropriate ID, in this case, `icmconct` (Figure 15-4 on page 558):
 - i. Click **Tools → Manage Database Connection ID → Change Database Shared Connection ID**.
 - ii. Click to clear the **Password is required for all users** option.
 - iii. Optionally reenter the shared database user ID if the `icmconct` user has changed.
 - iv. Click **OK** to save.

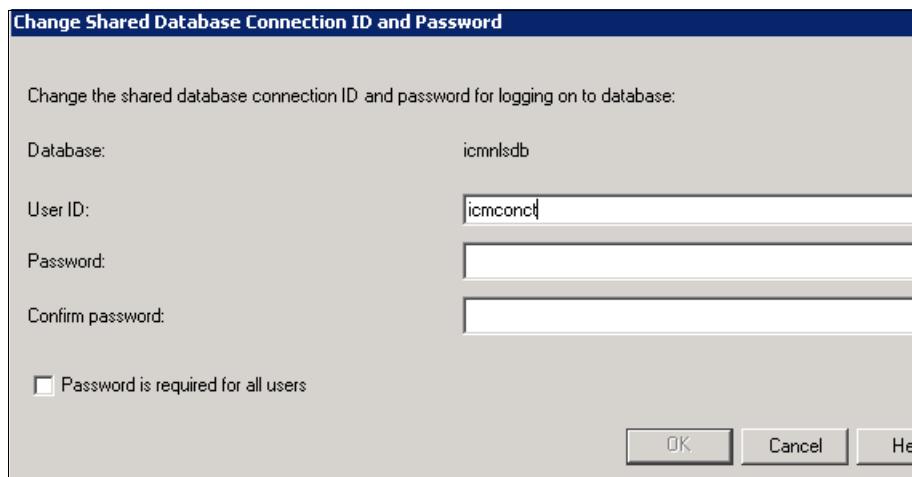


Figure 15-4 Change Shared Database Connection ID to *icmconct*

- b. Configure library server as shown in Figure 15-5 on page 559:
 - i. In the navigation pane, expand **Library Server Parameters** and select **Configurations**.
 - ii. Right-click **Library Server Configuration** in the right pane and select **Properties**.
 - iii. Select **Allow logon without warning** for the **Max user action** field.
 - iv. Select **Allow trusted logon**.
 - v. Click **OK** to save the changes.

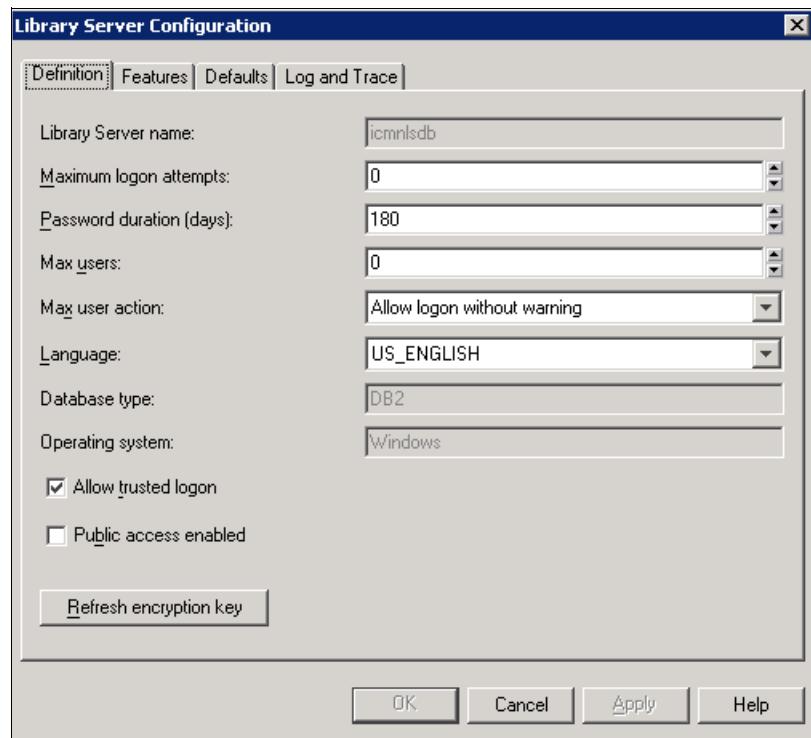


Figure 15-5 Configuring the library server

- c. Configure LDAP:
 - i. Click **Tools** → **LDAP Configuration**.
 - ii. Click the **LDAP** tab and select **Enable LDAP User import and authentication**.
 - iii. Click the **Server** tab and enter your LDAP server information.

iv. Click **OK** to save it (Figure 15-6).

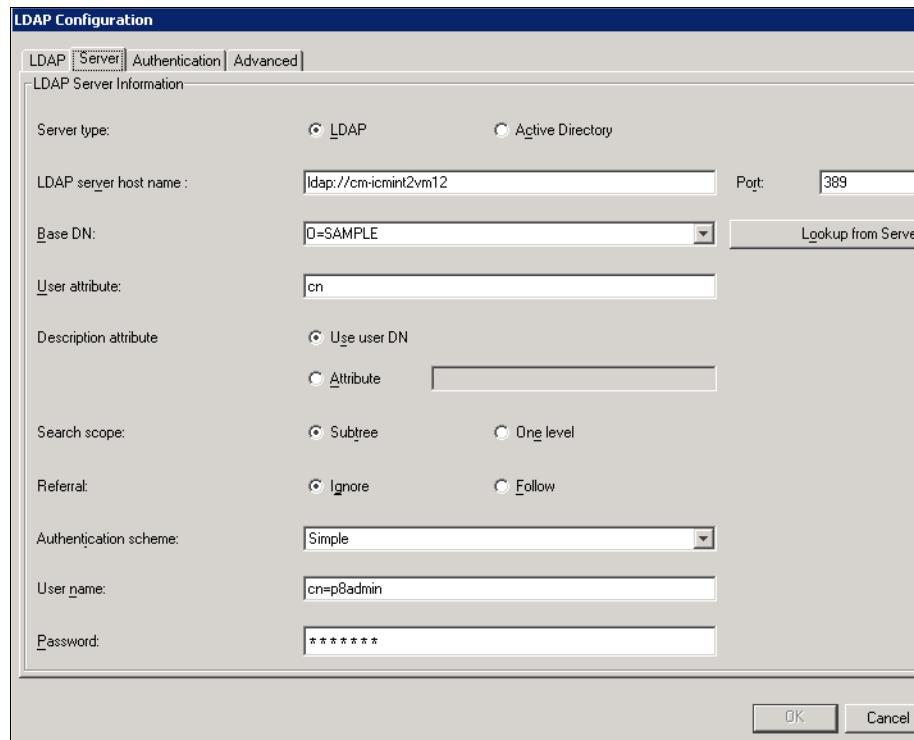


Figure 15-6 Configuring LDAP

d. Create a privilege set:

- i. Expand the **Authorization** node and select **Privilege Sets**.
- ii. Right-click the **ClientUserAllPrvs privilege** set and click **Copy** → **Advanced**.
- iii. Enter a new name for the privilege set (ex: SSOPrivSet).
- iv. Select the **AllowTrustedLogon** privilege.

- v. Click **OK** to save it (Figure 15-7).

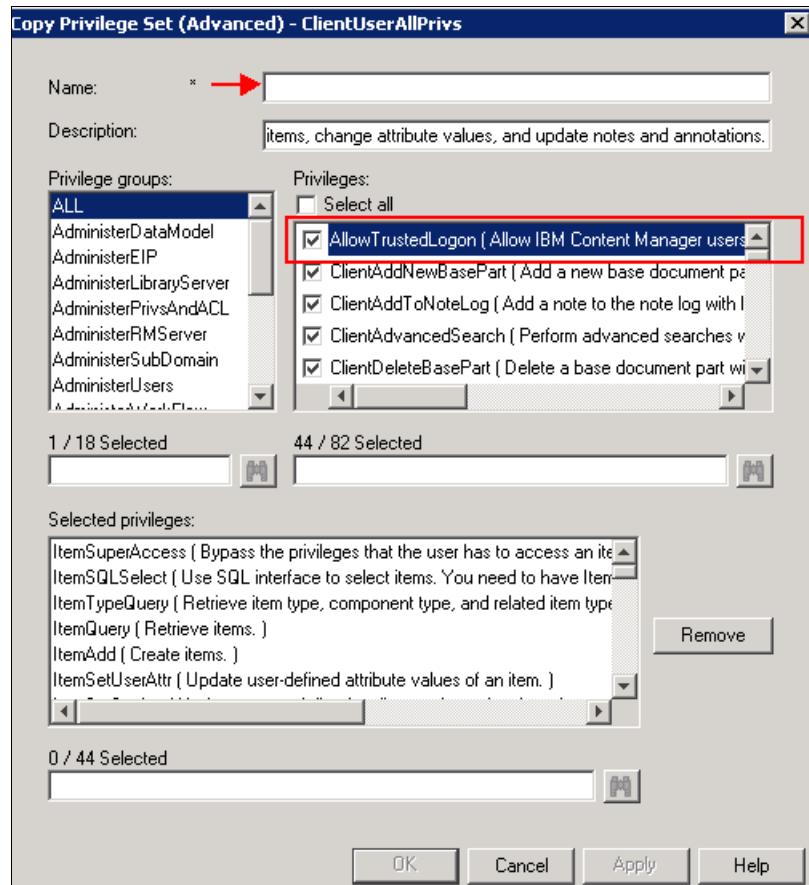


Figure 15-7 Copy privilege set (advanced)

- e. Configure authentication settings for the imported user:
- Expand the **Authentication** node, right-click **Users** and select **New**.
 - Set Maximum privilege set field to the privilege (SSOPrivSet) set in the previous step.
 - Set **Password expiration** to **Never expires**.
 - Click **LDAP** and select the user that you want to import. Then click **OK**.
 - Click the **Set Defaults** tab and select a **Default access control list**.

vi. Click **OK** to save the user (Figure 15-8).

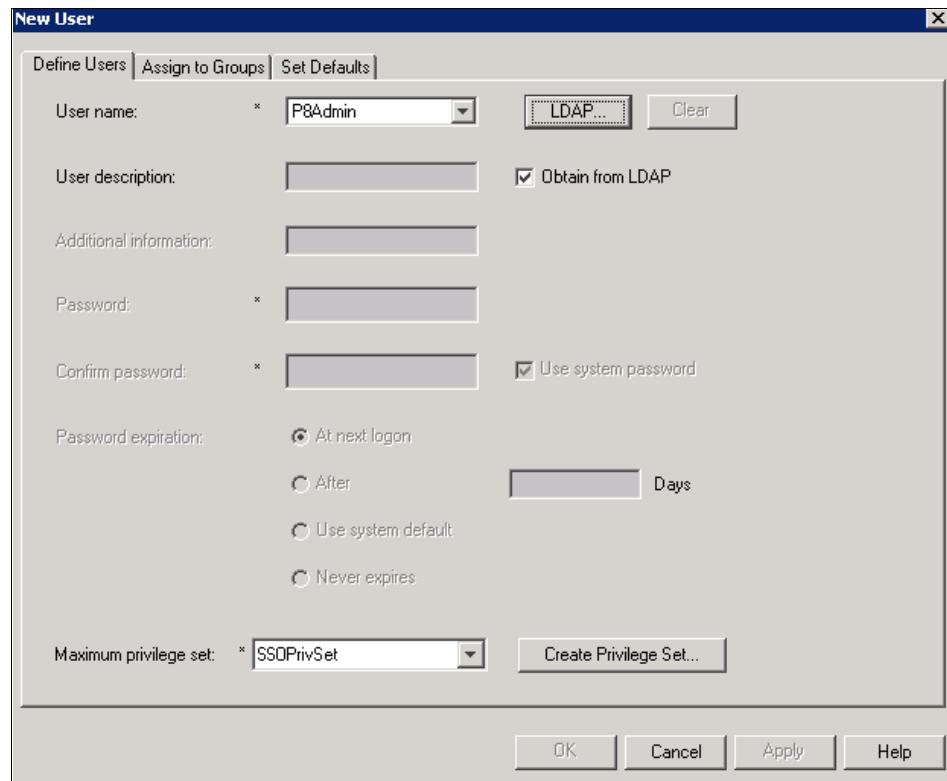


Figure 15-8 Configuring the authentication setting of an imported user

4. Configure DB2 or Oracle for Content Manager system administration authentication from IBM Case Manager administration client for the bootstrap process.

The Content Manager administrator user must have sufficient rights on the Content Manager databases for authentication from IBM Case Manager administration client to perform all read and write operations.

For Linux/UNIX, make sure that the user is added to the primary group of the DB2 or Oracle instance owner.

15.3.4 Detailed configuration steps on IBM Case Manager server

Follow these steps to configure the IBM Case Manager server side for the integration:

1. If you use Oracle for your Content Manager server database, complete the steps 1 and 3 from the document at:

<http://publib.boulder.ibm.com/infocenter/casemgmt/v5r1m0/topic/com.ibm.casemgmt.installng.doc/acmcp023.htm>

2. Add the tasks in the IBM Case Manager administration client profile.

The steps here assume that you have already created a default project area profile and configured the design and target object stores. These steps configure it to work with Content Manager.

If you want to maintain a separate project area for Content Manager configuration, create a project area and a separate target object store, and configure them. You can maintain different project area configurations, one for FileNet P8 repository and one for Content Manager repository. You do not have to use the default project area (`dev_env_connection_definition`) for the Content Manager repository.

Make sure that the WebSphere Application Server is running on both IBM Case Manager and on Content Engine.

In the IBM Case Manager administration client profile that you created, make sure that the following tasks are added:

- Configure the Content Manager host properties.
- Create the Content Manager CMIS WAR file.
- Deploy the Content Manager CMIS application.

3. Configure the Content Manager host properties:

- a. Double-click the **Configure Content Manager Host Properties** task. This action opens a panel in the right pane.

This task does the following tasks:

- Creates the required Content Engine event actions and subscription
- Creates the `CmAcmCM8IntegrationData` custom object in the IBM Case Manager folder that contains the Content Manager host connectivity properties
- Runs the bootstrapping configuration of the Content Manager server to create the attributes for the proxy case folder. It also creates the `ICMPROXYFDRS` item type, and assigns the event subscriptions to it.

- b. Enter the appropriate field values for the task as shown in Figure 15-9 on page 565:
- **Deployment type:** Leave it as the default, **Standard**, unless you have IBM Case Manager configured for Network Deployment or Cluster.
 - **Database type:** If you select Oracle RAC for a cluster environment, manually configure the JDBC URL to include the correct attribute values. These values include host name and port number. This string is set in the Content Engine custom object.

If you specify DB2 or Oracle, the JDBC URL is constructed automatically at run time, so the Content Engine custom object's CmAcmsJDBCURL property is empty.
 - **Connection Definition:** Select the connection definition that points to the project area and target object store that are related to the Content Manager library server. This target object store houses the Content Manager host connection properties.
 - Enter the Content Manager administration user name and password. The user has the AllPrivils privilege set configured in the Content Manager server. This user is authenticated for the Content Manager bootstrap sequence.
 - Enter the library and schema name for your Content Manager server. This information can be found in the Content Manager System administration client, or in the /cmgmt/connectors/cmbicmsrvs.ini file in the Content Manager server installation folder.
 - Enter the Integration ID, icmintid, created earlier. This user is used by both the Case Manager REST API and the Content Engine event handler to authenticate into the Content Manager server.
 - Enter the Shared database connection ID and password fields. The Content Manager default is icmconct.
 - Do not change the JDBC driver's default value of com.ibm.db2.jcc.DB2Driver for DB2 or oracle.jdbc.driver.OracleDriver for Oracle.
 - Enter the maximum size in megabytes of the log file. If the log file exceeds this size, a new log file is created with an incremental numeric suffix.
 - Enter the remaining Content Engine and WebSphere Application Server node/cell settings that are applicable to your environment.
 - Leave the **Restart** option cleared. You can restart the WebSphere Application Server after all the tasks are completed.

Configure IBM Content Manager Host Properties ▾

This task configures the connection to the IBM Content Manager host server. This task is required if you use IBM Content Manager as the content repository for IBM Case Manager. Save your changes and run the task to apply your settings.

Deployment type:	Standard
Database type:	DB2
Connection definition:	dev_env_connection_definition
IBM Content Manager administration ID:	icmadmin
IBM Content Manager administration password:	***** Confirm: *****
IBM Content Manager host name:	cm-icmint1vm18
Port:	50000
Library server name:	icmnlsdb
Schema name:	icmadmin
IBM Content Manager integration ID:	icmintid
IBM Content Manager integration password:	***** Confirm: *****
Shared database connection ID:	icmconct
Shared database connection password:	***** Confirm: *****
Log file size (MB):	5
JDBC driver:	com.ibm.db2.jcc.DB2Driver
Content Engine Application server administrator user name:	P8Admin
Content Engine application server administrator password:	***** Confirm: *****
Content Engine application server name:	cm-icmint2vm12
Content Engine application server SOAP port:	8880
Content Engine application server cell:	P8Node01Cell
Content Engine application server node:	P8Node01
Content Engine application server name:	server1
Script:	C:\Program Files (x86)\IBM\CaseManagement\configure\scripts\configCEWAS.tcl
Temporary directory:	C:\Program Files (x86)\IBM\CaseManagement\configure\tmp

Figure 15-9 Content Manager host properties configuration in IBM Case Manager administration client

- c. Save the configuration.
- d. Run the configuration by either right-clicking the task or clicking **Run toolbar** in the upper right.

The console output indicates whether there are any errors.

This task sets the integration type of the target object store to Content Manager for either a development environment or a production target object store.

This task can be run again if you must configure a different target object store. It can also update the same target object store with different Content Manager host connection values.

4. Create Content Manager WAR file:

- a. Double-click the **Create Content Manager CMIS WAR File** task.

This task has the following functions:

- Update the property values of the `cmppathservice.properties` file within the `cmcmis.war` file.
 - Base64 encode (per Content Manager Java API requirement) the Shared database user ID and password into the `cmbicmenv.ini` file from the `cmcmis.war` file.
- b. Enter the appropriate field values for the task as shown in Figure 15-10. Enter the same Library and Schema names (names are case-sensitive) that you entered in the previous task.

Create IBM Content Manager CMIS WAR File ▾

This task adds the IBM Content Manager library server name and library server schema name to the IBM Content Manager after you run the Configure IBM Content Manager Host Properties task, and before you run the Deploy Content Manager task to apply your settings.

Library server name:	<input type="text" value="cmnlisdb"/>
Schema name:	<input type="text" value="icmadmin"/>
Shared database connection ID:	<input type="text" value="icmconct"/>
Shared database connection password:	<input type="password" value="*****"/> Confirm: <input type="password" value="*****"/>

Figure 15-10 Creating the CMIS war file

- c. Save and run the task.
5. Deploy the Content Manager CMIS application.

This task packages up all the `cmcmis.war` components into a `cmcmis.war` file and deploys it into the WebSphere Application Server.

To deploy the `cmcmis.war` file:

- a. Double-click the **Deploy Content Manager CMIS Application** task.

- b. Enter the appropriate field values for the task (see Figure 15-11):
- Enter the path to where the war file will be created.
For the example, enter `c:\Program Files (x86)\IBM\CaseManagement\configure\deploy\cmcmis.war`
 - Enter the name of the application that will be represented inside the WebSphere Application Server applications list.
For the example, enter `cmcmis`.
 - Enter the application server node and name that you entered in the Configure Content Manager Host Properties task.
For the example, select **P8Node01**.
 - Enter the appropriate application server virtual host, path to the deployment script, and temp directory.
For the example, select **server1** as the application server, and **default_host** as the application server virtual host. Select the script file `\IBM\CaseManagement\configure\scripts\deployCMISApplication.tcl`.
 - Leave the **Restart** option selected unless you plan on running other tasks.

Deploy IBM Content Manager CMIS Application ▾

You must run the Create IBM Content Manager CMIS WAR file task before you run this task. This task deploys the IBM server. Deploying the WAR file enables Case Manager applications to retrieve information from the Content Manager Manager.

Deployment type:	Standard
IBM Content Manager CMIS WAR file path:	<code>C:\Program Files (x86)\IBM\CaseManagement\configure\deploy\cmcmis.war</code>
IBM Content Manager CMIS application name:	<code>cmcmis</code>
Application server node:	<code>P8Node01</code>
Application server name:	<code>server1</code>
Application server virtual host:	<code>default_host</code>
Script:	<code>C:\Program Files (x86)\IBM\CaseManagement\configure\scripts\deployCMISApplication.tcl</code>
Temporary directory:	<code>C:\Program Files (x86)\IBM\CaseManagement\configure\tmp</code>
<input type="checkbox"/> Restart Application Server	

Figure 15-11 Deploying the CMIS application

- c. Save and run the task to deploy the `cmcmis.war` file.

6. Install the Content Manager Java API and properties files for FileNet Content Engine:
 - a. Run the IBM Case Manager-client installer for your operating system type. For example, Windows is 5.1.0-ICM-CLIENT-WIN.EXE.
 - b. Click **Next** through the options and stop at **Select Products** option.
 - c. If you select the **Content Engine** option, the client installer uses the default Content Engine installation location. If you select the **Other Applications** option, you are prompted for the installation location.
 - d. Click **Next** through the subsequent options to complete the installation.
 - e. Run the Configuration Manager for Content Engine to deploy the FileNetEngine.ear that contains the Content Manager Java API .jar and property files.

15.3.5 Starting event monitor and case handler on Content Manager

The case event handler of IBM Case Manager watches and processes item type events and it runs on Content Manager. The Content Manager v8.4.3 Installer deploys the necessary .jar and configuration files, and scripts to support the case event handler.

For the case event handler to run, you must start the event monitor on Content Manager, then start the case event handler.

Starting and stopping event monitor on Content Manager

To start the event monitor, run the following commands in the *<IBM Content Manager install path>/bin* directory (for example /opt/IBM/db2cmv8/bin in UNIX and c:\Program files\IBM\db2cmv8\bin for WINDOWS):

1. **cmbenv81.bat** on Windows or **cmbenv81.sh** on UNIX to set up the required class path.
2. **ICMEvent -d <database> -u <userid> [-s <schema>]**
where:
 - <database> is the library server database name.
 - <userid> is the administrator user ID of the library server database name.
 - <schema> is an optional parameter in the command line. If the administrator user does not set it, the event monitor uses the user ID as the schema name.
3. Enter the user password.

To stop the event monitor, enter **QUIT** the window console.

Starting and stopping case event handler on Content Manager

To start the case event handler in IBM Case Manager, run the following commands in the /IBM/db2cmv8/bin directory:

1. `CaseHandler -d <Content Manager database> -u <user ID> -cu <IBM Case Manager user ID>`
2. Enter the Content Manager password of the user ID to connect to the Content Manager server.
3. Enter the password of the IBM Case Manager user ID to connect to an IBM Case Manager server.

Have the case handler run test connections to Content Manager and IBM Case Manager, by using the URI and login configuration file that is defined in `cmbemconfig.properties`.

To stop the case event handler, enter `QUIT` in the window console.

15.4 Using Content Manager documents in case solutions

This section addresses the process of how Content Manager documents (item types) can be used in IBM Case Manager case solutions.

15.4.1 Using project area configured with Content Manager

As part of the configuration steps that are described in 15.3, “Integration configuration steps” on page 553, the user’s selected project area is configured with Content Manager integration. After the user logs in, Case Manager Builder automatically detects the user’s project area and is placed into the corresponding project area where Content Manager is configured. You can then use Content Manager documents (item types) in the case solutions.

15.4.2 Using document type from the Content Manager item types

Case Manager Builder has been enhanced to use Case Manager REST API to communicate to the Content Manager server and retrieve the available Content Manager item types. When you try to add document types for a case solution in the Document Types tab, the window displays a list of Content Manager item types. Figure 15-12 shows an example.

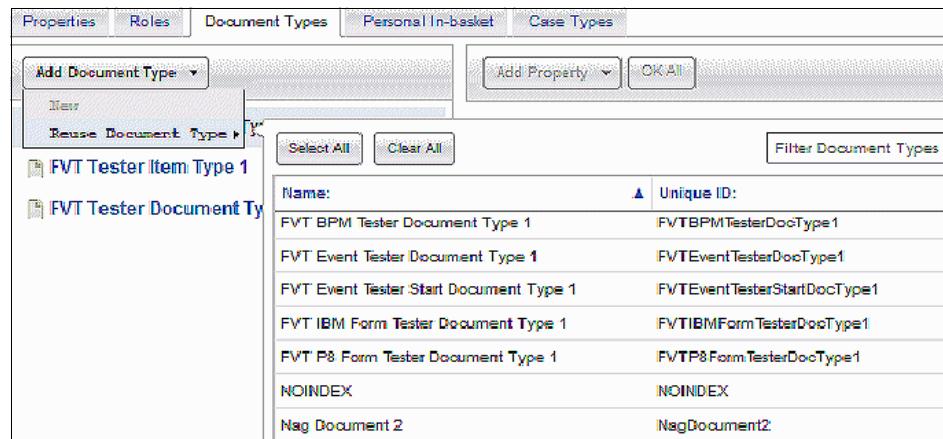


Figure 15-12 Content Manager item type listing in the Add Document Type window

After the document type is created with an Content Manager item type, only its name and unique ID are stored. These values cannot be modified in Case Manager Builder. The document type can then be used in case type initiation or task preconditions.

Restriction: IBM Case Manager does not support case folder hierarchy creation in CaseTypes when it is configured with the Content Manager system. Therefore, Case Manager Builder is modified to disable folder creation in Case types.

15.4.3 Adding Content Manager documents in cases

After document types are created in solution and deployed to target object store, Case Manager Client can be used to add Content Manager documents to cases. This integration ensures that the documents added to cases by using “Add document” are created in the Content Manager repository. There are no additional changes in Case Manager Client to handle documents specific to Content Manager, and they can be used like any FileNet P8 documents in cases.

15.4.4 Security implications

User IDs and groups must be created in LDAP for all case workers who use the Case Manager Client to start and manage cases. You can also use existing users. These case worker IDs must have read and file/unfile access to proxy case folders.

When a case worker logs in to Case Manager Client, it in turn logs in to IBM Case Manager and Content Manager on behalf of the user. Case and workflow security are maintained by IBM Case Manager, and document security is maintained by Content Manager.

15.4.5 Resetting the test environment

Each IBM Case Manager development environment requires its own separate Content Manager system for test environment purpose.

When the test environment is reset by using Case Manager Builder, it deletes all event subscriptions for item types that are configured in solutions in the project area in IBM Case Manager. However, this action does not remove the proxy docs and folders in the Content Manager environment. The Content Manager administrator must perform manual steps on the Content Manager test system. For more information, see the IBM Case Manager information center at:

<http://publib.boulder.ibm.com/infocenter/casemgmt/v5r1m1/topic/com.ibm.casemgmt.install.doc/acmcp027.htm>

For more information about limitations in integration between IBM Case Manager and Content Manager, see the IBM Case Manager information center at:

<http://publib.boulder.ibm.com/infocenter/casemgmt/v5r1m1/topic/com.ibm.casemgmt.install.doc/acmcp024.htm>

15.4.6 Troubleshooting

Check these settings when users experience problems with integration:

- ▶ Verify that the JVM class path of the IBM Case Manager WebSphere server instance is correctly set for the Content Manager Java API OEM Toolkit /cmgmt properties folder.
- ▶ Verify that the JVM class path of the Content Engine WebSphere server instance is correctly set to the location of the following files and folder:
 - cmbicmsdk81.jar
 - cmoutil81.jar
 - cmutilicm81.jar

- db2jcc.jar
- db2jcc_license_cisuz.jar
- db2jcc_license_cu.jar
- log4j-1.2.15.jar
- /cmgmt

If Content Manager item type retrieval fails in Case Manager Builder, make sure that the ICM SERVER property in the /cmgmt/connectors/cmbicmsrvs.ini file is the first entry of all the properties.

To help with troubleshooting, acquire all the logs from WebSphere Application Server, Content Engine, and Content Manager. Common log file location examples:

- ▶ Content Engine (Linux):
/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/FileNet/server1
- ▶ IBM Case Manager:
C:\IBM\WebSphere\AppServer\profiles\AppSrv01\logs\server1
- ▶ Content Manager Java API:
C:\IBM\WebSphere\AppServer\properties\cmlog\dklog.log
- ▶ The user-defined DKLogOutputFileName property in
\\cmgmt\\connectors\\cmblogconfig.properties

For more information about troubleshooting Content Manager integration with IBM Case Manager, see the technotes at:

<http://www.ibm.com/support/docview.wss?uid=swg21570837>



External Data Service Framework

IBM Case Manager offers an improved and easier-to-use data integration across systems that provides a seamless and consolidated view of a case. You can configure an IBM Case Manager solution to reference data that is stored in an external system, such as a customer relationship management (CRM) system.

This chapter introduces the external data service framework and describes how it can be used to initialize case data in IBM Case Manager.

This chapter includes the following sections:

- ▶ Implementing an External Data Service
- ▶ Registering the External Data Service

16.1 Implementing an External Data Service

Case Management systems typically must access data from external sources when they create a case. These sources include databases, web services, file systems, and systems of record such as CRM and enterprise resource planning (ERP). For example, in the Customer Complaints scenario, a case worker uses an e-form to create a case. The case worker enters the Customer Number in the e-form. However, the customer details such as Customer Name, Address, and Contact Number are retrieved from a CRM system through the External Data Service. A custom application can use a lookup field such as account number or claim number to retrieve and populate dependent case properties when the case is created.

Another common requirement among Case Management applications is the ability to define dependent properties for dynamic behavior and load them from an external data source. For example, you might populate a city choice list differently based on another choice list item, the state. When a case worker selects a state, the choice list that is associated with the city property contains only cities that are in that state.

External Data Service can also be used to run custom validation of data that are entered in the case. The service can return a custom error message when the validation fails, or invalid values can be replaced without user intervention. For example, the external data service can check whether a certain date value for the case property lies within a certain range. If it does, external data service can return a custom message that the validation failed.

To facilitate these integration scenarios, IBM Case Manager provides an external data service Representational State Transfer (REST) protocol. This service that is implemented on the REST protocol acts as the communication bridge between IBM Case Manager and the external data source. To populate the case properties with this external data, you must implement an external data service that adheres to this REST protocol.

The external data service can also be used to modify property attributes such as a minimum or maximum value of a case property. A solution with a case property Claim Amount with a minimum value constraint of \$8000 can be overridden by any amount larger than that. However, the external data service cannot make this value smaller because that violates the constraint to make it less restrictive.

Figure 16-1 provides a high-level view of the architecture of the IBM Case Manager and External Data Service integration.

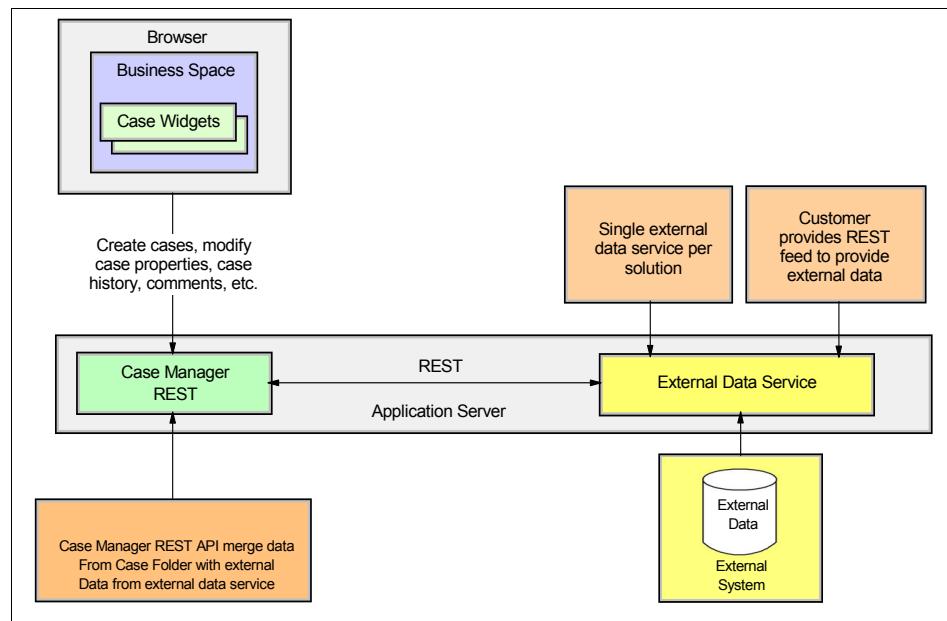


Figure 16-1 External Data Service integration architecture

Tip: Use the external data service only for *retrieving data* from an external source. For example, when a case worker creates a case, Case Manager Client saves the data that was received from the external data source in IBM FileNet Content Engine. If the case worker modifies this data, Case Manager Client does not update the corresponding data in the external data source.

Use the following procedures to implement the external data service:

- ▶ Implement the HTTP POST method for the case type. This method returns any changes in values or attributes to the case properties. It allows the external data service to influence the values that are populated when the case is saved.
- ▶ The external data service can use the Content Engine Java API or CMIS REST API to retrieve property attributes for case properties that it manages.
- ▶ If the external data service needs to authenticate users, it must participate in the same single sign-on authentication configuration as the other IBM Case Manager components.

16.1.1 Sequence for case creation using external data service

Figure 16-2 shows the sequence diagram and the flow of steps that are involved between the Case Manager Client, Case Manager REST, and external data service when creating a case.

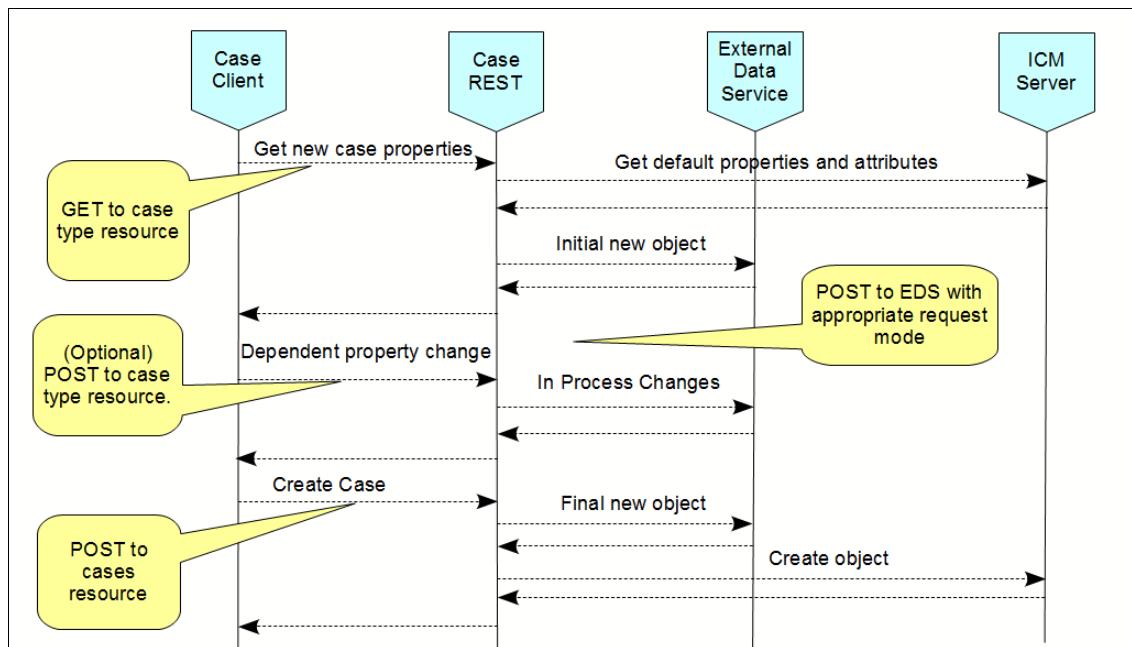


Figure 16-2 Sequence diagram for case creation using External Data Service

The different components interact in the following ways:

1. The case worker clicks **Add Case** on the Case Manager Client and selects the applicable case type to create a case. When the case type is selected, the Case Manager Client starts a GET HTTP request to retrieve the list of case properties. The Customer Complaints example has this resource URL:
GET
`http://icmlab1:9080/CaseManager/CASEREST/v1/casetype/CC_CustomerComplaint`
2. The IBM Case Manager REST API sends an HTTP POST request to the external data service. The POST request is sent to the case type resource in the external data service.
3. The external data service responds with changes to the set of properties it manages and sets the external data identifier. See the following tip on external data identifier.

4. The IBM Case Manager REST API merges the response payload from the external data service with the default case data and returns the data to the Case Manager Client.
5. A dependent property change, such as selection of state choice list item, loads the cities choice list from external data service. In this case, the Case Manager Client determines that the State property has dependent properties and makes an extra POST call to the IBM Case Manager REST API. The IBM Case Manager REST API makes the POST call to the external data service with the requestMode parameter in the payload that is set to `inProgressChanges`. The external data service returns the Cities choice list in the response payload. The IBM Case Manager REST API merges this payload with the case data and presents it to the Case Manager Client.
6. When the user clicks **Add** on the Add Case page, a final call is made from the IBM Case Manager REST API to the external data service. The IBM Case Manager REST API sends all case data properties that are about to be persisted in the Content Engine. This POST call sets the requestMode parameter in the payload to `finalNewObject`. If an existing case is being updated, then the requestMode parameter is set to `finalExistingObject`.
7. The external data service responds with the response payload to the IBM Case Manager REST API. The IBM Case Manager REST API then persists the case and its associated case data properties to the Content Engine. If the case creation fails, no notification is sent to the external data service. Avoid implementing custom updates to external data based on the final POST call from the IBM Case Manager REST API.

Tip: The external data identifier is a string that provides contextual information to indicate the state of the data that was returned by the External Data Service. The value of this parameter is set by an external data service.

Typically, the service sets the parameter to reference the specific configurations that were used to define the attributes other than the property value. These attributes include settings for the minimum value, maximum value, and choice list.

Case Manager Client maintains the value of the parameter, but it does not change its value. Include the `ExternalDataIdentifier` parameter in the payload whenever a method creates or updates a case. Also, include this parameter in subsequent requests if a value was provided in response to a previous call to get data from the external data service.

16.1.2 Implementing the POST method for external data service

The POST method provides the means for obtaining data from an external data source for a case of a specific case type. Only one external data service can be registered per solution, so all case types must use the same external data service. When a case worker performs any of these tasks from the Case Manager Client, the IBM Case Manager REST API automatically makes an HTTP POST call. This call goes to the case type resource in the external data service.

- ▶ Creates a case
- ▶ Opens an existing case
- ▶ Starts a task
- ▶ Splits a case

The POST method passes the request payload, which contains the current value for each case property. Each property can have one of the following values:

- ▶ The default value that is specified in Case Manager Builder for that case property
- ▶ The value persisted in Content Engine for the case property
- ▶ The working value that is entered by the case worker for that property

Upon receipt of the request payload, the service retrieves the properties that are managed by the payload. The service then builds the response payload with the updated case property values or attributes and returns this payload to the IBM Case Manager REST API. The IBM Case Manager REST API merges the data from the response payload with the case data and returns it to the Case Manager Client.

In the Customer Complaints scenario, an external data service implements the CC_Complaint case type resource with the following resource URL:

http://icmlab1:9080/test/ICMEDREST/type/CC_Complaint

The request JSON payload for CC_Complaint case type resource is shown in Example 16-1.

Example 16-1 Request JSON payload

```
{  
    "repositoryId": "CMTOS",  
    "objectId" : "",  
    "requestMode" : "initialNewObject",  
    "externalDataIdentifier" : "",  
    "properties":  
    [  
        {  
            "name": "Case Type",  
            "value": "Customer Complaint"  
        },  
        {  
            "name": "Case Status",  
            "value": "Open"  
        }  
    ]  
}
```

```

{
    "symbolicName" : "CC_CustomerNumber",
    "value"         : "667888909",
},
{
    "symbolicName" : "CC_CustomerName",
    "value"         : null,
},
{
    "symbolicName" : "CC_Address",
    "value"         : null,
},
{
    "symbolicName" : "CC_Telephone",
    "value"         : null,
},
{
    "symbolicName" : "CC_Email",
    "value"         : null,
}
// More properties ...

],
"clientContext":
{
}
}

```

16.1.3 Request parameters in the POST payload

The requestMode parameter in the JSON payload refers to the action requested by the client application, which in the example is the Case Manager Client. The external data service introspects this requestMode parameter and returns the appropriate response. For example, if the request is to create a case, the service builds the response payload with the initial property values for the case type. The RequestMode parameter can have these values:

- ▶ `initialNewObject`: The RequestMode parameter is set to this value in the first call made to the external data service during new case creation. As shown in Example 16-1 on page 578, do not set the `externalDataIdentifier` parameter in the request payload. The `externalDataIdentifier` is populated by the external data service in the response payload. This process allows

- externalDataIdentifier to be used in subsequent calls to reference the current state of data to the external data service.
- ▶ initialExistingObject: The RequestMode parameter is set to this value when modifying an existing case. Because this parameter applies to existing cases, the objectID field in the request payload refers to the GUID for the case. As in the previous case, the externalDataIdentifier is populated by the external data service in the response payload.
 - ▶ inProgressChanges: The RequestMode parameter is set to this value when a property that has dependent properties is changed on the Case Manager Client. In addition, those dependent properties are to be retrieved from the external data service. The request payload includes the current working value for each property in the case. It also includes the externalDataIdentifier that is sent by the external data service in one of the previous response payloads, and the objectID for the case. The objectID is only set if you are updating an existing case.
 - ▶ finalNewObject: The RequestMode parameter is set to this value when the IBM Case Manager API is ready to persist the new case and the properties in the Content Engine. This call is made before the case object is persisted.
 - ▶ finalExistingObject: The RequestMode parameter is set to this value when the IBM Case Manager API is ready to persist an existing case and the associated properties in the Content Engine. After this call, the case properties are persisted.

The clientContext parameter provides contextual information about the work item that is associated with a case. The Case Manager Client automatically includes the clientContext parameter in the request when a case is opened. For more information, see the IBM Case Manager Information Center topic “Implementing an external data service” at:

<http://publib.boulder.ibm.com/infocenter/casemgmt/v5r1m0/topic/com.ibm.casemgmt.installng.doc/acmdv101.htm>

16.1.4 Case data persistence

The IBM Case Manager REST API makes a final call to the external data service with the RequestMode parameter set to finalNewObject or finalExisting object when creating or updating a case. Depending on the response payload from the external data service to this final POST call, the IBM Case Manager REST API determines whether to persist the case to the Content Engine.

The IBM Case Manager REST API evaluates each property in the payload response as follows:

- ▶ If a value for a case property is explicitly passed from the Case Manager Client, that value is validated against the property attributes returned by the external data service. If the value passed validation from the external data service, the value is persisted in the case property in the Content Engine.
- ▶ If a value for the case property is not explicitly passed from the Case Manager Client and the external data service populated that value, the value is persisted in the Content Engine.
- ▶ If a value for the case property is not explicitly passed from the Case Manager Client and the external data service did not return a value in the response payload for that case property, the default value is persisted for new cases. The default value is defined in the Content Engine. For existing cases, the current value of the case property remains unchanged. If the external data service modified any property attributes, however, the default value or existing value is validated against those property attributes.

16.1.5 Response payload and HTTP response codes

For the Customer Complaint scenario, the response payload for the HTTP POST operation is show in Example 16-2.

Example 16-2 HTTP POST response payload

```
{
  "externalDataIdentifier": "",
  "properties": [
    {
      "symbolicName": "CC_CustomerName",
      "value": "John Taylor",
      "displayMode": "readonly",
      "hasDependentProperties": false
    },
    {
      "symbolicName": "CC_Address",
      "value": "3565 Harbor Blvd, Costa Mesa, CA 92626",
      "displayMode": "readonly",
      "hasDependentProperties": false
    },
    {
      "symbolicName": "CC_Telephone",
      "value": "714-327-7800",
      "displayMode": "readonly",
      "hasDependentProperties": false
    }
  ]
}
```

```

    {
      "symbolicName": "CC_Email",
      "value": "john@us.ibm.com",
      "displayMode": "readonly",
      "hasDependentProperties": false
    }
  ]
}

```

As mentioned in 16.1.3, “Request parameters in the POST payload” on page 579, the externalDataIdentifier parameter must be used in subsequent calls to the external data service. In the example, external data service manages a list of cities that is dependent on a State case property. In this case, the InProgressChanges requestMode parameter must reference the externalDataIdentifier parameter in the external data service HTTP POST call. The Case Manager Client passes the externalDataIdentifier to the IBM Case Manager REST API, which then passes it to the external data service.

The response payload can also return a property attribute named customValidationErrors if the current value of the property failed to validate in the external data service. In the Customer Complaint scenario, if the customer number is invalid or not found, the customValidationErrors property attribute gets populated in the response payload as “Customer Number not found.”

For a complete list of property attributes that can be returned in the response payload, see the IBM Case Manager Information Center topic “Implementing an external data service” at:

<http://publib.boulder.ibm.com/infocenter/casemgmt/v5r1m0/topic/com.ibm.casemgmt.install.doc/acmdv101.htm>

Table 16-1 shows the response codes that are returned by the IBM Case Manager REST API for the HTTP POST calls.

Table 16-1 HTTP response codes

Error Code	Description
200	OK. Successful.
400	Bad Request. Indicates that a required parameter was not specified or the parameter specified was invalid.
404	Not Found. Indicates that the requested resource or the case type was not found
500	Internal Server error

The userMessage field in the JSON response also has more information about the error condition.

16.2 Registering the External Data Service

After you implement the external data service based on the IBM Case Manager REST protocol, you must register this service. The registration is done through IBM Case Manager administration client. You can register only one external data service per IBM Case Manager solution.

After the external data service URL is registered by using IBM Case Manager administration client, the solution must be deployed. Registering the external data service URL registers it only for that solution in the Project Area (connection definition). The runtime environment is not configured until the solution is deployed.

Figure 16-3 shows the Register external data service task within IBM Case Manager administration client.

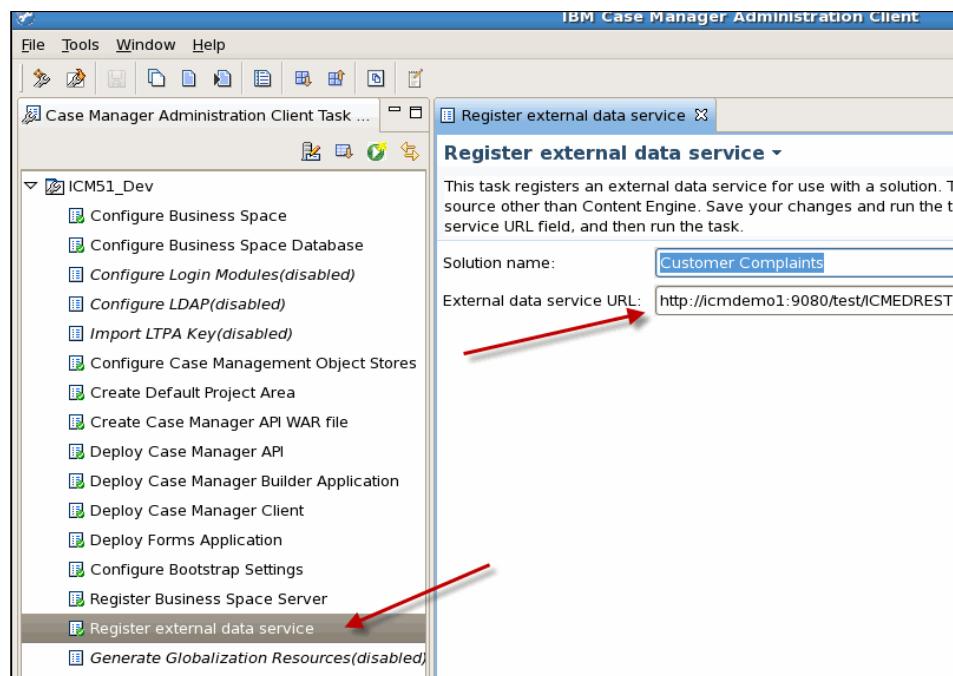


Figure 16-3 Registering external data service by using IBM Case Manager administration client

Perform these steps to register the external data service:

1. Start IBM Case Manager administration client and open the profile for your case manager deployment.
2. Right-click **Register external data service task** and select **Enable Task**.
3. Open the Register external data service task by double-click the task.
4. Select the Solution name for which you are registering an external data service.
5. Enter the external data service URL. The sample external data service has a context root of test. The ICMEDREST must be appended to the deployed context root of the external data service. The example uses the following URL:
`http://icmdemo1:9080/test/ICMEDREST/`
6. Deploy the solution from the Case Manager Builder.

Important: The register external data service step must be run in IBM Case Manager administration client to integrate an external data source with IBM Case Manager. Only one external data service can be registered per IBM Case Manager solution.



Integration with IBM Business Process Manager

IBM Case Manager supports IBM Business Process Manager (IBM BPM) to improve configuration flexibility and productivity. Case workers can view both IBM Case Manager and IBM BPM work items simultaneously from the same integrated view. Solution designers can define case tasks that were previously implemented in IBM BPM and managed in the Process Center.

This chapter describes how to use the capabilities of IBM Case Manager and IBM BPM to manage human-centric, integration-centric, and content-centric business processes. These processes can achieve end to end process control and visibility across your business network.

The chapter includes the following sections:

- ▶ Introduction to IBM Case Manager and IBM BPM
- ▶ Implementing a task as IBM BPM process application
- ▶ Implementing a task as an IBM BPM automated process
- ▶ Configuring an Integrated Inbox

Important: Two IBM products are designed for business process management: IBM Business Process Manager and IBM FileNet Business Process Manager. To avoid confusion, this chapter uses IBM BPM as the abbreviation for IBM Business Process Manager.

17.1 Introduction to IBM Case Manager and IBM BPM

Most companies look at their business network as composed of processes within the walls of their business. However, the processes involved in the network span across partners, suppliers, and customers. The dynamic and unpredictable nature of the relationships between partners, suppliers, and customers affects the business network in unforeseen ways. Supporting and improving these processes requires robust and flexible connections across the business network.

IBM offers business process management capabilities that support a range of patterns such as human interaction, system interaction, case management, and activity-based workflows. Multiple entry points of support include advanced case management, process discovery and compliance, process automation and integration, and business monitoring and decision management.

17.1.1 IBM BPM components

IBM BPM unifies the capabilities of IBM WebSphere Lombardi Edition and IBM WebSphere Process Server into a single product offering. It provides a highly scalable business process management platform for delivering solutions to complex business problems. IBM BPM can simplify operations, centralize visibility, and improve the overall experience for managing business processes.

Figure 17-1 on page 589 shows the different components in IBM BPM, which have these capabilities:

- ▶ **Process Center:** The Process Center provides a repository that is used for storing process definitions and the tools that are used to govern the lifecycle of business processes. The Process Center includes a Process Center server (the playback server) and a performance data warehouse. These systems allow users who work in the authoring environments to run processes and store performance data for testing and playback purposes.
- ▶ **Process Server:** The Process Server is the runtime environment for business process applications. It includes Process Administration Console for management of the Process Server and deployed applications, and Process Portal to run and manage the tasks of business processes.
- ▶ **Process Designer:** Formerly known as Lombardi Authoring Environment, the Process Designer is used to model and implement business processes as process applications.
- ▶ **Integration Designer:** Formerly known as WebSphere Integration Developer, the Integration Designer is used to author complex integrations and automated processes that support the process applications that are designed in Process Designer.

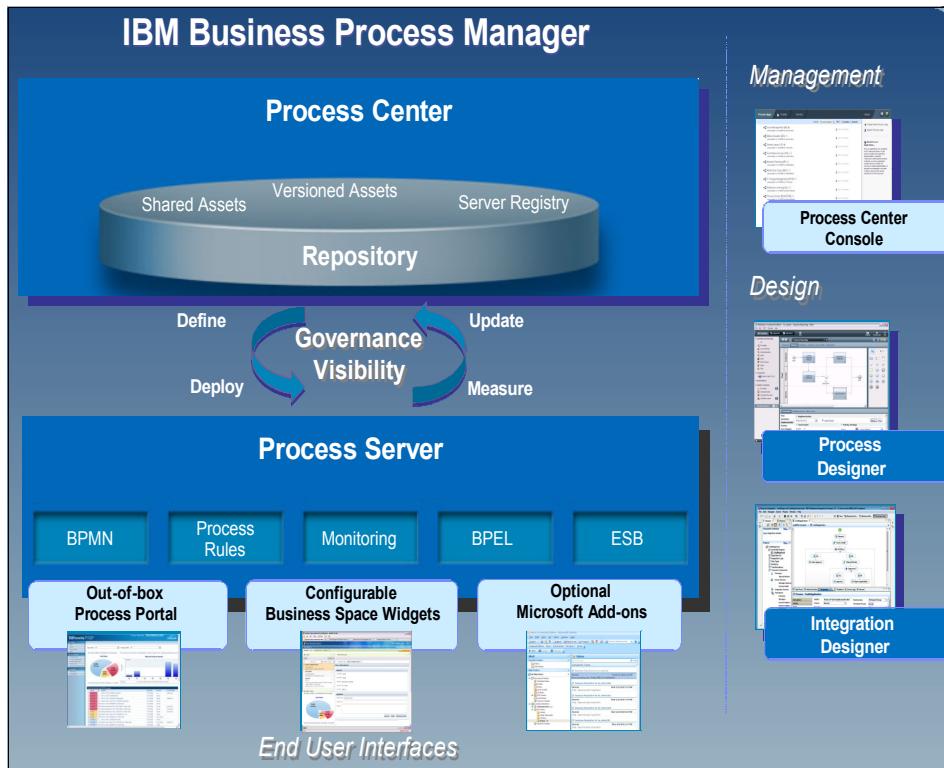


Figure 17-1 Component overview of IBM BPM

More resources: For more information about IBM BPM offerings, see the following resources:

- ▶ IBM BPM configurations:
http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r5mx/topic/com.ibm.wbpm.main.doc/topics/cbpmp_configurations.html
- ▶ IBM BPM configuration capabilities:
http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r5mx/index.jsp?topic=/com.ibm.wbpm.main.doc/topics/cbpmp_wbpm_pkg.html

17.1.2 IBM Case Manager integration for IBM BPM

IBM Case Manager enables invocation of IBM BPM processes to support process scenarios such as:

- ▶ Provide transactional integrity and enterprise class qualities of service
- ▶ Provide web service choreography for transactions and data movement among applications
- ▶ Use pre-built connectivity to high demand applications by using WebSphere Adapters

These processes allow case workers to act on decisions based on the context of documents and other content objects. The business processes can be implemented in IBM BPM in two ways:

- ▶ A case task implementation that starts an IBM BPM process application
- ▶ A case task implementation that starts an IBM BPM automated process

17.2 Implementing a task as IBM BPM process application

A business process application is a component that allows people and services to interact. In this integration approach, complete the following steps:

- ▶ Design the process by using the IBM BPM Process Designer
- ▶ Store the business process definitions in the Process Center
- ▶ Deploy the definitions to the Process Server

In Case Manager Builder, the case task is mapped to the exposed business process definitions in the Process Center.

Remember: This chapter does not show you how to implement IBM BPM process applications or configure IBM Case Manager and IBM BPM so that they communicate with each other. Rather, it shows you how to use the IBM BPM process applications within IBM Case Manager. This description assumes that the required configurations such as SSO and Federated LDAP are complete. Therefore, the focus is on exposing IBM BPM process applications to IBM Case Manager and using them in IBM Case Manager solutions.

17.2.1 Configuring IBM Case Manager for IBM BPM integration

Use the IBM Case Manager administration client to configure the IBM Case Manager system to integrate with IBM BPM. Before you configure IBM BPM, create a development profile and select the **Configure IBM Business Process Manager** configuration option (Figure 17-2).

The screenshot shows the 'Create Profile' dialog box with the 'Profile Information' tab selected. The interface is titled 'Create Profile' at the top. The 'Profile Information' section contains the following fields:

- * Profile type: Development environment profile
 Production environment profile
- * Installation type: New installation
 Upgrading from IBM Case Manager 5.0
- Configuration options:
 Configure IBM Content Manager
 Configure IBM Business Process Manager
- * Profile name:
- * Directory name that will contain the new profile directory and files:

A note at the bottom indicates: * denotes a required field.

Figure 17-2 Creating a profile for IBM Case Manager integration with IBM BPM

Important: You must configure Secure Sockets Layer (SSL) and single sign-on (SSO) on both the IBM Case Manager and the IBM BPM WebSphere profiles. Only federated Lightweight Directory Access Protocol (LDAP) that is configured for SSO is supported. IBM Case Manager uses an integration user account to look up the business process definitions in Process Center and start the process at run time. Before you run these tasks, identify this user on the Process Server and make sure that the user has the necessary privileges to start IBM BPM processes.

Perform the following tasks in the development environment profile:

1. Configure IBM BPM.
2. Deploy the IBM BPM integration service.

After you successfully complete these tasks, you can reuse IBM BPM Process Definitions as case task implementations.

17.2.2 Reusing an IBM BPM process definition

The Customer Complaints example needs a customer outreach process whenever a high value client files a complaint. The case management application is designed to start the customer outreach process automatically when a customer with a Platinum rating files a complaint. The ILOG rules engine determines the customer rating (Platinum, Gold, or Silver) and stores the rating as a case property. The customer outreach process is implemented as an IBM BPM process to demonstrate the integration capability.

Before you reuse the IBM BPM process in IBM Case Manager, the business process definition must be defined in Process Designer and stored in the Process Center. The business process definition and process application have these considerations:

- ▶ The business process definition must be stored in the Process Center and a snapshot of the process application must be installed on the Process Server.
- ▶ The process application name must *exactly match* the IBM Case Manager solution name.
- ▶ More than one IBM Case Manager task can point to the same business process definition in IBM BPM.

Figure 17-3 shows the mapping between the IBM FileNet P8 domain and the IBM BPM deployment domain.

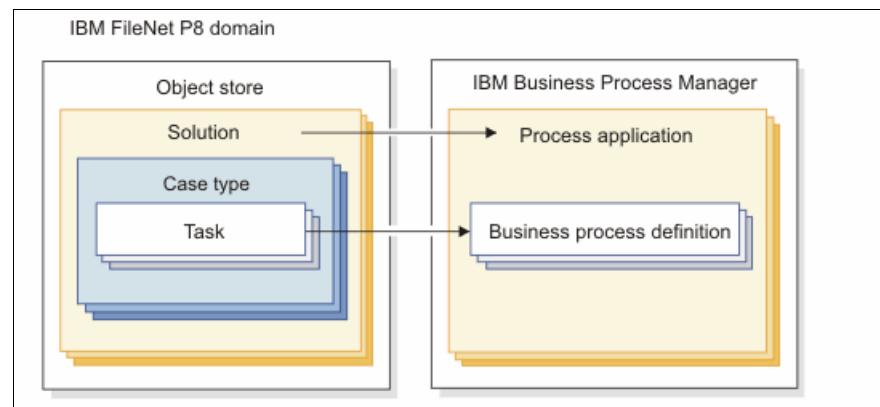


Figure 17-3 Mapping between IBM FileNet P8 domain and IBM BPM deployment

Figure 17-4 shows the Customer Complaints process application in IBM BPM Process Designer. The process application name is set to Customer Complaints, which matches with the IBM Case Manager solution name. The process application snapshot is saved to the Process Center. The revision history on the lower left corner shows the versions of the process application that are also saved as snapshots in the Process Center.

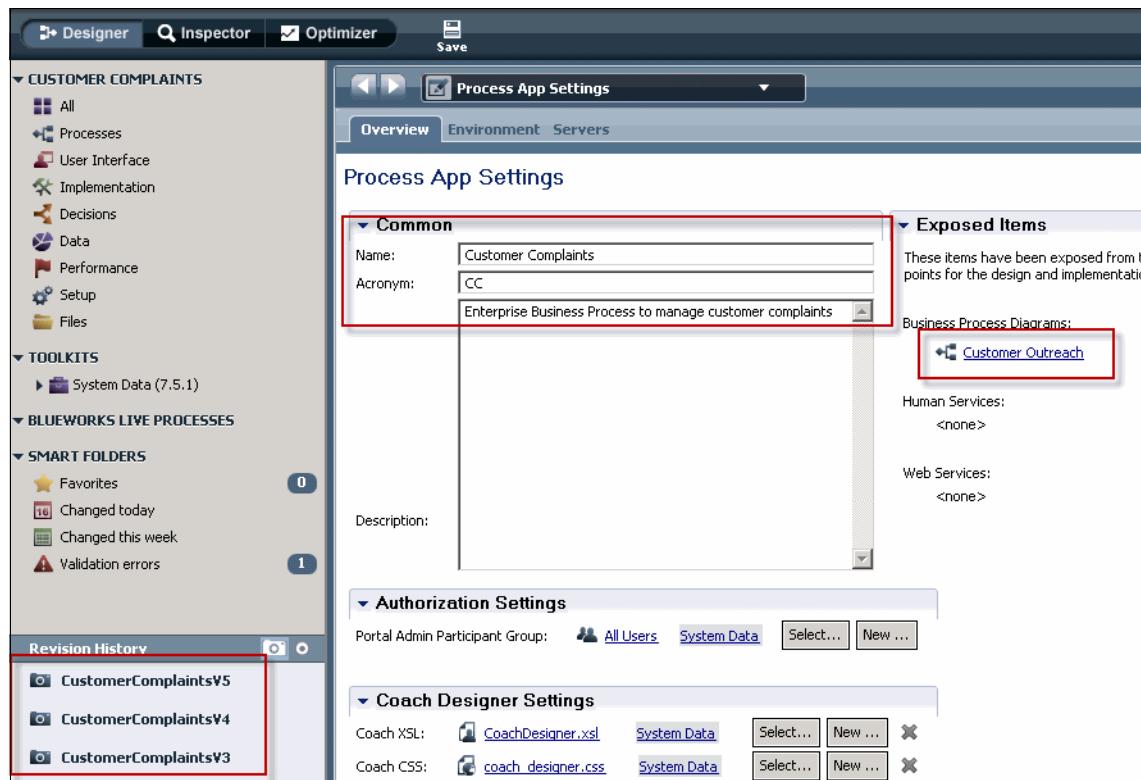


Figure 17-4 Customer Complaints business process definition

Before you install the snapshot to the Process Server, test it in the Process Center playback server. After the testing is complete, save the process application snapshot to the Process Center, select the snapshot, and install it to the Process Server.

After you install the snapshot in the Process Server, log in to the Process Administration console. Select the installed snapshot and set it to **Active** and **Default** as shown in Figure 17-5.

Snapshots	Status
Commercial Banking Solution (BPM) - v3	Active
Commercial Banking Solution (BPM) - v4	Active
Customer Complaints (CC) - CustomerComplaintsV1	Active
Customer Complaints (CC) - CustomerComplaintsV2	Active, Default
Hiring Sample (HSS) - Hiring Sample 3	Active, Default
Process Portal (TWP) - 7.5.0	Active, Default
Test (TEST) - Test Snapshot	Active

Figure 17-5 Customer Complaints snapshot settings

After you set the snapshot to **Active** and **Default**, make sure that the Customer Outreach business process definition is exposed to **All Users** (Figure 17-6).

BPDs	Access Level
Customer Outreach(as Startable BPD)	All Users
Customer Outreach(Business Data)	All Users

Figure 17-6 Exposing Customer Outreach business process definition

17.2.3 Mapping a task to business process definition

The Customer Outreach business process definition exposed in the Customer Complaints process application can now be used as a task implementation in the Customer Complaints IBM Case Manager solution. Navigate to the Tasks for the Complaint Case type and selecting **Add Task**. The **Add IBM BPM Process** option is enabled as shown in Figure 17-7.

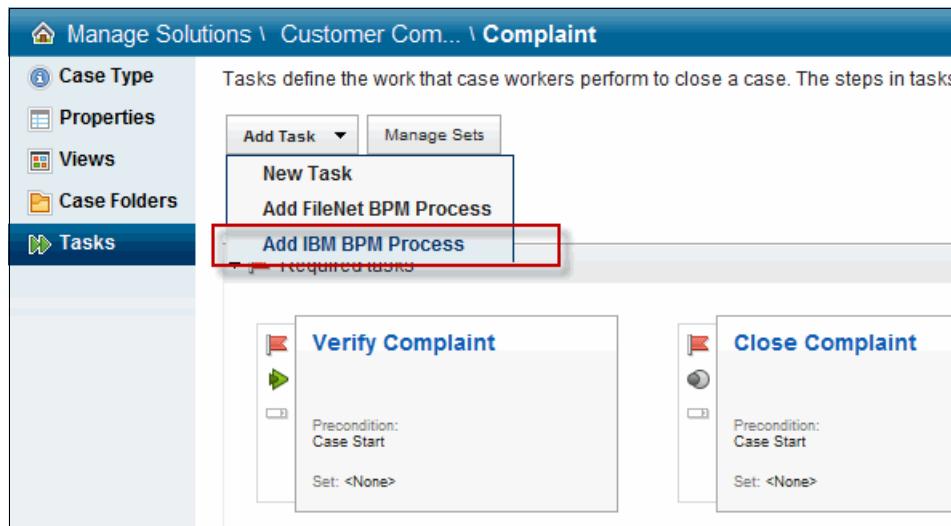


Figure 17-7 Adding task to reuse Customer Outreach business process definition

Requirements: When you enable **Add IBM BPM Process**, the Case Manager Builder makes an HTTP Representational State Transfer (REST) application programming interface (API) call to the Process Center. This call gets a list of exposed business process definitions when the process application name matches the IBM Case Manager solution name. The matches must meet these requirements:

- ▶ The solution and the process application names (not the solution prefix) match exactly.
- ▶ The process application is set to **Active** and **Default**.
- ▶ The business process definitions in the process application are exposed to **All Users**.

When these requirements are met, **Add IBM BPM Process** in Case Manager Builder is enabled.

To map an IBM Case Manager task to the exposed business process definition, complete these steps:

1. Select **Add IBM BPM Process**, which starts the task wizard.
2. On the General window, enter the task name and define the task properties, for example, automatic, manual, or user initiated.
3. Select the **Customer Outreach** business process definition that you want to reuse from the process application as shown in Figure 17-8. This window shows all exposed business process definitions for your process application.

The screenshot shows a 'Select Process' dialog box. At the top, it says 'Add Task' and 'Select Process'. Below that, it says 'Select the process to reuse:'. A table lists a single process definition:

Process Name	Description
Customer Outreach	Customer Outreach

At the bottom right, there are three buttons: 'Back', 'Cancel', and 'Next'.

Figure 17-8 Selecting the Customer Outreach business process definition

4. Map the process data field name to the case property name as shown in Figure 17-9.

Add Task

Map Properties

Customer Outreach (Reused Business Process Man)

Map process data fields to solution properties.

Click the "+" button to save the data field to property mapping.

Process data field name: Case type property name:

CustomerRating (string, input) Customer Rating string +

Single Valued, input

Property map:

CustomerName = Customer Name (string, input)
Address = Address (string, input)
Telephone = Telephone (string, input)
ComplaintDescription = Complaint Description (string, input)

-

Back Finish Next

Figure 17-9 Mapping the process data field to the case property

5. After you map the fields, click the + icon, which adds the fields to the property map.
6. Repeat these steps until all process data fields are mapped to the corresponding case type properties.
7. Save the solution.

After you create the task and save the solution, deploy the solution from Case Manager Builder. Test the task by creating a case from Case Manager Client and starting the task.

Consideration: The selected business process definition can be changed in the task later. If you select a different business process definition, you must map the process data field to the case property again.

17.3 Implementing a task as an IBM BPM automated process

An automated business process is a set of business-related activities that are started to achieve a business goal. Use the Integration Designer to map the task to a new web-service-enabled Business Process Execution Language (BPEL) process. The BPEL process implements the work flow for that task.

For each such task implementation, Case Manager Builder creates an interface process fragment in FileNet Process Engine. This interface process starts the IBM BPM automated process by using a web services call. The IBM BPM process might be called synchronously by using a single invoke step. Otherwise, it is called asynchronously with a sequence of invoke and receive steps in the Process Engine interface process.

The invoke message includes the service endpoint where IBM BPM Process Server must send the reply message. This message also includes the correlation identifier that allows the Process Engine to match the incoming message to the correct waiting work item. After you receive the reply message, a system step updates the case properties that are mapped to the values returned by the Process Server service call.

To implement an IBM BPM automated task, create an empty task by using Case Manager Builder. Use Integration Designer to discover this empty task and to implement the Process Server process. After being pointed to the correct IBM Case Manager design object store, Integration Designer lists all solutions available in that object store. From there, navigate to the appropriate case type and select the task to implement.

Integration Designer creates a skeleton module that you can then use to implement the process. From the list of properties available for the selected case type, map your required properties as input or output parameters to the process implementation. After defining the process, Integration Designer updates the XML Process Definition Language (XPDL) in the IBM Case Manager solution package. This updated XPDL includes the interface process with the required invoke step for the task. It also includes the Web Services Description Language (WSDL) for the web services call to start the IBM BPM process.

At run time, the web service invoke requests in the Process Engine WSRequest queue are managed by the Process Engine Component Manager. Component Manager can be used to configure Process Server credentials if the default Component Manager credentials are not valid for the Process Server connection.

17.4 Configuring an Integrated Inbox

An Integrated Inbox provides a consolidated view of all outstanding tasks. The integration of IBM Case Manager and IBM BPM achieves this consolidation by using business space widgets. In business space, the Integrated Inbox displays a federated task list that contains tasks and work items from both IBM BPM and IBM Case Manager.

IBM BPM items in the list are called *tasks*. IBM Case Manager items in the list are called *work items* or *steps*. If you open a work item, the work item is displayed in an IBM Case Manager Work Details window. If you open a task, the task is displayed in an IBM BPM widget. You can only edit or reassign work items in an Integrated Inbox. A work item opened from the Integrated Inbox into a Work Details window allows the same actions as a Work Details window opened from an in-basket. The only action that is not available is Get Next.

Restriction: If the Work Details window is opened from an Integrated Inbox, the Get Next action does not return the next item. The Get Next action does not work even if the option is enabled in the Work Item Toolbar widget

Figure 17-10 shows the architecture for the Integrated Inbox. The Integrated Inbox widget calls the Federated REST API. You can retrieve work items and tasks from IBM Case Manager and Process Server by using the Federated REST API. The API combines them into a single task list.

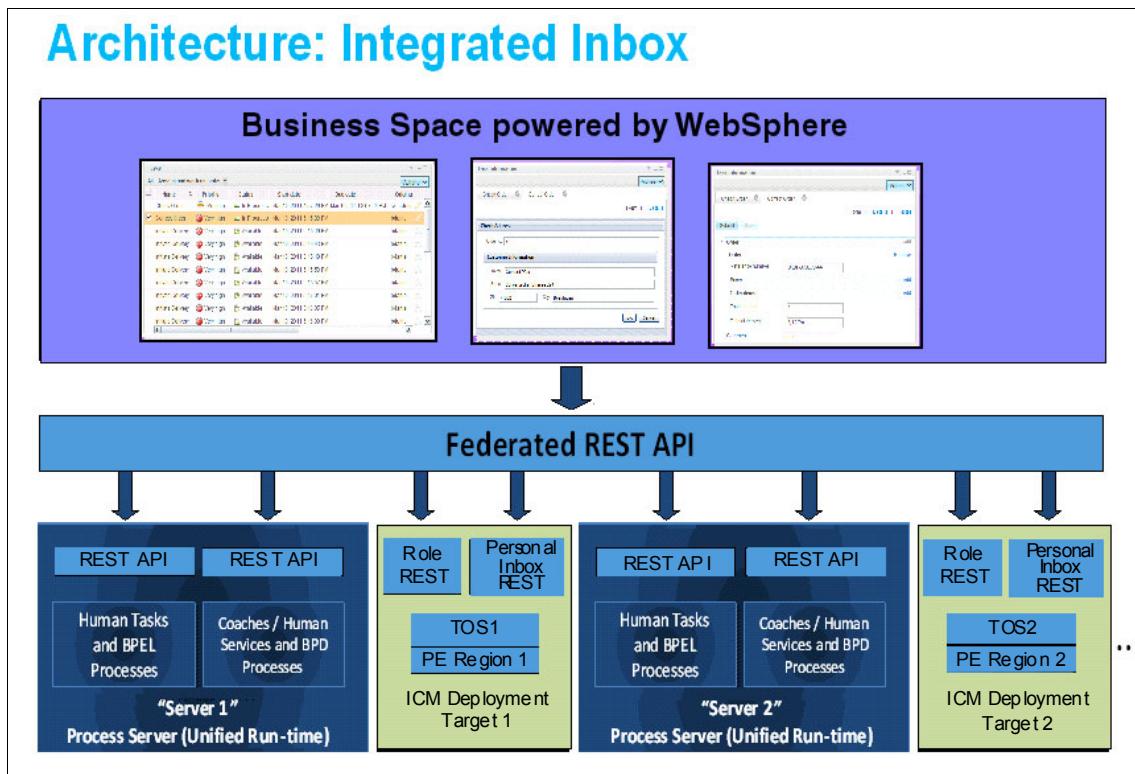


Figure 17-10 Integrated Inbox architecture

17.4.1 Setting up business space for IBM Case Manager

Before you can use the Integrated Inbox widget in a case management solution, you must complete the following steps:

1. Configure cross cell security for IBM BPM and IBM Case Manager, including a federated realm, single-sign-on, and Secure Sockets Layer. Because both the applications are deployed in different cells in WebSphere Application Server, you must configure a federated LDAP repository. This configuration allows both cells to have access to the same users.

Figure 17-11 shows the federated LDAP setup.

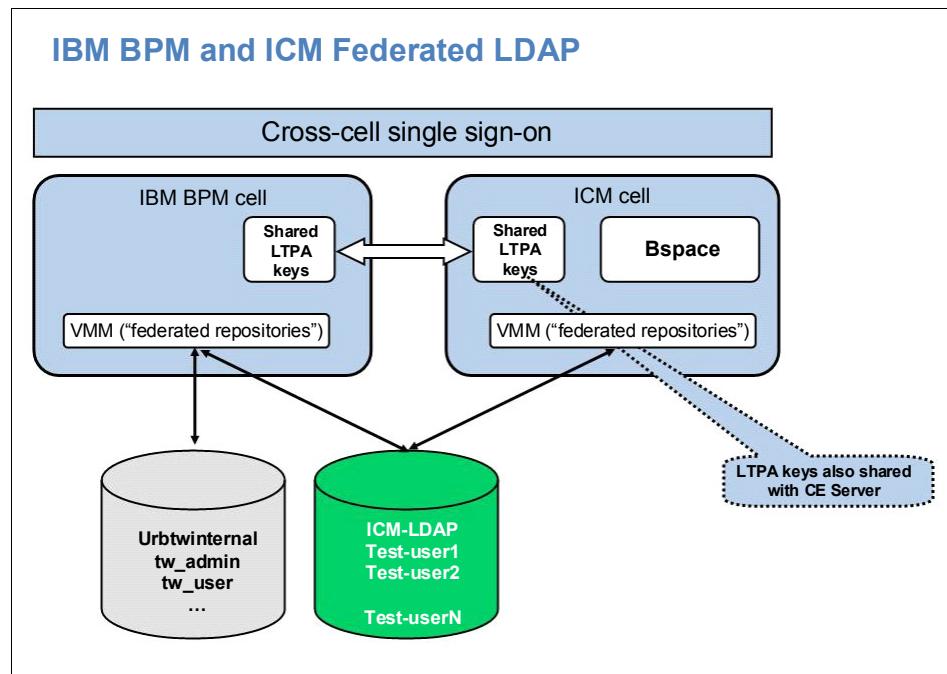


Figure 17-11 Federated LDAP for IBM BPM and IBM Case Manager

2. Register the IBM BPM widgets in IBM Case Manager, including the widget catalog and endpoints.
3. Register the IBM Case Manager REST services in IBM BPM by running the `addICMSystem` command where the IBM BPM Federated REST API is deployed.

After successfully setting up business space for IBM Case Manager, you are ready to configure the Integrated Inbox in your case management solution.

Tip: Set the `federateSystem` option to true in the `addICMSystem` command when you are registering the IBM Case Manager REST services in IBM BPM. A federation domain that contains the IBM Case Manager and IBM BPM cells is required to configure the Integrated Inbox. If you set the `federateSystem` option to false, you must run the `createBPMApiFederationDomain` administration command to create a federated domain.

17.4.2 Creating a saved search query in IBM BPM Process Portal

Create a saved search query in the IBM BPM Process Portal by completing these steps:

1. Select the search conditions as shown in Figure 17-12. The task status is set to **New** or **Received** and process application to **CC**. For the Customer Complaints Process Application example, the alias is set to **CC**.

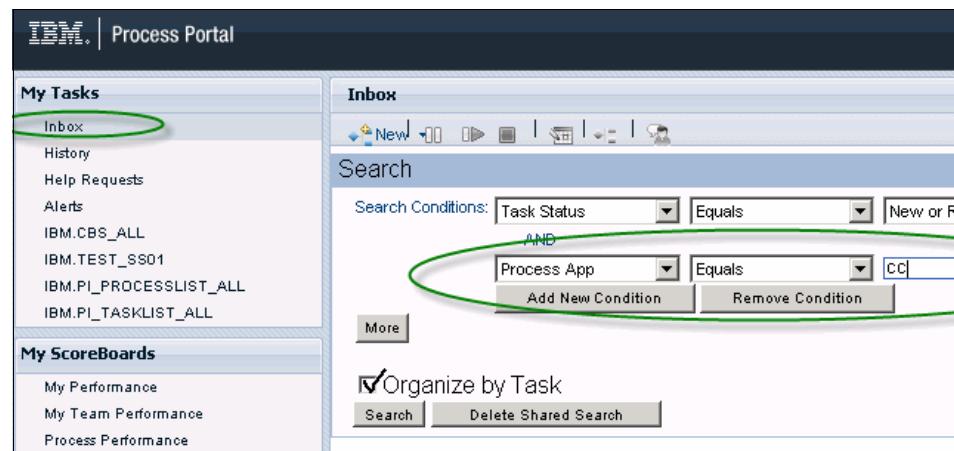


Figure 17-12 Saved search query with task status and process application

2. Select **Organize by Task** and run the search.

- Customize the search query by including the columns that you want to display in the Integrated Inbox as shown in Figure 17-13.

The screenshot shows the 'Search' configuration screen. At the top, there are two search condition boxes: 'Task Status Equals New or Received' and 'Process App Equals CC'. Below these are buttons for 'Add New Condition' and 'Remove Condition'. The next section, 'Choose Columns To Display:', contains two panes. The left pane, 'Candidate Columns:', lists various fields like 'Instance:Instance Name', 'Task:Task Subject', etc. The right pane, 'Columns To Display:', lists specific columns: 'Task:Task Subject', 'Task:Task Priority', 'Task:Task Due Date', 'Business Data:CC_Telephone', 'Business Data:CC_CustomerName', 'Business Data:CC_ComplaintDescription', 'Business Data:CC_Address', and 'Business Data:CC_CustomerRating'. A green oval highlights this list. Between the panes are arrows for moving columns between them. To the right, under 'Choose Columns to Sort On', are sections for 'Primary: Column Name' (set to 'Task Due Date') and 'Secondary: Column Name' (set to 'Task Priority'). At the bottom right is a 'Items Per Page:' dropdown.

Figure 17-13 Column display for search query

- Click **Share the Search** to share the search query with all users. See Figure 17-14.

This screenshot shows the 'Search' interface after sharing. It includes the same search conditions and primary/secondary sort options as Figure 17-13. A green oval highlights the 'Share this Search' button in the bottom navigation bar, which is positioned between 'Delete Search' and 'More'. Below the search bar is a section titled 'Tasks' with a header row containing 'Task Subject', 'Task Priority ▲', 'Task Due Date ▲', 'CC_Telephone', 'CC_CustomerName', and 'CC_C'. At the bottom, it displays '0 results (0.02 seconds)'.

Figure 17-14 Sharing the search query

5. Click **Save Search** and enter the search query name as shown in Figure 17-15. Save the name as uppercase characters.

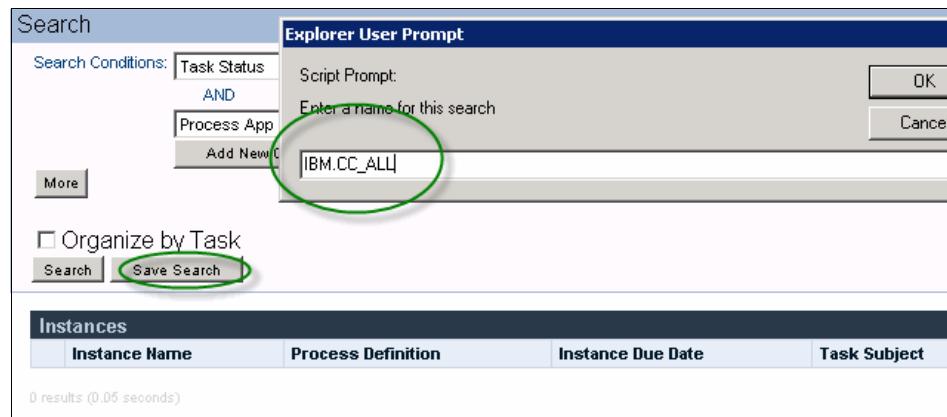


Figure 17-15 Saving the search query

17.4.3 Configuring In-baskets in the Customer Complaints solution

To configure In-baskets in the Customer Complaints solution, complete these steps:

1. Start the FileNet Process Designer and open the solution by clicking **File** → **Solution** → **Edit**.
2. Open the solution configuration by clicking **View** → **Configuration**. Select the **CC_Manager** queue under the Work queues.
3. Modify the in-basket by selecting the columns that you want to display in this in-basket.

- Add the **OneBPMQuery** and **OneBPMSSolution** parameters in the Custom Attributes tab as shown in Figure 17-16.

Name	Type	Value
OneBPMQuery	String	IBM.CCALL
OneBPMSSolution	String	Customer Complaints
OneBPMSPrefix	Boolean	true

Figure 17-16 Custom Attributes in Complaints In-basket for CC_Manager queue

The **OneBPMQuery** is set to **IBM.CCALL**, which is the saved search query that was created and shared in Process Portal for Customer Complaints Process Application. The **OneBPMSSolution** is set to **Customer Complaints**, which is the solution name in IBM Case Manager. The values *must match* the query name and solution name exactly, including uppercase and lowercase characters.

- Include the work items from the Personal Inbox of the case worker into the Integrated Inbox. Copy the personal in-basket by clicking **User Queues** → **InBox** → **In-baskets**. In addition to **OneBPMQuery** and **OneBPMSSolution** that were defined in the Complaints in-basket, add a third attribute, **OneBPMSPrefix** (Figure 17-17).

Name	Type	Value
OneBPMSPrefix	String	CC_
OneBPMQuery	String	IBM.CCALL
OneBPMSSolution	String	Customer Complaints

Figure 17-17 Custom attributes for personal in-basket

As in the previous step, **OneBPMQuery** and **OneBPMSolution** are set to **IBM.CCALL** and **Customer Complaints**. The custom attribute **OneBPMPrefix** is set to **CC_**. A trailing underscore is required after the solution prefix CC.

6. Save and close the solution in FileNet Process Designer.
7. Open the solution in Case Manager Builder and deploy it.

17.4.4 Adding Integrated Inbox to the Customer Complaints solution

To use the Integrated Inbox in business space, add and configure the Integrated Inbox widget to a window by using these steps:

1. For the Customer Complaints solution, provide a separate space for the managers to view their integrated inbox. Create a separate space called ICM IBPM integrated space to get an integrated view of work items and tasks in both IBM Case Manager and IBM BPM.
2. In this space, create two new windows:
 - An Inbox window
 - Work on tasks windowAlternatively, you can also use the Integrated Inbox space template to create a space with an Inbox window and a Work on tasks window.
3. Add the following widgets to the Inbox window:
 - Inbox
 - Connector for IBM Business Process Manager
 - Task Filter
 - Page Navigator
4. Add the following widgets to the Work on tasks window:
 - Task information
 - Task filter
 - Page Navigator

17.4.5 Configuring the Inbox window for the integrated space

Configure the Inbox window for the ICM IBPM integrated space by using these steps:

1. Go into the edit mode for the Inbox window.
2. Configure the Inbox widget from the **Edit Settings** menu as shown in Figure 17-18.

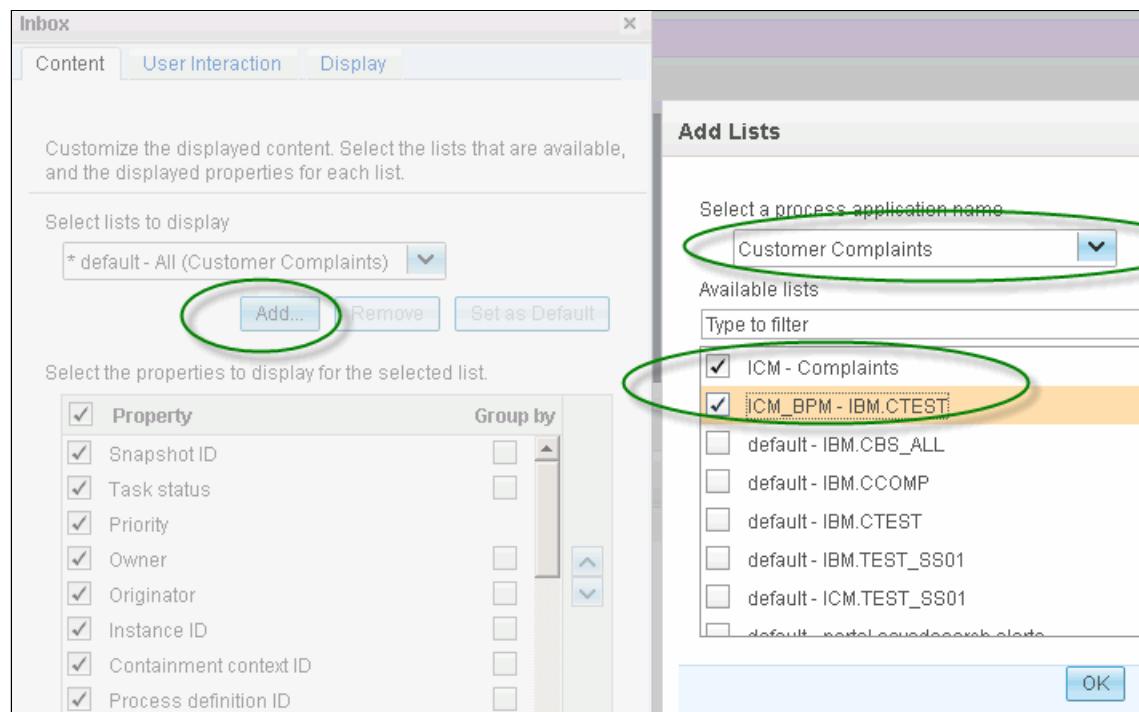


Figure 17-18 Adjusting Inbox settings

3. Click **Add** on the Content tab. This process displays the Add Lists pane.
 4. Select the process application name, **Customer Complaints**. From **Available lists**, select **ICM.Complaints** and **ICM_BPM-IBM.CCALL**.
- ICM.Complaints** is the in-basket list for the Manager Role (CC_Manager queue) in IBM Case Manager. **ICM_BPM-IBM.CCALL** retrieves the federated list of work items and tasks from in-baskets in IBM Case Manager and IBM BPM.

5. Customize each list by selecting the properties that you want to be displayed in the inbox. For the example shown in Figure 17-19, select the Customer Name, Customer Rating, and Case Number properties to be displayed in the inbox column. They are displayed when the user views the ICM.Complaints in-basket.

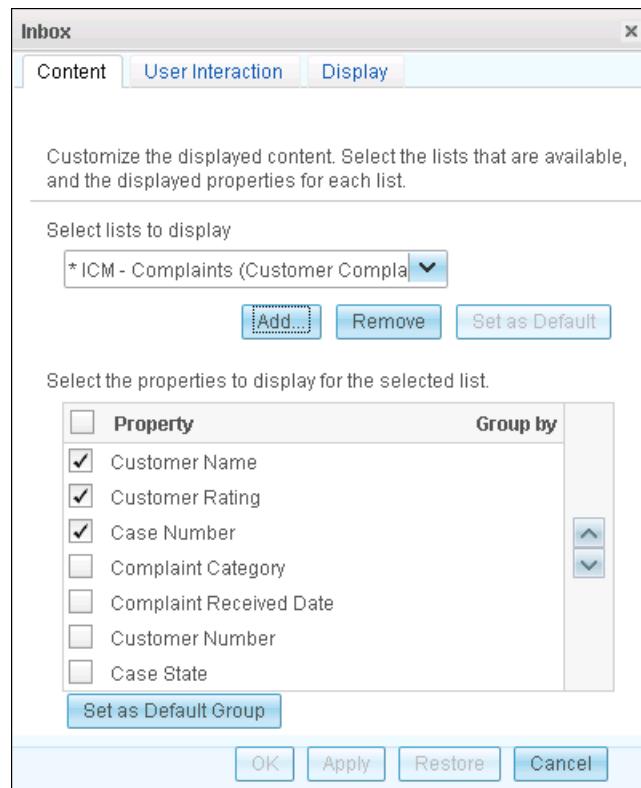


Figure 17-19 Customizing the IBM Case Manager inbox list

- For user interaction filters that allow users to filter work items and tasks, select the views and actions you want available. For the example shown in Figure 17-20, select **My work**, **Unassigned work**, and **All items** as the views for the list.

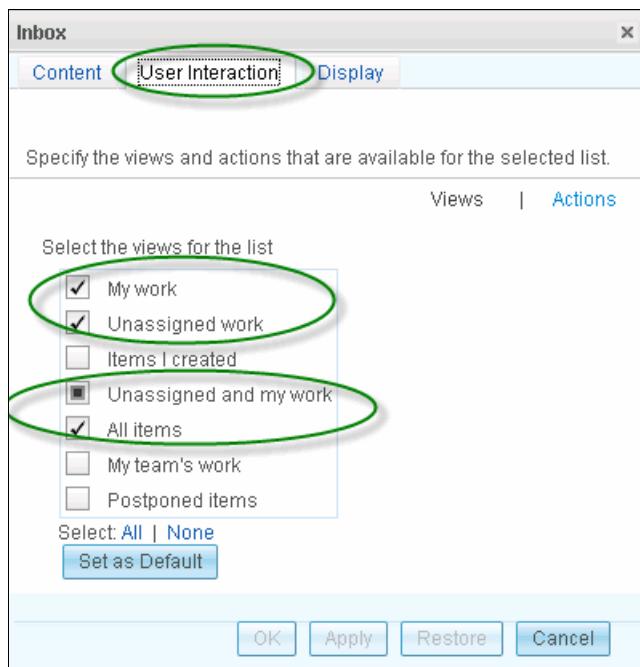


Figure 17-20 User interaction filter views

Table 17-1 describes the available views in the user interaction filters. It explains how they are applicable to work items in IBM Case Manager and tasks in IBM BPM.

Table 17-1 User interaction filters

View	IBM BPM	IBM Case Manager
Work on [my tasks]	Lists tasks that are assigned to the user: That is, in a claimed state and owner set to current logged in user.	Lists work items that are assigned to the user: That is, in the Personal Inbox of current logged in user, which is filtered by solution.
Assess available [tasks]	Lists all tasks available to be worked on and not already assigned.	Lists work items in the Role queue that are not locked by other users.

View	IBM BPM	IBM Case Manager
Check status of [tasks]	Lists tasks that are started or drafted to check status.	Not applicable.
Assess and work on my [tasks]	Combines both Work on [my tasks] and Assess available [tasks].	All work items in logged in users' Personal Inbox and in the role queue that are not locked by other users.
Manage Tasks	Lists tasks that the user can manage.	Not applicable.
Resume Backlog	Lists tasks that are pending.	Not applicable.

7. Wire the Inbox widget to the Connector for IBM Business Process Manager widget as shown in Figure 17-21.
 - Wire the **Action Requested** event for the Inbox widget to the **Action Requested** event for the Connector for IBM Business Process Manager widget.
 - Hide the Connector for IBM Business Process Manager widget.

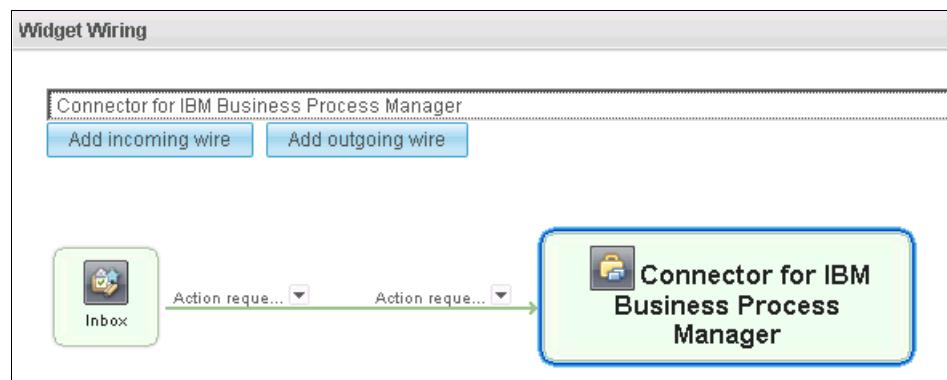


Figure 17-21 Wiring for Inbox and Connector for IBM Business Process Manager widgets

- Wire the Task Filter widget to the Inbox and the Connector for IBM Business Process Manager widgets as shown in Figure 17-22.
 - Wire the outgoing items-selected, **Action requested** and **Focus changed** events from the Inbox to the corresponding incoming events on the Task Filter widget.
 - Wire the **Action requested** event from the Task Filter widget to the **Trigger page** event of the Page Navigator widget.
 - Hide the Task Filter widget.

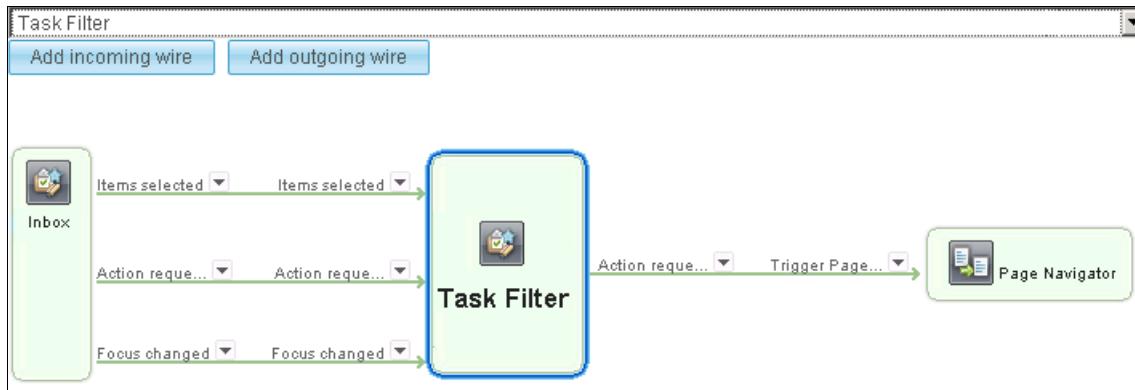


Figure 17-22 Wiring for Task Filter, Inbox, and Page Navigator widgets

17.4.6 Configuring the Work on tasks window for the integrated space

Configure the Work on tasks window by completing these steps:

- Open the Work on tasks window in edit mode. Wire the task Information widget to the Task Filter and the Page Navigator widgets as shown in Figure 17-23 on page 613:
 - Wire the outgoing **Action requested** event from the Task Filter widget to the incoming **Action requested** event on the Task Information widget.
 - Wire the outgoing **Tab changed** event from the Task Information widget to the incoming **Tab changed** event on the Task Filter widget.
 - Wire the outgoing **Tab changed** event from the Task Filter widget to the incoming **Trigger Page** event on the Page Navigator widget.

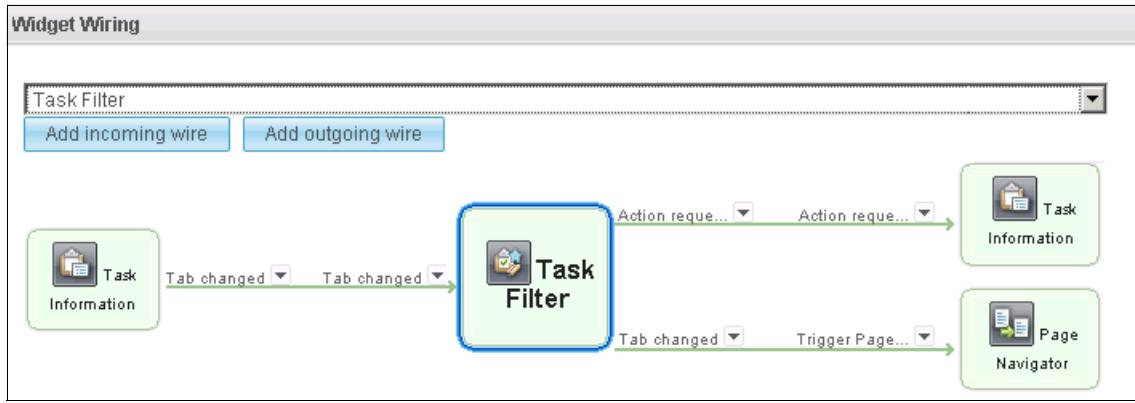


Figure 17-23 Wiring for Task Filter, Task Information, and Page Navigator widgets

2. Configure the Page Navigator widget settings on the Work on tasks window as shown in Figure 17-24. The Page Navigator widget is used to switch to the Inbox window after the step or the task is completed on the Work on tasks window.

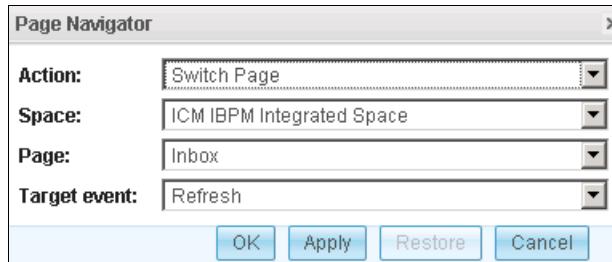


Figure 17-24 Page Navigator widget settings on the Work on tasks window

3. Hide the Page Navigator and the Task Filter widgets in the Work on tasks window.

- On the Inbox window, configure the Page Navigator widget to load the Work on tasks window when a work item or task is selected for editing. Configure the Page Navigator widget settings on the Inbox window as shown in Figure 17-25.

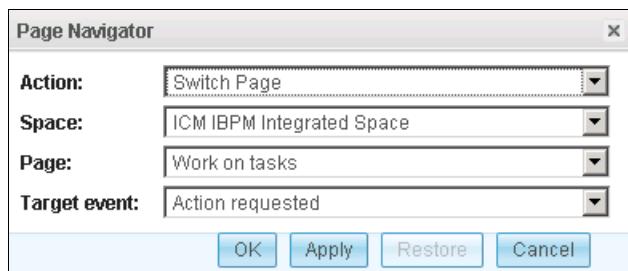


Figure 17-25 Page Navigator widget settings in the Inbox

Configuration for the Integrated Inbox is complete, and it can display both work items and tasks from IBM Case Manager and IBM BPM. When the Customer Outreach task is started, a task is created in IBM BPM. This task is displayed in the Inbox window of the ICM IBPM Integrated space as shown in Figure 17-26.

Inbox				
	CC_CustomerName	CC_CustomerRating	Complaint Category	Case Number
<input checked="" type="checkbox"/>	Tim Doe	Platinum		
	John	Gold	Product	1111
	Jim Jackson	Platinum	Service	9098
	Tammy Jones	Platinum	Product	1900

Figure 17-26 Inbox showing IBM BPM tasks and IBM Case Manager work items

If the internal names of the properties in IBM BPM and IBM Case Manager match, those property values are displayed in the same column. To enhance usability, certain system fields are automatically matched. These fields must be selected in the property list and in the saved search query (Table 17-2).

Table 17-2 System fields that are automatically mapped in Inbox

Field Name	IBM BPM	IBM Case Manager
assignedToUser	OWNER	User name associated with the F_LockUser. Sorting on this field is not available
instanceName	PI_NAME	F_Subject

Field Name	IBM BPM	IBM Case Manager
taskActivityName	NAME	F_StepName
taskReceivedDate	ACTIVATED	F_CreateTime

IBM BPM task widgets support several built-in actions for the tasks. If an action is not supported for IBM Case Manager, it is not available. These actions can be configured on the Inbox widget as shown in Figure 17-27.

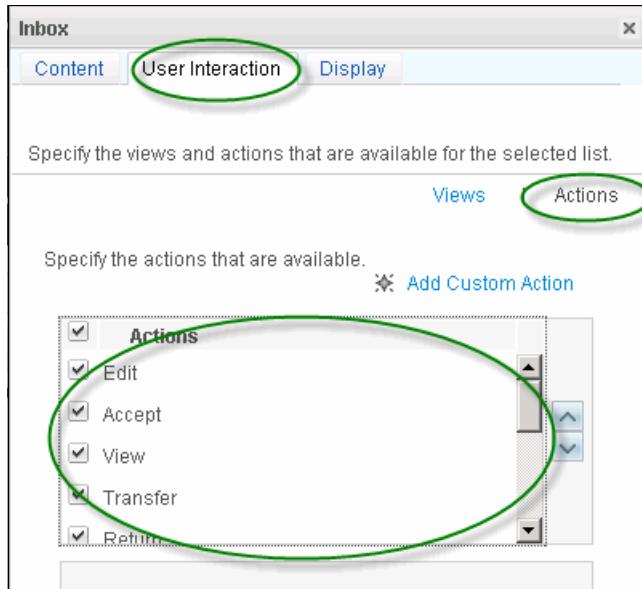


Figure 17-27 Actions in the Inbox tasks

IBM Case Manager supports actions as a right-click menu on the work item in the In-basket widget, but not on the Integrated Inbox widget. Table 17-3 shows the actions that are available on the tasks in the Inbox widget and their behavior in IBM BPM and IBM Case Manager.

Table 17-3 Meaning of actions in the Inbox widget

Action	IBM BPM	IBM Case Manager
Edit (Open item)	Opens the task and makes changes. Only enabled if user has claimed or manages the task.	Opens the work item in the Work Details window.
Accept (Move to inbox and open item)	Claim the task and open it for editing.	Never enabled.

Action	IBM BPM	IBM Case Manager
View	Open and view the task in read-only mode.	Never enabled.
Return (Return item)	Cancel the claim.	Never enabled.
Transfer (Reassign item)	Assign the task to a different person.	Assign the work item to a different person. This action is only enabled if the item is assigned to a user or not yet assigned or locked.

To edit the task, complete these steps:

1. Select the IBM BPM task.
2. Accept and edit the task. This action opens the task in the Work on tasks window for IBM BPM as shown in Figure 17-28.

The screenshot shows the 'Task Information' window for the 'VP Customer Engagement' task. The window has a title bar 'Task Information' and a toolbar with 'Actions' and 'Details' buttons. Below the toolbar is a step indicator 'Step: VP Customer Engagement'. The main area contains a form titled 'VP Customer Engagement' with the following fields and values:

Customer Name:	Tim Doe
Telephone:	8887776654
Address:	100 Beverly Drive
Customer Rating:	Platinum
Complaint Description:	Overbilled for product X1000

An 'Ok' button is located at the bottom right of the form area.

Figure 17-28 Work on tasks window for Customer Outreach process

3. Complete the task. The Work on tasks window closes and control switches back to the Inbox window.



Integration with IBM Content Analytics

IBM Content Analytics helps gather and mine unstructured and structured data to gain actionable insights that improve or enhance business operations. IBM Case Manager can use the natural language processing techniques of Content Analytics to discover patterns, trends, and correlations between various cases. These discoveries help the case workers and executives with insights into their business and an opportunity to respond with appropriate actions.

This chapter includes the following sections:

- ▶ IBM Content Analytics overview
- ▶ Installing Content Analytics Version 2.2
- ▶ Installing IBM FileNet Content Engine Client
- ▶ Configuring script to resolve Content Engine Client paths
- ▶ Configuring crawlers to access case solutions
- ▶ Rapid insights detection with the Text Mining application

18.1 IBM Content Analytics overview

Content Analytics processes your textual data to help you search, discover, and run the same analytics that you can currently run on structured data. With Content Analytics, you can use your text in ways that were only previously attainable from your structured data.

Content Analytics delivers new business understanding and visibility from the content and context of textual information. For example, you can identify patterns, view trends over time, and reveal unusual correlations or anomalies. You can explain why events are occurring and find new opportunities by aggregating the voices of customers, suppliers, and the market. You can track and drive improvement in non-quantitative business metrics through content dashboards, reports, and scorecards. In addition, Content Analytics helps reduce costs by exposing irrelevant or obsolete content for deletion.

With Content Analytics, you can define many facets (aspects) of your data, with each facet potentially leading to valuable insights for various users. For example, you might define a weekend destination facet that consists of major places where people travel over the weekend. You might also define an activity facet that consists of typical activities people do during their weekend travel. With such facets, a tourist industry analyst can analyze which types of people tend to travel to which specific locations. These people can be categorized based on their age, profession, gender, and other aspects. You can further identify the types of activities they engage in over the weekend.

Content Analytics is a tool for reporting statistics and for obtaining actionable insights. *Actionable insights* refers to insights into data that lead to action. Content Analytics provides far more value than merely reducing the workload of manual analysis. Content Analytics brings the power of business intelligence to all of your enterprise information, not just your structured information. The result helps you achieve the most value from all your data, regardless of its structure.

IBM Case Manager integrates with Content Analytics to analyze the structured and unstructured contents of cases to identify actionable insights. A case in a solution consists of structured and unstructured attributes. Default attributes such as the following make up the *structured components* of a case data:

- ▶ Case ID
- ▶ Case State
- ▶ Date the case was opened
- ▶ Date the case was closed

- ▶ Task that is related to a case
- ▶ Custom attributes defined during the solution building process and other fields

Documents and other artifacts that are gathered from the client and comments exchanged by the knowledge workers are the *unstructured components* of a case data. As the number of cases grows in the system, business analysts must analyze cases to measure the quality and quantity of case processed by the hosting organizations. The analysis helps the organization to measure its delivery of business commitments and identify shortfalls to pursue opportunities. Because the cases also include case documents, artifacts, and comments, you must acquire information from unstructured artifacts.

The integration inherits the structured attributes, applies Natural Language Processing and Named Entity Extractors to extract entities such as Parts Of Speech (POS), Names, Organizations, and Locations. The product can be customized to extract entities relevant to a vertical industry. The *Text Miner application* that is bundled in Content Analytics helps bringing the attributes and entities extracted from unstructured artifacts for discovering and reporting insights. The following sections addresses the installation and configuration that are required to integrate Content Analytics and IBM Case Manager.

The purpose of this chapter is not to explain in-depth how Content Analytics works and how to use it to discover actionable insights. The focus is mainly on the integration piece with IBM Case Manager. For more information about Content Analytics, see *IBM Content Analytics Version 2.2: Discovering Actionable Insight from Your Content*, SG24-7877.

18.2 Installing Content Analytics Version 2.2

This section provides the instructions to run a default installation of Content Analytics on a Windows XP system. This is a single server installation with default ports for internal communication. A pre-bundled web application server, namely JETTY, is used to host the administration and the Text Mining applications. For advance installation of the product, see *IBM Content Analytics Version 2.2: Discovering Actionable Insight from Your Content*, SG24-7877.

18.2.1 Installing Content Analytics

To install Content Analytics with the GUI installer, complete the following steps:

1. Extract the installation image file. Find the launchpad command in the extracted files. For AIX and Linux, use **launchpad.sh**. For Windows, use **launchpad.exe**.
2. Run the launchpad command to start the installation.
3. Select the language to use during the installation (Figure 18-1). For the scenario, select **English**. Then click **OK**.

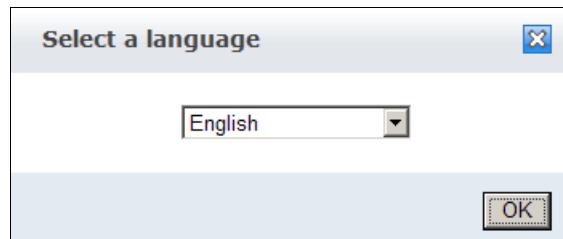


Figure 18-1 Selecting the language that is used during installation

4. In the Welcome window (Figure 18-2) that opens, click **Install Product**.

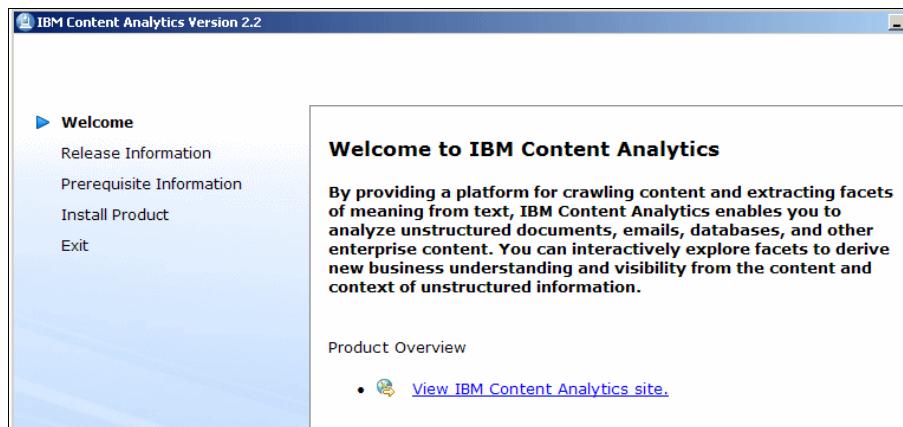


Figure 18-2 Content Analytics installation: Welcome window

5. In the Install Product window that shows the links to run the installer (Figure 18.2.2), click **Launch Content Analytics installation program**. For more information, see 18.2.2, “Installing Content Analytics V2.2 FixPack 2” on page 624.

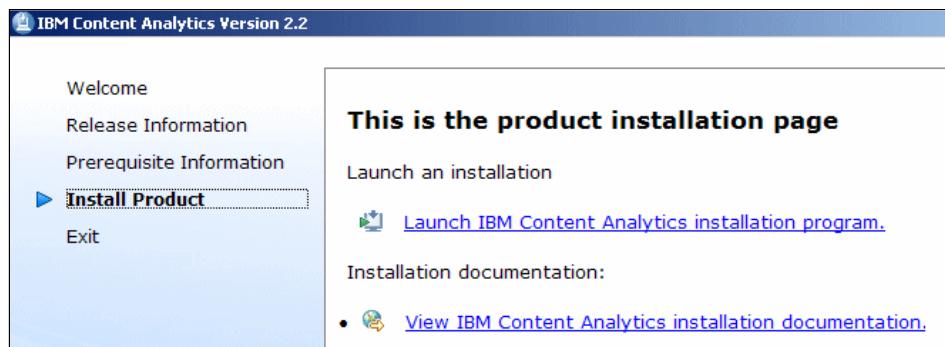


Figure 18-3 Content Analytics installation: Install Product window

6. In the window that opens with a banner, click **OK** to proceed.
7. In the Software License window (Figure 18-4), accept the license agreement, and click **Next**.

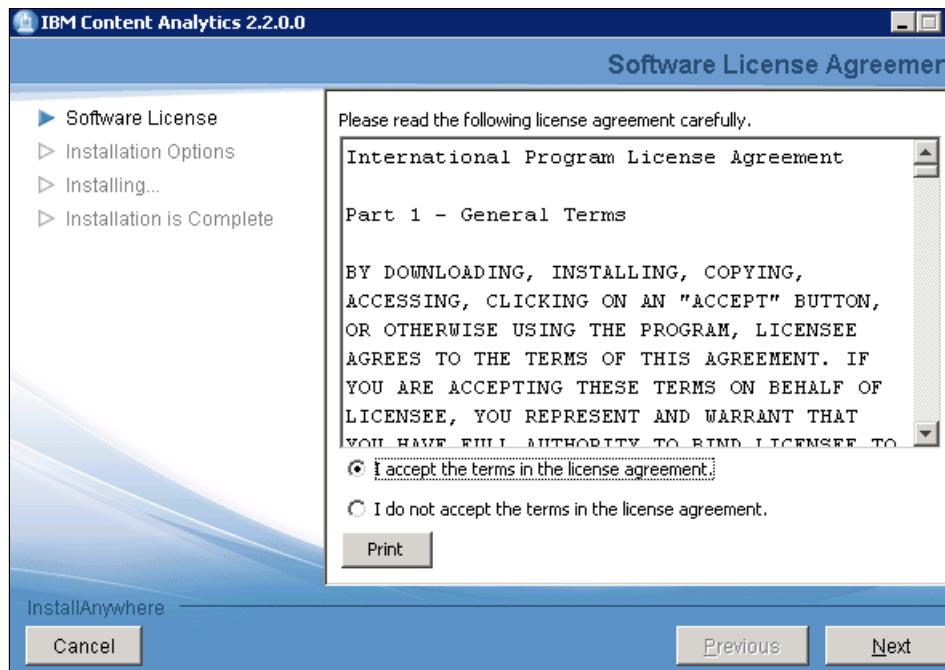


Figure 18-4 Content Analytics installation: Software License window

8. In the installation options window (Figure 18-5), specify the following parameters:

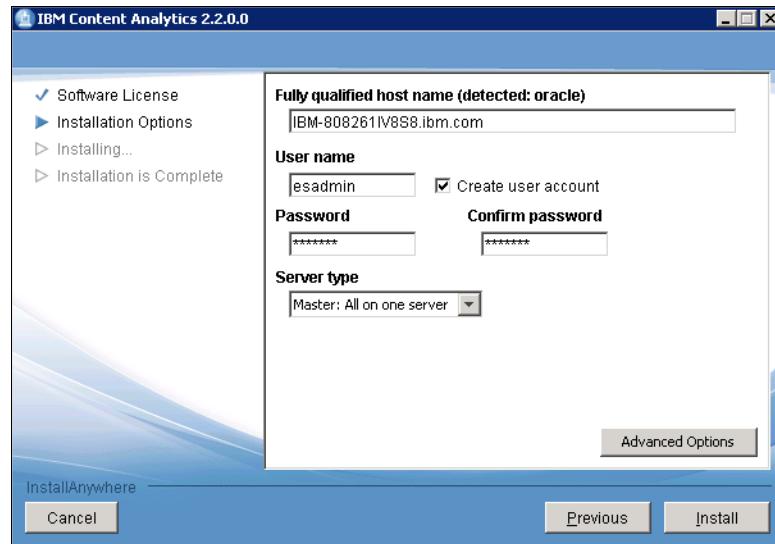


Figure 18-5 Content Analytics installation: Installation Options window

- Fully qualified host name.
- User name and password. You can create a user or use an existing user for the administrator user.
- Server type. If you plan to perform a distributed server installation or extra server installation, select the appropriate server type. For the scenario, select **Master: All on one server**.
- For a default configuration, click **Install** to start the installation program. See IBM Content Analytics to learn about advance configuration options.

As the product installs, the progress window is displayed (Figure 18-6).

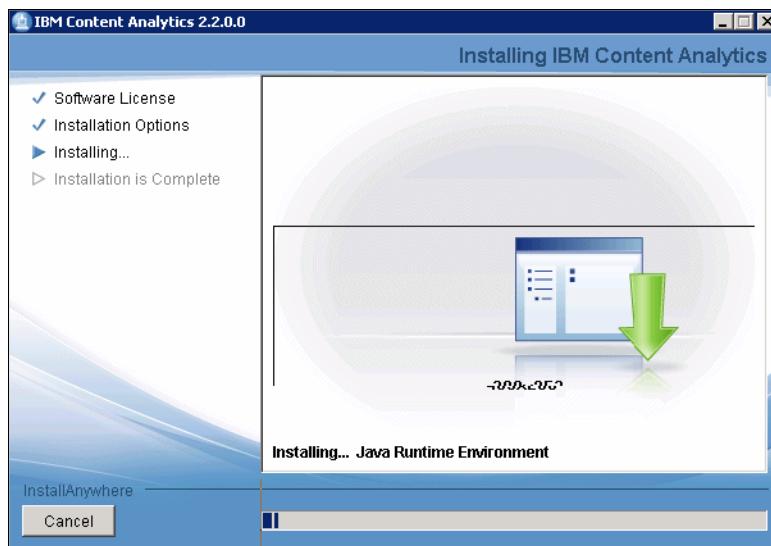


Figure 18-6 Installation progress window

9. After the installation is complete, you are prompted to reboot the system (Figure 18-7). Select the restart option and click **Done**. The system reboots and starts the required services to start IBM Content Analytics. Ignore the First Steps window that is displayed after the reboot.

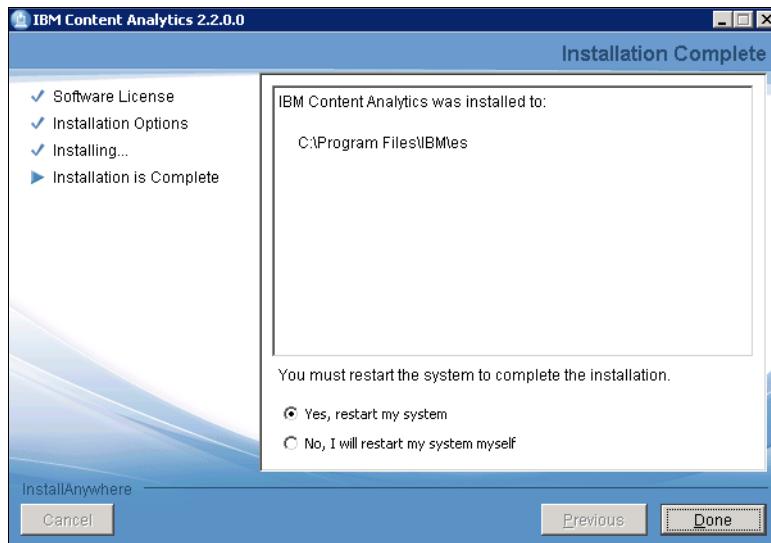


Figure 18-7 Rebooting the system after installation

18.2.2 Installing Content Analytics V2.2 FixPack 2

After Content Analytics is successfully installed, make sure that you also install the latest fix pack of the product. At the time of writing, Fix Pack 2 is available. For more information, see the IBM Fix Central for the latest available fix pack at the time of installation:

<http://www.ibm.com/support/fixcentral>

Perform the following steps to install FixPack 2 for Content Analytics v2.2:

1. After you unpack the image, click **install.exe** to start the installation program.
2. At the initial installation window, select **English** and click **OK**.
3. On the Software License Agreement window, accept the license agreement and click **Next**.
4. Confirm the directory of the product installation and click **Next** as shown in Figure 18-8.

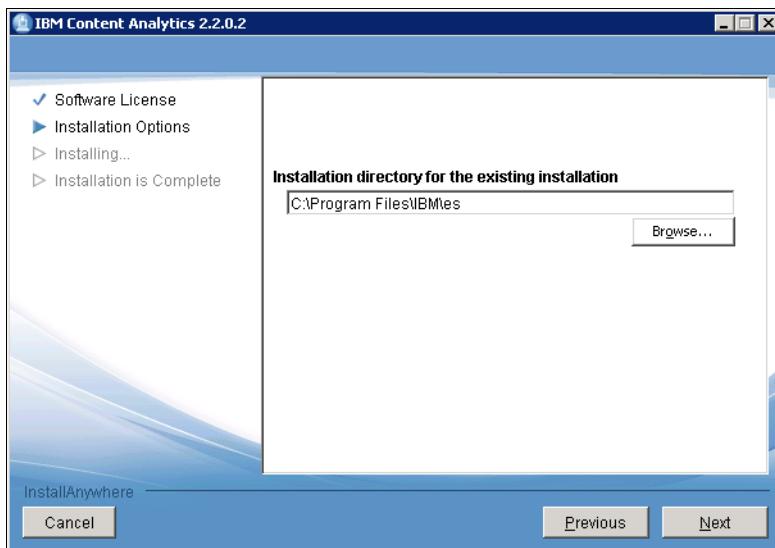


Figure 18-8 Confirming the directory of product installation

5. Select **Yes, restart my system** and click **Done** as shown in Figure 18-9.

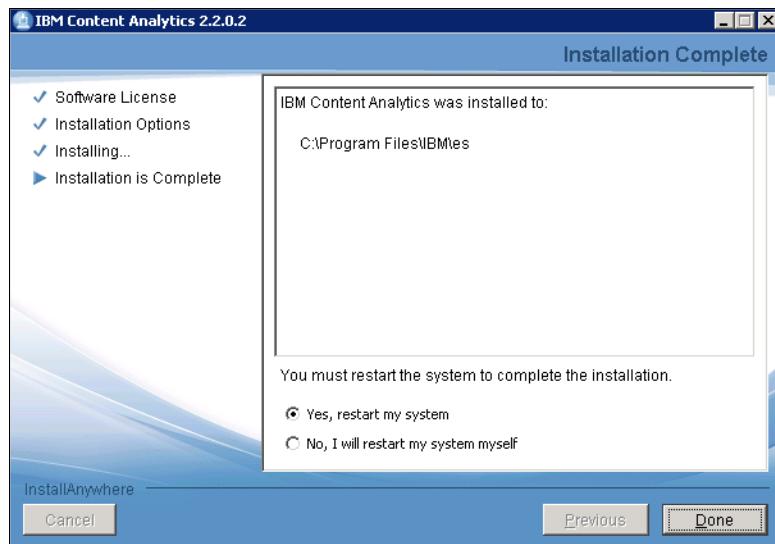


Figure 18-9 Restarting the system

6. After the system reboots, start the First Steps application by clicking **Start** → **IBM Content Analytics** → **First Steps**.
7. Select the **Start Server** option as shown in Figure 18-10. This action starts the services that are required to access the product capabilities. After the successful start of the server, click **Exit** to exit the application.

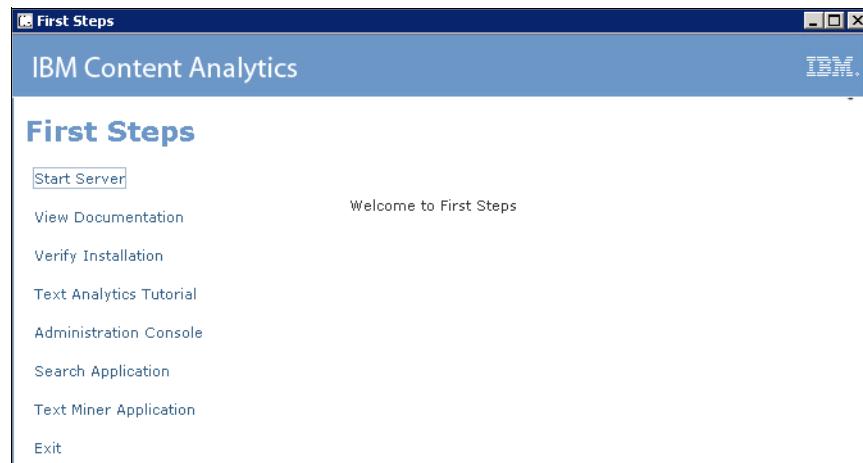


Figure 18-10 Using the First Steps application to start the server

18.3 Installing IBM FileNet Content Engine Client

Content Analytics uses a stand-alone crawler to crawl the FileNet Content Engine that is hosting the case solution. The crawler that is pre-bundled in the product requires access to the FileNet client libraries, which is not part of the product distribution. You must install the software for Content Engine Client on the server where you installed Content Analyzer.

To install the IBM FileNet Content Engine Client, complete these steps:

1. After you unpack the software on the server where you installed Content Analytics, click the **5.1.0-P8CE-Client-Win.exe** program.
2. On the Welcome window, click **Next** to continue.
3. On the Software License Agreement window, accept the license terms and click **Next** to continue.
4. Accept the default location as shown in Figure 18-11.

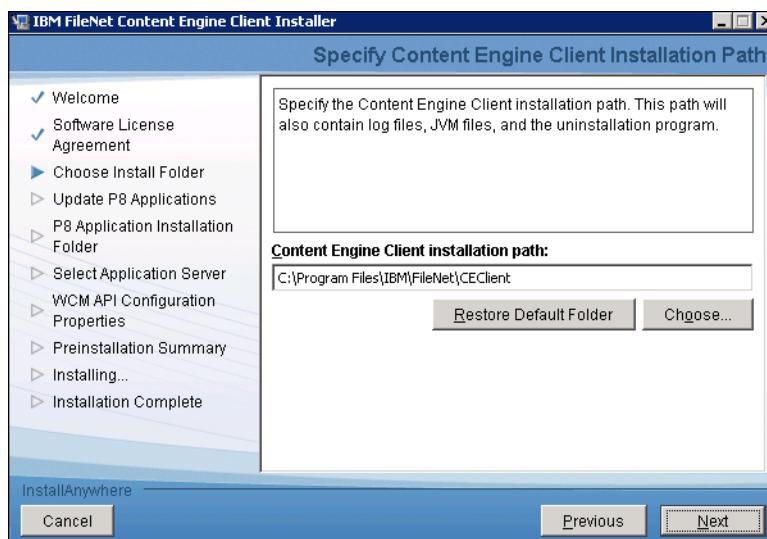


Figure 18-11 Default location for IBM FileNet Content Engine Client

5. In the window shown in Figure 18-12 on page 627, consult your Content Engine administrator to get the following parameters:
 - Content Engine server host name
 - Port number
 - Protocol

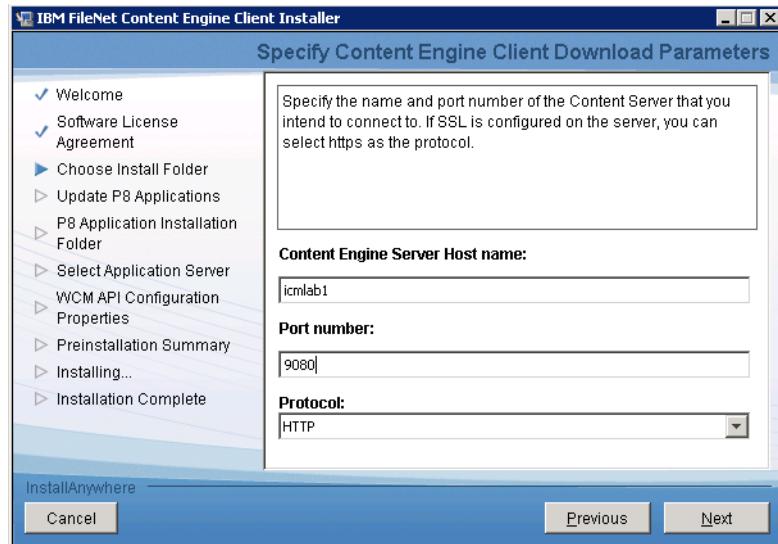


Figure 18-12 Connection parameters

The scenario uses the default installation of Content Engine on WebSphere Application Server.

6. Do not select any check boxes as shown in Figure 18-13. Click **Next**.

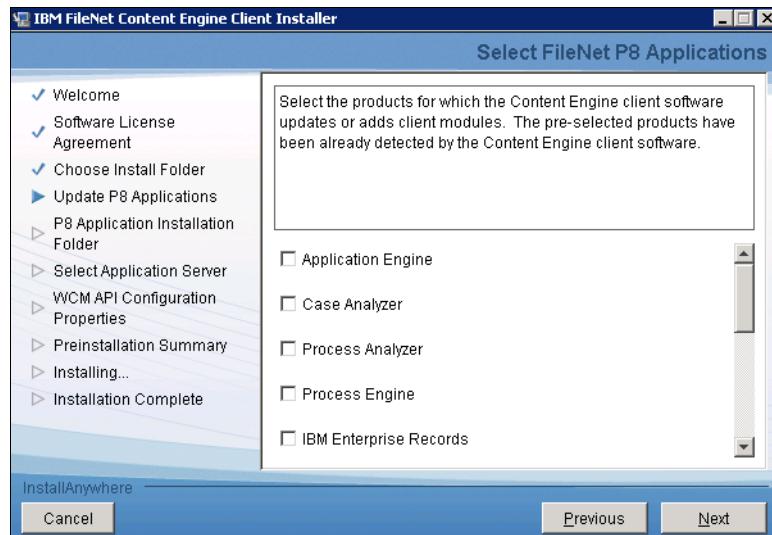


Figure 18-13 Optional selections

7. Click **Install** to continue the unpacking of the software onto the local disk as shown in Figure 18-14.

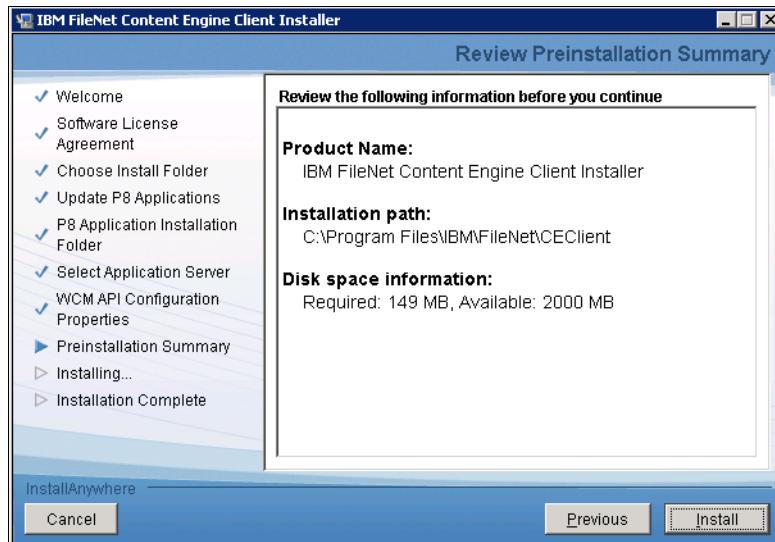


Figure 18-14 Continuing the installation

8. Click **Done** to finish the installation of IBM FileNet Content Engine Client.

18.4 Configuring script to resolve Content Engine Client paths

Start a script in Content Analytics to update the path for resolving the required client libraries. Perform these steps to resolve the Content Engine Client paths:

1. Start a DOS command window on the server that hosts Content Analytics.
2. From the command window, run the **escrfilenet.vbs** script. For UNIX OS, there is an equivalent script called **escrfilenet.sh**. This script must be run to configure the case crawler.

3. Enter the path selected for installing the Content Engine Client components as shown in Figure 18-15. For more information, see 18.3, “Installing IBM FileNet Content Engine Client” on page 626.

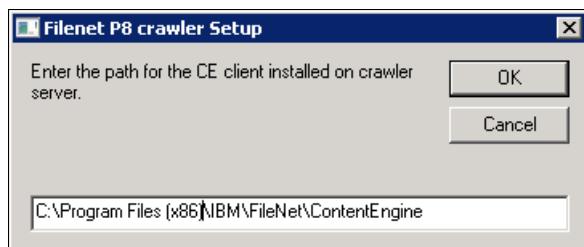


Figure 18-15 Specifying the path to the content engine client components

4. Click **OK** to complete the script.
5. Now that the execution of the script is complete. Stop and restart the services that are related to IBM Content Analytics. Use the First Steps application.

18.5 Configuring crawlers to access case solutions

This section describes the administration and configuration of Content Analytics to gather and parse the structured and unstructured artifacts of a case.

18.5.1 Starting the administration console and creating collections

Content Analytics provides an administration user interface to configure the collections and crawlers that are required to gather and parse artifacts. Follow these steps to start and log in to the administration console and create a collection:

1. Start the administration console from the system startup by clicking **Start** → **IBM Content Analytics** → **Administration Console**. For UNIX OS, start the internet browser and enter the following URL:
`http://<your server name>:8390/ESAdmin`
2. Log in with the user and password you identified during the installation.

3. Click **Create Collection** as shown in Figure 18-16.

The screenshot shows the IBM Content Analytics interface. The top navigation bar includes links for Collections, System, Security, Search Customizer, Analytics Customizer, Log Out, and Help. The main content area is titled "Collections". It features a "Create Collection" button and a message indicating the page was last refreshed on Monday, November 28, 2011, at 12:39:58 PM EST. Below this is a "Search Collections" section with a table header for "Collection name", "Crawl", and "Parse". A "Text Analytics Collections" section follows, also with a similar table header. At the bottom of the page is a search bar with a magnifying glass icon and a help link.

Figure 18-16 Creating the collection

4. Enter the name of the collection and the collection type as shown in Figure 18-17. In the example, these are **Complaints case solution** for the name and **Text analytics collection** for the type. Click **OK** to continue.

The screenshot shows the 'Create Collection' dialog box from IBM Content Analytics. At the top, there's a navigation bar with links for Collections, System, Security, Search Customizer, Analytics Customizer, Log Out, and Help. Below the navigation bar, the title 'Collections > Create Collection' is displayed. The main content area is titled 'Create a Collection' and includes a link to 'Help for this page'. A note states: 'A collection contains the various sources that users can search with a single query. After you click OK, you return to the Collections view. From the Collections view, click the Edit icon for your new collection to add content to it.' There are two required fields: 'Collection name:' containing 'Complaints Case Solution' and 'Collection type:' containing 'Text analytics collection'. Below these, under 'General options', there are two dropdown menus: 'Document cache (required for thumbnail generation, rebuilding from cache, and content export)' set to 'Enable the document cache' and 'Thumbnail generation' set to 'Do not enable thumbnail generation'. At the bottom of the dialog are 'OK' and 'Cancel' buttons.

Figure 18-17 Specifying collection attributes

Figure 18-18 shows the created collection with a default instance of Parser and Search.

The screenshot shows the IBM Content Analytics interface. At the top, there's a navigation bar with links for Collections, System, Security, Search Customizer, Analytics Customizer, Log Out, Help, and About. Below the navigation bar, a message box displays: "FFQM0204I The collection was successfully created. To add content to the collection, click the Edit icon next to the crawler." A link "Help for this page" is also present. The main content area is titled "Collections". It shows two sections: "Search Collections" and "Text Analytics Collections". Under "Search Collections", there's a table with one row for "Collection name" (Complaints Case Solution) with columns for "Crawl" (containing a blue square icon) and "P" (containing a blue square icon). Under "Text Analytics Collections", there's a similar table with one row for "Collection name" (Complaints Case Solution) with columns for "Crawl" (containing a blue square icon) and "P" (containing a blue square icon). A "Create Collection" button is located at the top left of the collections section. The status bar at the bottom indicates "Last refreshed: Monday, November 28, 2011 12:43:24 PM EST" and a "Refresh" button.

Figure 18-18 Results of a created collection

Crawler is an instance that connects to a data source and gathers data. *Parser* is an instance that tokenizes documents and extracts entities. *Search* is an instance that exposes the extracted entities and facets for Text Mining.

18.5.2 Configuring crawler and parser for the case solution

After the collection is created, you can configure the crawler and parser for the case solution.

Perform these steps to configure a crawler instance and a parser instance to annotate case-related artifacts:

1. Click the **Crawl** icon next to the newly created collection as shown in Figure 18-19.

This screenshot is identical to Figure 18-18, showing the IBM Content Analytics interface with the "Complaints Case Solution" collection created. The "Crawl" column for this collection contains a blue square icon with a white square inside, indicating it is selected or active. The rest of the interface, including the "Parser" column, the "Text Analytics Collections" section, and the "Create Collection" button, remains the same.

Figure 18-19 Selecting the Crawl icon

2. Click **Create Crawler** as shown in Figure 18-20.



Figure 18-20 Creating the crawler instance

3. Select **IBM Case Manager** from the drop-down menu as shown in Figure 18-21.

A screenshot of the "Create a Crawler" wizard. The title bar says "IBM Content Analytics". The breadcrumb navigation shows "Collections > Complaints Case Solution : Crawl > Create Crawler". A link "Help for this page" is available. The main section is titled "Create a Crawler" and contains the instruction: "Select the type of data that you want to add to the collection. Next, specify which default values you want to use with the new crawler. Complete the pages of the wizard to create the crawler." A required field "Crawler type:" has a dropdown menu open, showing "Case Manager" as the selected option. Below this is a section titled "Crawler properties" with two radio buttons: "Use the system default values for the new crawler" (selected) and "Clone the values of an existing crawler for the new crawler". At the bottom of the wizard are four buttons: "Back", "Next", "Finish", and "Cancel".

Figure 18-21 Creating crawler for IBM Case Manager

4. Accept the default name and click **Next** as shown in Figure 18-22.

The screenshot shows the 'Case Manager Crawler Properties' configuration screen. At the top, there's a navigation bar with tabs like 'Collections', 'System', 'Security', etc. Below that, a breadcrumb trail shows 'Collections > Complaints Case Solution : Crawl > Crawler type : Case Manager'. The main area is titled 'Case Manager Crawler Properties' and contains a note about applying changes to all documents crawled by this crawler. A field labeled 'Crawler name:' has 'Case Manager crawler 1' entered. There are 'Advanced options' and 'Help for this page' links. At the bottom are 'Back', 'Next', 'Finish', and 'Cancel' buttons.

Figure 18-22 Selecting the name for the crawler

5. In the window shown in Figure 18-23, enter the following arguments:

- Content Engine Web Service URL
- User name for FileNet P8 Administrator
- Password for FileNet P8 Administrator

The screenshot shows the 'Connection Information for the Case Manager Crawler' configuration screen. The title bar includes 'IBM Content Analytics' and various navigation links. The main content area is titled 'Connection Information for the Case Manager Crawler' and includes a 'Help for this page' link. It asks for connection information to access documents. A 'Content Engine Web Service URL' field contains 'http://icmlab1:9080/ws/FNCEWS40MTOM'. Below it, under 'Connection Information', there are two radio button options: 'Reuse configured connection information' (unchecked) and 'Specify new connection information' (checked). Under 'FileNet P8 User Information', the 'User name' field contains 'P8Admin' and the 'Password' field contains '*****'. At the bottom are 'Back', 'Next', 'Finish', and 'Cancel' buttons.

Figure 18-23 Connecting to the Content Engine

Contact your FileNet Administrator for connection details.

6. Figure 18-24 shows the case solutions deployed on the IBM Case Manager.

To showcase the Content Analytics capabilities, create a copy of the generic Customer Complaints solution and name it “Cell Phone Customer Complaints”. The rest of the chapter uses Cell Phone Customer Complaints as an example.

Select **Cell Phone Customer Complaints**, add it to the **Spaces to crawl**, and click **Next**.

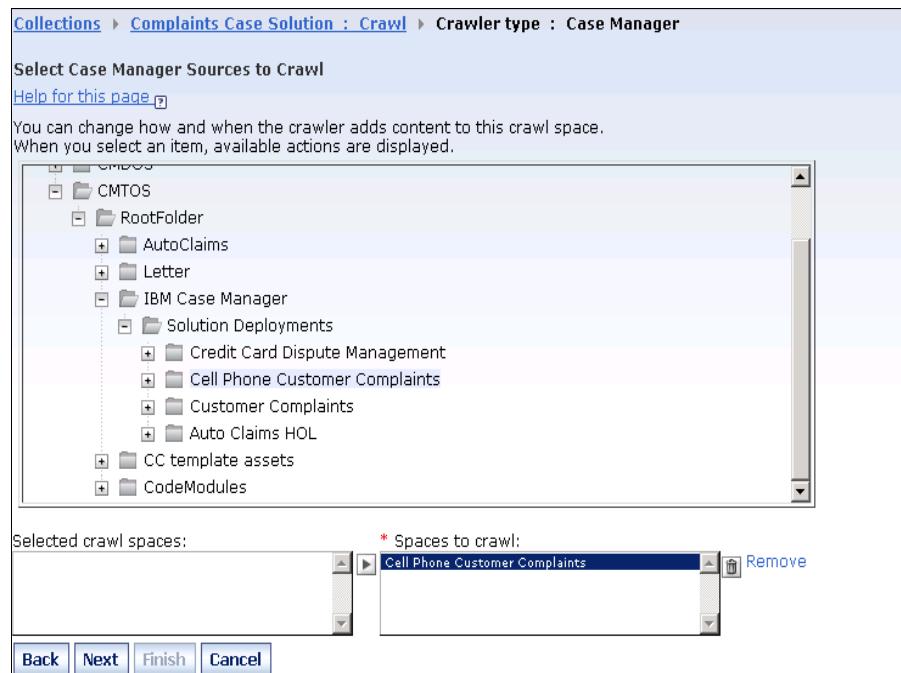


Figure 18-24 Configuring spaces to crawl

7. In the window as shown in Figure 18-25, click **Finish** to complete the creation of the crawler.



Figure 18-25 Completing the crawler

8. Click **Crawl space** as shown in Figure 18-26.

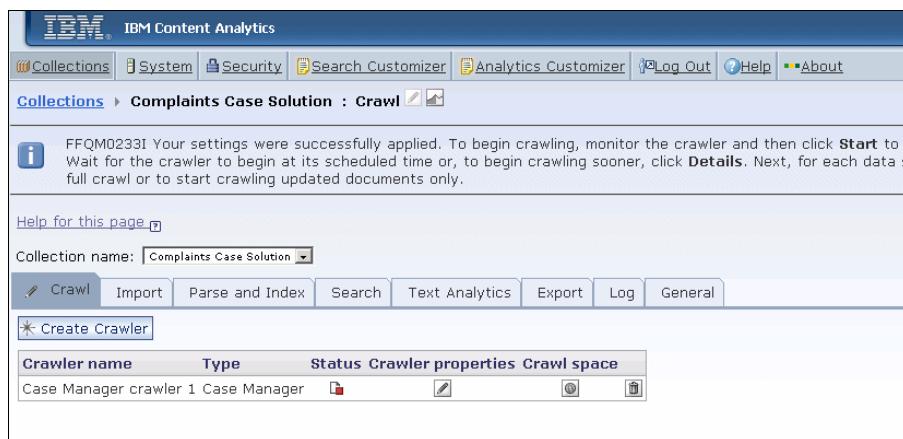


Figure 18-26 Configuring the crawl space

9. Click **Edit Metadata** to identify the list of default case attributes that are gathered by the crawler as shown in Figure 18-27.



Figure 18-27 Default metadata

10. Display the list of default case-related attributes that are crawled and indexed in Content Analytics as shown in Figure 18-28. Click **Cancel** to return to the previous window.

Crawl Name	Search field name	Returnable	Faceted search	Free text search	In summary	Fielded search
<input checked="" type="checkbox"/> ClassName	classname ▾ classname	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> CmAcmCaseIdentifier	caseid ▾ caseid	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> CmAcmCaseState	casestate ▾ casestate	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> DateCreated	date ▾ date	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> Folder.Annotations.CmAcmCommentText	body ▾ body	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> Folder.ClassName	classname ▾ classname	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Folder.CmAcmCaseIdentifier	caseid ▾ caseid	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Folder.CmAcmCaseState	casestate ▾ casestate	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Folder.CmAcmCaseTypeFolder.FolderName	casetype ▾ casetype	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Folder.Folder.CmAcmCaseTypeFolder.FolderName	casetype ▾ casetype	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> MimeType	mimetype ▾ mimetype	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> ObjectType	objecttype ▾ objecttype	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> PathName	path ▾ path	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> content_size	filesize ▾ filesize	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> crawled_date	[default] ▾ [default]	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> docname	title ▾ title	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> filename	[default] ▾ [default]	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> lastmodified_date	[default] ▾ [default]	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
OK Cancel						

Figure 18-28 Default metadata fields crawled and indexed

11. Click the **Edit options** link as shown in Figure 18-29. Use the resulting windows to add the crawling of metadata fields that are customized for the Cell Phone Customer Complaints solution. The case solution builder is used to identify the custom fields.

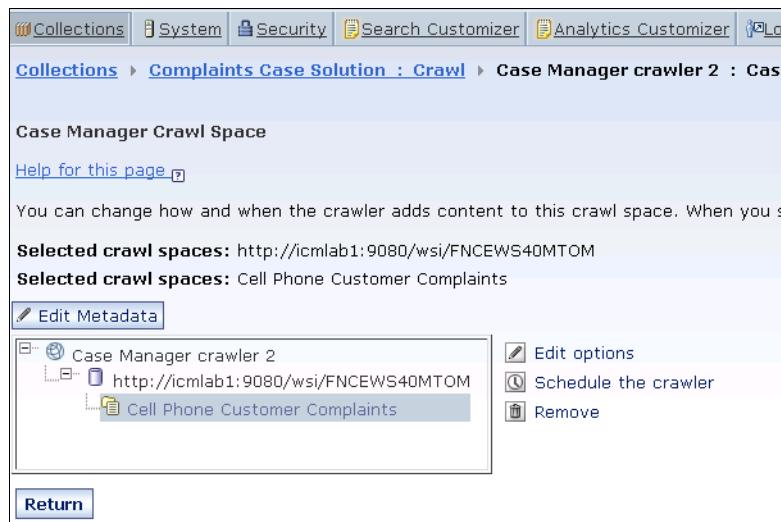


Figure 18-29 Editing options

12. Identify the symbolic names of the fields that were created for this solution when the Case Solution Builder is used. One way is to start the **FileNet Enterprise Manager Client** and access the case folders for the symbolic property names.

On the Content Engine server, start the FileNet Enterprise Manager Client. Coordinate with your FileNet Administrator for the details.

13. Right-click the case folder of one of the cases that was created for the example to display **Properties** as shown in Figure 18-30.

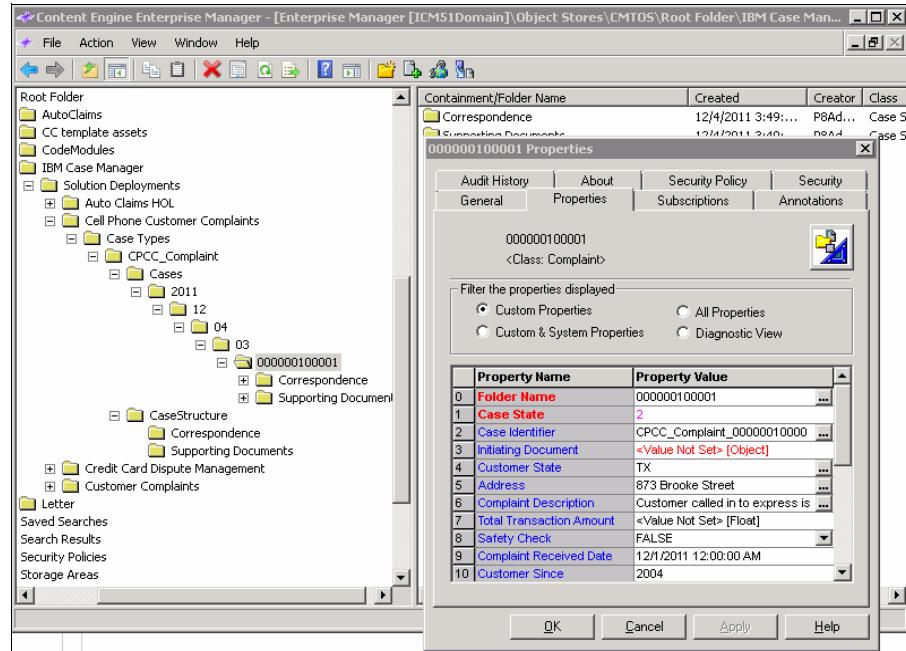


Figure 18-30 Case properties

14. Click **Customer State**. Figure 18-31 shows **CPCC_CustomerName** as the symbolic name of the display field called **Customer Name**.

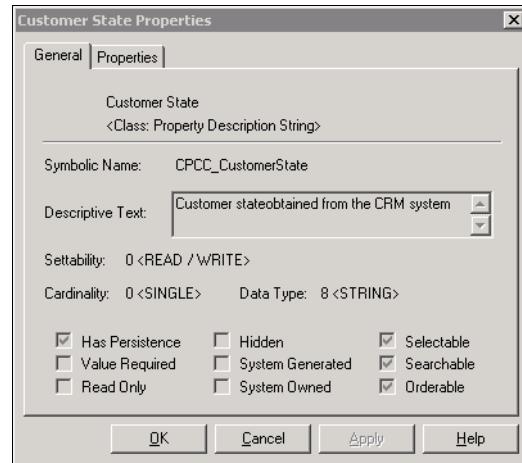


Figure 18-31 Symbolic name for the CustomerState field

CPCC_Address is the symbolic name of the display property called **Address** as shown in Figure 18-32.

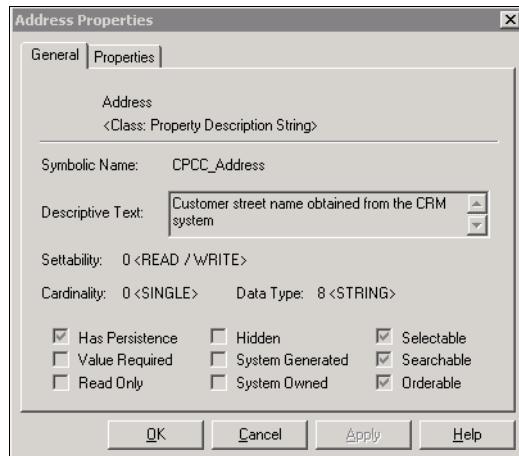


Figure 18-32 Symbolic name for the Address field

15. After you identify the symbolic names of the required case properties, return to the Administration console to update the crawl space for the crawler instance.
16. Click the **Edit Options** link
17. Click **Add Field**. Use the resulting prompts to add all the symbolic names you identified in the earlier steps. Make sure all the newly added fields are enabled for **Facet Searching** and the symbolic field labeled **CCPC_ComplaintDescription** is identified as the **body** of the case. The

contents of **CCPC_ComplaintDescription** is considered as the unstructured attribute and the natural language processing techniques are applied for extracting relevant entities. The results are shown in Figure 18-33.

Field name	Search field name	Returnable	Faceted search	Free text search	In summary	Fielded search	Exact match	Case-sensitive
CPCC_CaseSource	text	CPCC_CaseSourc	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
CPCC_CustomerCity	text	cpcc_customercity	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
CPCC_CustomerName	text	cpcc_customernam	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
CPCC_ComplaintCategory	text	cpcc_complaintca	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
CPCC_Telephone	text	cpcc_telephon	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
CPCC_CustomerRating	text	cpcc_customerrat	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
CPCC_ComplaintStatus	text	cpcc_complaintst	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
CPCC_CaseNumber	text	cpcc_casenumber	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
CPCC_Email	text	cpcc_email	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
CPCC_Valid	text	cpcc_valid	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
CPCC_CustomerSince	text	cpcc_customersin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
CPCC_ComplaintReceivedDate	text	cpcc_complaintre	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
CPCC_SafetyCheck	text	cpcc_safetycheck	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
CPCC_TotalTransactionAmour	text	cpcc_totaltransac	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
CPCC_ComplaintDescription	body	body	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
CPCC_Address	text		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
CPCC_CustomerState	text	cpcc_customersta	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Figure 18-33 All the symbolic names in the crawl space

18.Click **OK** at the bottom of the window to save the changes.

19. Enable a built-in annotator in the product to extract entities such as Names, Places, and Organizations. Click the Parse and Index tab to access a page as shown in Figure 18-34. Click the **Configure document processing pipeline** link.

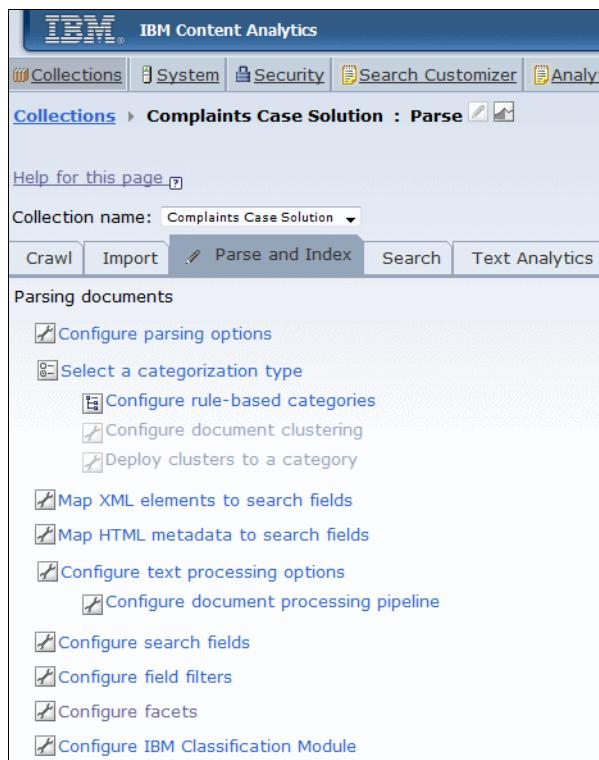


Figure 18-34 Configuring parser for annotations

20. Select the **Named Entity Recognition annotator** box and click **OK** as shown in Figure 18-35.

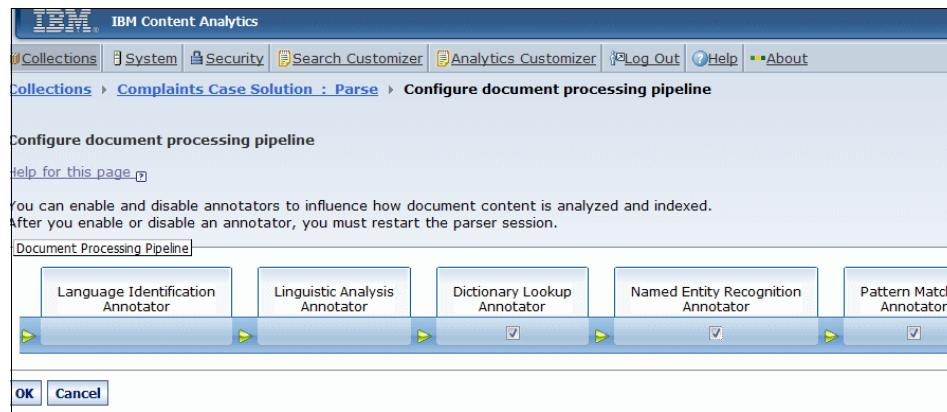


Figure 18-35 Enabling Named entity annotator

18.5.3 Gathering and parsing the case artifacts

After you configure the crawler and parser, complete these steps to start them:

1. On any of the administration pages, click **Monitor**.
2. Click **Crawl** and click the button that resembles a **Play** symbol. The status of the crawler is updated as shown in Figure 18-36.

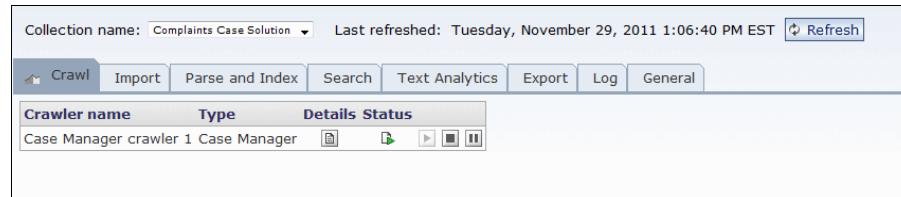


Figure 18-36 Status of the crawler

3. Click **Details** to observe the progress of the crawl.

- Likewise, click the **Parse and Index** tab to start the parser component. The status of the parser is updated as shown in Figure 18-37.

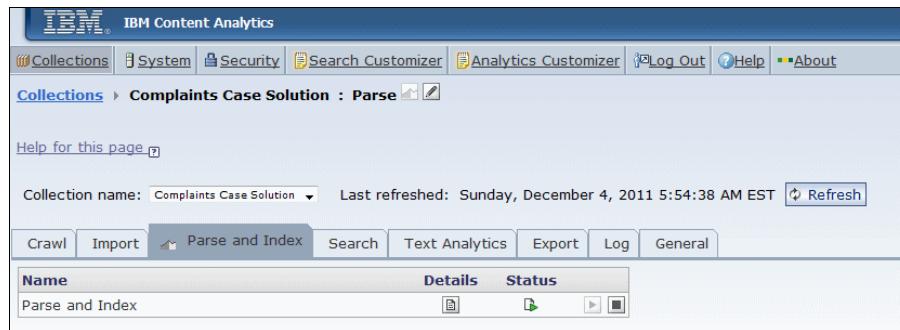


Figure 18-37 Status of the parser

- Click **Details** to observe the progress of the parsing.
- The search server should be started by default. If not, click the **Search** tab to start it. Start the search server to access the Text Mining application.

18.6 Rapid insights detection with the Text Mining application

IBM Content Analytics bundles a Text Mining application that helps with rapid insights detection. The tool showcases the values of the extracted entities in various views to identify frequency counts, correlations between facets, trends, and patterns. The views include Documents, Facets, Time series, Deviations, Trends, Facet pairs, and Connections. To aid with gleaning insights from large amounts of unstructured data, the application uses various color codes to isolate abnormal data. The views and the color coded graphs assist with identifying topics and dynamically build a query to narrow down documents of interest. A built-in dashboarding capability allows findings to be presented as printable reports. The product also integrates with IBM Cognos for a complete integration with Business Intelligence and reporting tools.

Remember: The purpose of this chapter is not to explain in-depth how Content Analytics works and how to use it to discover actionable insights. The focus is to highlight the IBM Case Manager artifacts that are inherited in the fact finding process. For more information about Content Analytics, see *IBM Content Analytics Version 2.2: Discovering Actionable Insight from Your Content*, SG24-7877.

18.6.1 Starting the Text Mining application

Text Mining is a web-based application that is bundled with the product. The web application is hosted on an open source container (JETTY) that is deployed with the installation of the product.

Start the Text Mining application from the system startup by clicking **Start** → **IBM Content Analytics** → **Text Miner Application**. For UNIX OS, you can start the internet browser and type in the following URL:

<http://<your server name>:8393/analytics>

Figure 18-38 shows the landing page of the Text Miner application that gets started on your default web browser.

IBM Content Analytics

Collection: Co... (change) Logged In as: No... | Preferences | My Profile

Documents Facets Time Series Deviations Trends Facet Pairs Connections Dashboard

Results 1-10 of 15 (15/15 results matched)

Facet Navigation Default order ▾

Filter: Clear

Part of Speech ²

Phrase Constituent ²

Named entity ²

My Keywords

Terms of Interest

Voice of Customers (sample)

casestate

casetype

caseid

objecttype

cpcc_casesource

cpcc_customercity

cpcc_customername

cpcc_complaintcategory

cpcc_telephone

cpcc_customerrating

Source	Date	Title	Thumbnail
casemanager	12/4/11	000000100001	
casemanager	12/4/11	000000100002	
casemanager	12/5/11	000000100101	
casemanager	12/5/11	000000100108	

Customer called in to express issues with the recent price raises for the service. The customer is unhappy to learn new service price of 60\$ for the 500 mins plan she has subscribed for. The customer complains an increase in service price but no increase in extra minutes for talk time. Also the customer has been a loyal member since 2004. The customer is objecting for the recent price raise on her plan..

The customer complains about the signal issue at Washington Dulles international airport. The customer can't receive or make calls once he is in the gate for boarding the plane. He has to use the pay phone services to call his customers and assistance to get in touch with his network. The customer has provided a bill for the company to reimburse for the charges he has incurred at this airport. Recently he has spent 125\$ from the airport gates to connect to his business related meetings.

On going problems. I have bought several phones with metro when I started a family service. When I got the phones we were having signal problems and one phone not working at all hardly. Took the phones back and complained about the problem and was told we would have to pay for new ones. We were still under a one year warranty. We only had to phone not even three months. We couldn't afford to buy new phones for all of us so we bought only one phone at that time. Needless to say the phone we bought is only two weeks old and we get no signal with them at John Wayne Airport. We took it back to the store we bought it from and they told us it was a bad phone and we needed to take it to their Corp store and they will have to fix it or will replace it.

Figure 18-38 Text Miner application

The application represents the crawled documents and the extracted entities as facets. The facet navigation integrates with various views to identify abnormal patterns and correlations. The following subsections address the structured and unstructured case artifacts that are used to detect actionable insights.

To better explain the concept and information in this section, the sample Customer Complaints solution is changed to cater to a cell phone company. The cell phone company uses this case management solution to handle complaints from its customers, including cell phone products, services, billing, and other issues. In the example, there are 15 cases in the solution. In real world scenario, you have hundreds, thousands, or even more cases for Content Analytics to crawl through. For this example, 15 cases are sufficient for demonstration purposes.

18.6.2 Default facets of case solutions

This subsection addresses the set of default case properties that are inherited by the Text Mining application. These default properties can be used to build a query to narrow down topics for analysis. Click the **Facets** tab to review the values that are assigned to various facets.

Click the **caseid** facet as shown in Figure 18-39 to list the values of the **Case IDs** assigned to individual cases.

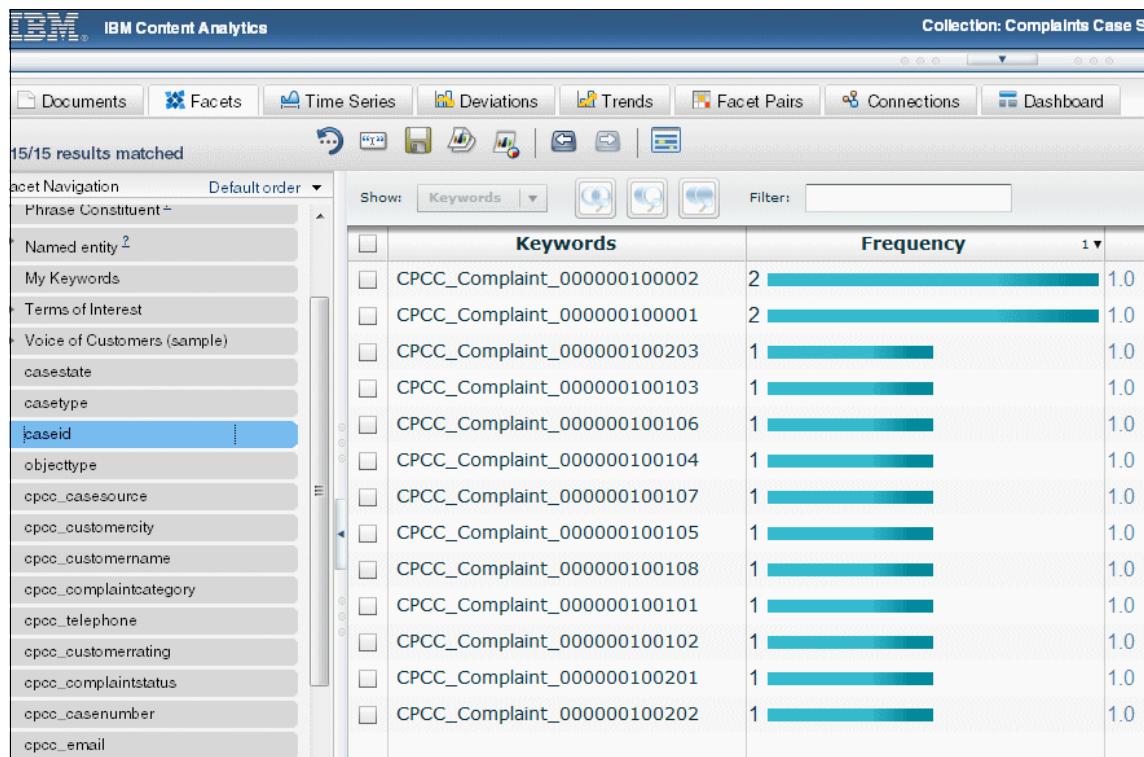


Figure 18-39 List of Case IDs across the cases

Click the **casestate** facet as shown in Figure 18-40 to list the values of the various states that are assigned to the cases.

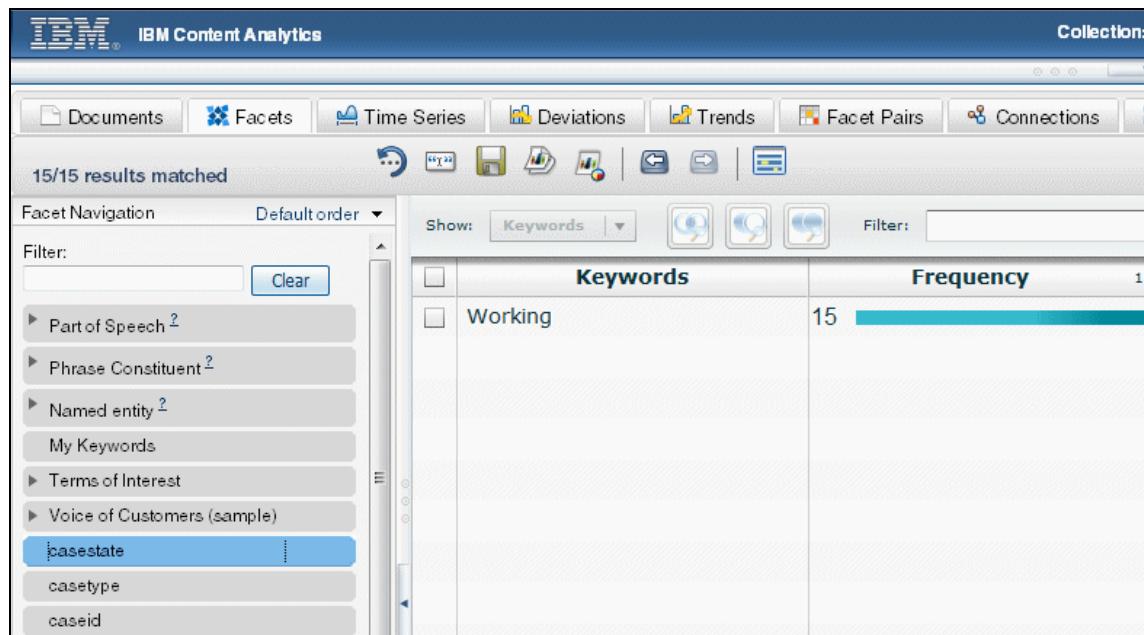


Figure 18-40 Values of the case states

Click the **casetype** facet as shown in Figure 18-41 to list the various types of cases in the solution.

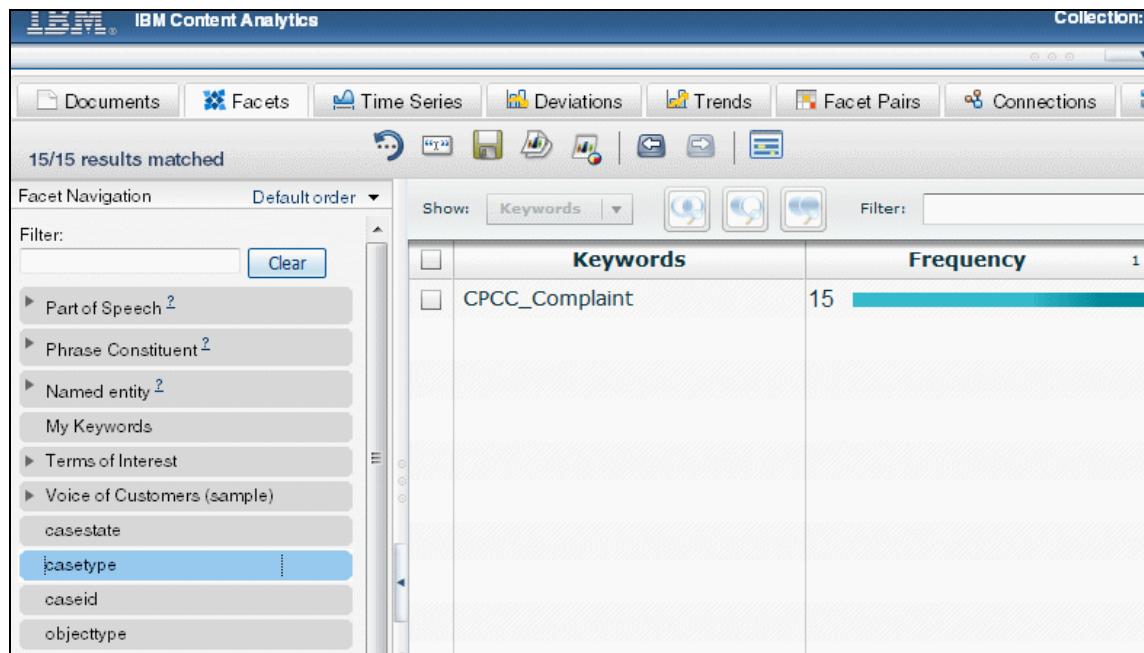


Figure 18-41 Case types for various cases

Click the **objecttype** facet as shown in Figure 18-42 to list the attachments and comments that are associated to the cases.

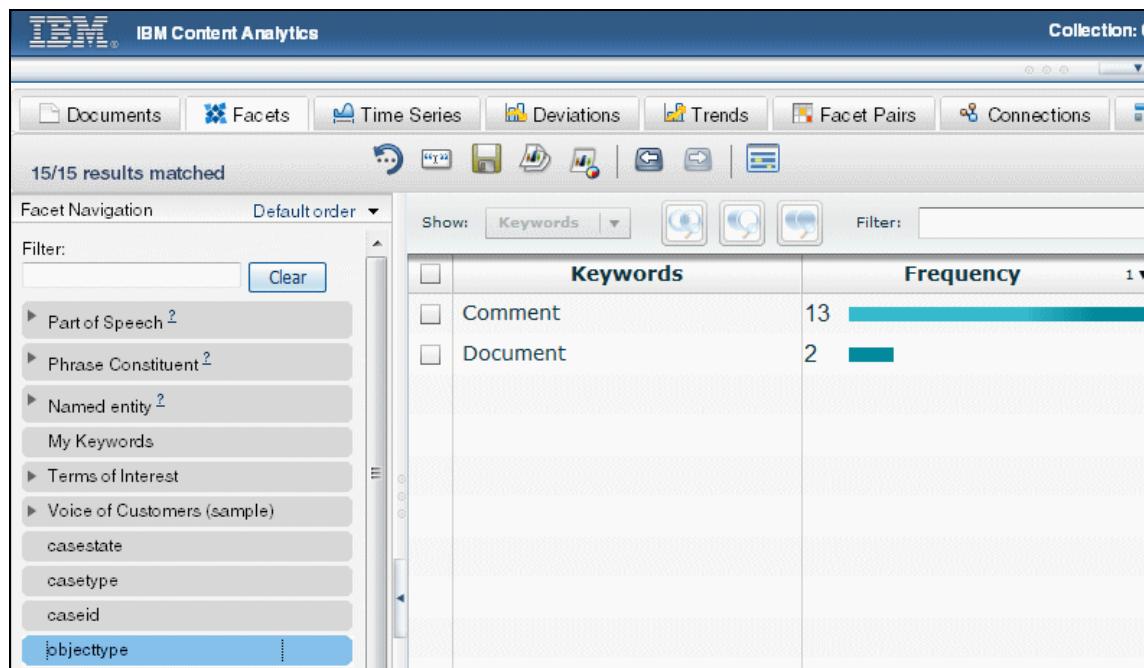


Figure 18-42 Comments and attachments that are associated to cases

Click the **Time Series** tab to view the **Dates** when complaints were registered. Figure 18-43 shows the time series representation of the cases.

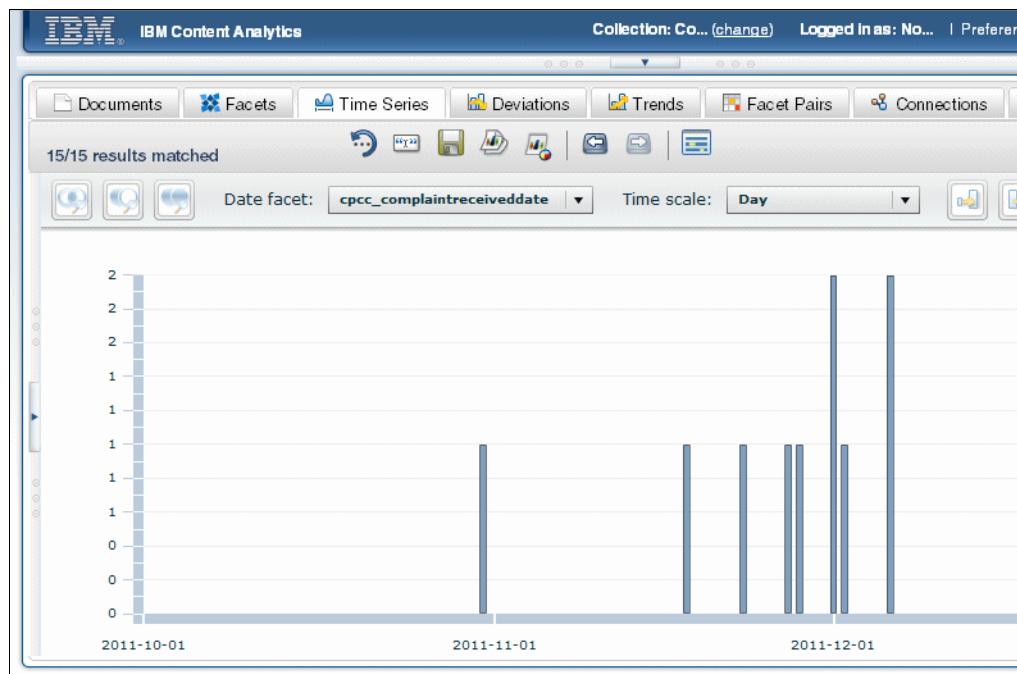


Figure 18-43 Time series representation of the case creation dates

18.6.3 Representation of case properties identified at the crawl time

This section shows a few of the other case properties that are configured for the crawl space and their availability for mining. Click the **Facets** tab to get started.

Click the **cpcc_complaintcategory** facet as shown in Figure 18-44 to list the various categories of the complaints.

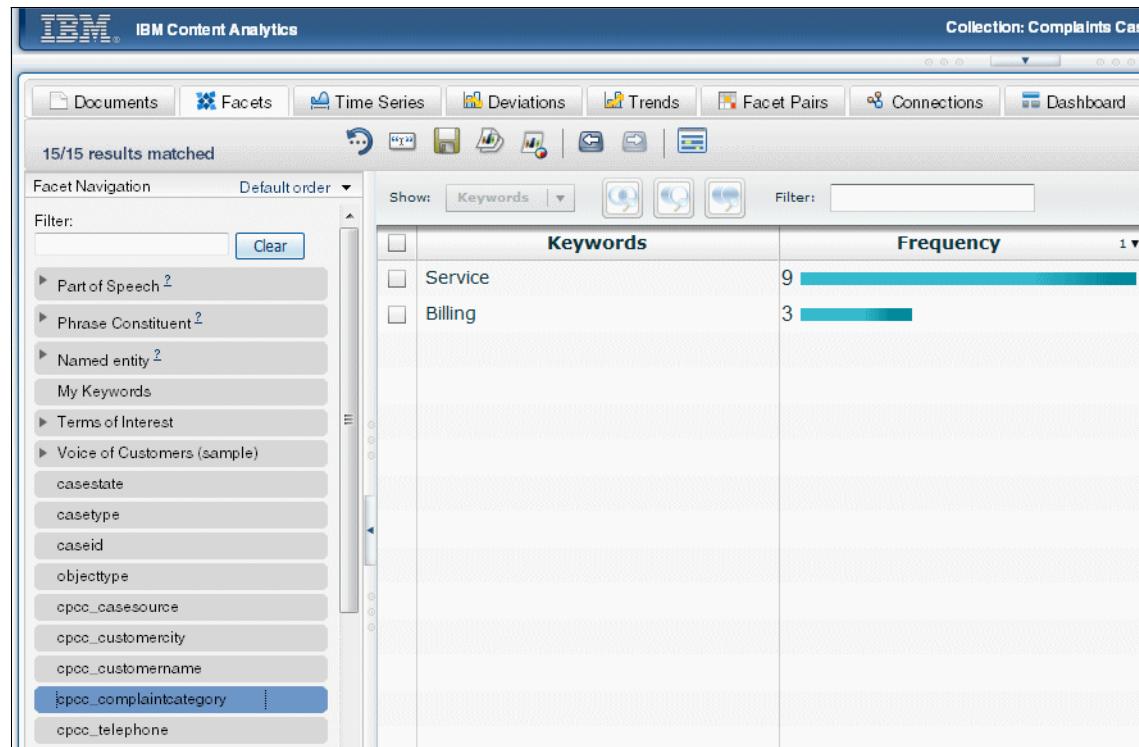


Figure 18-44 Categories of complaints

Expand the **Parts of Speech** facet and then click the **Noun** facet. The list of values represent the nouns extracted by applying Natural Language Processing on case artifacts like complaint description, comments, and attachments. Figure 18-45 shows the list of nouns identified across the 15 cases.

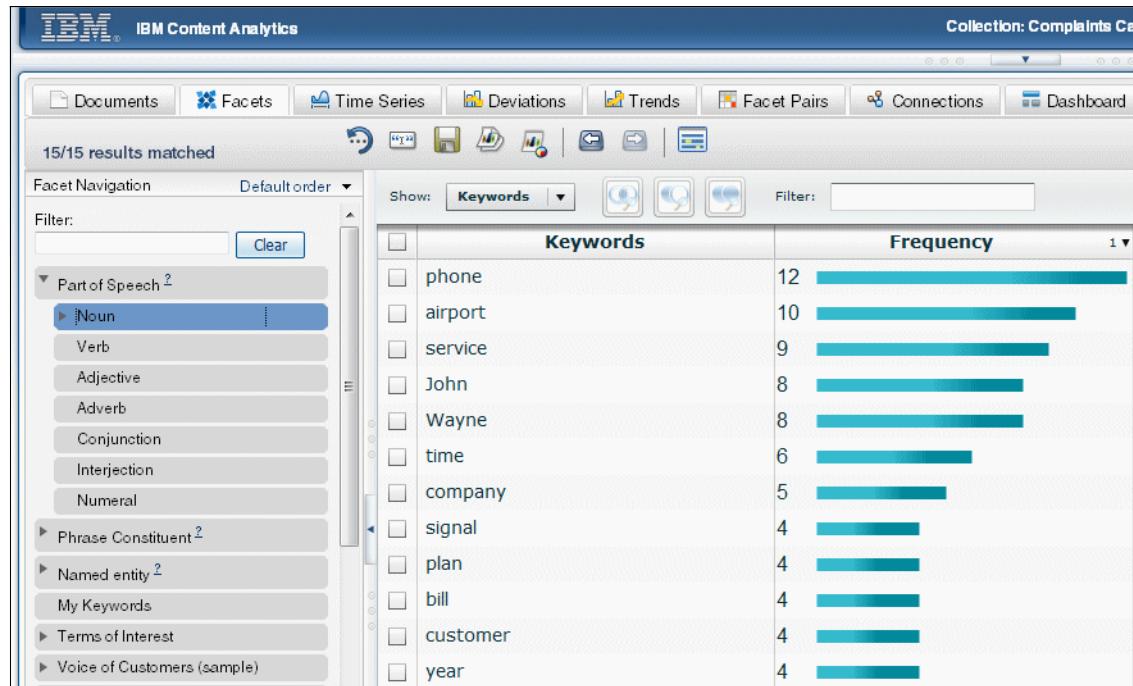


Figure 18-45 Nouns that are identified by natural language processing

In addition to identifying **Part of Speech**, the product can also use various patterns like **Noun sequence**, **Verb followed by Noun**, and **Noun followed by a Predicate**. Figure 18-46 shows the list of Noun sequences identified across the cases.

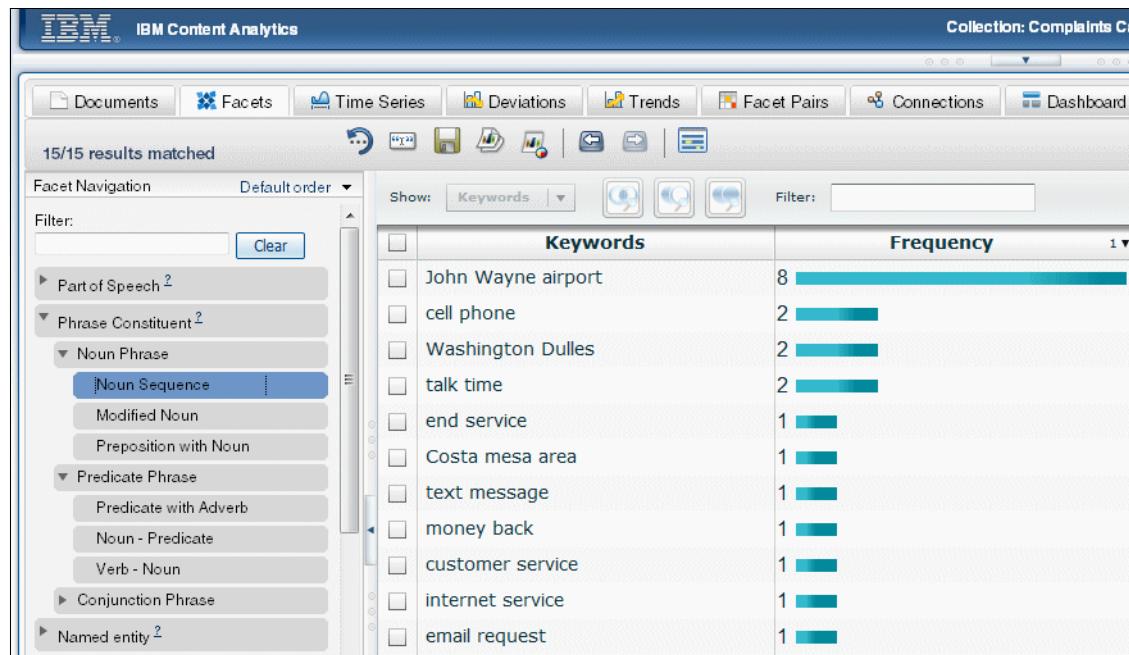


Figure 18-46 Noun sequences extracted across the cases

18.6.4 Insights that are related to customer service

This section addresses using the extracted entities to discover insights.

First, select a category for analysis. As shown in Figure 18-47, click the **cpcc_complaintcategory** facet to identify various categories of the cases.

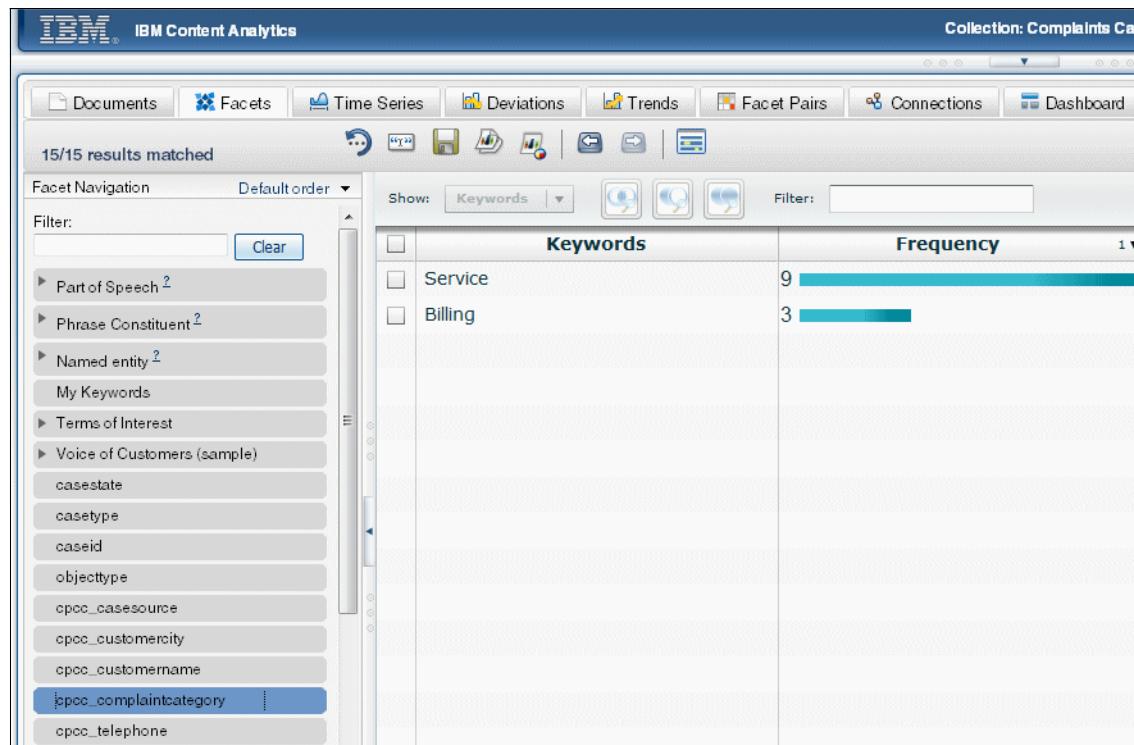


Figure 18-47 Categories of complaints

Select the **Service** keyword and then click **Add to search with boolean AND**. This action narrows down the documents to analyze from 15 to 9. See Figure 18-48 for the results of the **boolean AND** operation.

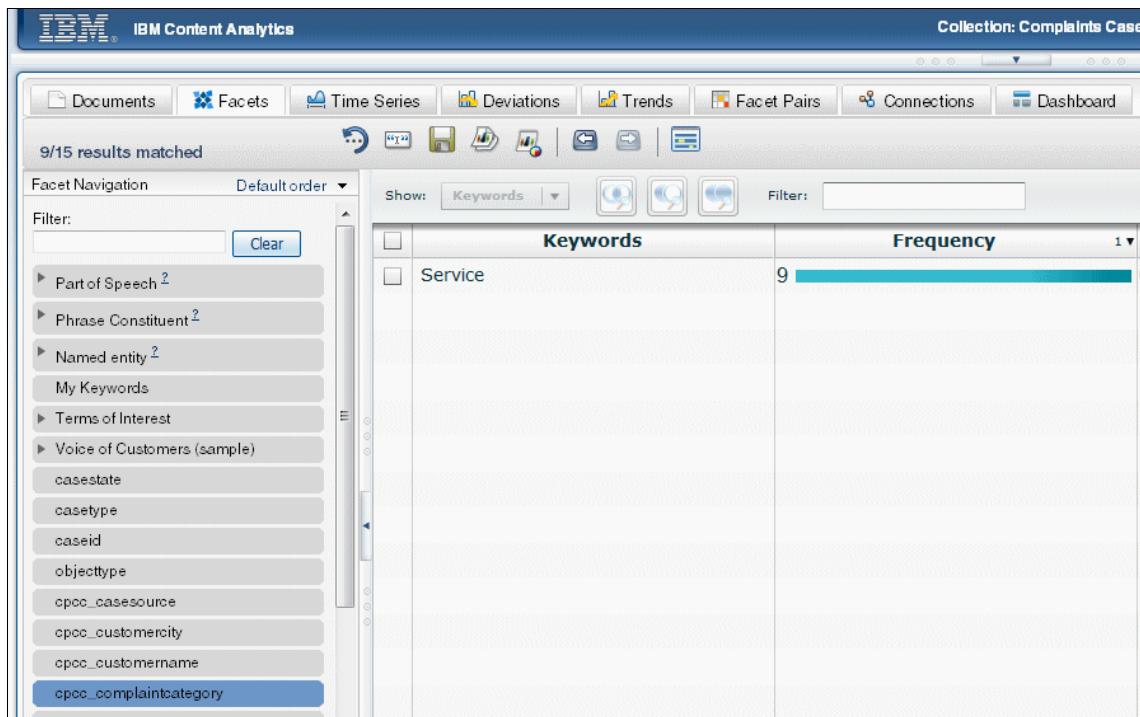


Figure 18-48 Results of the AND operation

Use the entities that are related to natural grammar to further identify the root cause for these service-related complaints.

Expand the **Part of Speech** facet and then click the **Noun** facet. Figure 18-49 shows that the noun called **airport** as the most frequency entity. Select **airport** and add the term to the query with **boolean AND** operation.

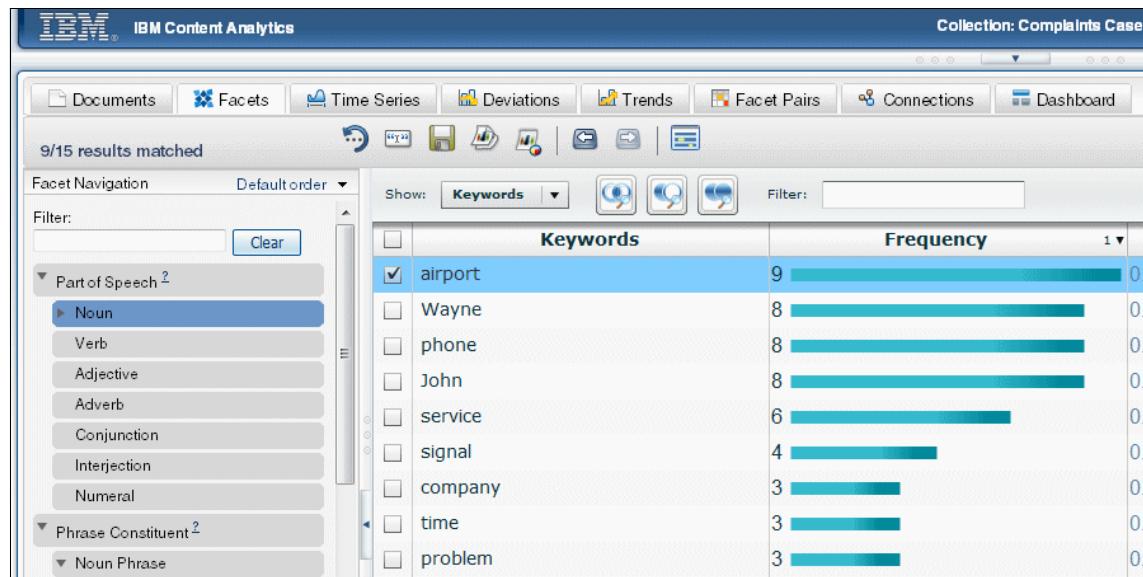


Figure 18-49 The term *airport* occurs with the most frequency

Now that you have identified service-related issues with **airport**, use the grammar-related entities to discover deeper insights. Expand the **Noun Sequence** to identify the most frequent **Modified Nouns**. Figure 18-50 shows **signal ... problem** with the most frequency. Select the most frequency modified noun and add with the **boolean AND** operation.

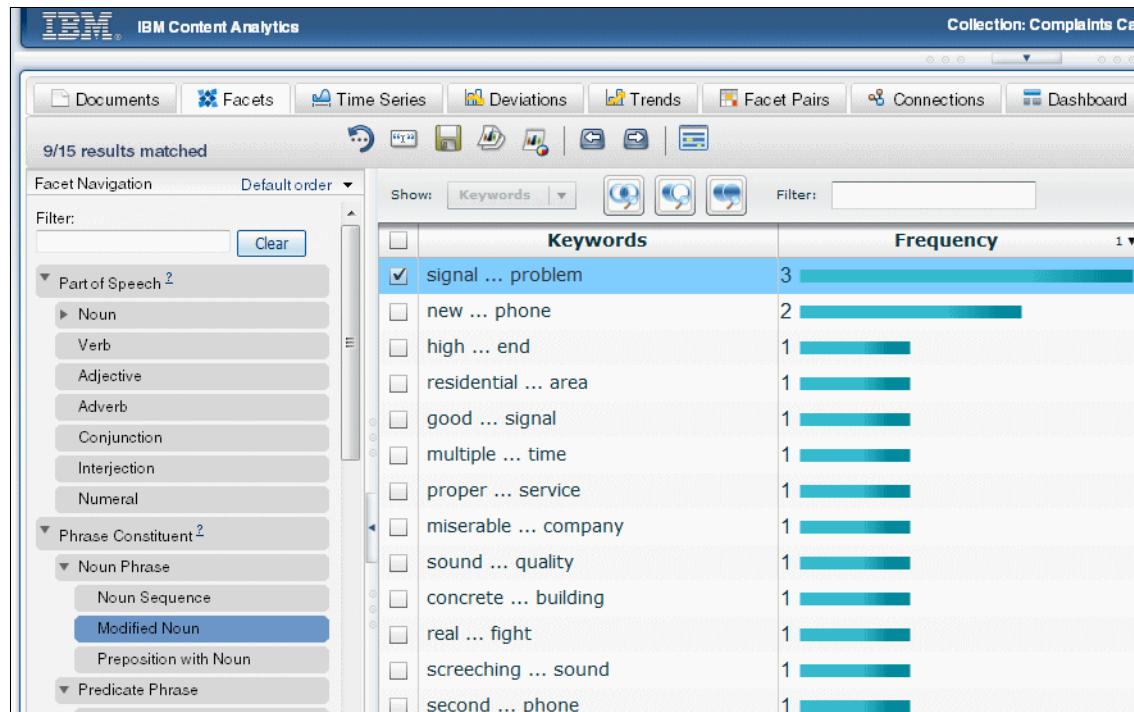


Figure 18-50 Signal problems

The analytics show that there are **signal problems** in association with an **airport**. The analyst can look at the grammatical entities to discover the associated airports. Click the **Noun Sequence** facet to see that **John Wayne airport** occurs with the highest frequency. See Figure 18-51 for the list.

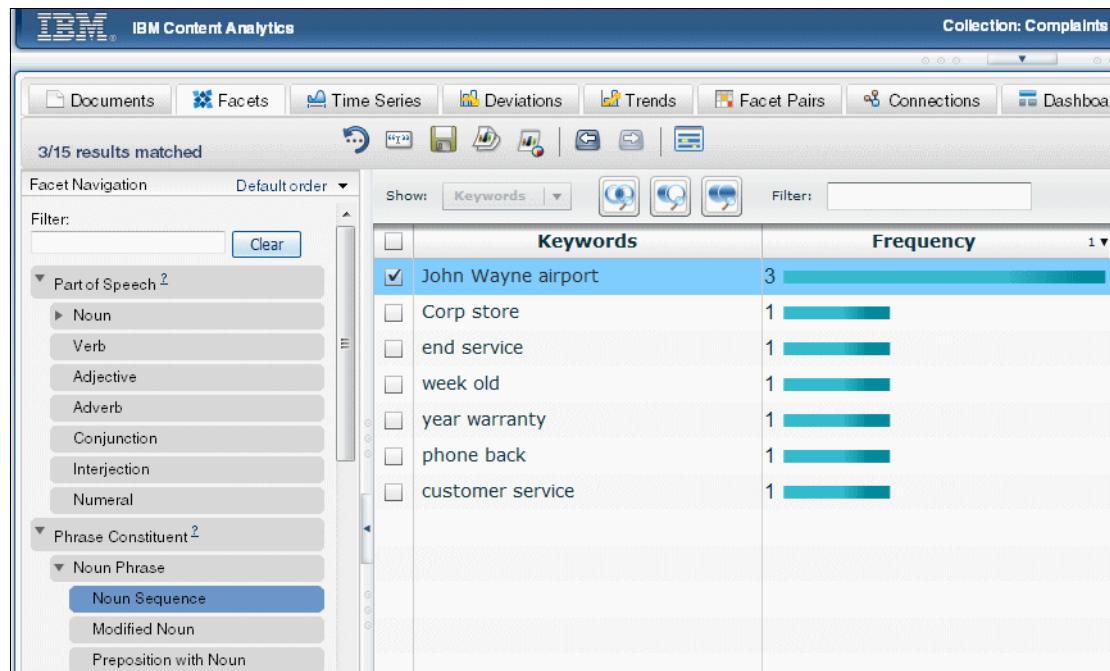


Figure 18-51 List of noun sequences

Select **John Wayne airport** to add to the query. Narrow down the investigation to a small subset of documents to be reviewed manually. Click the **Documents** view to access the documents. See Figure 18-52 for the documents view.

The screenshot shows the IBM Content Analytics interface with the following details:

- Header:** IBM Content Analytics, Collection: Complaints Case Solution (change), Logged in as: Not logged in
- Toolbar:** Documents, Facets, Time Series, Deviations, Trends, Facet Pairs, Connections, Dashboard
- Facet Navigation:** Results 1-3 of 3 (3/15 results matched)
- Filter:** Clear
- Part of Speech:** Noun, Verb, Adjective, Adverb, Conjunction, Interjection, Numeral
- Phrase Constituent:** Noun Sequence (selected), Modified Noun, Preposition with Noun
- Table:** Shows three documents from 'casemanager' source, ordered by Date (12/5/11).

Source	Date	Title
casemanager	12/5/11	000000100101
casemanager	12/5/11	000000100107
casemanager	12/5/11	000000100102

 The table contains the following rows:
 - ... phones we where having **signal problems** and one phone not working at all hardly. Took the phones back ... Needless to say the phone we bought is only two weeks old and we get no signal with them at **John** ... We took it back to the store we bought it from and they told us it was a bad phone and we needed to ...
 - ... I'm having a **signal problem** with my phone. So I went to a corporate office / store in Irvine, and ... They said the phone for me was at **John Wayne Airport**.
 - ... I've been with your company, then another for over 10 years in two locations. I've always ... ve had nothing but **signal problems** at **John Wayne Airport**. We still have problems with the all the ...

Figure 18-52 List of documents narrowed down by mining

Click the **casemanager** link to display the Document Analysis window. Figure 18-53 shows the document analysis dialog for one of the identified documents.

The screenshot shows the 'Document Analysis' window with the following details:

- Analytics Facet:** A table showing various parts of speech and their values. Some values are highlighted in blue.
- Metadata Facet:** A table showing metadata fields and their values.
- Text Summary:** A large text area containing a customer complaint about service issues at John Wayne Airport.

Name	Value
Verb	have
Verb	be
Preposition with Noun	with ... company
General Noun	company
Adverb	then
Numerical	10
General Noun	year
Preposition with Noun	in ... location
Modified Noun	two ... location
Adjective	two
General Noun	location
Verb	have
Adverb	always
Verb	subscribe
Preposition with Noun	to ... end service
Modified Noun	high ... end
Adjective	high
Noun Sequence	end service
General Noun	end
General Noun	service
Verb	have
Adverb	never
Verb	be
Adjective	happy
Preposition with Noun	with ... service
General Noun	service
Adverb	technically
Noun Sequence	customer service
General Noun	customer
General Noun	service

Name	Value
caseid	CPCC_Complaint_000000100102
casestate	Working
casetype	CPCC_Complaint
cpcc_casenumber	1205-0620
cpcc_complaintcategory	Service
cpcc_customername	Mary L. Staples
cpcc_safetycheck	false
cpcc_valid	false
objecttype	Comment

I've been with your company, then another for over 10 years in two locations. I've always subscribed to the higher end services. I've never been happy with the service technically. Customer service is getting much, much worse. We moved to our house nearly seven years ago. Since then, we've had nothing but signal problems at John Wayne Airport. We still have problems with the all the above and now the phone (a very bad joke) is pretty much useless.

Figure 18-53 Document Analysis window

The identified document describes a complaint where a customer reports a **service** issue that is related to **signal problems at John wayne airport**. This insight provides useful information to the mobile service provider. The provider can scan the indexed cases to discover similar complaints and work on a plan to fix this outage. The service provider might rent out a cell phone tower closer to this airport as a step to improve customer satisfaction and retain business.

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.

- ▶ *Disaster Recovery and Backup Solutions for IBM FileNet P8 Version 4.5.1 Systems*, SG24-7744
- ▶ *Federated Content Management: Accessing Content from Disparate Repositories with IBM Content Federation Services and IBM Content Integrator*, SG24-7742
- ▶ *IBM Content Analytics Version 2.2: Discovering Actionable Insight from Your Content*, SG24-7877.
- ▶ *IBM FileNet Content Manager Implementation Best Practices and Recommendations*, SG24-7547
- ▶ *IBM FileNet P8 Platform and Architecture*, SG24-7667
- ▶ *IBM High Availability Solution for IBM FileNet P8 Systems*, SG24-7700
- ▶ *Introducing IBM FileNet Business Process Manager*, SG24-7509
- ▶ *Understanding IBM FileNet Records Manager*, SG24-7623

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

Online resources

These websites are also relevant as further information sources:

- ▶ IBM Case Manager main information page
<http://www.ibm.com/software/advanced-case-management/case-manager>
- ▶ IBM FileNet P8 Platform main information page
<http://www.ibm.com/software/data/content-management/filenet-p8-platform>
- ▶ IBM FileNet P8 Version 5.0 Information Center:
<http://publib.boulder.ibm.com/infocenter/p8docs/v5r0m0/index.jsp>
- ▶ IBM FileNet P8 Platform product documentation
<http://www.ibm.com/support/docview.wss?rs=3247&uid=swg27010422>

This URL includes links to all expansion IBM FileNet P8 products.
- ▶ IBM FileNet Content Manager
<http://www.ibm.com/software/data/content-management/filenet-content-manager>
- ▶ IBM FileNet Business Process Manager
<http://www.ibm.com/software/data/content-management/filenet-business-process-manager>
- ▶ IBM Case Manager Version 5.1 Information Center:
<http://publib.boulder.ibm.com/infocenter/casemgmt/v5r1m0/index.jsp>

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

IBM



Redbooks

Advanced Case Management with IBM Case Manager

(1.0" spine)
0.875" <-> 1.498"
460 <-> 788 pages



Advanced Case Management with IBM Case Manager



Redbooks®

Understanding case management and IBM Case Manager

Developing IBM Case Manager solution with use case example

Integrating with external products and components

Organizations face case management challenges that require insight, responsiveness, and collaboration. IBM Case Manager, Version 5.1.1, is an advanced case management product that unites information, process, and people to provide the 360-degree view of case information and achieve optimized outcomes. With IBM Case Manager, knowledge workers can extract critical case information through integrated business rules, collaboration, and analytics. This easy access to information enhances decision making ability and leads to more successful case outcomes. IBM Case Manager also helps capture industry best practices in frameworks and templates to empower business users and accelerate return on investment.

This IBM Redbooks publication introduces the case management concept. It includes the reason for and benefits of case management, and why it is different from the traditional business process management or content management. In addition, this book addresses how you can design and build a case management solution with IBM Case Manager, and integrate that solution with external products and components.

This book is intended to provide IT architects and IT specialists with the high-level concepts of case management and the capabilities of IBM Case Manager. In addition, it serves as a practical guide for IT professionals who are responsible for designing, building, and deploying IBM Case Manager solutions.

**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**