

Course Guide

Developing workflow solutions using IBM Business Automation Workflow V20.0.0.1

Course code WB835 / ZB835 ERC 1.0



December 2020 edition

Notices

This information was developed for products and services offered in the US.

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

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

*IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
United States of America*

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 jurisdictions 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 provide 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 actual people or business enterprises is entirely coincidental.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

© Copyright International Business Machines Corporation 2020.

This document may not be reproduced in whole or in part without the prior written permission of IBM.

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

Trademarks

The reader should recognize that the following terms, which appear in the content of this training document, are official trademarks of IBM or other companies:

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide.

The following are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide:

Bluemix®
SPSS®

Cognitive Era™
Watson Avatar®

Cognos®
Worklight®

Windows is a trademark of Microsoft Corporation in the United States, other countries, or both.

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

Other product and service names might be trademarks of IBM or other companies.

Contents

Trademarks	iii
Course description	xv
Agenda	xix
Unit 1. Introduction to IBM Business Automation Workflow	1-1
How to check online for course material updates	1-2
Unit objectives	1-3
Key concepts in this unit	1-4
Topics	1-5
1.1. IBM Business Automation Workflow key capabilities	1-6
IBM Business Automation Workflow key capabilities	1-7
IBM Business Automation Workflow key capabilities	1-8
Case Management and Business Process Management comparison	1-9
The process spectrum	1-10
How BAW covers the process spectrum	1-11
1.2. The Workflow Center	1-12
The Workflow Center	1-13
The Workflow Center: The center of process development	1-14
The Workflow Center: Process applications	1-15
The Workflow Center console (1 of 3)	1-16
The Workflow Center console (2 of 3)	1-17
The Workflow Center console (3 of 3)	1-18
The classic Workflow Center console	1-19
Creating a process application	1-20
Creating a process application	1-21
Process application tip, snapshots, and tracks	1-22
1.3. Managing snapshots	1-24
Managing snapshots	1-25
Creating a snapshot	1-26
Exporting a snapshot	1-27
Importing a snapshot	1-28
1.4. IBM Process Designer	1-29
IBM Process Designer	1-30
IBM Process Designer	1-31
Starting the IBM Process Designer (1 of 2)	1-32
Starting the IBM Process Designer (2 of 2)	1-33
IBM Process Designer interface	1-34
IBM Process Designer: Web interface	1-36
IBM Process Designer: Main toolbar	1-37
IBM Process Designer editor (1 of 2)	1-38
IBM Process Designer editor (2 of 2)	1-39
1.5. The Process Portal	1-40
The Process Portal	1-41
The Process Portal overview	1-42
The Process Portal main page	1-44
Viewing processes in the Process Portal (1 of 2)	1-45
Viewing processes in the Process Portal (2 of 2)	1-46
The Process Portal: Social features	1-47

The Process admin console	1-49
Process admin console (1 of 2)	1-50
Process admin console (2 of 2)	1-51
Process inspector	1-52
The WebSphere application server administrative console	1-54
WebSphere application server administrative console	1-55
IBM Business Automation Workflow architecture	1-56
Authoring and runtime environments	1-57
The Workflow Center: The center of the development effort	1-59
The Workflow Server databases	1-60
The Performance Data Warehouse	1-61
The Workflow Center: A unique Workflow Server	1-63
Architecture: Workflow Server tools	1-65
1.6. Overview of IBM Cloud Pak for Automation	1-66
Overview of IBM Cloud Pak for Automation	1-67
IBM has six Cloud Paks	1-68
IBM Cloud Paks	1-69
IBM Cloud Pak for Automation	1-70
IBM Cloud Pak for Automation capabilities	1-71
IBM Business Automation Workflow on Cloud	1-72
IBM Business Automation Workflow on Cloud (1 of 2)	1-73
IBM Business Automation Workflow on Cloud (2 of 2)	1-74
Unit summary	1-75
Review questions	1-76
Review answers	1-77
Exercise: Creating your first process application	1-78
Exercise introduction	1-79
Unit 2. Introduction to Business Process Management	2-1
Unit objectives	2-2
Topics	2-3
2.1. Business Process Management (BPM)	2-4
Business Process Management (BPM)	2-5
Business Process Management is about value	2-6
What is Business Process Management?	2-7
Three themes	2-8
BPM lifecycle	2-9
The modeling phase	2-11
2.2. Introduction to process modeling	2-12
Introduction to process modeling	2-13
What is a business process?	2-14
What is process modeling?	2-15
Three-phase approach	2-16
What is a process model?	2-17
The right process model development strategy	2-19
2.3. Business Process Management project development	2-20
Business Process Management project development	2-21
Business Process Management project components	2-22
Project development phases	2-23
Business Process Management project teams	2-24
2.4. The Playback methodology	2-25
The Playback methodology	2-26
What is a Playback?	2-27
Playback methodology	2-28
Playback cycles	2-30
Playback 0	2-32

Playback 1	2-33
Playback 2	2-34
Playback 3	2-35
2.5. Playback 0	2-36
Playback 0	2-37
Playback 0	2-38
Playback 0: Process discovery (1 of 2)	2-40
Playback 0: Process discovery (2 of 2)	2-41
Playback 0: Process analysis	2-43
Playback 0: Iteration	2-45
Business data	2-47
Unit summary	2-48
Review questions	2-49
Review answers	2-50
Unit 3. Playback 0: Modeling the As-Is and To-Be business processes.....	3-1
Unit objectives	3-2
Key concepts in this unit	3-3
Topics	3-4
3.1. Modeling a process.	3-5
Modeling a process	3-6
Creating the As-Is process model	3-7
Automation	3-9
Where to start	3-10
Process modeling guidelines in IBM Process Designer	3-12
Guidelines for modeling activities	3-13
Connecting flow objects	3-14
Process design considerations	3-15
3.2. Nested processes.	3-16
Nested processes	3-17
Activity: Nested process	3-18
Activity: Nested process types (1 of 3)	3-19
Activity: Nested process types (2 of 3)	3-20
Activity: Nested process types (3 of 3)	3-22
Introduction to Business Process Model and Notation	3-23
About BPMN	3-24
BPMN elements	3-26
BPMN example (1 of 2)	3-27
BPMN example (2 of 2)	3-28
3.3. Pool and lanes	3-29
Pool and lanes	3-30
Pool	3-31
Lanes	3-32
System lane	3-33
3.4. Flow objects	3-34
Flow objects	3-35
Events: Start events	3-36
Events: Multiple start events	3-37
Events: End events	3-39
Events: Multiple end events	3-40
Activity: Task types (1 of 2)	3-41
Activity: Task types (2 of 2)	3-42
Gateway: types	3-44
3.5. Performing process discovery with Blueworks Live	3-45
Performing process discovery with Blueworks Live	3-46
Full model to analysis lifecycle	3-47

Blueworks Live and SaaS	3-49
Using IBM Blueworks Live and IBM BAW (1 of 2)	3-50
Using IBM Blueworks Live and IBM BAW (2 of 2)	3-51
3.6. The Hiring Requisition Process	3-53
The Hiring Requisition Process	3-54
The Hiring Requisition Process	3-55
Core requirements (1 of 3)	3-56
Core requirements (2 of 3)	3-58
Core requirements (3 of 3)	3-59
Translating business process work steps into activities	3-60
Unit summary	3-62
Review questions	3-63
Review answers	3-64
Exercise: Playback 0: Creating the To-Be process	3-65
Exercise introduction	3-66
Unit 4. Playback 0: Controlling process flow.	4-1
Unit objectives	4-2
Topics	4-3
Key concepts in this unit	4-4
4.1. Process sequence flow.	4-5
Process sequence flow	4-6
Normal sequence flow	4-7
Conditional sequence flow	4-8
Default sequence flow	4-9
4.2. Introduction to tokens	4-10
Introduction to tokens	4-11
What is a token?	4-12
Traverses the flow objects	4-13
Diverted through alternative paths	4-14
Split into parallel paths	4-15
4.3. Directing sequence flow by using gateways	4-16
Directing sequence flow by using gateways	4-17
Represented as a diamond	4-18
Can be thought of as a question	4-19
Has a defined set of alternative answers	4-20
Two distinct modes: Split	4-21
Two distinct modes: Join	4-22
Exclusive: Diamond shape with no internal marker	4-23
Exclusive: One or more outgoing conditional sequence flows	4-24
Exclusive: Default sequence flow	4-25
Process narrative	4-26
Submit auto claim example (1 of 6)	4-27
Submit auto claim example (2 of 6)	4-28
Submit auto claim example (3 of 6)	4-29
Submit auto claim example (4 of 6)	4-30
Submit auto claim example (5 of 6)	4-31
Submit auto claim example (6 of 6)	4-32
Parallel gateway: Diamond with an internal plus	4-33
Parallel split: No conditional or default flows	4-34
Process narrative	4-35
New hire onboarding example (1 of 4)	4-36
New hire onboarding example (2 of 4)	4-37
New hire onboarding example (3 of 4)	4-38
New hire onboarding example (4 of 4)	4-39
Parallel join	4-40

Evaluating conditions: Decision logic in the outgoing sequence flow	4-41
Evaluating conditions: Externalized decision logic	4-42
4.4. Interrupting the process by using intermediate events.....	4-43
Interrupting the process by using intermediate events	4-44
Intermediate events	4-45
Intermediate events: Timer	4-46
[No title]	4-47
[No title]	4-48
Intermediate events: Content	4-49
[No title]	4-50
Attached and sequence flow intermediate events	4-51
Sequence flow intermediate event	4-52
Process stops until an intermediate event occurs	4-53
Process continues on sequence flow when an intermediate event completes	4-54
A common requirement (1 of 5)	4-55
A common requirement (2 of 5)	4-56
A common requirement (3 of 5)	4-57
A common requirement (4 of 5)	4-58
A common requirement (5 of 5)	4-59
Attached intermediate event	4-60
Process flow reaches an activity	4-61
Attached intermediate event generates a separate token	4-62
Attached intermediate event can create a parallel flow	4-63
Attached intermediate event can close an activity	4-64
Attached intermediate event example (1 of 4)	4-65
Attached intermediate event example (2 of 4)	4-66
Attached intermediate event example (3 of 4)	4-67
Attached intermediate event example (4 of 4)	4-68
Attached intermediate event is active only when the activity it is attached to is active	4-69
A way to model an escalation	4-70
Timer intermediate event	4-71
Unit summary	4-72
Review questions (1 of 2)	4-73
Review answers	4-74
Exercise: Playback 0: Controlling process flow	4-75
Exercise introduction	4-76

Unit 5. Playback 1: Controlling process flow with business data	5-1
Unit objectives	5-2
Topics	5-3
Key concepts in this unit	5-4
Playback 1: Controlling process flow	5-5
5.1. Managing variables and data flow	5-6
Managing variables and data flow	5-7
Two types of process data	5-8
What is flow data?	5-9
What is business data?	5-11
Business objects, variables, and data mapping	5-12
Business objects	5-13
Example of business objects and variables	5-14
Create the business objects in the lab exercise	5-15
Example of data mapping	5-16
Declaring three kinds of variables	5-17
Standardizing variable names	5-18
Understanding namespaces	5-19
5.2. Implementing the intermediate event: Timer	5-20

Implementing the intermediate event: Timer	5-21
Timer intermediate event	5-22
Attached timer intermediate event	5-23
Timer intermediate event implementation (1 of 2)	5-24
Timer intermediate event implementation (2 of 2)	5-25
5.3. Implementing gateways	5-26
Implementing gateways	5-27
Implementing a gateway	5-28
5.4. Routing tasks	5-29
Routing tasks	5-30
Creating a team	5-31
Team and team members	5-32
Routing activities	5-33
Team filter service	5-35
Team retrieval and filter service templates	5-36
Identifying expert users for an activity	5-37
Unit summary	5-38
Review questions	5-39
Review answers	5-40
Exercise: Playback 1: Controlling process flow with business data	5-41
Exercise introduction	5-42
Unit 6. Playback 1: Business data, services, and coaches	6-1
Unit objectives	6-2
Playback 1: Business data and services	6-3
Topics	6-4
Key concepts in this unit	6-5
6.1. Creating a data model	6-6
Creating a data model	6-7
Defining the business object data model	6-8
Building a complex business object	6-9
Building the data model	6-10
Standardizing the variable naming convention	6-11
Refactoring	6-12
Complex objects and lists	6-14
Initializing complex objects and lists	6-15
Nested processes and variables	6-17
Sharing a complex business object	6-18
6.2. Building services	6-19
Building services	6-20
What are services?	6-21
Creating services	6-22
6.3. Building coaches	6-23
Building coaches	6-24
Coach overview	6-25
Build user interface for human services by using coaches	6-26
Laying out a coach by using the grid layout	6-27
The coach designer interface	6-28
Panels	6-29
Controls	6-30
View object settings	6-31
Adding controls from server-side variables	6-32
Buttons and sequence flow	6-33
Setting screen size	6-34
6.4. Implementing services in a process	6-35
Implementing services in a process	6-36

Implementing services	6-37
Business objects, variables, and data mapping	6-38
Data mapping	6-39
Mapping input and output data for an activity or step	6-40
Understanding object methods	6-41
Unit summary	6-42
Review questions	6-43
Review answers	6-44
Exercise: Playback 1: Business data, services, and coaches	6-45
Exercise introduction	6-46
Unit 7. Playback 1: Enhancing coaches.....	7-1
Unit objectives	7-2
Playback 1: Enhancing coaches	7-3
Key concepts in this unit	7-4
Topics	7-5
7.1. Enhancing coaches	7-6
Enhancing coaches	7-7
Basics of coach enhancement	7-8
Implementing a select control (1 of 2)	7-9
Implementing a select control (2 of 2)	7-10
Configuring the select list	7-11
7.2. Views	7-12
Views	7-13
Views	7-14
Create a View	7-16
Create a View	7-17
View definition page	7-18
Defining View behavior	7-20
UI Toolkit	7-21
7.3. Theme support	7-22
Theme support	7-23
Theme support	7-24
Create custom theme	7-25
Create custom theme	7-26
WYSIWYG Theme Editor: Create and edit theme (1 of 3)	7-27
WYSIWYG Theme Editor: Create and edit theme (2 of 3)	7-28
WYSIWYG Theme Editor: Create and edit theme (3 of 3)	7-29
Applying custom theme to process apps	7-30
7.4. Conducting the Playback session.....	7-31
Conducting the Playback session	7-32
Playback 1: Business data and services	7-33
Expose a process to a team	7-34
Creating a snapshot	7-35
Guidelines for using snapshots	7-36
Toolkits	7-37
Unit summary	7-38
Review questions	7-39
Review answers	7-40
Exercise: Playback 1: User interface design and implementation	7-41
Exercise introduction	7-42
Exercise: Playback 1: Conducting the Playback session	7-43
Exercise introduction	7-44
Unit 8. Playback 2: Integrations	8-1
Unit objectives	8-2

Playback 2: Integrations	8-3
Key concepts in this unit	8-4
Topics	8-5
8.1. Creating a decision service.....	8-6
Creating a decision service	8-7
Building a decision service (1 of 2)	8-8
Building a decision service (2 of 2)	8-10
Adding a BAL rule	8-11
Building a BAL rule	8-12
8.2. Implementing message events.....	8-13
Implementing message events	8-14
Message event introduction	8-15
Message start events	8-16
Implementing message start events	8-17
Mapping message start event variables	8-19
8.3. Accessing and manipulating external data.....	8-20
Accessing and manipulating external data	8-21
Integrating with other systems	8-22
Integration service flows	8-23
Using service components	8-24
Prebuilt Service Flows	8-25
Advanced Integration service	8-26
8.4. Exposed process variables (EPVs) and environment variables (ENVs).....	8-27
Exposed process variables (EPVs) and environment variables (ENVs)	8-28
Environment variables (ENVs)	8-29
Exposed process variables (EPVs)	8-30
EPV versus environment variable comparison (1 of 2)	8-31
EPV versus environment variable comparison (2 of 2)	8-32
8.5. Applying asset tagging	8-33
Applying asset tagging	8-34
Creating favorites	8-35
Tagging library items	8-36
Playback 2: Integrations	8-37
Unit summary	8-39
Review questions	8-40
Review answers	8-41
Exercise: Playback 2: Integrations	8-42
Exercise introduction	8-43
Unit 9. Playback 3: Error handling and deployment	9-1
Unit objectives	9-2
Playback 3: Hardening processes and services	9-3
Topics	9-4
Key concepts in this unit	9-5
9.1. Handling errors in a process.....	9-6
Handling errors in a process	9-7
Using error events to handle errors	9-8
Handling errors in a process	9-10
Catching errors	9-12
9.2. Handling errors in services.....	9-13
Handling errors in services	9-14
Catching errors in services	9-15
Client-side human services: Throwing errors by using error end events (1 of 2)	9-17
Client-side human services: Throwing errors by using error end events (2 of 2)	9-18
Playback 3: Hardening processes and services	9-20
Unit summary	9-21

Review questions	9-22
Review answers	9-23
Exercise: Playback 3: Handling errors and deploying your process application	9-24
Exercise introduction	9-25
Unit 10. Advanced routing.....	10-1
Unit objectives	10-2
Topics	10-3
Key concepts in this unit	10-4
10.1. Team data and lanes	10-5
Team data and lanes	10-6
Team data and lanes	10-7
Organizational data	10-9
Types of routing	10-10
10.2. Custom routing	10-11
Custom routing	10-12
Custom routing (1 of 3)	10-13
Custom routing (2 of 3)	10-14
Custom routing (3 of 3)	10-15
10.3. Team filters	10-16
Team filters	10-17
Creating team filters	10-18
Applying a team filter service	10-19
10.4. Experts team.....	10-20
Experts team	10-21
Designating an experts team	10-22
10.5. Four eyes	10-23
Four eyes	10-24
Four eyes	10-25
Unit summary	10-26
Review questions	10-27
Review answers (1 of 2)	10-28
Review answers (2 of 2)	10-29
Exercise: Implementing a custom routing solution	10-30
Exercise introduction	10-31
Exercise: Implementing the “four eyes” policy by using a team filter	10-32
Exercise introduction	10-33
Unit 11. Managing complex tasks and process interactions.....	11-1
Unit objectives	11-2
Topics	11-3
11.1. Parallel tasks and messaging.....	11-4
Parallel tasks and messaging	11-5
Parallel tasks	11-6
Data flow	11-7
Sharing data among tasks	11-8
How variables are passed (1 of 2)	11-9
How variables are passed (2 of 2)	11-10
Passing complex objects	11-11
Shared objects	11-12
Variable number of tokens	11-14
Canceling parallel tasks	11-15
Messaging patterns	11-16
Message events	11-17
Parallel tasks and messaging	11-18
Nested process cancellation pattern	11-19

11.2. Multi-instance loops	11-20
Multi-instance loops	11-21
Multi-instance loops (MILs)	11-22
Simple loop configuration	11-24
Multi-instance loop configuration options	11-25
Setting up a multi-instance loop (1 of 2)	11-26
Setting up a multi-instance loop (2 of 2)	11-27
Multi-instance loop end conditions	11-28
Multi-instance loops and performance	11-29
Unit summary	11-30
Review questions	11-31
Review answers	11-32
Exercise: Building a cancellation pattern	11-33
Exercise introduction	11-34
Unit 12. Integrating with external systems.	12-1
Unit objectives	12-2
Key concepts in this unit	12-3
Topics	12-4
12.1. Building outbound integrations	12-5
Building outbound integrations	12-6
Building an outbound integration	12-7
Integration types	12-8
Java integration	12-9
Web service integration	12-10
Connecting to a web service	12-12
12.2. System toolkit services	12-13
System toolkit services	12-14
System toolkit services	12-15
Guidelines (1 of 2)	12-16
Guidelines (2 of 2)	12-17
12.3. Building inbound integrations	12-18
Building inbound integrations	12-19
Building inbound integrations	12-20
Event-based undercover agents (UCA)	12-21
Unit summary	12-22
Review questions	12-23
Review answers	12-24
Exercise: Building web service connections	12-25
Exercise introduction	12-26
Unit 13. Handling content events in a process	13-1
Unit objectives	13-2
Topics	13-3
Key concepts in this unit	13-4
13.1. CMIS system integration.	13-5
CMIS system integration	13-6
Goals of integration with enterprise content management systems (ECM)	13-7
Internal document store	13-8
Defining ECM server properties	13-9
13.2. Document controls	13-11
Document controls	13-12
Document List view (1 of 2)	13-13
Document List view (2 of 2)	13-14
Running a coach from the Process Portal	13-15
Running a coach as a startable service	13-16

13.3. Content events	13-17
Content events	13-18
Modeling content events on a process	13-19
Content Event Subscription for BPM Documents	13-20
Creating the attached service	13-21
Case integration	13-22
Case integration (1 of 2)	13-23
Case integration (2 of 2)	13-24
Running a Case solution (1 of 2)	13-25
Running a Case solution (2 of 2)	13-26
Unit summary	13-27
Review questions	13-28
Review answers (1 of 2)	13-29
Review answers (2 of 2)	13-30
Exercise: Handling content events in a process	13-31
Exercise introduction	13-32
Unit 14. Course summary, badge, and other resources	14-1
Unit objectives	14-2
Course objectives (1 of 7)	14-3
Course objectives (2 of 7)	14-4
Course objectives (3 of 7)	14-5
Course objectives (4 of 7)	14-6
Course objectives (5 of 7)	14-7
Course objectives (6 of 7)	14-8
Course objectives (7 of 7)	14-9
IBM credentials: Badges and certifications	14-10
Learn more about this product	14-11
Additional resources (1 of 3)	14-12
Additional resources (2 of 3)	14-13
Additional resources (3 of 3)	14-14
Unit summary	14-15
Course completion	14-16

Course description

Developing workflow solutions using IBM Business Automation Workflow V20.0.0.1

Duration: 5 days

Purpose

This course integrates training in business process management (BPM) methods and implementation with IBM Business Automation Workflow V20. You learn core process modeling and implementation skills, the project development approach, process model implementation fundamentals, and delivery patterns. These skills improve the speed and quality of implementing an executable process.

IBM Business Automation Workflow is a comprehensive environment that provides the visibility and insight that is required to effectively manage the business processes of an organization. The course begins with an overview of the product and then covers business process management, emphasizing the concepts of reuse, ease of maintenance, and high-quality development strategies. You create a structured process using IBM Process Designer from business requirements that are identified during process analysis. You learn how to make team collaboration more efficient by using standard Business Process Model and Notation (BPMN) elements.

The course provides an overview of the architecture of IBM Business Automation Workflow and describes the use of process applications and toolkits within the tool. You create business objects and variables, implement gateways, and demonstrate process flow on your diagrams. You build customized user interfaces (coaches) to enable business and process data flows throughout the process model.

The course uses an interactive learning environment, with class review questions to reinforce concepts and check understanding. Lab exercises throughout the course provide hands-on experience with IBM tools. This course is intended to be collaborative, and you can work in teams to complete class activities.

Audience

This course is designed for project members who design and implement detailed logic, data models, and external system integrations for an executable business process. These roles include process owners, BPM analysts, BPM authors, BPM developers, BPM administrators, and BPM project managers.

Prerequisites

Before taking this course, you should have:

- Practical knowledge of data structures
- Understanding of SQL syntax and JavaScript

- Basic understanding of web services
- Experience with modern programming techniques

Objectives

- Understand the key capabilities of Business Automation Workflow
- Describe how to use IBM Business Automation Workflow to accomplish process modeling goals
- Describe the purpose of the Process Portal, Process Designer, and Workflow Center repository
- Describe the high-level architecture for Business Automation Workflow
- List and describe the core notation elements that are used in the IBM Process Designer
- Describe the purpose and function of Blueworks Live
- Explain the benefits of using IBM Business Automation Workflow on Cloud
- Define Business Process Management (BPM)
- Understand the process spectrum
- List and describe the phases in the IBM Playback methodology
- Describe Playback 0 and the achievements that are reached during this stage
- Examine a defined workflow from detailed process requirements and identify the interrelated process activities
- Describe how to model a structured process
- Decompose activities into a nested process
- Describe process sequence flow and the runtime use of process tokens
- Explain how to evaluate and model conditions for a gateway
- List and describe intermediate event types that are used in the IBM Process Designer
- Model a business process escalation path with an attached timer intermediate event
- Describe the differences between process flow data and business flow data
- Add variables and business objects to a process
- Describe teams and process lanes
- Explain user distribution in a process application
- Implement routing for tasks
- Create a routing design by using a team filter service
- Assign an expert group to an activity
- Expose a process application to a team
- Use coaches to define and implement guided user interactions
- Implement a service for an activity in a process
- Enhance coaches by applying a theme and adding tabs

- Create a reusable view
- Create a snapshot
- Share your assets by using a toolkit, and exporting your process application
- Organize assets with favorites, tagging, and smart folders
- Conduct a Playback session
- Explain how to create a decision service
- Describe how to create and configure an undercover agent (UCA)
- Describe how to start a process with a message start event
- Define the basic function of services
- Configure and define services for outbound integration
- Create an inbound web service
- Describe the differences between an environment variable and an exposed process variable
- Catch an error in a process and service
- Explain when to use team filter services to support business policy
- Use parallel tasks and messaging in a BPMN model
- Use Multi-instance loops to efficiently route work
- Explain how to integrate with external systems in IBM Business Automation Workflow
- Explain how to handle content events in a process
- Understand how the Target Object Store (TOS) can be used to share documents between a BPM solution and a Case solution
- Understand how to build a simple Case solution that integrates a process from a BPM solution

Contents

- Introduction to IBM Business Automation Workflow
- Introduction to Business Process Management
- Playback 0: Modeling the As-Is and To-Be processes
- Playback 0: Controlling process flow
- Playback 1: Controlling process flow with business data
- Playback 1: Business data, services and coaches
- Playback 1: Enhancing coaches
- Playback 2: Integrations
- Playback 3: Error handling and deployment
- Advanced routing
- Managing complex tasks and processes

- Integrating with external systems
- Handling content events in a process

Curriculum relationship

This course is an update of course WB828 Developing workflow solutions with Business Automation Workflow v19.0.0.1 - ERC 1.1

Agenda



Note

The following unit and exercise durations are estimates, and might not reflect every class experience.

Day 1

- (00:15) Course introduction
- (01:30) Unit 1. Introduction to IBM Business Automation Workflow
- (01:00) Exercise 1. Creating your first process application
- (01:30) Unit 2. Introduction to Business Process Management
- (01:30) Unit 3. Playback 0: Modeling the As-Is and To-Be business processes
- (02:00) Exercise 2. Playback 0: Creating the To-Be process

Day 2

- (01:30) Unit 4. Playback 0: Controlling process flow
- (01:30) Exercise 3. Playback 0: Controlling process flow
- (01:30) Unit 5. Playback 1: Controlling process flow with business data
- (01:30) Exercise 4. Playback 1: Controlling process flow with business data

Day 3

- (02:00) Unit 6. Playback 1: Business data, services, and coaches
- (01:30) Exercise 5. Playback 1: Business data, services, and coaches
- (01:30) Unit 7. Playback 1: Enhancing coaches
- (01:30) Exercise 6. Playback 1: User interface design and implementation
- (01:00) Exercise 7. Playback 1: Conducting the Playback session

Day 4

- (01:30) Unit 8. Playback 2: Integrations
- (01:30) Exercise 8. Playback 2: Integrations
- (01:00) Unit 9. Playback 3: Error handling and deployment
- (01:00) Exercise 9. Playback 3: Handling errors and deploying your process application
- (01:00) Unit 10. Advanced routing
- (01:00) Exercise 10. Implementing a custom routing solution
- (01:30) Exercise 11. Implementing the “four eyes” policy by using a team filter

Day 5

- (01:30) Unit 11. Managing complex tasks and process interactions
- (01:00) Exercise 12. Building a cancellation pattern
- (01:00) Unit 12. Integrating with external systems
- (01:00) Exercise 13. Building web service connections
- (01:00) Unit 13. Handling content events in a process
- (01:30) Exercise 14. Handling content events in a process
- (00:30) Unit 14. Course summary and other resources

Unit 1. Introduction to IBM Business Automation Workflow

Estimated time

01:30

Overview

This unit is an overview of IBM Business Automation Workflow, its key capabilities, high-level architecture, and its primary components. It focuses on how to create a process application in the Workflow Center, provides a look at the Designer and Inspector views of the IBM Process Designer, and introduces the Process Portal and Workflow repository.

How you will check your progress

- Review
- Exercise

How to check online for course material updates



Note: If your classroom does not have internet access, ask your instructor for more information.

Instructions

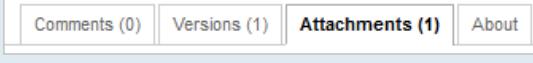
1. Enter this URL in your browser:
<http://ibm.biz/CloudEduCourses>
2. On the wiki page, locate and click the **Course Information** category.
3. Find your course in the list and then click the link.
4. The wiki page displays information for the course. If the course has an errata document, this page is where it is found.
5. If you want to download an attachment, such as an errata document, click the **Attachments** tab at the bottom of the page.

6. To save the file to your computer, click the document link and follow the dialog box prompts.

Figure 1-1. How to check online for course material updates

Unit objectives

- Understand the key capabilities of IBM Business Automation Workflow
- Describe how to use IBM Business Automation Workflow to accomplish process modeling goals
- Explain how to create process applications in the Workflow Center
- Explain how to design process models by using the IBM Process Designer
- Describe the purpose of the Process Portal, Process Designer, and the Workflow Center repository
- Describe the high-level architecture for IBM Business Automation Workflow
- Explain the benefits of using IBM Business Automation Workflow on Cloud

Key concepts in this unit

- **Process app / application:** A process application is a container for process models and their supporting implementations, and it is stored in the repository
- **Toolkit:** A toolkit is a collection of assets that are shared between process apps or other toolkits during development.
- **Snapshot:** Snapshots record the state of items within a workflow project at a specific point in time.

Topics

- IBM Business Automation Workflow key capabilities
- Primary components of IBM Business Automation Workflow
- The Workflow Center
- Creating a process application
- Managing snapshots
- IBM Process Designer
- The Process Portal
- The Process admin console
- The WebSphere application server administrative console
- IBM Business Automation Workflow architecture
- Overview of IBM Cloud Pak for Automation
- IBM Business Automation Workflow on Cloud

1.1. IBM Business Automation Workflow key capabilities

IBM Business Automation Workflow key capabilities

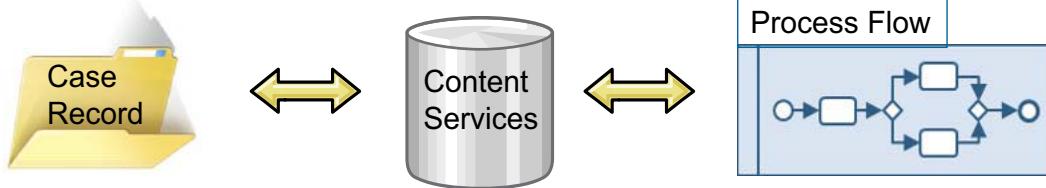
Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-5. IBM Business Automation Workflow key capabilities

IBM Business Automation Workflow key capabilities

- IBM Business Automation Workflow (BAW) provides three key capabilities
 1. **Case Management:** To track the data and work associated with a business entity such as a person, investigation, or product
 2. **Business Process Management:** To orchestrate human and system tasks in a predefined and structured flow
 3. **Content Services:** To store, manage, govern, and secure structured and unstructured content
- This course covers the Business Process Management portion of the product only. Case management is not covered in this course.



Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-6. IBM Business Automation Workflow key capabilities

IBM Business Automation Workflow provides three key capabilities. This course covers the Business Process Management portion.

Case Management and Business Process Management comparison

	Case Management	Business Process Management
Goal is to:	Optimize case outcomes	Optimize process efficiency
Focus is on:	Why tasks are completed	How tasks are completed
Emphasis:	Flexibility	Efficiency
Relies on:	Knowledge workers to determine when a business goal is reached	System to determine when a business goal is reached
Solution is:	A case that consists of pieces that are chosen to fit the situation	A process that consists of steps that apply to every situation
Level of control:	Case workers have considerable control over which tasks must be done	Process participants have little control over which tasks must be done
Design Flexibility	Cases can be modified at run time	Adding complexity to the workflow design requires development changes

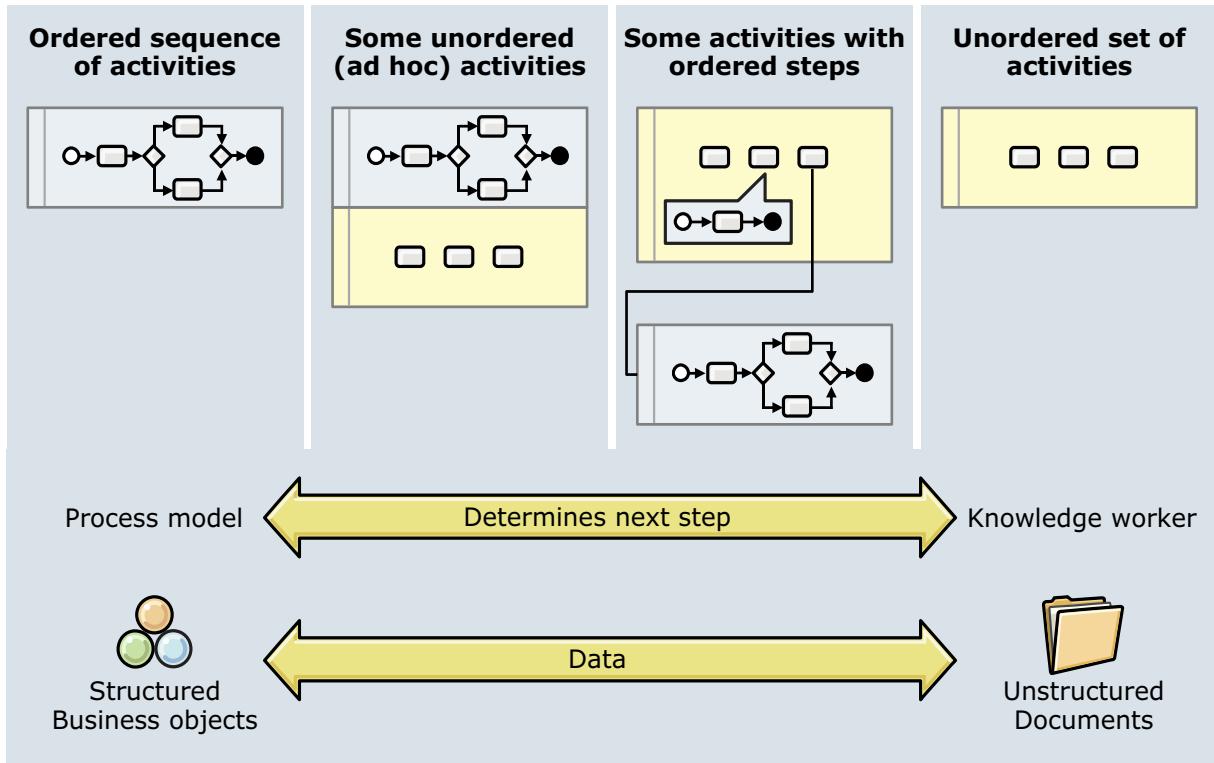
INTRODUCTION TO IBM BUSINESS AUTOMATION WORKFLOW

© Copyright IBM Corporation 2020

Figure 1-7. Case Management and Business Process Management comparison

- This slide covers the differences between Case Management and Business Process Management. This course focuses only on Business Process Management. Be sure that you are taking the correct course.
- Business process management involves optimizing processes whereas the primary goal of Case Management is to optimize case outcomes. To enable this optimization, Case Management solutions are more flexible.
- This course covers the building of structured processes to enable a Business Process Management implementation.

The process spectrum



Introduction to IBM Business Automation Workflow

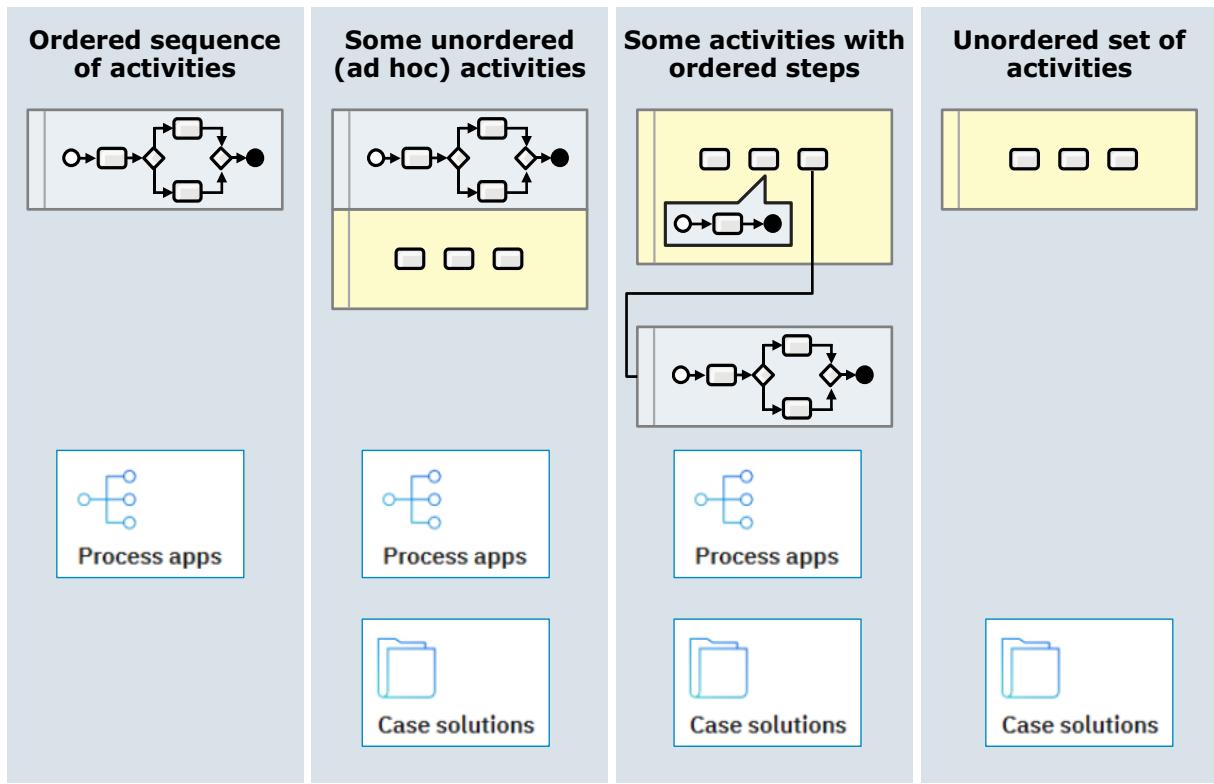
© Copyright IBM Corporation 2020

Figure 1-8. The process spectrum

IBM BAW supports a broad spectrum of process types. They include structured process, ad hoc, basic process with ordered steps, and basic process with unordered set of activities.

This course covers the development of structured processes for a Business Process Management implementation.

How BAW covers the process spectrum



Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-9. How BAW covers the process spectrum

As displayed, IBM BAW supports the full spectrum of process types. This slide cross-references functionality to solution types.

- **Ordered sequence of activities**
 - This is the process type that you create in this course.
- **Some unordered (ad hoc) activities and some activities with ordered steps**
 - Process Applications that are built in IBM Process Designer can also access information regarding related Case solutions.
 - An example use case is when documents that are part of a Case solution are also used in a structured process.
 - For instance, a structured process is started as a result of the receipt of a document that is part of a Case solution.
 - You build this solution later in the course.
- **Unordered set of activities**
 - In this scenario, there are no ordered sets of activities. Only a Case solution would be appropriate.

1.2. The Workflow Center

The Workflow Center

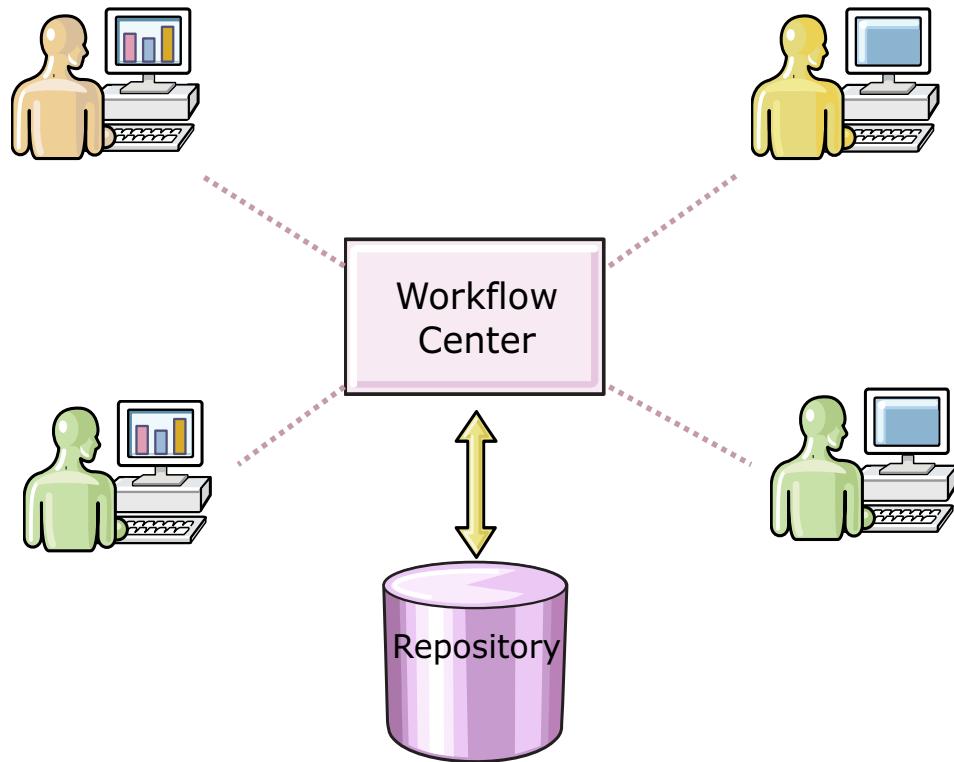
Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-10. The Workflow Center

The unique design environment of IBM Business Automation Workflow includes a central repository, which is called the Workflow Center.

The Workflow Center: The center of process development



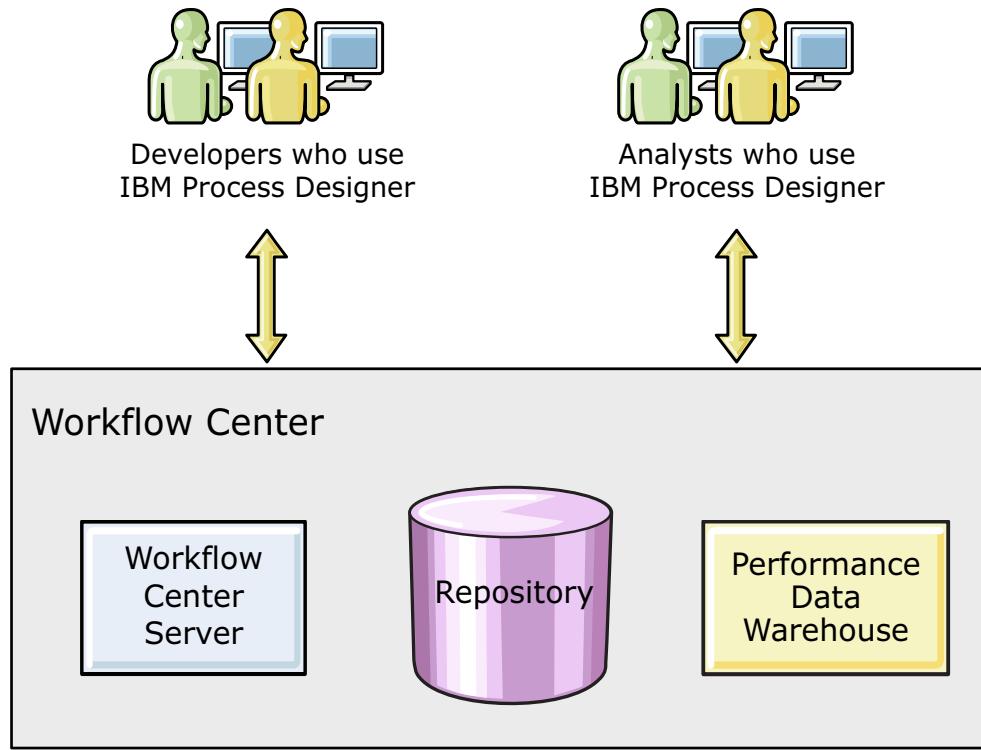
Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-11. The Workflow Center: The center of process development

- The Workflow Center provides a central development environment and repository for multiple process authors who are working in IBM Process Designer. The Workflow Center includes a Workflow Center server and a Business Performance Data Warehouse, which you can use to build and run process applications. You can also use them to store process performance data for testing and Playback purposes during development efforts.
- Multiple authors can connect to the Workflow Center and concurrently view the same processes or assets.

The Workflow Center: Process applications



Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-12. The Workflow Center: Process applications

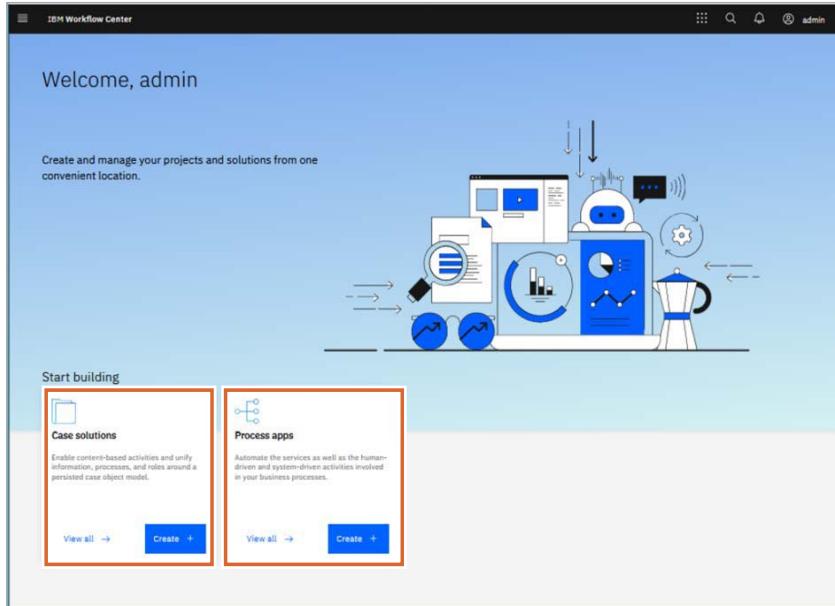
The Workflow Center contains three components.

1. Workflow Center repository, which is responsible for managing the artifacts. The Workflow Center can be accessed through IBM Process Designer.
2. Workflow Server, which enables running the process application in the development environment
3. Performance Data Warehouse, which collects and aggregates process data from running processes.

The Process Application is the highest-level “container” for a set of assets. When importing, exporting, and deploying your assets into a different environment, you use the entire Process Application and all the assets that it contains.

The Workflow Center console (1 of 3)

- The Workflow Center is the central place in IBM Business Automation Workflow where developers create Process Apps, Case solutions, and Toolkits.
- This course only covers Process Applications and Toolkits. Case Solutions are not covered in this course.



Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-13. The Workflow Center console (1 of 3)

The Workflow Center is the central place in IBM Business Automation Workflow where developers create Process Applications, Case Solutions, and Toolkits. Workflow Center is a central repository for all project assets that are created using Process Designer. Users can share items, such as processes and services. In Workflow Center, the same repository is used for both process and case solutions.

The Workflow Center unifies case and process solutions at a Solution level

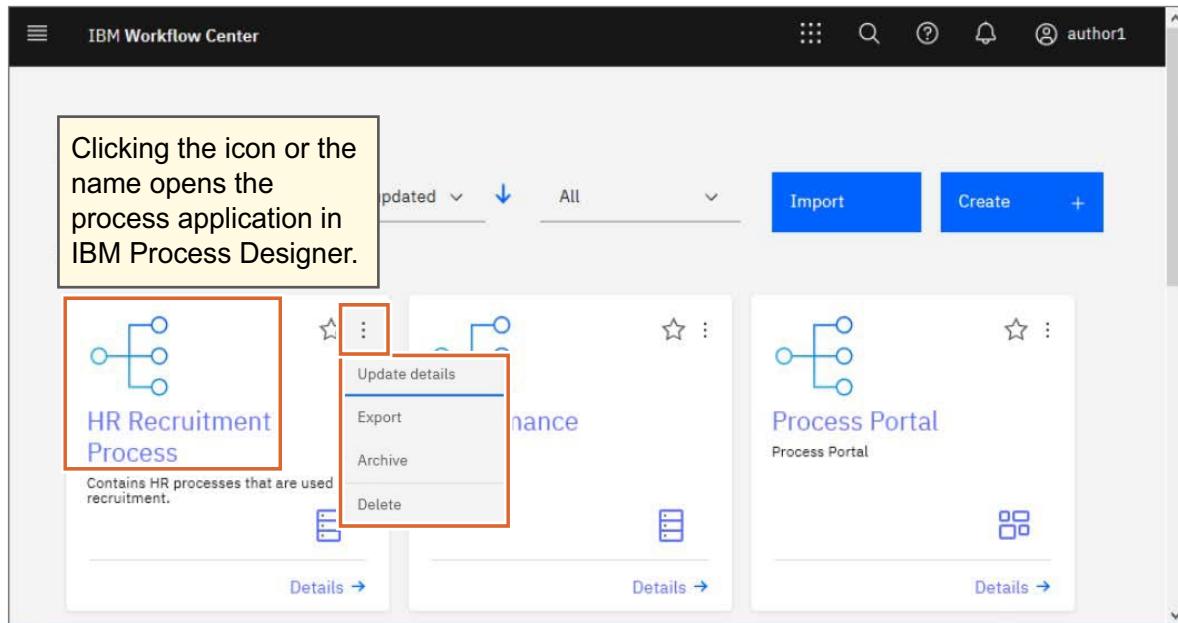
- View and manage your workflow projects from Workflow Center
- Create a single, consolidated workflow project that combines both process and case artifacts
- Add and orchestrate locally managed case activities that are backed by processes to your workflow projects
- Processes are aware of cases that orchestrate them

The look-and-feel is based on IBM Carbon Design.

This course only covers Process Applications and Toolkits. Case Solutions are not covered in this course.

The Workflow Center console (2 of 3)

When you open Process Apps in the Workflow Center, you can select one to open in IBM Process Designer. You can also view details, update details, export, archive, and delete process applications.



Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-14. The Workflow Center console (2 of 3)

When you open the Process Apps in the Workflow Center, you can select one to open in IBM Process Designer. You can also view details, update details, export, archive, and delete process applications.



The Workflow Center console (3 of 3)

- You can access the Classic Workflow Center by clicking the icon at the top.
- Icons that are displayed on the process app tile denote whether it is for a traditional deployment or combination of traditional and container.

The screenshot shows the 'Process apps (8)' section of the Workflow Center. At the top right, there is a yellow callout box with the text 'Click this icon to access the Classic Workflow Center' pointing to a small square icon with a gear and a checkmark. Below this, there are three process app tiles:

- HR Recruitment Process**: Contains HR processes used for recruitment. It has a blue square icon with three circles and a gear.
- Performance**: Contains performance-related processes. It has a blue square icon with three circles and a gear.
- Business Portal**: Contains business portal processes. It has a blue square icon with four smaller squares arranged in a 2x2 grid.

Each tile has a yellow callout box with deployment information:

- The first two tiles have a yellow callout box with the text 'This icon denotes a traditional deployment.' pointing to their respective icons.
- The third tile has a yellow callout box with the text 'This icon denotes a traditional or container deployment.' pointing to its icon.

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-15. The Workflow Center console (3 of 3)

In this course, you work with traditional deployment.



The classic Workflow Center console

The screenshot shows the classic Workflow Center console interface. At the top, there are five navigation tabs: Process Apps (highlighted), Toolkits, Servers, Admin, Preferences, Logout, and a search bar. The main content area displays a list of process applications:

Process Application	Last updated	Action
HR Recruitment Process (HRR)	3/24/19 by author1	Open in Designer
Performance (SYSPERFDB)	2/3/19 by author1	Open in Designer
Process Portal (SYSRP)	2/3/19 by author1	Open in Designer
Heritage Process Portal (deprecated) (TWP)	2/3/19 by author1	Open in Designer
Procurement Sample (STPPS1)	2/1/19 by author1	Open in Designer
Hiring Sample Advanced (HSAV1)	2/1/19 by author1	Open in Designer
Discover BPM UI Sample (DBPMUIS)	2/1/19 by author1	Open in Designer
Hiring Sample (HSS)	2/1/19 by author1	Open in Designer

To the right of the list is a sidebar with the following options:

- Create New Process App
- Import Process App
- Download desktop Process Designer (deprecated) [Learn more](#)
- Launch Getting Started
- Process Applications: A detailed description of what process applications are.
- [Managing process applications](#)

Callouts numbered 1 through 8 are overlaid on the interface:

- Process Apps tab
- Toolkits tab
- Servers tab
- Admin tab
- Logout button
- Create New Process App option
- Import Process App option
- Download desktop Process Designer (deprecated) option

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-16. The classic Workflow Center console

You do not work with the classic Workflow Center in this course. This slide is presented for reference only.

- Process Apps:** You can create, clone, and import process applications and do other maintenance tasks on the process applications.
- Toolkits:** You can create toolkits to enable Process Designer users to share library items across process applications.
- Servers:** Administrators can manage the IBM Business Automation Workflow servers in their environments.
- Admin:** Administrators can manage user access to the Workflow Center repository from the Workflow Center console.
- Search:** This field allows users to conduct searches on the Workflow Center repository.
- Create New Process App:** This option allows user to create a process application. You have a chance to create a new process application in the lab exercise at the end of this unit.
- Import Process App:** This option allows users to import a process application.
- Download desktop Process Designer (deprecated):** You can use this option to download the Process Designer installation file. However, keep in mind this is deprecated.

Creating a process application

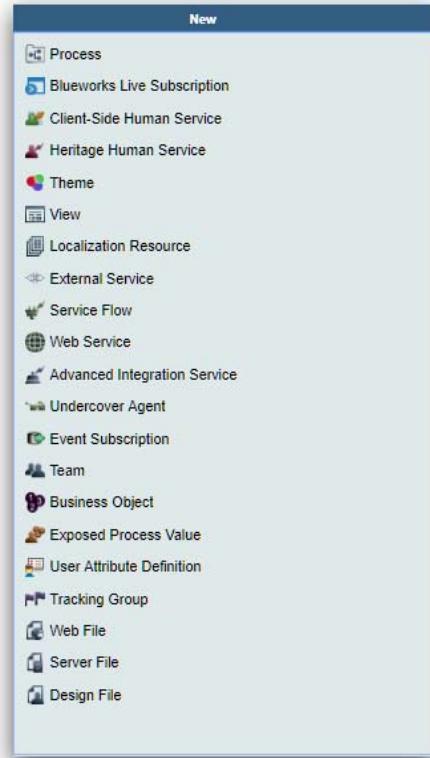
Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-17. Creating a process application

Creating a process application

- A process application is a container for process models and their supporting implementations, and it is stored in the repository
- Process applications contain some or all of the following artifacts:
 - One or more processes
 - References to toolkits
 - The services that are required to implement activities or integrate with other systems, including Advanced Integration services
 - One or more tracks
 - Service Component Architecture (SCA) modules and libraries (authored in IBM Integration Designer)
 - Other items that are required to run the process



Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-18. Creating a process application

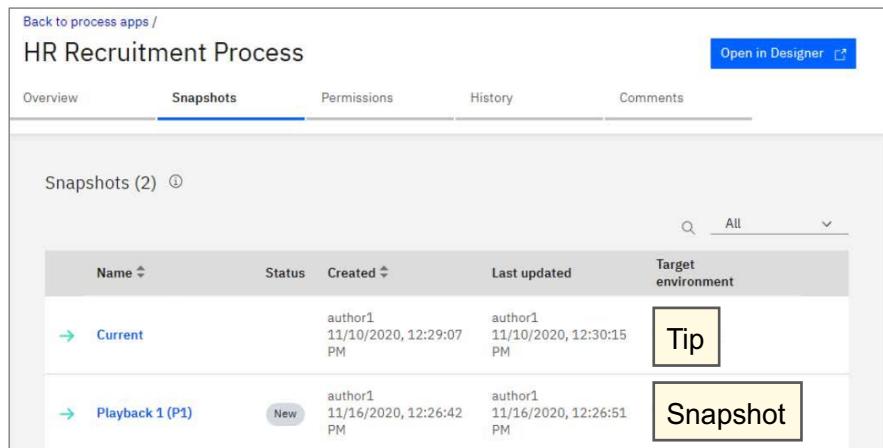
A process application is a container for process models and their supporting implementations. All the process models and supporting implementations are stored in the repository. After the artifacts are authored or otherwise created, they are assembled into a process application. The screen capture shows the list of new artifacts you can create when building a process application.

A track is an optional subdivision in a process application based on team tasks or process application versions. If more tracks are necessary for a process application, you can enable them at any time.

IBM Integration Designer is not covered in this course.

Process application tip, snapshots, and tracks

- Any changes to a process application are saved to the workflow center repository at the tip
- Tip is the current working version of the process application
- Run your application on the tip to instantly test and manage the current working version of the process application
- Tip is a special snapshot:
 - The only type of snapshot in which you can change contents
 - It runs only on the workflow center server
 - Cannot install a tip on a Workflow Server



Name	Status	Created	Last updated	Target environment
Current	author1 11/10/2020, 12:29:07 PM	author1 11/10/2020, 12:30:15 PM		Tip
Playback 1 (P1)	New	author1 11/16/2020, 12:26:42 PM	author1 11/16/2020, 12:26:51 PM	Snapshot

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-19. Process application tip, snapshots, and tracks

- For the selected process application or toolkit, you can manage user access to the project, allow users to create tracks, and add or edit the description.
- Any changes that you make to a process application are dynamically saved to the workflow center repository at the tip, which is the current working version of the process application. You can use Playback sessions on the tip to instantly test and manage the current working version of the process application.
- The process application remains at that tip level until you decide to create a snapshot, which records the state of library items within a process application or track at a specific point in time. Typically, you take a snapshot whenever you are ready to test the integration or want to install the process application on a workflow center server or a Workflow Server for development, test, staging, or production.
- The tip is a special snapshot; it is the only type of snapshot in which you can change contents, but you can run it only on the workflow center server. You cannot install a tip on a Workflow Server.
- Tracks allow parallel development to occur with isolation from changes in other tracks. For example, tracks enable one team to fix the current version of a process while another team builds a new version based on new external systems and a new corporate identity.

- By default, each process application has a single track, called Main. If you want to allow parallel development on a process application, you can create more tracks. These optional subdivisions in the process application keep changes isolated. For example, imagine that your company is in the midst of rebranding; during this transition, the current process applications must be maintained while new versions are being developed based on the updated corporate identity. In this situation, one team might be making minor fixes on the current version of a process application (in the Main track). At the same time, another team is building a new version of the process application in a separate track.

1.3. Managing snapshots

Managing snapshots

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-20. Managing snapshots



Creating a snapshot

- A snapshot captures the state of the library items within a process application or toolkit at a specific point in time.
- To create a snapshot in the Workflow Center:
 1. Click **Details** on the process app tile
 2. On the Snapshots tab, open the list of Options by clicking the three dots on the right side of the Current tip.
 3. Select **Create a snapshot**

The screenshot shows the 'HR Recruitment Process' application in the Workflow Center. The 'Snapshots' tab is selected. A context menu is open on the 'Current' snapshot, indicated by a yellow circle with the number 3. The menu options are 'Create a snapshot' (highlighted with a red box) and 'Open in Designer'. The interface includes a search bar, a filter for 'All', and a dropdown menu.

Name	Status	Created	Last updated	Target environment
→ Current	author1	9/22/2020, 12:51:50 PM	author1 9/22/2020, 12:51:50 PM	Traditional

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-21. Creating a snapshot

Before exporting your process application you need to take a snapshot.



Exporting a snapshot

- After you create a snapshot, you can export it.
- To export a snapshot in the Workflow Center,
 1. Click **Details** on the process app tile
 2. On the Snapshots tab, open the list of Options for the snapshot
 3. Select **Export**

Snapshots (1) ⓘ

Name	Status	Created	Last updated	Target environment
→ Current		author1 9/22/2020, 12:51:50 PM	author1 9/22/2020, 12:51:50 PM	Traditional
→ Exercise 1 solution (E1S)	New	author1 9/22/2020, 3:35:42 PM	author1 9/22/2020, 3:35:42 PM	Traditional

2

3

Status
Update details
Clone
Activate
Archive
Export
Install
Open in Designer

Introduction to IBM Business Automation Workflow

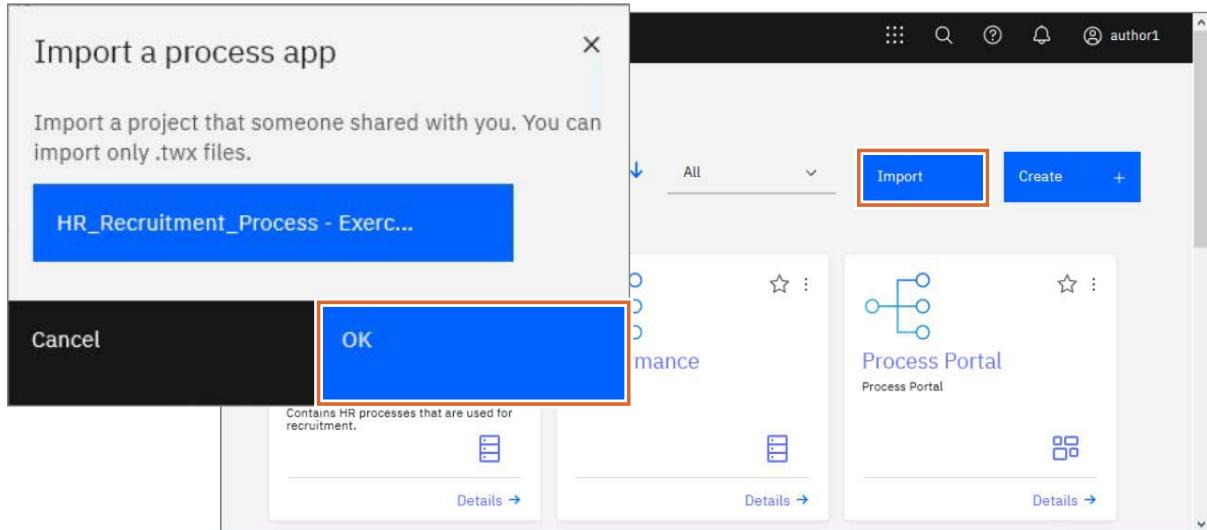
© Copyright IBM Corporation 2020

Figure 1-22. Exporting a snapshot

When you export your snapshot, it is saved as a .twx file. Only snapshots can be exported and imported (the Tip cannot).

Importing a snapshot

- You can import snapshots you previously exported.
- To import a snapshot in the Workflow Center,
 1. Click **Import** on the process apps page
 2. Browse to the location of the **.twx** file and open it
 3. Click **OK**



Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-23. Importing a snapshot

1.4. IBM Process Designer

IBM Process Designer

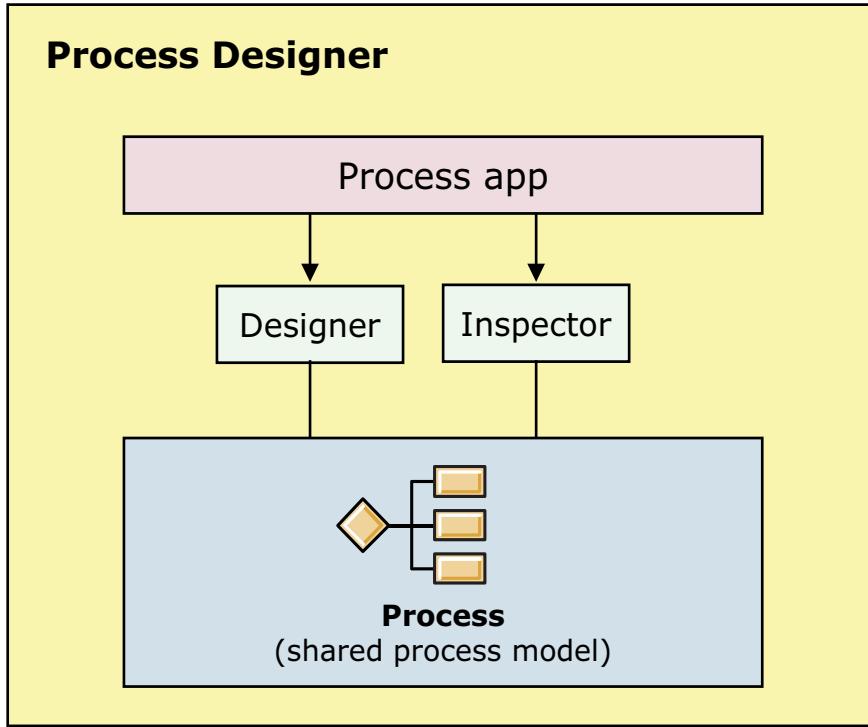
Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-24. IBM Process Designer

- After a process application is in place in the Workflow Center, the next step is to create and edit the process that will be contained within. This step is accomplished through the IBM Business Automation Workflow interface: IBM Process Designer.
- Now that you created a process application, the next section looks at the options available in IBM Process Designer to create and manage assets for business processes that are part of the process application.

IBM Process Designer



Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-25. IBM Process Designer

Process modeling in IBM Business Automation Workflow is accomplished through the IBM Process Designer views or interfaces. These interfaces allow developers or authors to create, manage, and test process models.

IBM Process Designer is composed of two key interfaces:

- Designer: to model your processes, services, and other process application assets
- Inspector: to run and debug your processes and services

To access these interfaces, an author goes through the central repository, the Workflow Center, and opens a process app.



Starting the IBM Process Designer (1 of 2)

The screenshot shows a Windows application window titled "Business Automation Workflow Quick Start". On the left, there is a sidebar with four items: "ProcessCenter Quick Start" (highlighted with a red box), "Profile Management Tool", "Start the deployment manager", and "Start the node agent". The main content area is titled "Business Automation Workflow Enterprise Quick Start for ProcessCenter profile DmgrProfile". It contains sections for "Deployment environment administration" (with a link "Start the Workflow Center Deployment Environment" highlighted with a red box), "Administration consoles & tools" (with a link "WebSphere Application Server Administrative Console"), and "Documentation" (with a link "Knowledge Center"). At the bottom of the main content area, it says "IBM".

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-26. Starting the IBM Process Designer (1 of 2)

IBM Process Designer is accessed through a browser. Before you can access it in the browser, you need to ensure that the Workflow Center is running. You do this by clicking the Quick Start icon and then clicking the link to start the Workflow Center Deployment Environment. Once started, you can access IBM Process Designer by clicking the Process Center Console link.

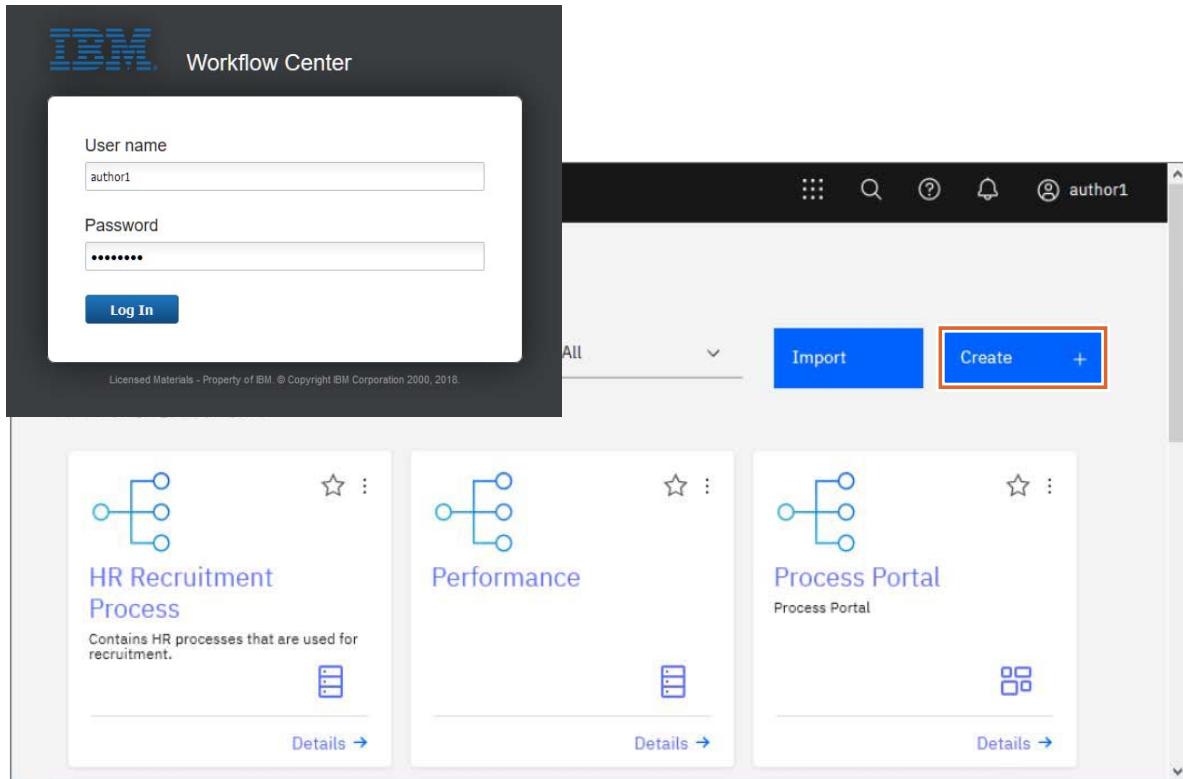
Process Center is the legacy term for Workflow Center. Note that this link points to the Classic Workflow Center. To access Workflow Center, access the link in the browser.

The Workflow Center repository URL is: <https://<server:port>/WorkflowCenter/>

The Classic Workflow Center repository URL is: <https://<server:port>/ProcessCenter/>



Starting the IBM Process Designer (2 of 2)



Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-27. Starting the IBM Process Designer (2 of 2)

The Workflow Center window displays the list of process applications that are available. To create a new process application, click **Create**



IBM Process Designer interface

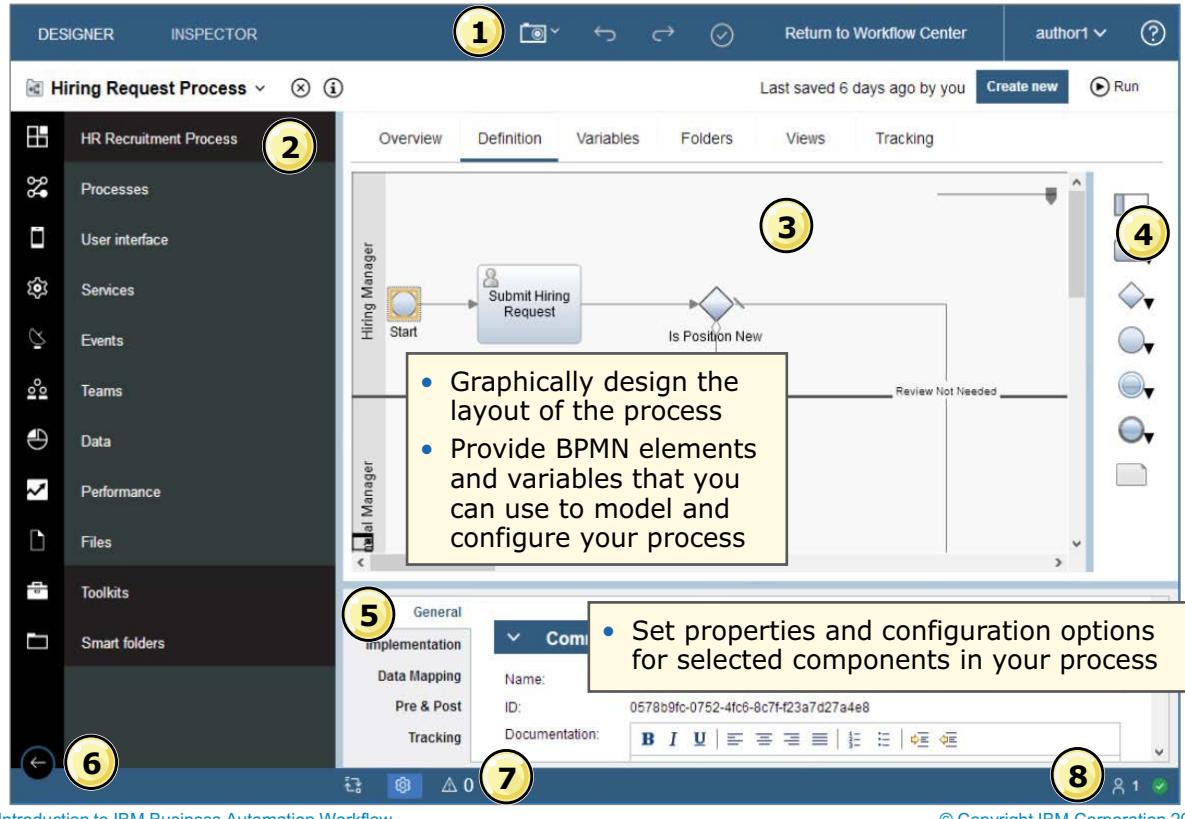


Figure 1-28. IBM Process Designer interface

You use IBM Process Designer to model and implement your business processes and easily demonstrate process design and functionality during development efforts.

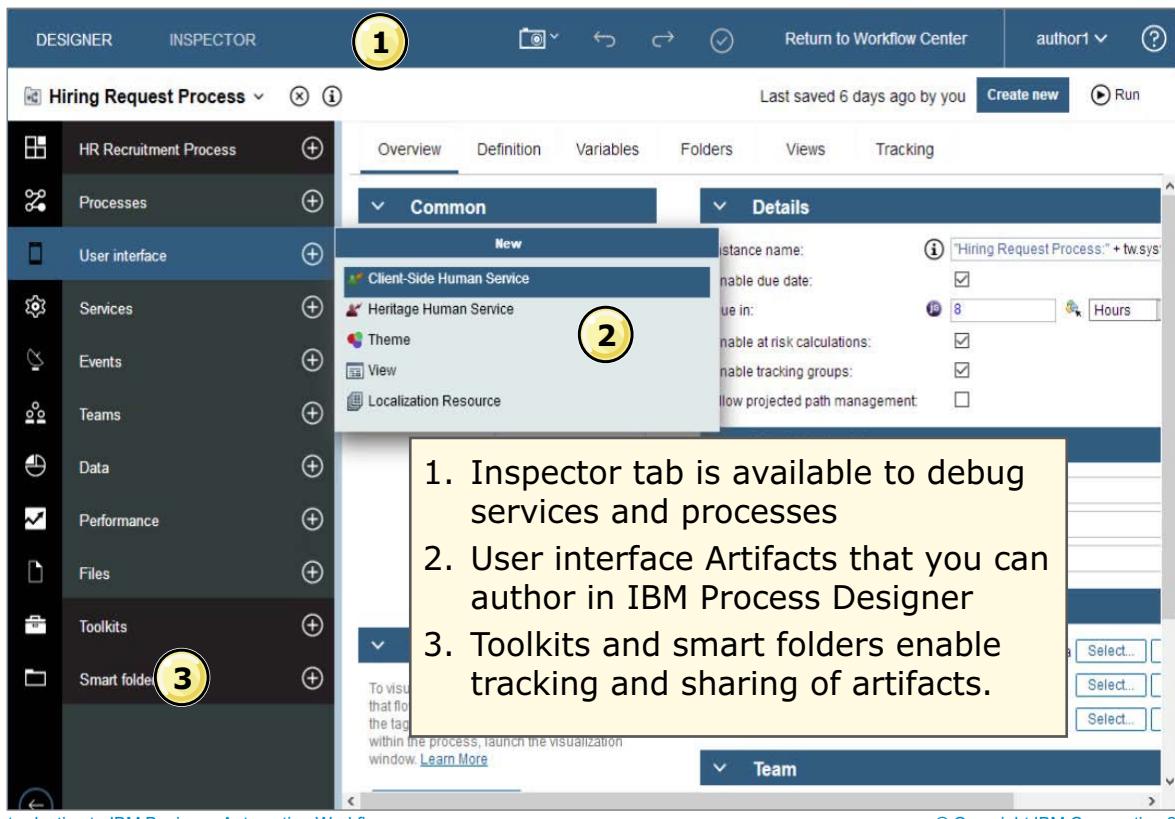
- Main toolbar:** Provides access to Designer, Inspector, and Workflow Center. The main toolbar is also where you save all open assets and view help.
- Process library:** Provides access to the library items for the current process application.
- Main canvas:** The area in which you can graphically model your process. Each process automatically includes a start event and an end event. Two default lanes are included for user and system tasks.
- Palette:** When you develop the process diagram in IBM Process Designer, the web components are available from the palette.
- Properties:** Opens the view to set the properties and configuration options of the item that is selected on the canvas.
- Process Library Expand:** Toggle this arrow to expand the Process Library or to minimize it.
- Status bar:** You can toggle the following on or off:
 - Artifact references

- View artifact dependencies, such as the artifacts that use a particular library item and the artifacts that the library item uses in process applications and across toolkits.
 - Properties
 - You can toggle the properties panel off in the main canvas to provide more room for modeling.
 - Validations
 - Because validation occurs on the workflow server, it can slow down performance for the user. For this reason, you might choose to enable it only when you want to import and test a process application for errors before you deploy the application on a production system.
8. **Server status:** These two icons represent: Number of viewers currently in the project and connection status to the Workflow Center (green equals connected, red equals disconnected).

IBM Training



IBM Process Designer: Web interface



Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

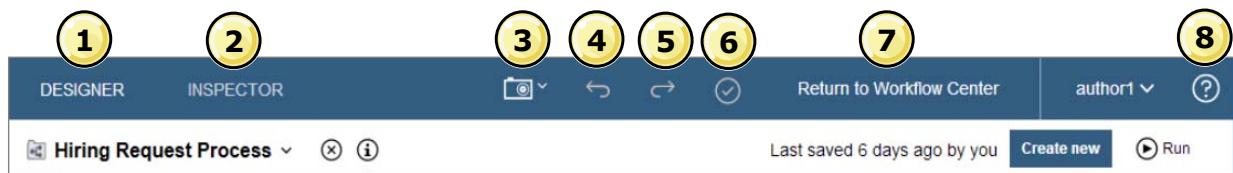
Figure 1-29. IBM Process Designer: Web interface

The interface of the IBM Process Designer supports creating new artifacts, debugging services and tracking and sharing artifacts.

You can arrange library items in smart folders for quick and easy access in IBM Process Designer.



IBM Process Designer: Main toolbar



- 1. Designer:** Provides the interface to model your processes
- 2. Inspector:** Run and debug instances of processes
- 3. Snapshot:** Record the state of library items within a process application or track at a specific point in time
- 4. Undo:** Revert to the last state in the process
- 5. Redo:** Redo the most recent changes that were made in the process
- 6. Save:** Save your work
- 7. Return to Workflow Center:** Takes you back to the Workflow Center
- 8. Help:** Open the IBM Business Automation Workflow Help system

Introduction to IBM Business Automation Workflow

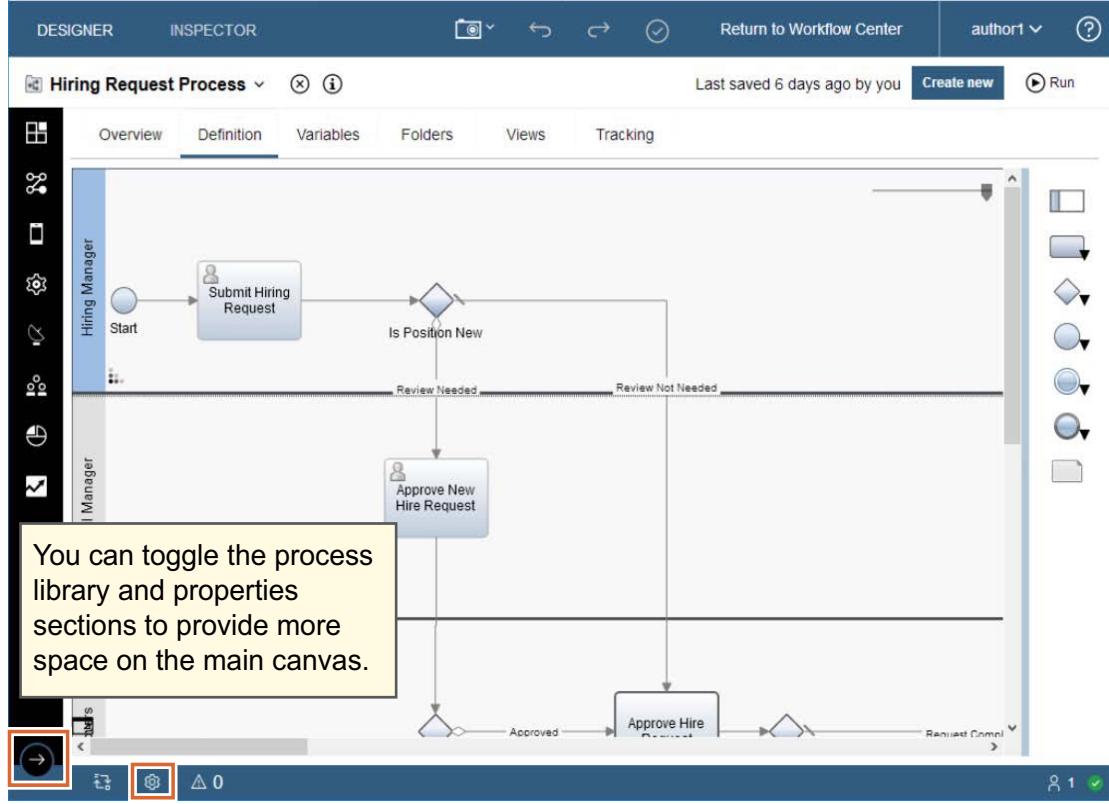
© Copyright IBM Corporation 2020

Figure 1-30. IBM Process Designer: Main toolbar

This slide provides a description of the toolbar functions. You have the opportunity to work with IBM Process Designer at the end of the next Unit.



IBM Process Designer editor (1 of 2)



Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-31. IBM Process Designer editor (1 of 2)

The IBM Process Designer user interface is configurable. You can toggle the process library and properties sections to provide more space on the main canvas.



IBM Process Designer editor (2 of 2)

- When building user-interfaces, the IBM Process Designer is designed for WYSIWYG (what you see is what you get).
- You can change the screen size to match the form factor you are implementing

Icon	Size	Resolution
	Small	640 pixels or less
	Medium	641 - 1024 pixels
	Large	More than 1024 pixels

The screenshot shows a mobile application interface for a requisition form. At the top, there are tabs for 'Requisition Details', 'Position Details', 'Recruiting Details', 'Compensation Details', and 'Department Details'. Below these are sections for 'Requisition Number' (text input), 'Requester' (text input), 'Date Details' (with a calendar icon), 'Date Of Request' (text input), 'Date Position Available' (text input), and 'Hiring Manager Comments' (text input). At the bottom left is a 'Submit' button, and at the bottom right are navigation icons for back, forward, and search.

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-32. IBM Process Designer editor (2 of 2)

The Process Designer user interface uses the WYSIWYG principle. This makes it easy to design the user interface by using web-based components that can respond to various screen sizes.

1.5. The Process Portal

The Process Portal

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-33. The Process Portal

- The Process Portal is the main tool that business process users interact with to complete tasks and processes. Other tools, such as the IBM Business Automation Workflow mobile application, can also be used to complete tasks and processes.
- The Process Portal also has use for project development, especially in terms of validation. BPM teams and business stakeholders want to reach consensus in the Playback session to end a stage of development. When consensus is the goal, the Process Portal allows the team to view the process performance as it would function in a user environment.



The Process Portal overview

The screenshot shows the IBM Business Automation Workflow Process Portal. On the left, there's a sidebar with a user profile (author1), edit profile, log out, and a search bar. Below that are links for Dashboards, Work, Processes, Process Performance, Team Performance, Launch, Advanced HR Open New Position, Discover BPM UI, Hiring Request Process, ReplenishmentBPD, and Standard HR Open New Position. The main area is titled "Work" and shows a list of work items:

Step	Process	Due Date
Step: Complete Hire Request	Hiring Request Process.211	Mar 24, 2019, 10:01:10 PM
Step: Override Hire Request	Hiring Request Process.254	Mar 26, 2019, 12:59:57 PM
Step: Review Posting	Hiring Request Process.256	Mar 26, 2019, 1:02:48 PM
Step: Check Hire Request	Hiring Request Process.257	Mar 26, 2019, 1:50:17 PM

At the bottom, there are pagination controls (1, 25, items per page) and a note "1 - 4 of 4 items".

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-34. The Process Portal overview

Process Portal Overview

- Process Portal is the user interface that is provided with IBM Business Automation Workflow for participating in processes. It can be configured for a single IBM Business Automation Workflow system or federated environments that include IBM® Process Federation Server. From Process Portal, users can start processes, dashboards, and startable services, and work on their tasks.
- Because Process Portal is based on coaches, it is highly customizable. For example, the coaches use the Classic theme, which means that you can easily change the overall styling to fit your organization's look and feel. In addition, a set of configuration options is available for you to adapt Process Portal behavior without having to customize the Process Portal application itself. The coach-based design also makes it easier to use Process Portal as the baseline for your own custom portals.
- Because it's critical to be able to search business data, Process Portal has a robust search capability. You can also create customized views of your tasks, for example, with specific business data, by saving your task-based searches for later use. You can also share your saved search with the members of your team.
- Dashboards are a critical piece of Process Portal. To analyze and manage the work on your business processes, you can use the ready-to-use dashboards that Process Portal includes or the company-specific dashboards that your installation might provide.

- If Process Portal is configured for a federated environment, you can access processes and tasks on any IBM Business Automation Workflow system in the federated environment running V8.0.1.3 or later, or on an IBM Business Automation Workflow system, from a single Process Portal sign-on.

The screenshot shows the main page of the IBM Process Portal. At the top left is the "IBM Training" logo. At the top right is the "IBM" logo. The main header is "The Process Portal main page". On the left is a sidebar with a user profile for "author1" (number 1), a "Dashboards" section (number 2), a "Processes" section (number 3), and a "Work" section (number 4). The "Work" section displays a list of tasks:

- Step: Complete Hire Request (number 5)
- Step: Override Hire Request
- Step: Review Posting
- Step: Check Hire Request

At the top right of the main area, there are statistics: "4 Total Open", "2 On Track", "1 At Risk", and "1 Overdue". Below the tasks are pagination controls: "< 1 > 25 items per page" and "1 - 4 of 4 items".

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-35. The Process Portal main page

The Process Portal has a main page that you can use to run and manage tasks.

1. User can edit their profile and log out of here.
2. Users view their current process performance or the team's performance by selecting the tab for each dashboard.
3. You can start your process.
4. **Search** allows users to search tasks.
5. On the Work page, **Overdue**, **On Track** and **At Risk** headings are used to separate tasks. Users manage tasks by using the menu directly to the right of the task.
 - On the screen capture, there is one work item overdue, one item at risk, and two items on track.
6. To complete a task, click the Step and claim the task. The user-interface that is designed for the step appears.



Viewing processes in the Process Portal (1 of 2)

The screenshot shows the IBM Process Portal interface. On the left, there's a sidebar with various links like Dashboards, Work, Processes (which is highlighted with a red box), and others. The main area is titled 'Processes' and shows 'Process Instances'. It has a search bar at the top. Two instances are listed:

- Hiring Request Process:211** (Due: March 25, 2019 4:55 AM)
- Hiring Request Process:253** (Due: March 26, 2019 7:52 PM)

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-36. Viewing processes in the Process Portal (1 of 2)

Along with completing work items and viewing the status of them, the user can also check the status of the process they are participating in. When you click **Processes** in the Process Portal, the currently running process instances are displayed with their status. Process instances are the runtime implementation of a particular process. Items are sorted by Active and Completed. When you click the process name, you can view details of the process instance. The number next to the name is the process instance ID. This ID is unique for each process instance.



Viewing processes in the Process Portal (2 of 2)

The screenshot shows the IBM Process Portal interface for a "Hiring Request Process:255". The main title is "Hiring Request Process:255". The interface is divided into several sections:

- 1 Hiring Request Process:255**: The main process instance.
- 2 Data**: No details found.
- 3 Documents**: No documents or folders were found at this level.
- 4 Tasks**: A list of tasks, including "Step: Approve New Hire Request" which is due on March 26, 2019, 12:59 PM, assigned to General Managers, and is marked as overdue.
- 5 GANTT CHART**: Shows the timeline from Mar 26, 2019 11:59 AM to Mar 26, 2019 7:59 PM.
- 6 Stream**: A timeline stream showing a comment "Stream" posted at Mar 26, 2019 11:59 AM.
- 7 Activities**: No activities were found.

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-37. Viewing processes in the Process Portal (2 of 2)

The details of the process instances are displayed on this page. From this page, you can view the following:

1. The process instance with the instance ID.
2. Data flowing between the tasks. Only data flagged as being visible will display.
3. Documents that are attached to the process.
4. Open and completed tasks are listed. You can claim an open task from here.
 - The status of the task is displayed by using appropriate icons. The task in the screen capture is overdue.
5. You can view the Gantt chart of the process.
6. The stream shows the actions that occur for this instance. You can also post a comment to the stream that appears immediately in the stream.
7. Activities are shown under the Activities section.

The Process Portal: Social features

The screenshot illustrates the social features of the IBM Process Portal. On the left, a task stream is displayed with a 'Stream' button highlighted (1). A comment box (2) is shown with a 'Post' button, and a Gantt chart (3) is visible. On the right, there are 'Following' and 'Mentions' sections, each with a count of 0 and a 'View' button.

1. The Stream shows the actions that occur for this instance
2. You can also post a comment to the stream. The comment appears immediately in the stream
3. You can view the Gantt chart for the process

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-38. The Process Portal: Social features

The Process Portal has several social features. When completing a task, you can use a menu in the right window to access task details, the process activity stream, and experts of the task.

Activity streams provide a way for you to monitor and complete ad hoc actions on your “favorite” processes and tasks.

Experts, either predefined or discovered dynamically (by using social analytics), help you complete process tasks. An expert’s advice can be taken while working on a task. Experts can also be invited for collaborative working on tasks. Experts can be either of two types:

- **Experienced experts:** Experts that are discovered dynamically by using social analytics. For example, business users who claim the tasks and complete on time are discovered as experienced experts.
- **Subject matter experts:** The experts that are predefined during the process-modeling phase are shown under the “Subject Matter Experts” label.

You can use the stream to comment and post, read, and comment on what actions the process completes immediately after they occur. You can upload a photo, and each post displays your photo. You can also create an IBM Connections Integration, which enables the default Process Portal business card to be replaced with the Connections business card. Automatic system posts are designated by an icon with blue gears.

In addition to text posts, you can also post attachments and links. Mention other users by using the @ symbol and typing the first few characters of the username.

The Process admin console

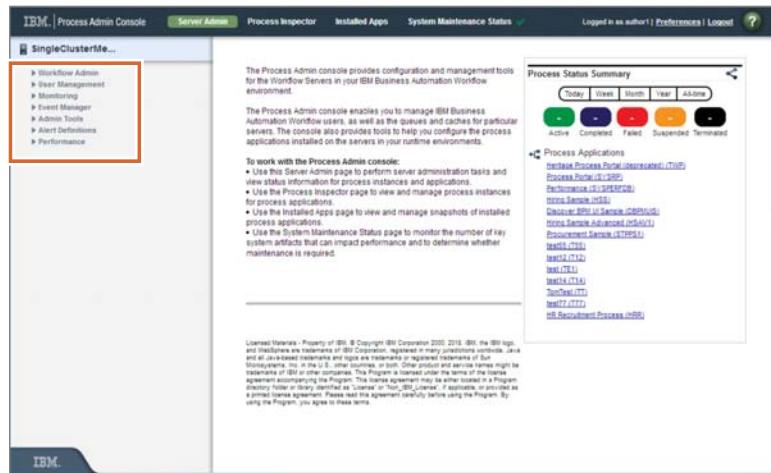
Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-39. The Process admin console

Process admin console (1 of 2)

- The Process Admin Console is used to administer the process servers in your environment, including the users and installed snapshots for each server.
- In addition, it provides tools to help you manage queues and caches.
- When the Process Admin console is displayed, note the various links in the left pane.
 - Workflow Admin
 - User Management
 - Monitoring
 - Event Manager
 - Admin Tools
 - Alert Definitions
 - Performance



Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-40. Process admin console (1 of 2)

The Process Admin console provides configuration and management tools for the Workflow Servers in your IBM Business Automation Workflow environment.

The Process Admin console enables you to manage IBM Business Automation Workflow users, as well as the queues and caches for particular servers. The console also provides tools to help you configure the process applications installed on the servers in your runtime environments.



Process admin console (2 of 2)

- The Process Admin Console is commonly used to:
 - Add users and groups
 - Manage caches and EPVs
 - Perform process and task cleanup
 - Monitor events and queues
 - Monitor performance
- The Process Inspector is accessed from the Process Admin console and is used to manage running process instances.

The screenshot shows the IBM Process Admin Console interface. At the top, there are tabs: 'Process Admin Console' (selected), 'Server Admin', 'Process Inspector' (highlighted with a red box), and 'Installed Apps'. Below the tabs, a sidebar titled 'SingleClusterMe...' lists several categories: Workflow Admin, User Management, Monitoring, Event Manager, Admin Tools, Alert Definitions, and Performance. The main content area has two sections. The left section, under 'Workflow Admin', contains text about the Process Admin console for Workflow Servers. The right section, under 'Performance', contains text about the Process Admin console for Business Automation Workflow users and applications installed on the server. A copyright notice at the bottom right reads '© Copyright IBM Corporation 2020'.

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-41. Process admin console (2 of 2)

To work with the Process Admin console:

- Use the Server Admin page to perform server administration tasks and view status information for process instances and applications.
- Use the Process Inspector page to view and manage process instances for process applications.
- Use the Installed Apps page to view and manage snapshots of installed process applications.
- Use the System Maintenance Status page to monitor the number of key system artifacts that can impact performance and to determine whether maintenance is required.



Process inspector

- As you develop your solution, you can use the Process Inspector to inspect the status of your applications.

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-42. Process inspector

As you develop your solution, you can use the Process Inspector to inspect the status of active, completed, suspended, terminated, and failed process applications. You can also view the task last executed and the data that is associated with the task.

If you expect to see a task or process display in Process Portal and it does not, you can use the Process Inspector to determine what state the process application is in.

The three main sections of the Process Inspector include:

1. Search panel

Returns the process instances that contain the provided text within the instance name or process application acronym. The search is not case-sensitive.

2. Results panel

The results panel displays one of two things. If you have not yet searched for process instances, the results panel displays a welcome message. If you have done a search, the results panel displays a list of process instances and cases that fulfill the search criteria. The list displays the first 20 instances automatically. If there are more than 20 instances, you can scroll down the list and the next 20 instances load automatically.

3. Details panel

The details panel displays information about what you have selected. If you select multiple process instances in the results panel, the details panel displays how many instances you have selected and how many have a particular status. If there are actions that you can do on all of the instances, the details panel displays those actions. If you select a single instance in the results panel, the details panel displays the information for that instance.

The WebSphere application server administrative console

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-43. The WebSphere application server administrative console



WebSphere application server administrative console

The administrative console is used to administer applications, services, and other resources at a cell, node, server, or cluster scope.

The screenshot shows the WebSphere administrative console interface. On the left, there is a navigation tree with various categories like Welcome, Guided Activities, Servers, Core Groups, Applications, Jobs, Services, Resources, Runtime Operations, Security, Operational policies, Environment, Integration Applications, System administration, Users and Groups, Monitoring and Tuning, Troubleshooting, Service Integration, and UDDI. A red box highlights the 'Deployment Environments' option under the 'Servers' category. The main panel displays the 'Deployment Environments' page with a table showing one entry: ProcessCenter, IBM Business Automation Workflow, Advanced Workflow Center, Single Cluster. Below this is another 'Deployment Environments' window showing the same information. At the bottom, there is a table titled 'You can administer the following resources:' with one row for 'Training Course DB'.

Select	Name	JNDI name	Scope	Provider	Description	Category
<input type="checkbox"/>	Training Course DB	jdbc/TrainingDB	Cluster=SingleCluster	DB2 Using IBM JCC Driver	DB2 JCC data source using the IBM Data	

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-44. WebSphere application server administrative console

The administrative console is used to administer applications, services, and other resources at a cell, node, server, or cluster scope. You can use the console with stand-alone servers and with deployment managers that manage all servers in a cell in a networked environment. If you have installed a stand-alone profile, you have a single node in its own administrative domain, known as a cell. Use the administrative console to manage applications, buses, servers, and resources within that administrative domain. Similarly, if you have installed and configured a network deployment cell, you have a deployment manager node and one or more managed nodes in the same cell. Use the administrative console to manage applications, set up managed nodes in the cell, and monitor and control those nodes and their resources.

IBM Business Automation Workflow architecture

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-45. IBM Business Automation Workflow architecture

Authoring and runtime environments

Business Automation Workflow can be implemented on premises or on cloud, in a containerized environment.

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-46. Authoring and runtime environments

Business Automation Workflow combines business process management and case management capabilities in an integrated workflow solution. The Business Automation Workflow container runtime environment provides an enterprise workflow platform to run cloud-native solutions. However, the authoring environment is on the traditional VM or on-premises system. The diagram on this slide shows the main elements for authoring workflows and running them in a containerized environment.

Although the runtime environment can be implemented either on-premises or on-cloud (containerized), the authoring environment can only be implemented on-premises.

This course uses an on-premises implementation of both the authoring and runtime environments.

Workflow Center

Workflow Center is the place where you manage the projects that you create in your authoring environments (Case Builder and Process Designer). Workflow Center has a Workflow Server for iterative development and playback of process applications, toolkits, and case solutions. When you are ready to deploy your process application or case solution to production, you can deploy it to the containerized Workflow Server.

Workflow Server (WebSphere)

In the container environment, Workflow Server is a Liberty-based server that is used for deploying and running process applications and case solutions. Workflow Server provides a single runtime environment that can support a wide range of business processes, cases, and service orchestrations. You can designate Workflow Server as one of the following server types:

- Development
- Test
- Staging
- Production

Process Federation Server

Process Federation Server enhances the collaboration between Workflow Server and IBM Process Portal. It is mandatory for Workflow Server in a container environment. Process Federation Server provides task workers with a single point of access to all of their tasks from a single Process Portal, regardless of the type of process that they are working on and the Business Automation Workflow back-end system on which the process artifacts are stored. Tasks can be federated across the following entities:

- Multiple Business Automation Workflow and IBM Business Process Manager cells
- Multiple Business Automation Workflow and IBM Business Process Manager releases
- Multiple WebSphere Application Server Network Deployment and container-based runtime architectures
- BPMN, case, and workstream tasks

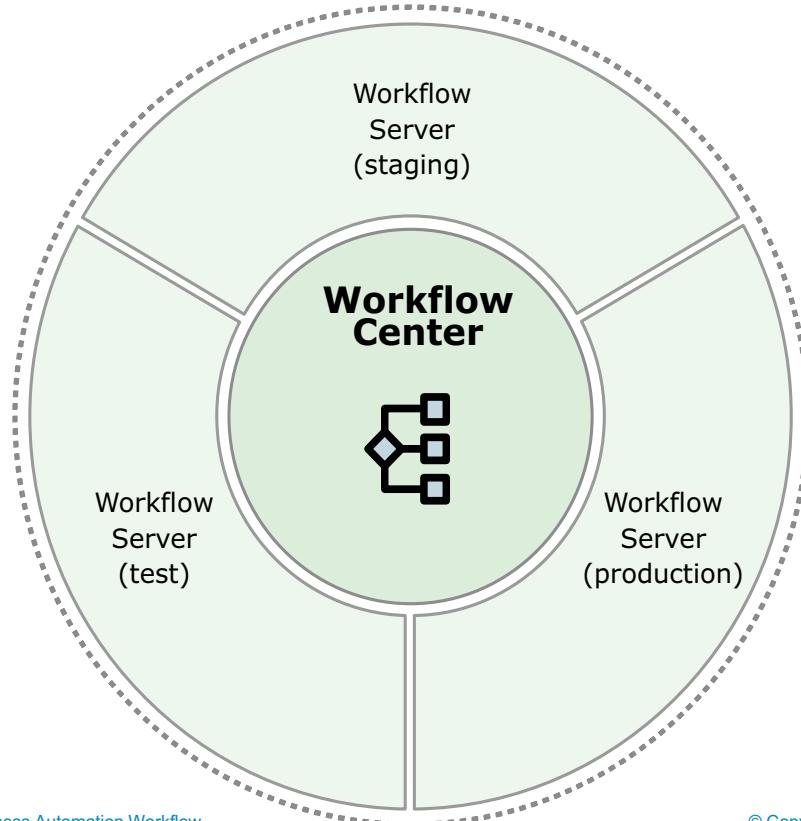
Process Portal

IBM Process Portal is a user interface that is provided with all editions of Business Automation Workflow for participating in processes. From Process Portal, you can start processes, dashboards, and services and work on tasks. Process Portal can be configured for a single Business Automation Workflow system or for federated environments that include Process Federation Server. If Process Portal is configured for a federated environment, you can access processes and tasks on any Business Automation Workflow system or IBM Business Process Manager system (V8.0.1.3 or later) from a single Process Portal sign-on. Process Portal features process instance indexing and comprehensive search support that is provided through federated search REST APIs and the Elastic search engine.

Business Automation Insights

IBM Business Automation Insights captures BPMN events and case events that are generated by Business Automation Workflow, aggregates them into business-relevant KPIs, and displays them on meaningful dashboards so that you can analyze your business data.

The Workflow Center: The center of the development effort



Introduction to IBM Business Automation Workflow

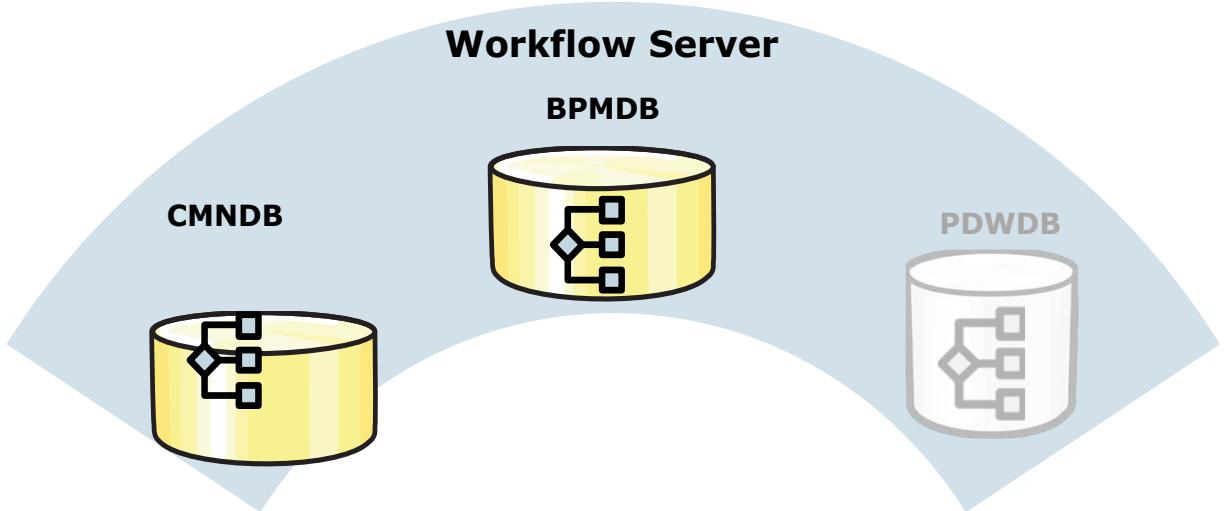
© Copyright IBM Corporation 2020

Figure 1-47. The Workflow Center: The center of the development effort

The Workflow Center not only is the center of the development effort, but also provides governance over the entire lifecycle of a business process. It includes the central repository for multiple process authors who work with process application artifacts. You can see that the different Workflow Servers surround the Workflow Center. The Workflow Center is where process applications are created and stored, whereas the Workflow Servers (also called runtime environments) are where those process applications are tested and processes are run. When the development effort reaches a milestone, a process application is installed on a Workflow Server to test, stage, or move into a production environment.

This topology is similar to a hub and spoke deployment configuration, where all development occurs inside the Workflow Center and then the assets are deployed out to each environment.

The Workflow Server databases



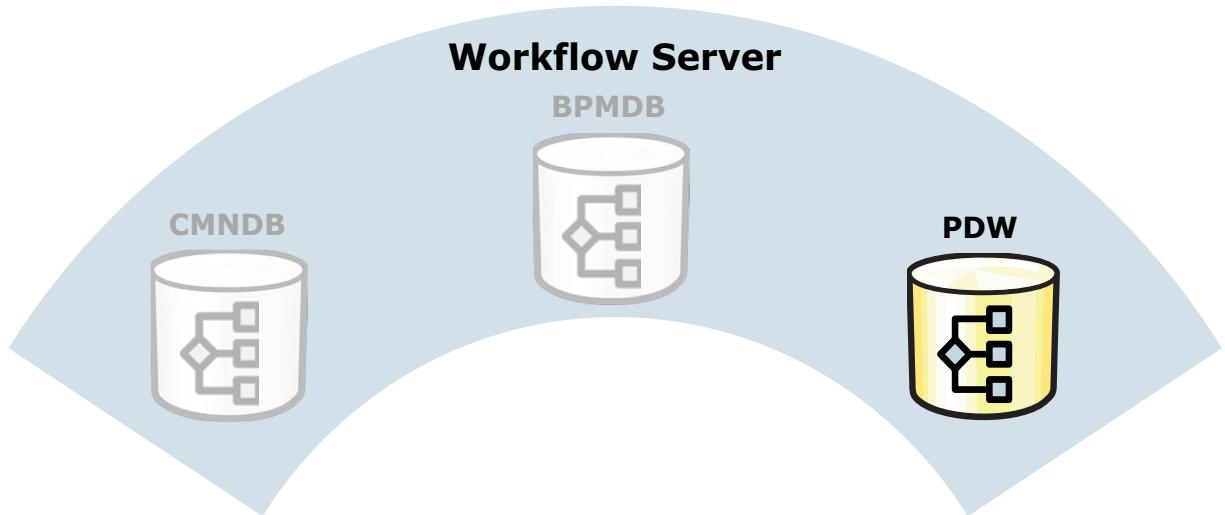
Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-48. The Workflow Server databases

- Every Workflow Server contains a common database (CMNDB), BPM database (BPMDB), and Business Performance Data Warehouse database (PDWDB). Both the BPM database and the common database contain the tables that the server uses to store the process artifacts and run the processes, along with all the other requirements to run the server. The system might need to be queried for data about in-flight instances, and this data comes from these tables. Do not query the tables directly, as the schema might change from version to version. IBM BPM provides a robust set of JavaScript and REST APIs to retrieve the data from these databases.
- Although the data in these databases might contain business data about the process, these databases should not be considered a business data system of record. You should persist the business data to an enterprise system of record during or after the process is complete.
- The databases that are covered here are specific to Business Process Management solutions. The database supporting Case solutions (Content Platform Engine Database or CPEDB) is not displayed.

The Performance Data Warehouse



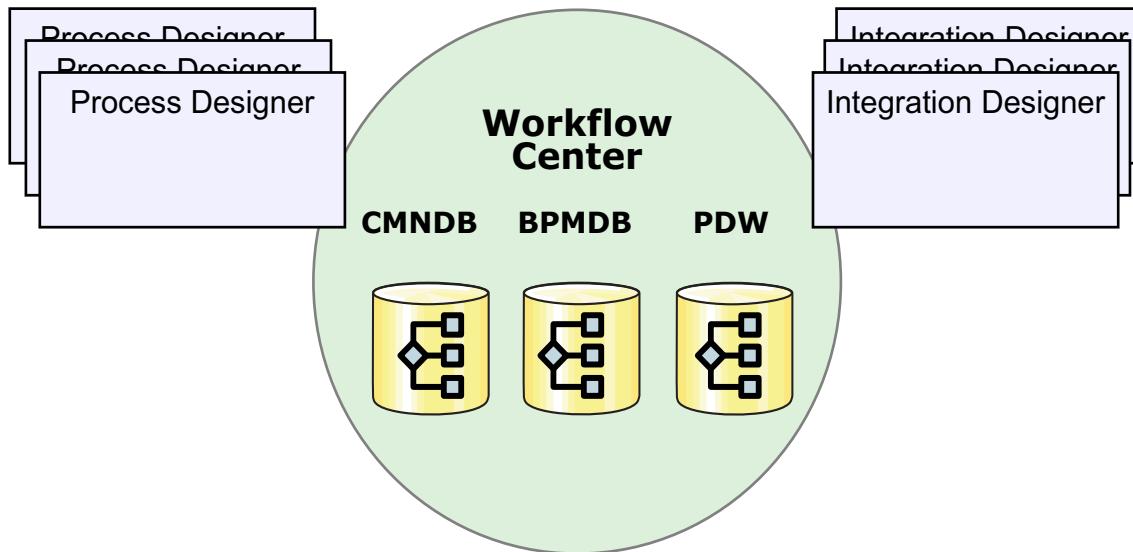
Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-49. The Performance Data Warehouse

- The Performance Data Warehouse collects performance data that represents key business events and metrics when processes are run in the development environment. You can use this collection to examine reports and data in development before a process is tested or put into production. It acts as a system of record for all production process data of an organization. The typical configuration contains a Performance Data Warehouse, which is run as a service on the Workflow Server. For some high-demand organizations, the Performance Data Warehouse can be configured to run on a separate node or cluster to meet the performance needs of the environment as configured by WebSphere.
- The Performance Data Warehouse uses the process model to correlate the business events in real time and aggregate raw performance data into a single database view for reporting and auditing. This database is the system of record for process data for the organization. Developers can create queries against the Performance Data Warehouse database to retrieve information on process performance. Information on total time or average time for a process to complete, and the percentage of processes that follow a certain flow, can be retrieved from the Performance Data Warehouse database.
- IBM Workflow Server automatically installs a Dashboards toolkit for developers to create quick reports with the data from the Performance Data Warehouse. However, third-party reporting packages and environments can integrate with the Performance Data Warehouse to extract process metrics for their own reporting needs.

The Workflow Center: A unique Workflow Server



Introduction to IBM Business Automation Workflow

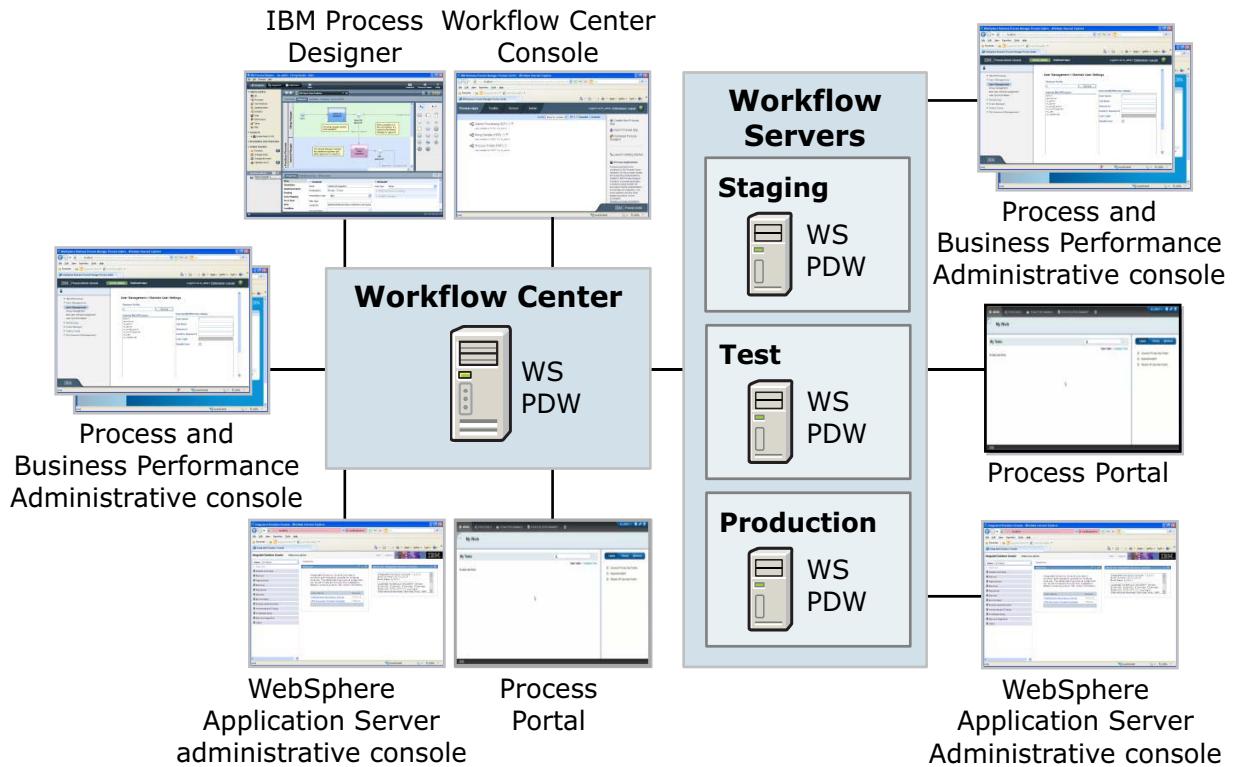
© Copyright IBM Corporation 2020

Figure 1-50. The Workflow Center: A unique Workflow Server

- Every Workflow Center is a Workflow Server with more capabilities. The Workflow Server manages and runs all process model definitions in the development environment. It also stores the versions and history for process development and deployment. The Workflow Server is an application server that is running on WebSphere, and it can run on a single node or cluster to meet the performance requirements of the environment as configured by WebSphere.
- The Workflow Server relies upon a database to store the data for process instances, similar to the runtime environments. The Workflow Center also uses database tables to store and version the process development assets that developers create by using the Process Designer. Because the development occurs on the Workflow Center, it is sometimes called the development environment.
- The Workflow Center is where developers create the toolkits, rules, integrations, and services to implement a solution. Complex process interactions like timers, message events, and error handling are part of the development effort and are implemented in the Workflow Center.
- The Workflow Center contains an integrated Workflow Server, which you can use to run and debug processes as you build them. When you are ready, you can install and run those same processes on the Workflow Servers in your runtime environments.
- The Business Performance Data Warehouse component collects and aggregates process data from processes that run on the Workflow Servers. You can use this data to improve your

business processes. You can use the Process Admin Console to manage the Workflow Servers in your runtime environments (development, test, staging, production) and the Workflow Server that is part of the Workflow Center.

Architecture: Workflow Server tools



Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-51. Architecture: Workflow Server tools

Workflow Servers each contain their own set of these applications (Process and Business Performance administrative console, WebSphere Application Server administrative console, and the Process Portal). These web applications are dedicated to the server that they are associated with.

Users use the runtime Process Portal to work on their process tasks and create instances of the processes necessary to do their jobs. For example, testers use the test environment to create process instances and to see the tasks that are assigned to them. The test environment is independent of the production environment, which can have a different version of the process with production tasks that are running on that server. In this screen capture, there are three Workflow Servers, one for each environment.

1.6. Overview of IBM Cloud Pak for Automation

Overview of IBM Cloud Pak for Automation

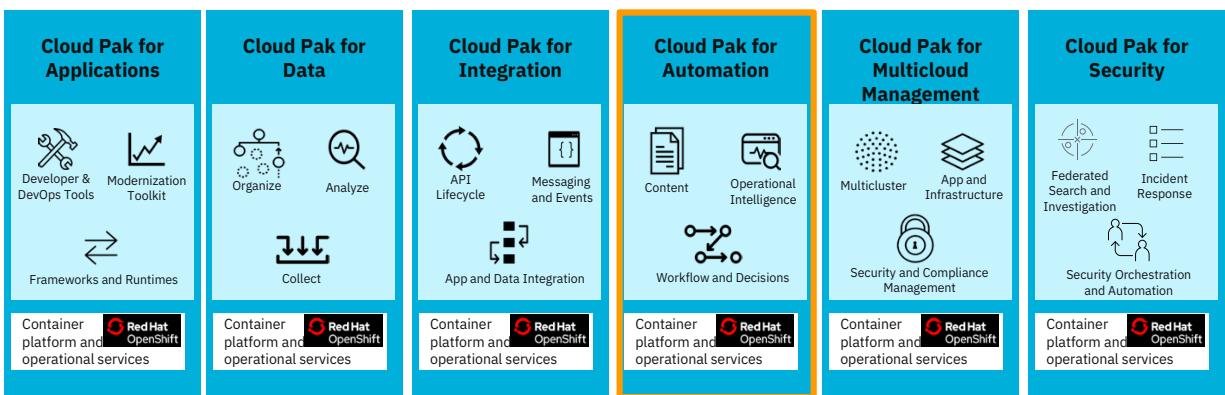
Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-52. Overview of IBM Cloud Pak for Automation

IBM has six Cloud Paks

- IBM Cloud Paks use a common set of operational services by default, such as security and identity services, logging, monitoring, auditing.
- IBM Cloud Paks bring together enterprise software container images that use Helm charts with intelligent defaults for simplified configuration and management and can include extra assets, such as Operators that intelligently manage software at run time.
- Business Automation Studio and Application Designer are components of the Cloud Pak for Automation.



Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

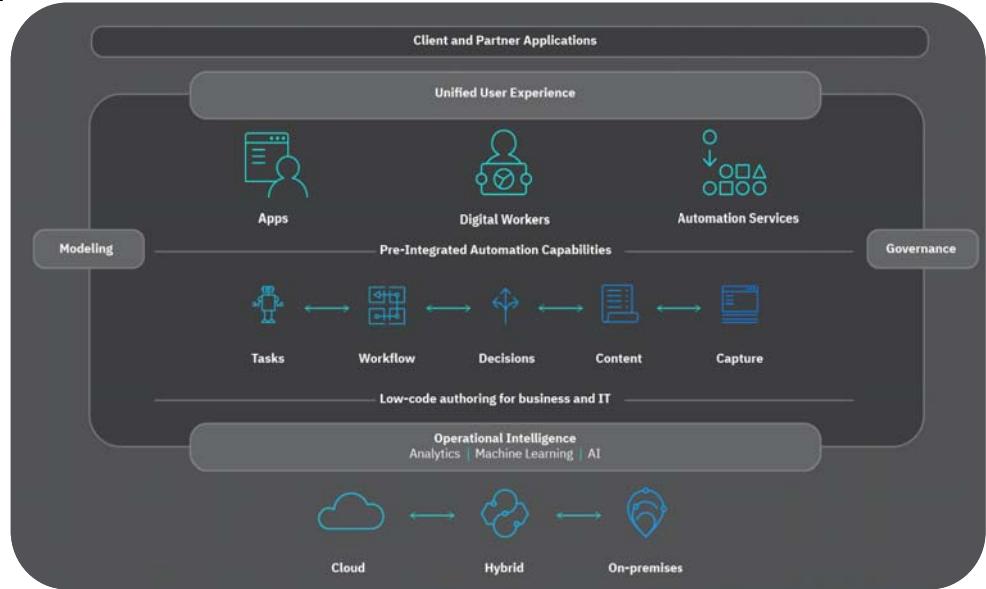
Figure 1-53. IBM has six Cloud Paks

IBM Cloud Pak for Automation is one of six IBM Cloud Paks that cover key workloads.

IBM Cloud Pak for Automation is one of six IBM Cloud Paks that cover key workloads.

IBM Cloud Paks

- IBM Cloud Paks are enterprise-grade containerized software combining container images with enterprise capabilities for deployment in production use cases with integrations for management and lifecycle operations.



Introduction to IBM Business Automation Workflow

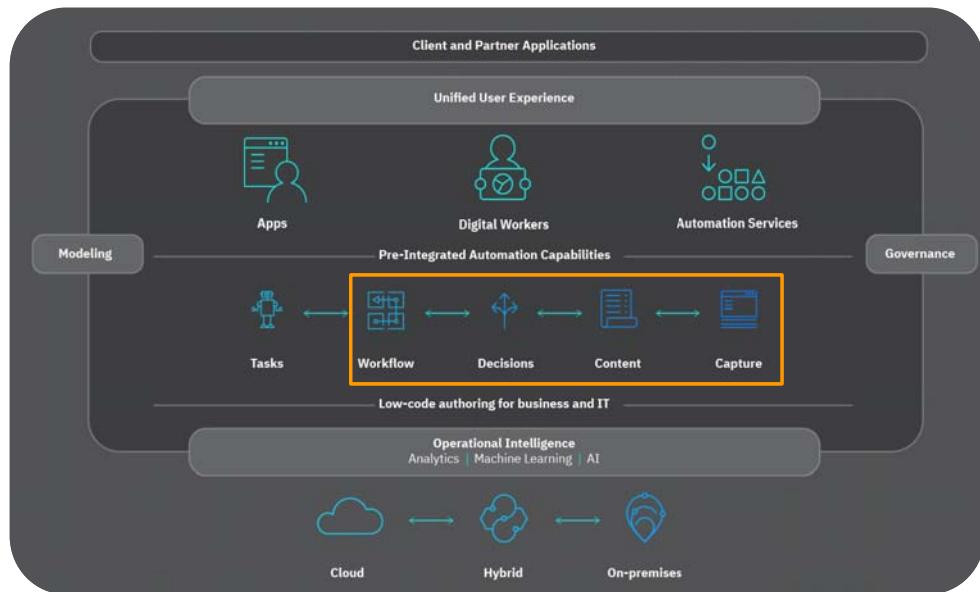
© Copyright IBM Corporation 2020

Figure 1-54. IBM Cloud Paks

IBM Cloud Paks are enterprise-ready, containerized software solutions that give clients an open, faster and more secure way to move core business applications to any cloud. Each IBM Cloud Pak includes containerized IBM middleware and common software services for development and management, on top of a common integration layer.

IBM Cloud Pak for Automation

- IBM Cloud Pak for Automation offers design, build, run, and automation services to rapidly scale your programs and fully run and operationalize an automation strategy.



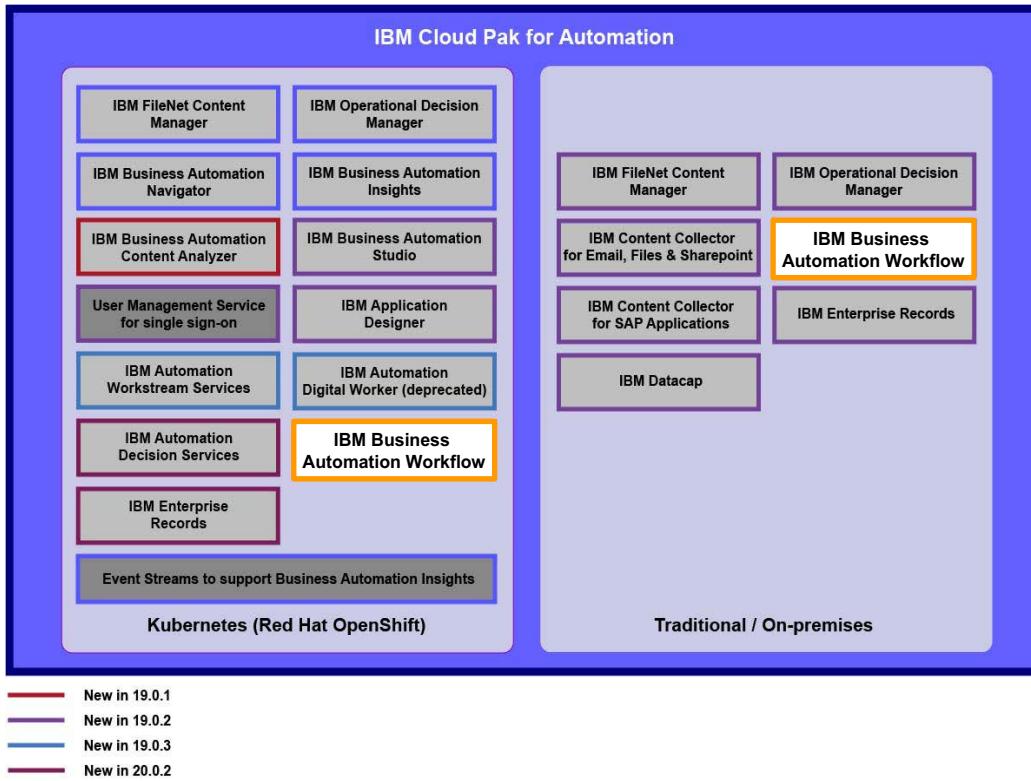
Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-55. IBM Cloud Pak for Automation

Cloud Pak for Automation includes certified container images that can be deployed together to automate processes, protect your business, and extract critical information from your business data. You can provision services on an on-demand basis and shut them down when you no longer need them.

IBM Cloud Pak for Automation capabilities



Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-56. IBM Cloud Pak for Automation capabilities

The diagram shows the capabilities of IBM Cloud Pak for Automation and the core program offerings that the platform provides. Each application can be installed independently of each other. However, to integrate certain components it might be necessary to install them in a specific order.

IBM Business Automation Workflow is a component of the IBM Cloud Pak for Automation.

In this course, you use the on-premises version of IBM Business Automation Workflow.

IBM Business Automation Workflow on Cloud

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

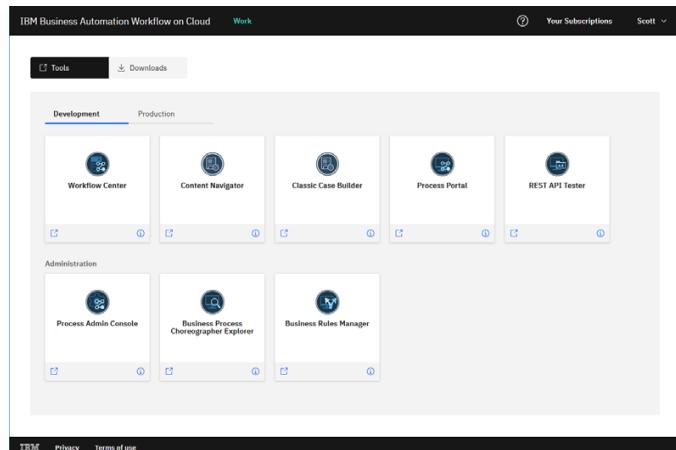
Figure 1-57. IBM Business Automation Workflow on Cloud



IBM Business Automation Workflow on Cloud (1 of 2)

- Enterprise-grade IBM Business Automation Workflow on cloud service for development, testing, and production Software as a Service implementation
- Ready-to-use development, test, and production environments are available
- Cloud-based, collaborative, and role-based environment
- Available exclusively on IBM Cloud infrastructure
- Monthly subscription plans
- Managed by IBM
- Free 30-day trial available

Access an array of tools in the different environments from the home page



Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-58. IBM Business Automation Workflow on Cloud (1 of 2)

IBM Business Automation Workflow on Cloud is a subscription-based cloud service.

IBM Business Automation Workflow on Cloud provides a full lifecycle IBM Business Automation Workflow environment that includes development, test, and production – with tools and runtime for process design, execution, monitoring, and optimization. It is designed to enable business users to get started with process improvement quickly without the need to build and maintain an IT infrastructure.

IBM Business Automation Workflow on Cloud (2 of 2)

IBM manages:

- Uptime
- Monitoring
- Backup
- High availability
- Disaster recovery
- Updates
- Maintenance



Customers manage:

- Application development
- Application integration
- Application support



Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-59. IBM Business Automation Workflow on Cloud (2 of 2)

IBM Business Automation Workflow on Cloud is a Software as a Service (SaaS) implementation. Customers can get started quickly without the need to build any infrastructure.

With IBM Business Automation Workflow on Cloud, IBM manages the infrastructure typical of a SaaS solution allowing customers to focus on application development, support, and integration.

Unit summary

- Understand the key capabilities of IBM Business Automation Workflow
- Describe how to use IBM Business Automation Workflow to accomplish process modeling goals
- Explain how to create process applications in the Workflow Center
- Explain how to design process models by using the IBM Process Designer
- Describe the purpose of the Process Portal, Process Designer, and the Workflow Center repository
- Describe the high-level architecture for IBM Business Automation Workflow
- Explain the benefits of using IBM Business Automation Workflow on Cloud

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-60. Unit summary

Review questions

1. True or False:
IBM Business Automation Workflow covers the full spectrum of Business Process Management and Case Management solutions.
2. True or False:
Users complete their tasks through the Workflow Center.
3. A process application is:
 - A. The model representation of the process
 - B. A container for process models and their supporting implementations
 - C. A human service implementation representing a structured process
 - D. None of the above
4. The following are components of IBM Business Automation Workflow (select all that apply):
 - A. Process Portal
 - B. Workflow Center
 - C. Process Designer
 - D. Data manager



Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2020

Figure 1-61. Review questions

Review answers

1. **True.**
2. **False:** Users complete their tasks through the Process Portal
3. **B:** A process application is a container for process models and their supporting implementations, and it is stored in the repository
4. **A, B, C**



Exercise: Creating your first process application

Introduction to IBM Business Automation Workflow

© Copyright IBM Corporation 2019

Figure 1-63. Exercise: Creating your first process application

Exercise introduction

- Start IBM Business Automation Workflow
- Create a process application in Workflow Center
- Use the Process Admin console and other tools to aid in troubleshooting
- Export and Import process applications



[Introduction to IBM Business Automation Workflow](#)

© Copyright IBM Corporation 2019

Figure 1-64. Exercise introduction

Unit 2. Introduction to Business Process Management

Estimated time

01:30

Overview

This unit explains the foundational concepts that establish the importance of process modeling. It includes a review of Business Process Management, the Business Process Management lifecycle, the basics of process modeling, the Playback methodology, and Business Process Management project development.

How you will check your progress

- Review

Unit objectives

- Understand the process spectrum
- Define Business Process Management (BPM)
- Define process modeling
- Understand the high-level project phases of a BPM project
- List and describe the phases in the IBM Playback methodology
- Describe Playback 0 and the achievements that are reached at this stage in the Playback methodology

Topics

- Business process management (BPM)
- Introduction to process modeling
- Business Process Management project development
- The Playback methodology
- Playback 0

Introduction to Business Process Management

© Copyright IBM Corporation 2020

Figure 2-2. Topics

2.1. Business Process Management (BPM)

Business Process Management (BPM)

Introduction to Business Process Management

© Copyright IBM Corporation 2020

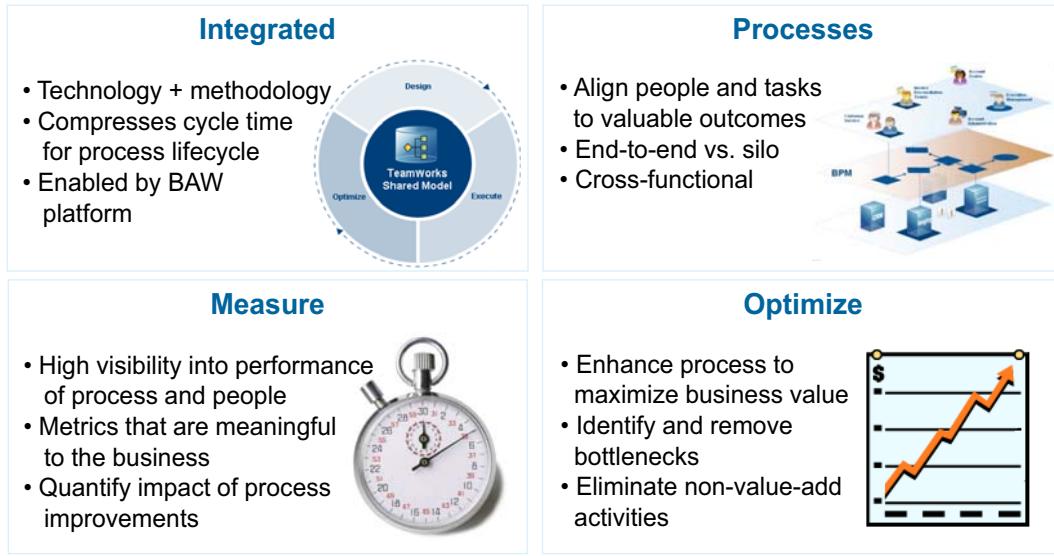
Figure 2-3. Business Process Management (BPM)

Organizations often seek ways to improve their organization to increase productivity, lower costs, and increase revenues. The challenge organizations face is that change is inevitable in business because of various factors, like market dynamics. To keep up with the climate of change in business, organizations must rely on efficient and effective business processes.

Many organizations try to implement different strategies to accomplish change management of processes with little to no disruption of customer service and employee productivity. But many times, those efforts fail outright or accomplish only a portion of the process improvement because performance measurement is limited.

Business Process Management is about value

- **BPM Defined:** An integrated approach to aligning the key activities of an organization into processes you can consistently measure to optimize value to your organization and its end customers.



Introduction to Business Process Management

© Copyright IBM Corporation 2020

Figure 2-4. Business Process Management is about value

Business Process Management is about value, not technology. BPM spans both technology and methodology. By itself, BPM is not a technology.

Business Process Management includes the services and tools that support process management (for example, process analysis, definition, processing, monitoring and administration), including support for human and application-level interaction. BPM tools can eliminate manual processes and automate the routing of requests between departments and applications.

What is Business Process Management?



Comprehensive change management of business processes that results in continuous process improvement

Introduction to Business Process Management

© Copyright IBM Corporation 2020

Figure 2-5. What is Business Process Management?

Business process management (BPM) excels at providing a comprehensive change management of business processes, which results in continuous process improvement.

Processes are meant to evolve as the organization or external conditions change.

Three themes

Goal	System	Results
The BPM goal is efficient and effective business processes with visibility	The BPM system is the management of people-to-people work steps, system-to-system communications, or person-to-system interactions	The BPM expected result is process improvement that brings about financial benefits and customer and employee satisfaction

Introduction to Business Process Management

© Copyright IBM Corporation 2020

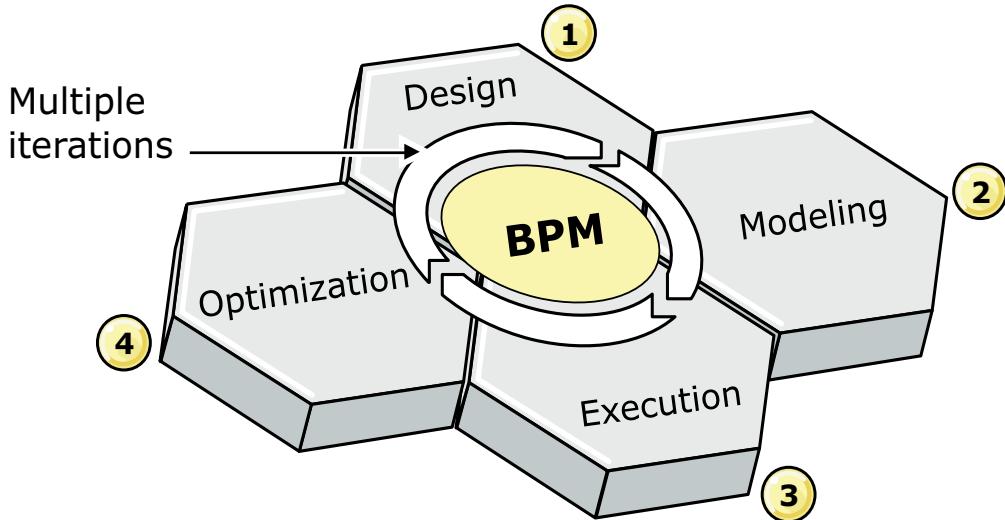
Figure 2-6. Three themes

BPM is also described with these common themes: the goal, the system, and the expected results.

All organizations have processes; however, the visibility of processes can be varied. BPM is a way to increase that visibility and hence give direction to the continued efficiency of the processes.

In true BPM, all aspects of a system are important, including human interactions. True BPM seeks to define and visualize all aspects of your process regardless of what role or system is conducting that part of the work. BPM results in continual process improvement, which provides many beneficial outcomes to the client.

BPM lifecycle



[Introduction to Business Process Management](#)

© Copyright IBM Corporation 2020

Figure 2-7. BPM lifecycle

The BPM lifecycle has four phases: design, modeling, execution, and optimization.

Looking at the BPM lifecycle, it becomes apparent that opportunities are available to use the expertise of business and IT to collaborate in each phase of the lifecycle. Using this approach to BPM, the business process is stable and on target. This stability is because of the overall iterative improvement cycles in keeping up with business goals, business change, and opportunities within each phase to make critical adjustments.

Business and IT working in concert throughout the BPM lifecycle require a clear set of goals for each phase. Matched against those lifecycle phase goals are the responsibilities for each group. Clearly, the governance of the business process varies at each phase for each group, but the involvement of both ensures that the process improvement is realized.

The goals of each phase are as follows:

1. Design goals:

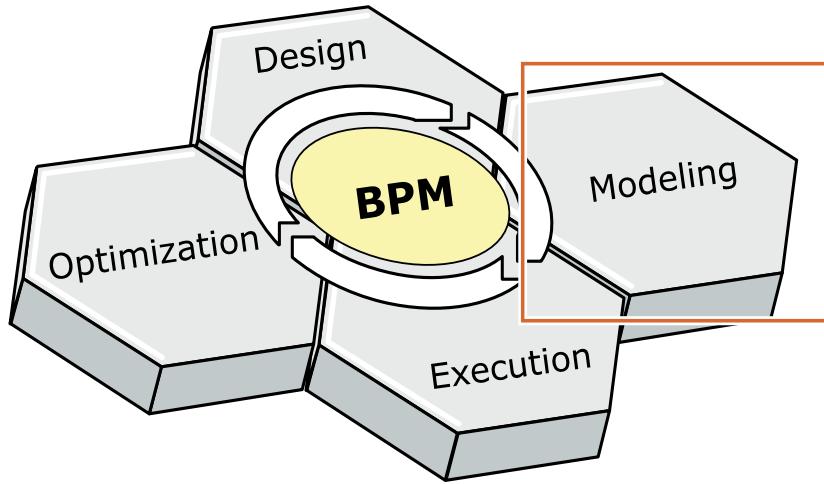
- Capture executive vision
- Process nomination
- Process prioritization
- Process discovery

- Process analysis
2. Modeling goals:
- Create a process model
 - Process adjustments
 - Process simulation
3. Execution goals:
- Implement the process model as a process application
 - Adjust business process requirements as needed
 - Deploy and monitor the process application
4. Optimization goals:
- Analyze and evaluate process performance data
 - Evaluate the business process ability to meet new business goals

After business practices or external conditions change, or the current process is no longer optimal, BPM iterates again through the lifecycle. This continual iteration allows the effective management of business processes.

A true BPM implementation tool allows these iterations to be easily applied.

The modeling phase



- Process discovery and documentation
- Process analysis
- Simulation and adjustment

[Introduction to Business Process Management](#)

© Copyright IBM Corporation 2020

Figure 2-8. The modeling phase

The modeling phase is more than just creation of the process model; it also requires an understanding of how to adjust the model to meet evolving business requirements. So, throughout this phase, the process model goes through continued analysis and a series of adjustments and refactoring efforts to obtain a model that can be implemented into a process application.

All the adjustments and testing allow for a process model that meets what the business expects in terms of an improved and efficient business process at the end of project development.

BPM done correctly results in business processes that are modeled, analyzed, and adjusted early and often. The BPM effort goes far beyond basically applying technology to a process to yield a changed process. Applying technology to automate a bad business process without regard to necessary analysis and adjustment efforts leads only to a more efficient, but still bad process.

2.2. Introduction to process modeling

Introduction to process modeling

Introduction to Business Process Management

© Copyright IBM Corporation 2020

Figure 2-9. Introduction to process modeling

An understanding of a business process is necessary before it can be modeled.

What is a business process?



- A set of activities that takes specific inputs and converts them into specific outputs in a defined, predictable fashion

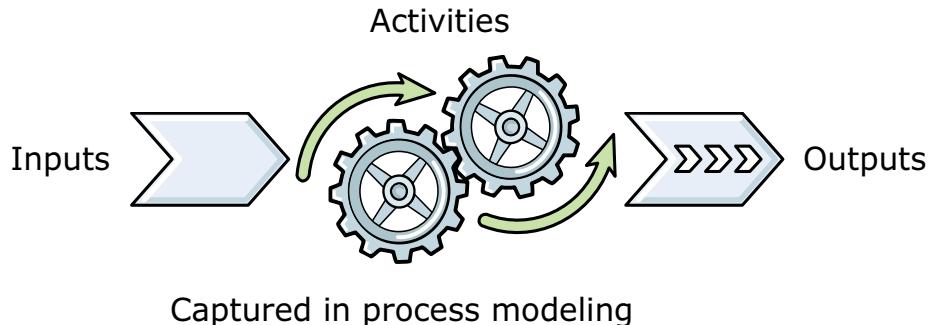
Introduction to Business Process Management

© Copyright IBM Corporation 2020

Figure 2-10. What is a business process?

- Every organization uses business processes to accomplish work. A *business process* is a set of business activities that represent the required steps to achieve a business objective. For example, you might have a business process that handles credit card disputes. In this case, the business objective is to resolve the dispute in an efficient and accurate way to minimize cost to your organization and to retain customer satisfaction. The process itself includes all of the steps that are needed to meet the objective (in this case, it might be activities like receiving the claim, examining the validity of the claim, deciding whether to remove the charge, and informing the customer of the decision).
- Business processes often require a combination of internal activities and activities that must be performed by humans. Therefore, we can look at Business Process Management as the intersection between people, processes, and technology.

What is process modeling?



- Captures the ordered sequence of the business process tasks or activities

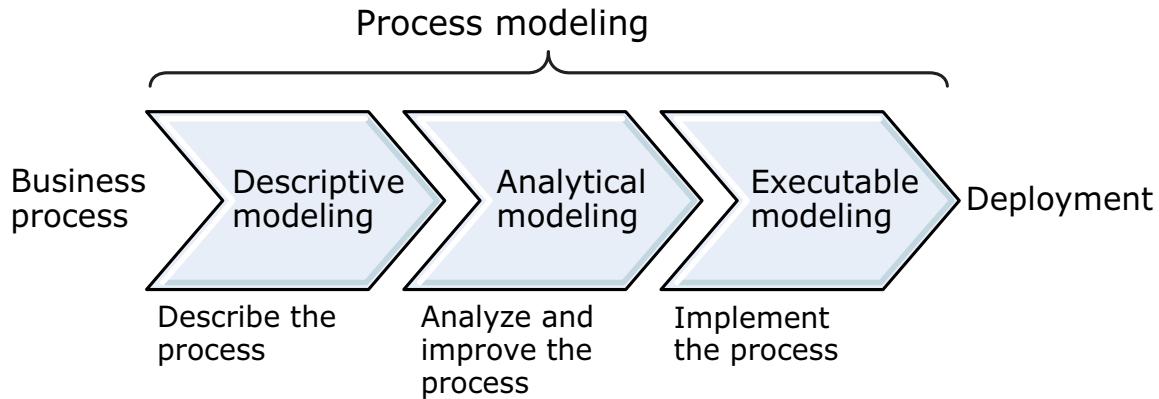
Introduction to Business Process Management

© Copyright IBM Corporation 2020

Figure 2-11. What is process modeling?

Process modeling captures the ordered sequence of the business process tasks or activities, and the responsible roles that are conducting the activities. It also captures the conditional branching and the sequencing of the flow of work between activities, along with the supporting information from start to end.

Three-phase approach



Introduction to Business Process Management

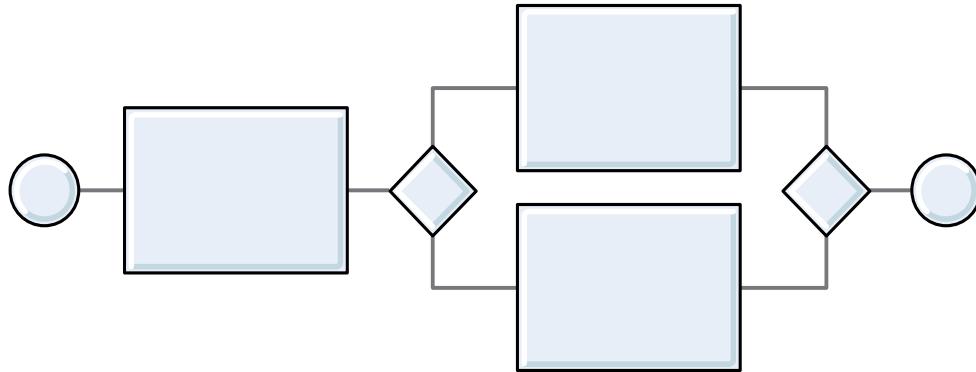
© Copyright IBM Corporation 2020

Figure 2-12. Three-phase approach

Process modeling can be described as having a three-phase approach:

- **Phase 1 or descriptive modeling:** Describe the process. This high-level model provides a description of the process that is based on business requirements. The model is easily communicated across the organization.
- **Phase 2 or analytical modeling:** Analyze and improve the process. This analytical, more detailed modeling shows all pertinent activities and flow that are used to detail process requirements.
- **Phase 3 or executable modeling:** Implement the process. This model details the functional requirements to implement the executable process application.

What is a process model?



A graphical representation, or diagram, of the business process that is universally understood and easily communicated

[Introduction to Business Process Management](#)

© Copyright IBM Corporation 2020

Figure 2-13. What is a process model?

A good process model is a graphical representation or diagram of the business process that is universally understood. Business people understand it easily, and it is directly implemented in a Business Process Management system (BPMS) such as IBM Business Automation Workflow.

For all parties to understand a process model universally, process owners, process participants (business), and the BPM development teams must easily understand each other and recognize the same concepts in the same context. IT does not need to redraw a process model to provide more clarity or a different point of view.

A good process model provides views into a process that are clearly and easily communicated in 5 minutes or less, at every level of granularity.

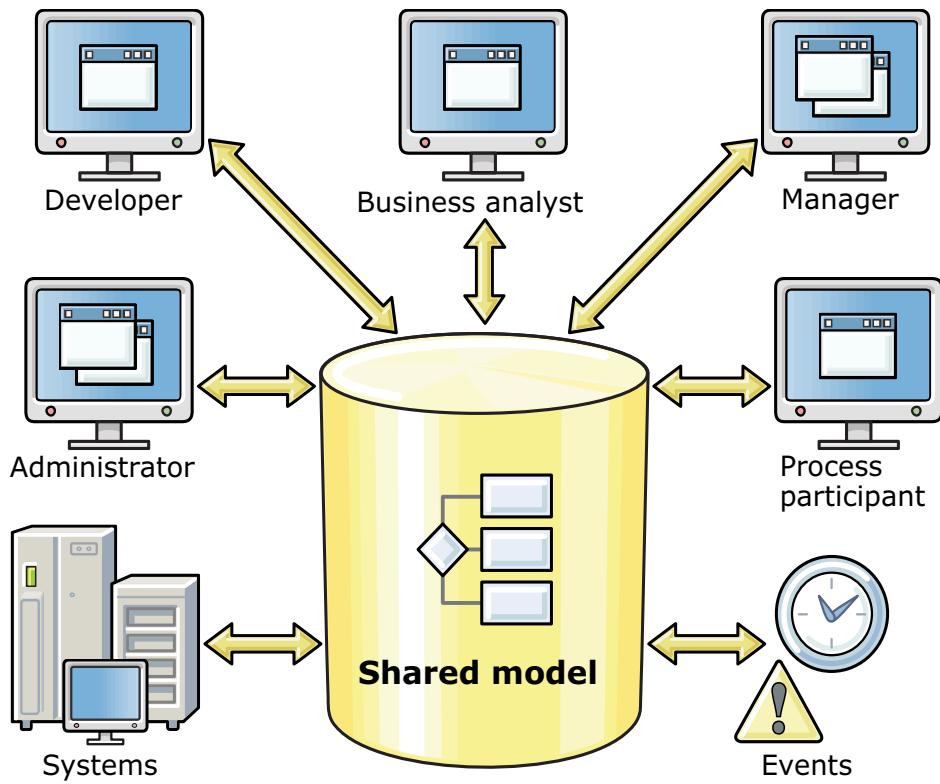
Process model development

Now that it is established when and what must be accomplished in process modeling, the next thing is to focus on is how process modeling is accomplished in terms of development methods.

Understanding how to model a process requires comprehension of the project development methods that are used for BPM. Realize that project development strategies for process models differ from standard methods, especially when you consider the usage of process models. In standard project development, the shelf life of diagrams that are derived from requirements is only from the business hand-off to the development teams. In essence, these requirements are converted into code, and their use ends at that point.

BPM process models are different. Going back to the BPM lifecycle, notice that the process model evolves in terms of usage. The prolonged shelf life of a process model allows for the iterative BPM lifecycle because it is data-driven and not code-driven. This setting provides the stability for a process application without fear of having to start from scratch when change is needed.

The right process model development strategy



Introduction to Business Process Management

© Copyright IBM Corporation 2020

Figure 2-14. The right process model development strategy

IBM Business Automation Workflow uses a single shared environment for project design and development. All process artifacts are stored in a single shared model architecture. All parties that are involved in the effort to define, model, implement, measure, and improve the process are working from a common shared model that encapsulates all of the various components. It helps maintain the vision of bringing business and IT together.

The following list shows various people who are all using the same process model:

- Business analysts who are modeling the process
- The IT developers who are constructing the detailed implementation of the model
- The responsible process participants who are completing their activities in the process
- The process owner and analysts who monitor the process performance and identify improvements

The model of the process that the analysts and developers build is the same one that completes at run time. It is the same one that is used to create reports on the performance and status of the process, and the same one that is used to implement process improvements. More details regarding BAW architecture are covered in the next Unit.

2.3. Business Process Management project development

Business Process Management project development

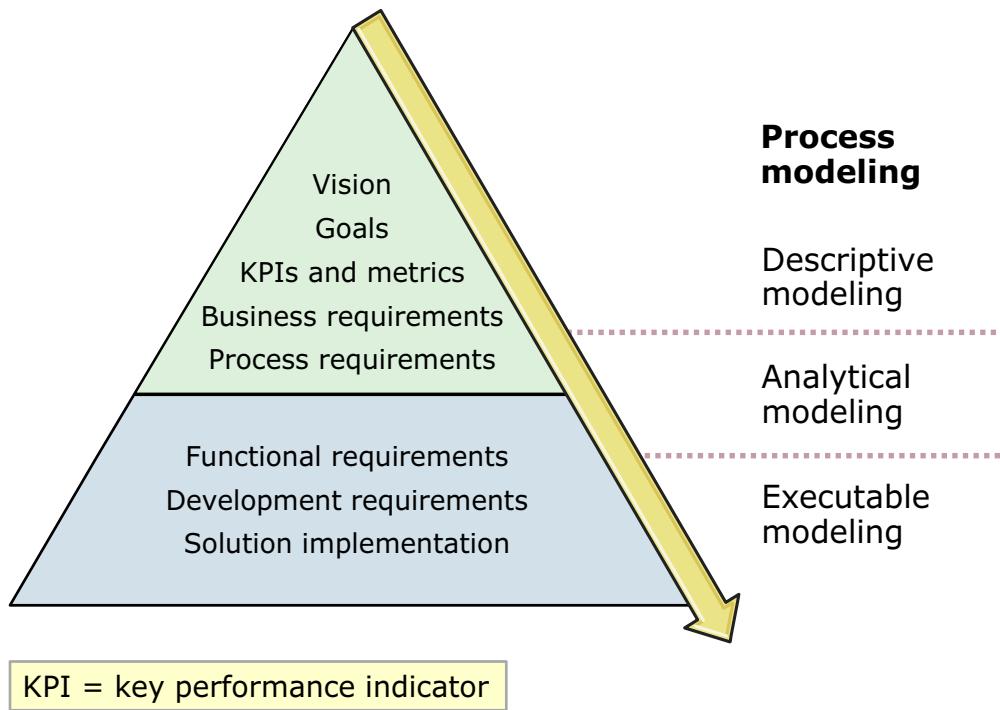
Introduction to Business Process Management

© Copyright IBM Corporation 2020

Figure 2-15. Business Process Management project development

The established standards and development methods typically reinforce the project development for any IT initiative. A BPM project, especially one that includes the broader vision and definitions that are provided, would not fit the typical project development standards because the key BPM project components are slightly different. This difference is because the BPM project components reinforce the process first, and the solution second. In BPM, everything is process-driven, so the development method that is used to develop process models and the eventual process application must be uniquely focused on the process needs. It is not a system that is traditionally used to develop projects.

Business Process Management project components



Introduction to Business Process Management

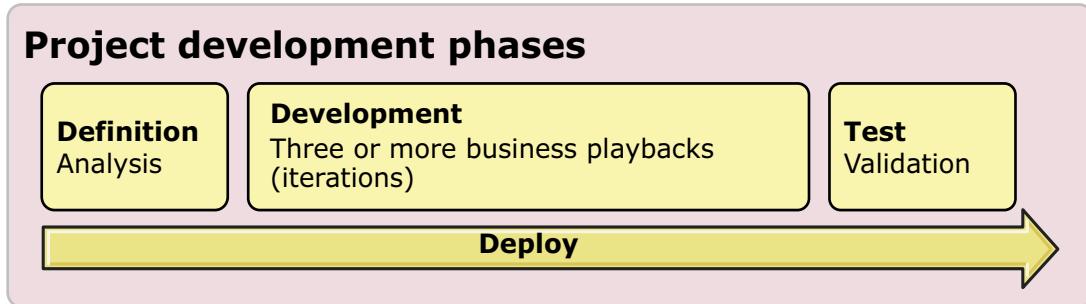
© Copyright IBM Corporation 2020

Figure 2-16. Business Process Management project components

The top-down diagram view of the BPM components provides a quick view of how a typical BPM project development evolves. Any of these components that are missing from a project would interrupt the effective design, definition, and creation of the process application, and curtail the engagement of business people.

KPI = key performance indicator

Project development phases



- **Definition:**
 - Discover and define the process
 - Analyze the process for improvements, model the process, and set the process performance measurement criteria
- **Development:**
 - Prepare the process application for deployment by using an iterative development with three or more playbacks
- **Test:**
 - Validate the process application performance in achieving expected business process goals in a production user environment

Introduction to Business Process Management

© Copyright IBM Corporation 2020

Figure 2-17. Project development phases

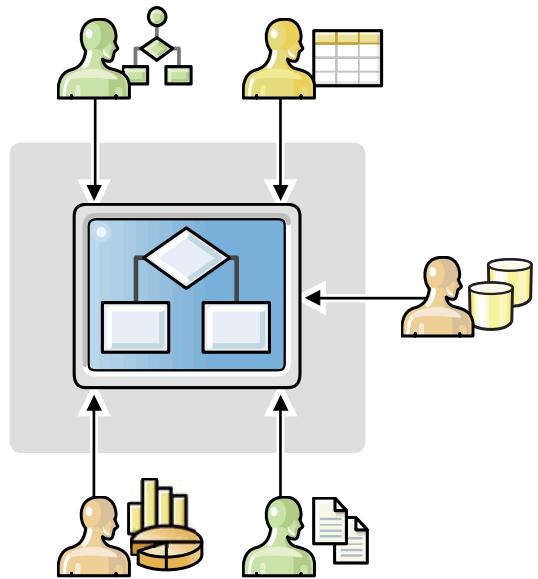
In contrast to the traditional IT application development approach stands the phased BPM project development approach that closely aligns to the overall BPM lifecycle. This approach also focuses on the BPM project components and allows for the different phases of process modeling.

The project development phases are:

- **Definition:** Discover and define the process. Analyze the process for improvements, model the process, and set the process performance measurement criteria.
- **Development:** Prepare the process application for deployment with an iterative development with three or more playbacks.
- **Test:** Validate the process application performance in achieving expected business process goals in a production or user environment.

Business Process Management project teams

- Process sponsor
- Process owner
- BPM project manager or program manager
- Subject matter experts
 - Core process activities
- Core team members
 - Analyst
 - Developer
 - Solution architect
- Administrators
- Facilitators
 - Ad hoc member (optional)



[Introduction to Business Process Management](#)

© Copyright IBM Corporation 2020

Figure 2-18. Business Process Management project teams

The unique phases and components of a BPM project require a specific set of project roles, including:

- **Process sponsor:** Responsible for establishing the project goals and scope, securing organizational support and resources, and ensuring alignment with organizational business goals
- **Process owner:** The person who is accountable for the successful execution of the process, knows the process from end to end at a high level, and can identify the subject matter experts
- **BPM project or program manager:** The person responsible for the success of the project
- **Subject matter experts:** People with knowledge of specific process resources, or systems
- **Core team members:** Business process management (BPM) development teams, typically including BPM analysts, BPM developers (includes integration designer developers and technical consultants), and solution architect (advanced role that can lead teams and serve as an analyst and developer)
- **Administrator:** Installs, updates, and configures the Business Process Management system
- **Facilitator:** (optional) Typically manages the collaboration meetings for a BPM team

All of these participants work together to collaborate in the design of the process model.

2.4. The Playback methodology

The Playback methodology

Introduction to Business Process Management

© Copyright IBM Corporation 2020

Figure 2-19. The Playback methodology

- A Playback between business and IT is a focused demonstration of a partially implemented process model at the designated development phase. This phase includes the goal of discussion, consensus building, collaborative improvement, and ultimate approval of the process model. Playbacks enable the iterative development of the process application.
- Playbacks provide early visibility and input from the business group on process application functions. The perspective of the business group at the early stages benefits development because the business quickly identifies adjustments to requirements well before the final product is implemented. The ability to shift direction during the definition and development phase is key to reaching the ultimate BPM project target.
- The Playback methodology is based on agile programming principles and techniques. IBM Business Automation Workflow was created from the ground up as an agile software development tool. IBM's clients sometimes attempt to use IBM Business Automation Workflow with other methodologies. However, it repeatedly happens that the most successful projects combine the right people (business and IT), agile software development practices, and IBM Business Automation Workflow to achieve their project and process goals.

What is a Playback?

- A Playback is a focused demonstration of a partially implemented process application, which is delivered to the business and IT communities for discussion, consensus-building and approval.
 - Iteration plan defines the set of user stories to be demonstrable
 - The business users should run the Playback
 - Each Playback provides validation that the solution is headed in the correct direction
 - Fosters business ownership, expectations, and sponsorship of the solution.
 - Create questions and suggestions that feed into subsequent playbacks.
- With each milestone reached and each Playback exercise that is completed, the business participants play a stronger role in the development.

Introduction to Business Process Management

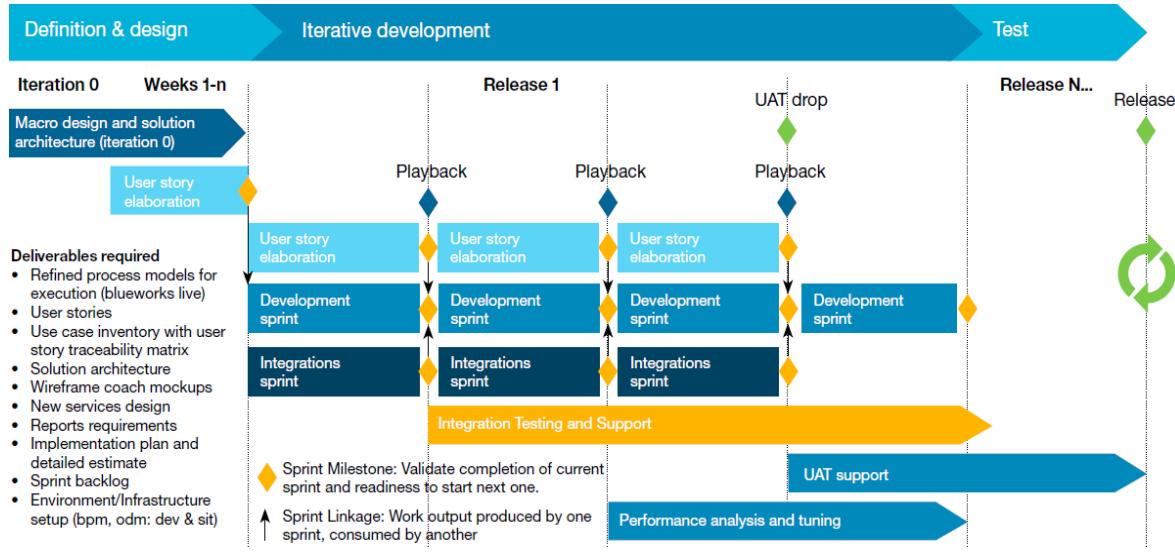
© Copyright IBM Corporation 2020

Figure 2-20. What is a Playback?

During a Playback for business stakeholders, users, and development, the process model is examined at a designated development phase to establish process goals, achieve consensus, collaboratively improve the process model, and, ultimately, approve the process model. By using playbacks, you can develop a process application iteratively.

Playback methodology

- Implementing a Business Process Management program is best performed in an iterative delivery cycle.
- IBM Business Automation Workflow supports playbacks to verify requirements often.



Introduction to Business Process Management

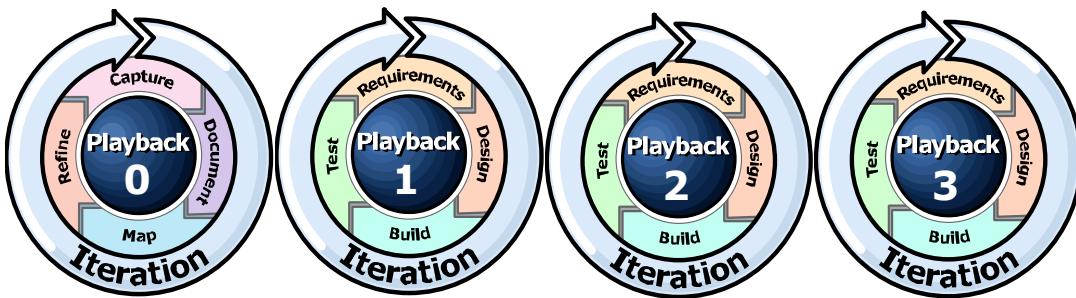
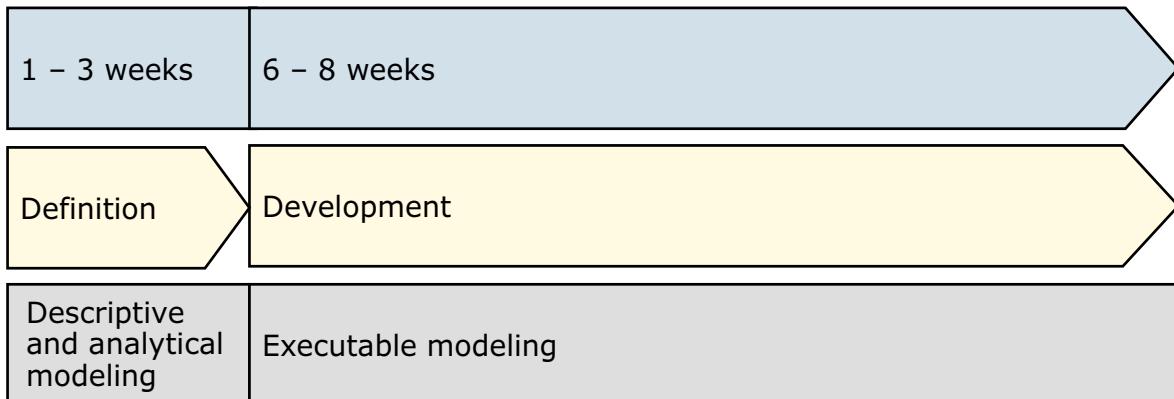
© Copyright IBM Corporation 2020

Figure 2-21. Playback methodology

- The Business Process Management approach is iterative; you don't implement it once, never to be touched again. Instead, you design, model, create, simulate, monitor, and optimize your processes on a regular basis. The feedback that you receive from testing and monitoring your processes drives continuous improvements to your organization's workflows.
- Iteration of your process is one critical key for BPM success.
- An “iteration” of a Playback allows users to focus on a part of the overall project and validate it before moving on to the next part.
- Most playbacks allow users to gather requirements, design and build solutions, and then validate (test) them. When the Playback is validated, you move to the next area of focus until this part of the project is complete.
- A Playback is a focused demonstration to clarify understanding and elicit consensus. An example of a Playback is an event where stakeholders come together and view the completed work up until that point. A business person conducts the Playback event by stepping through the completed work to verify that it meets the goals of this phase of the development process.
- “Playback” is also used when describing the phase of the project. As you see in the slide, the project lifecycle is split into multiple “playbacks.” These playbacks would be a phase of the project, and the culmination of that phase would be a final Playback of the completed code to

the stakeholders. Because each Playback event might require changes to the code to meet the requirements of the Playback phase, every Playback phase has multiple Playback events. This slide covers the Playback phases.

Playback cycles



Introduction to Business Process Management

© Copyright IBM Corporation 2020

Figure 2-22. Playback cycles

Often playbacks are conducted as themed stages. The number of actual collaboration meetings that are conducted inside of each stage can vary. Usually, multiple smaller playbacks are conducted and target individual groups with a specialized role (developers, administrators, and other roles), but these smaller playbacks build upon one another, leading to a final Playback. During this final Playback, make sure that you reach consensus before moving into the next stage.

This slide is an example of a project plan that contains a typical Playback timeline and content of each of the playbacks. In modeling and implementation, shorter cycles than what your company might currently be accustomed to are good practices. The agile development approach with shorter cycles is critical to BPM success. Larger projects are scoped down to smaller release cycles. Creating smaller chunks for a project has many benefits, as follows:

- Overall project risk is reduced.
- Working code can be released into production in a shorter amount of time.
- Changes to the project cause less rework when releases occur on a more regular basis.

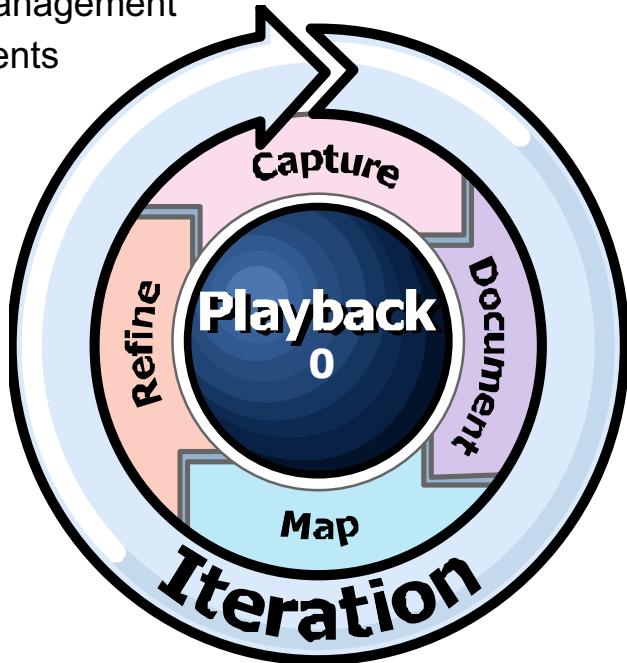
The following examples show what happens during these playbacks in the suggested project plan:

- Playback 0: Focus on high-level business process understanding and building consensus

- Process discovery, as-is model, process analysis, ending with a “to-be” model ready for implementation
- Playback 1: Focus on user interface design and implementation
 - Data model and process flow implementation
 - Human service or coach design, and data mapping
 - Business rules, process flow control implementation
- Playback 2: Focus on integrations
 - Creating a decision service
 - Implementing message events
- Playback 3: Focus on consolidation of the previous themes and producing an end-to-end solution
 - Handling errors in services
 - Deploying process applications

Playback 0

- Review business requirements & business process
- Define components and code management
- Review non-functional requirements
- Assess process characteristics
- Build Services catalog (50%)
- Assess caching needs
- Define system context
- Assess fit to purpose
- Select technology
- Define data model
- Define security approach
- Define naming convention



Introduction to Business Process Management

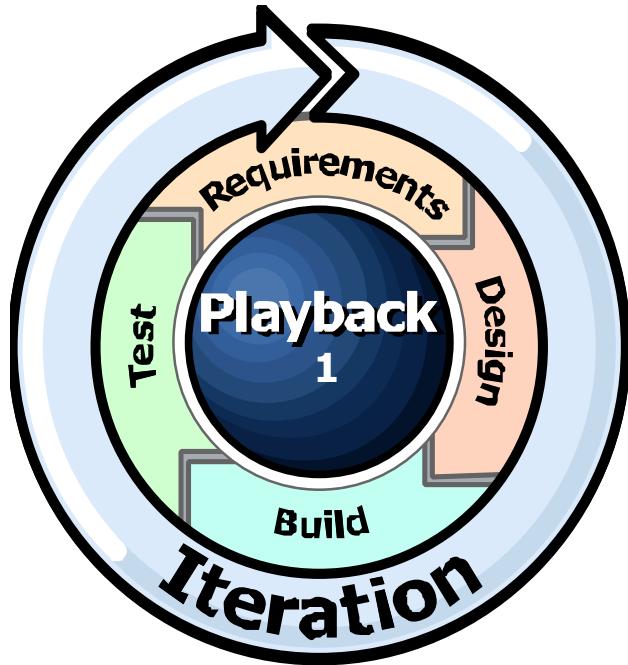
© Copyright IBM Corporation 2020

Figure 2-23. Playback 0

- Playback 0: Focus on high-level business process understanding and building consensus
 - Process discovery, as-is model, process analysis, ending with a “to-be” model ready for implementation
- This slide and the next three provide more details around the focus of each Playback. As development progresses through the playbacks, you want to verify artifacts in a specific order. Because the development process is iterative, it's possible to have to revisit playbacks to revise requirements.

Playback 1

- Review process definition
- Finalize logical architecture
- Address technical risk with Proof of Concept
- Address high availability
- Prepare performance architecture and test strategy
- Complete Services catalog (100%)
- Document architecture decision



Introduction to Business Process Management

© Copyright IBM Corporation 2020

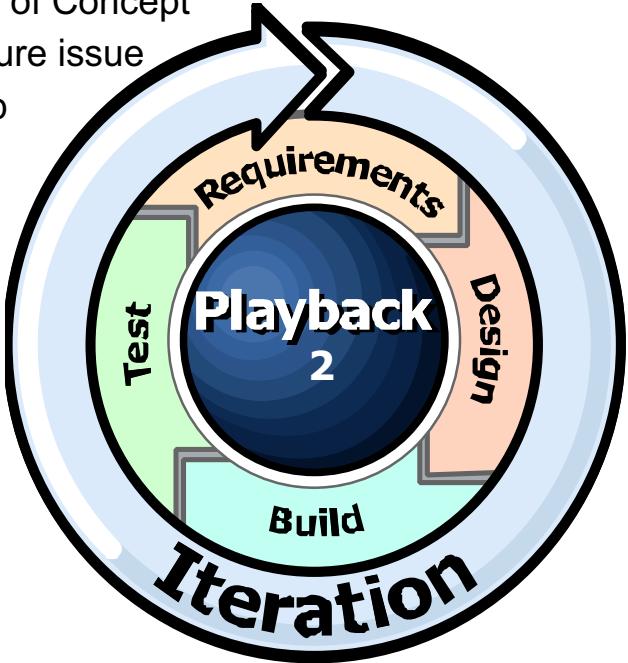
Figure 2-24. Playback 1

Playback 1: Focus on user interface design and implementation

- Data model and process flow implementation
- Human service or coach design, and data mapping
- Business rules, process flow control implementation

Playback 2

- Prepare non-functional requirements and performance testing
- Address technical risk with Proof of Concept
- Address any remaining architecture issue
- Technical tradeoff and leadership
- Design reusable logic, services
- Complete architecture decision
- Define Error Handling
- Enforce simplicity
- Ensure sustainable solution



Introduction to Business Process Management

© Copyright IBM Corporation 2020

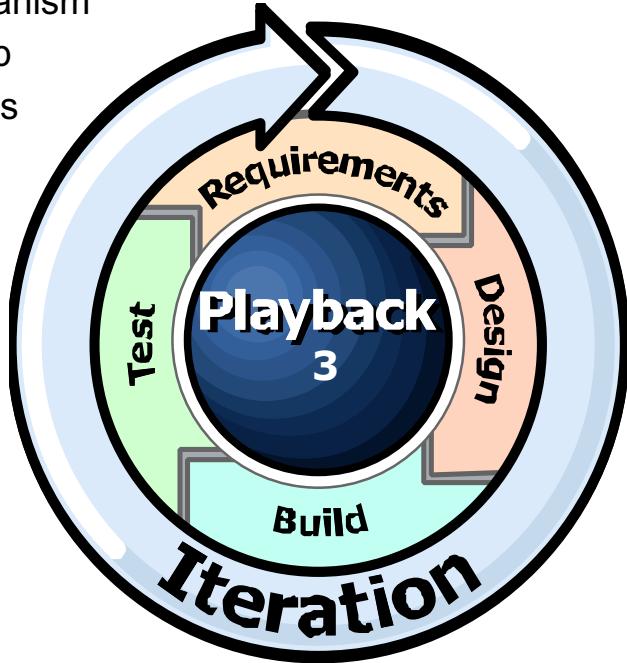
Figure 2-25. Playback 2

Playback 2: Focus on integrations

- Creating a decision service
- Implementing message events

Playback 3

- Non-functional requirements and performance testing
- Develop tracing & logging mechanism
- Technical tradeoff and leadership
- Address infrastructure challenges
- Develop deployment strategy
- Scope tradeoff / estimation
- Design Reporting
- Mentor resources



Introduction to Business Process Management

© Copyright IBM Corporation 2020

Figure 2-26. Playback 3

Playback 3: Focus on consolidation of the previous themes and producing an end-to-end solution

- Handling errors in services
- Deploying process application

Development starts at the end of Playback 0 when an executable process model is delivered. The next few slides cover more details around Playback 0.

2.5. Playback 0

Playback 0

Introduction to Business Process Management

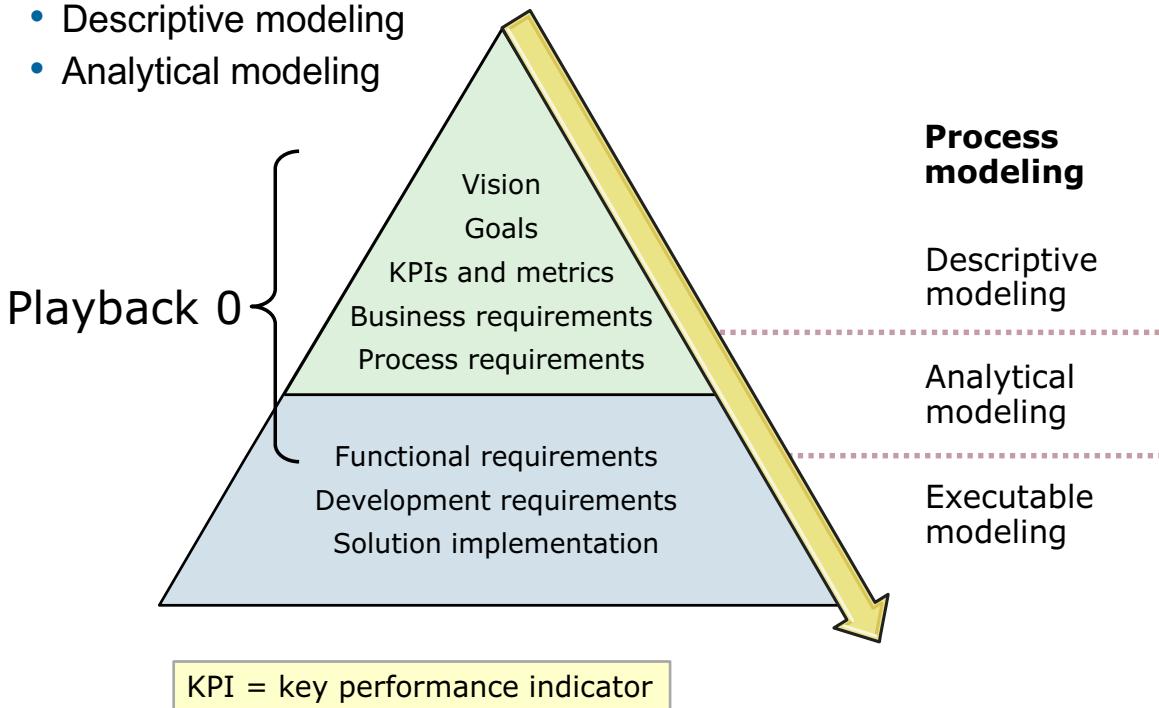
© Copyright IBM Corporation 2020

Figure 2-27. Playback 0

- Playback zero is an important milestone in a Business Process Management project-development lifecycle because it is during Playback 0 that a business process is identified and analyzed as a good candidate for a Business Process Management project. During Playback 0, the business process is defined, modeled, refined, and reviewed through discussion, discovery, and process analysis.
- Typically, a process-analysis phase also precedes Playback zero that takes two to three weeks. If the review is complete and the process model is approved, refining the process requirements can occur during the implementation playbacks, such as Playback One. Even if your team decides not to proceed with the development of the process, you have a documented process.

Playback 0

- Descriptive modeling
- Analytical modeling



Introduction to Business Process Management

© Copyright IBM Corporation 2020

Figure 2-28. Playback 0

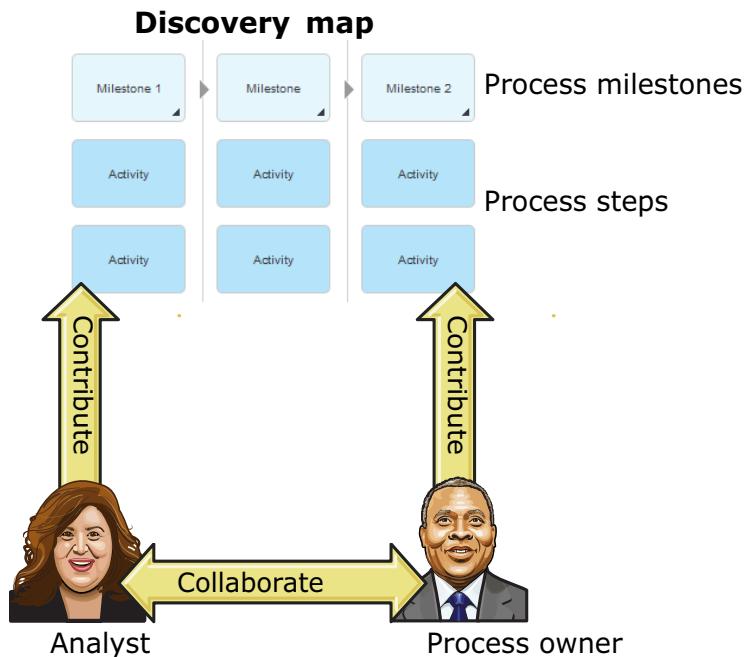
- Going back to the BPM project components, you can see Playback 0 covers Descriptive and Analytical Modeling. The definition stage of project development focuses primarily on these modeling efforts. The goal for Playback 0 is that the concerns and achievements at this stage of project development are iterated until a final solution is reached.
- The process must be discovered before it is modeled as a diagram. That requires sessions with the business process owner to uncover the particulars of the business process at a high level. As the process is defined, it is then necessary to start to analyze and create initial models if possible. Use an incremental approach from the current state to a future state business process that is accepted and agreed upon as a final “to-be” model.

Descriptive modeling

- The descriptive modeling that is accomplished at early stages can be done with various tools, including Bluworks Live. Bluworks Live provides the “best in industry” for process modeling that is focused on process discovery and analysis. A brief introduction to Bluworks Live is in the next unit.
- Early stage, or descriptive, modeling that is based on discovery and analysis has a specific outcome. Process model diagrams and documentation are refined until both the process owner (business) and the BPM team designate that the business process is completed and validated. The focus now changes from business requirements to actual process requirements.

- The milestones to get to the final stage in descriptive modeling are process discovery and the as-is model.
- Both process discovery and the as-is model can be completed by using either Bluworks Live or IBM Business Automation Workflow. When you get to the “to-be” model, it must then be in IBM Business Automation Workflow for implementation.

Playback 0: Process discovery (1 of 2)



- Document the current state of the business process
- BPM analyst, process owner, and BPM project manager

Introduction to Business Process Management

© Copyright IBM Corporation 2020

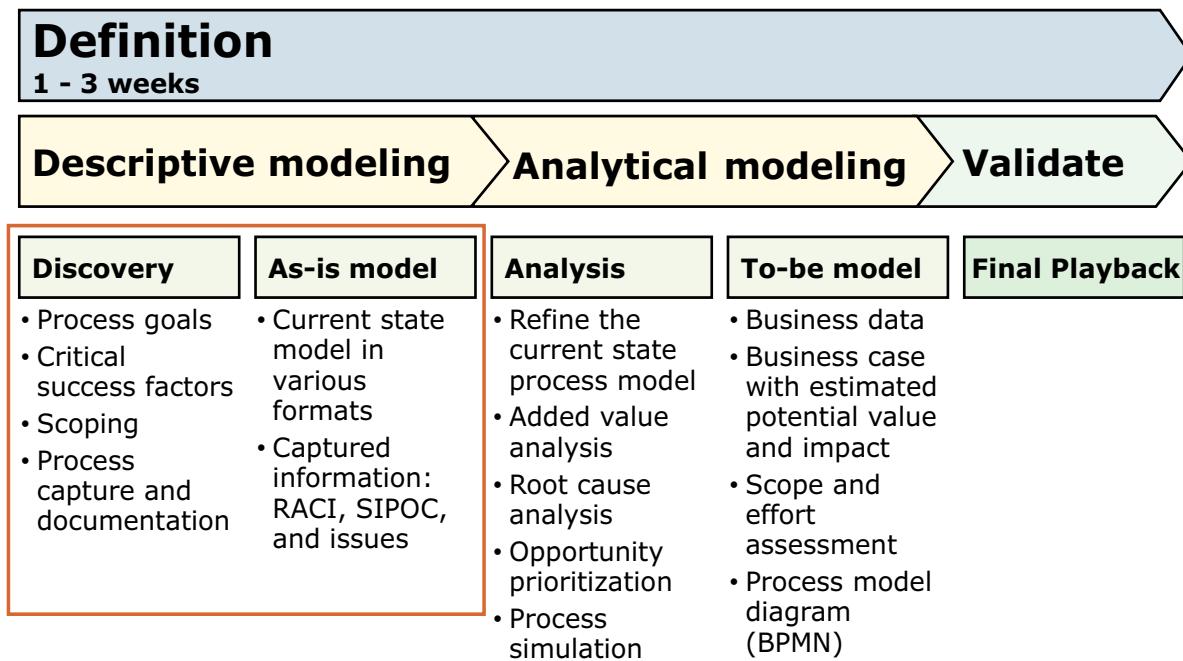
Figure 2-29. Playback 0: Process discovery (1 of 2)

The process discovery effort in Playback 0 allows for the capture of the initial process information that translates into the initial process model. It is typically an effort from process owners and BPM team members who want to make sure that the current state of the business process is documented. This documentation can be stored in various tools available to the team, including Blueworks Live. However, Blueworks Live has the advantage of providing connectivity to IBM Business Automation Workflow. With Blueworks Live, the documentation effort maintains a high level of usage, even beyond process discovery and analysis of the business process.

Agile software development places a greater value on working code than on comprehensive documentation. Although documentation is valuable, code that works is more valuable. Process discovery should attempt to capture the process as quickly and accurately as possible, but should not sacrifice project time to create documentation that does not lead to the project goals. Capturing and documenting processes inside of a tool like Blueworks Live and IBM Business Automation Workflow directly contribute to implementing the process.

Playback 0: Process discovery (2 of 2)

Playback zero



Introduction to Business Process Management

© Copyright IBM Corporation 2020

Figure 2-30. Playback 0: Process discovery (2 of 2)

Descriptive modeling is discovering what you currently have; it is providing visibility. The aim is to document the as-is model, or what you currently do. This action provides the baseline for going forward.

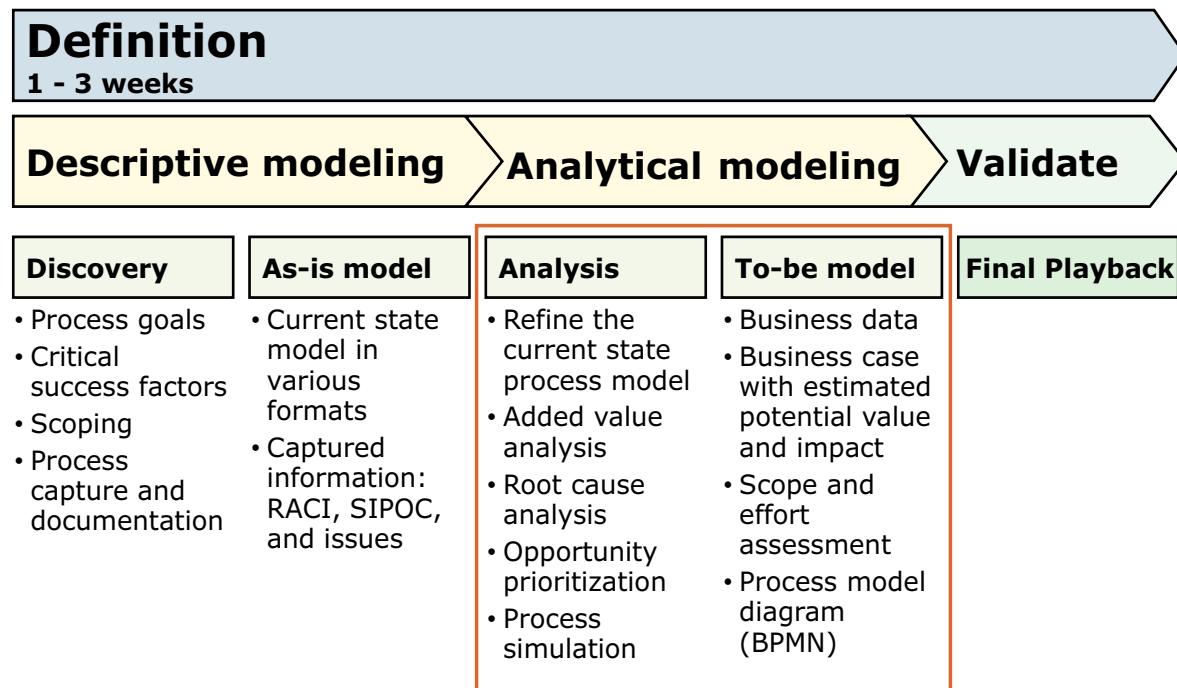
Moving from discovery to model:

- Business process discovery documentation, or mapping, with as much process detail as possible, is at times a quick process. The beginning of a process modeling effort can happen early. Therefore, the most common question when in the midst of a process discovery effort is: When do you move from process discovery to process modeling?
- Several aspects must be considered to answer this question.
- It is time to transfer to a process model when process discovery sessions exhaust all requirements to communicate the following concepts:
 - What a process is
 - Who is responsible for process task completions
 - Who is documenting the problems within the process
- Also, consider the conversations in the process discovery sessions during Playback 0 meetings. When the questions no longer center around "What does this process do?" and start

to center around “What does this process look like?”, then the move to a process model is at hand.

- Some acronyms and their expansions are listed:
 - RACI = responsible, accountable, consulted, informed
 - SIPOC = suppliers, inputs, processes, outputs, customers
- These abbreviations are used in many business design methodologies to capture relevant information. Business analysts in the class are probably familiar with these terms.

Playback 0: Process analysis



Introduction to Business Process Management

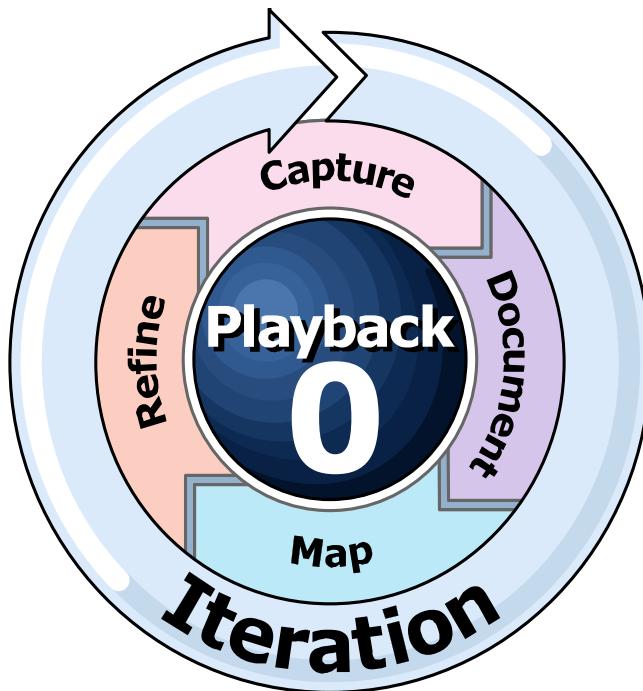
© Copyright IBM Corporation 2020

Figure 2-31. Playback 0: Process analysis

- The next phase in process modeling and Playback 0 is analytical modeling. The milestones to get to the final stage in analytical modeling are analysis and the “to-be” model.
- Discovery documentation is a “just the facts” effort, meaning the only thing that interests the BPM analyst and process owner is that the process in its current state is captured. A good example is the capture of process problems, or issues. During discovery, it is not important to try to solve process issues, only to document them. This action is where the next step in Playback 0 starts to take shape: process analysis.
- Process analysis is when the business process is continually refined until analysis goals are reached, such as solving process issues through root cause analysis. Other process analysis goals include added-value analysis of each process activity that is captured and analysis to ensure the correct priority for improvement opportunities. With process analysis, the business requirements are vetted and the process requirements are aligned so that the “to-be” process model is ready to be finalized.
- The successful outcome of Playback 0 for a process owner is to have the most efficient and effective business process that is modeled for continued use in implementation.
- This course does not cover in detail the “how to” for process discovery and analysis. IBM Cloud Education offers courses on process analysis methods for comprehensive training on process analysis.

- You might also notice that the third phase, executable modeling, is not shown here because executable modeling comes later; therefore, it is covered in the next part of the course.

Playback 0: Iteration



Introduction to Business Process Management

© Copyright IBM Corporation 2020

Figure 2-32. Playback 0: Iteration

Playback 0 has a unique set of achievements during this stage of project development. A BPM analyst handles Playback 0 when working in concert with the process owner and BPM project manager. BPM project managers need the data from Playback 0 to plan for the next project development cycles, or Playback stages. The process owner is interested in the business process. The successful outcome of Playback 0 for a process owner is to have the most efficient and effective business process that is modeled for continued use in implementation.

To achieve both objectives, the strategy that is used to gather Playback 0 data is the iterative phase approach of:

- **Capture:** Make sure that the business process information is shared fully
- **Document:** Refine documentation as the analysis continues because there exist more stakeholders than just the process owner
- **Map:** Create a discovery map that can clearly define the important information in an easy-to-read manner
- **Refine:** Allow for adjustment to a business process as a clear definition of the business process and process model is incrementally made

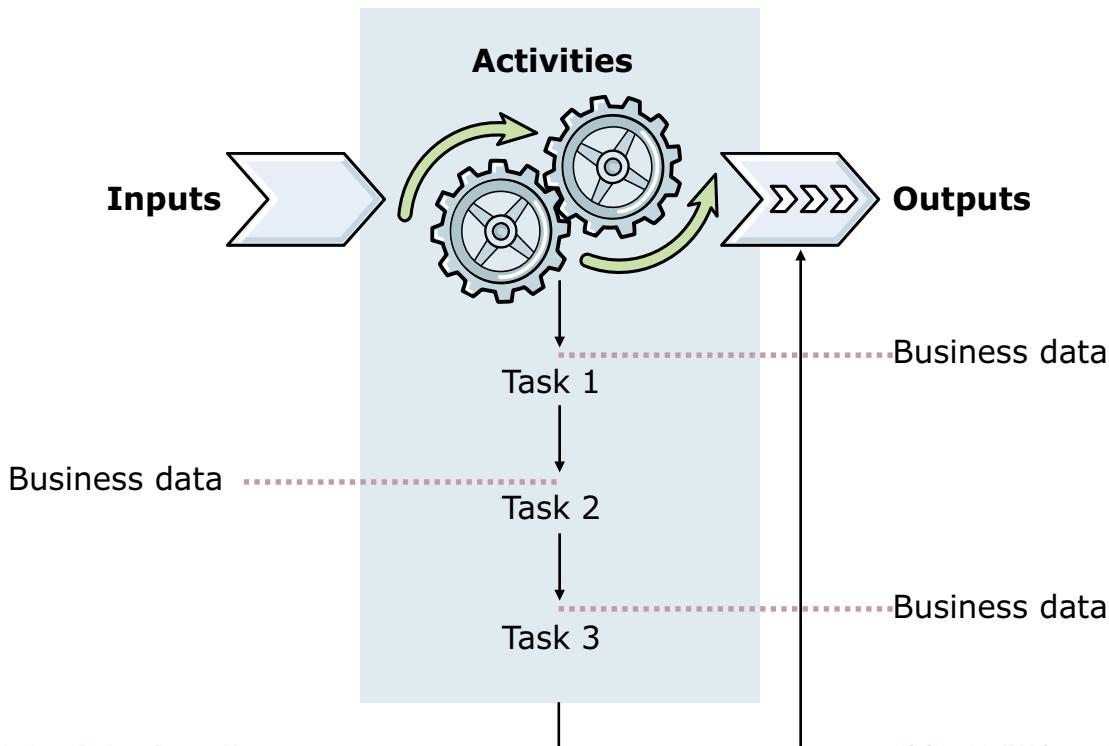
All requirements in Playback 0 are iterated, including the model that is finalized for validation. It is typical to encounter scope creep, or a broadening of the requirements to implement during this

iteration. Keep in mind the duration of the iteration and use the backlog to document those requirements that do not make this iteration.

Business data

Process:

Business process



Introduction to Business Process Management

© Copyright IBM Corporation 2020

Figure 2-33. Business data

Analytical modeling allows the process owner and BPM team to identify the business data that is needed for the process model. In general, business data provides the context of the process task for each responsible role for task completion. For example, if the business process is to process an insurance claim, the task verifies the claim, and the business data provides the claim type, claim number, claim description, and claim submitter.

When defining the business data for a process model, BPM teams look at the process as a whole. The question for each activity becomes: “What data does the process require to complete this task?” In the end, the process analysis produces the “to-be” process model and a business data model as well.

Business data is not implemented during Playback 0. Capture the inputs and outputs of activities or tasks in the documentation. Later in Playback 1, you implement the business data as part of the process. For now, business data is useful to conceptualize the process and to determine the following information: which activities must be part of the process, what occurs inside of the activity, what the outputs of the activity are, and what input data is necessary for the next activity.

At the completion of Playback 0, the “To-Be” process model is finalized and you move from analytical modeling into executable modeling. You create the “To-Be” model in the exercise at the end of the next unit.

Unit summary

- Understand the process spectrum
- Define Business Process Management (BPM)
- Define process modeling
- Understand the high-level project phases of a BPM project
- List and describe the phases in the IBM Playback methodology
- Describe Playback 0 and the achievements that are reached at this stage in the Playback methodology

Introduction to Business Process Management

© Copyright IBM Corporation 2020

Figure 2-34. Unit summary

Review questions

1. True or False:

The goal of Business Process Management is to optimize the efficiency of structured processes whereas the goal of Case Management is to provide flexibility in controlling the flow of the process.



2. A playback is:

- A. A focused demonstration of a partially implemented process application
- B. A demonstration of the components of Business Automation Workflow
- C. An in-depth technical overview of how the solution works
- D. Provided only at the end of the project lifecycle

3. At the end of Playback 0, the following is delivered.

- A. The “As-Is” process
- B. The “To-Be” process
- C. The analytical model
- D. The descriptive model

Review answers

1. **True.**
2. **A:** A Playback is a focused demonstration of a partially implemented process application, which is delivered to the business and IT communities for discussion, consensus-building and approval Design, Modeling, Execution, and Optimization
3. **B:** At the completion of Playback 0, the “To-Be” process model is finalized and you move from analytical modeling into executable modeling.



Introduction to Business Process Management

© Copyright IBM Corporation 2020

Figure 2-36. Review answers

Unit 3. Playback 0: Modeling the As-Is and To-Be business processes

Estimated time

01:30

Overview

In this unit, you learn how to create a structured process application based on an examination of requirements. You also learn how to identify process activities, decompose some activities into a nested process, and build the initial process. You learn the purpose of Blueworks Live and how it fits into the process development lifecycle. The unit also covers the core notation elements that are used in IBM Process Designer, and the categories of activities like processes and nested processes that contain process tasks. It concludes with a case-study scenario that begins the in-class development of a process that is based on the business requirements that are established within the process analysis - Playback zero phase of the project.

How you will check your progress

- Review
- Exercise

Unit objectives

- Describe how to model a process
- Decompose activities into a nested process
- List and describe the core notation elements that are used in IBM Process Designer
- Describe the purpose and function of Blueworks Live
- Examine a defined workflow from detailed process requirements and identify the interrelated process activities and the roles that are responsible for completing them
- Describe the requirements of the Hiring Request Process use case

Key concepts in this unit

- **BPMN:** A notation standard (Business Process Model and Notation)
- **Pool:** A BPMN element that represents the entire business process
- **Lanes:** A team or a responsible role of the process task
- **Activity:** Represents a single task that a process participant accomplishes from start to end
- **Flow objects:** Represent either process task assignments or process controls
- **Nested process:** Series of processes that are connected at a high level to child definitions

Topics

- Modeling a process
- Nested processes
- Introduction to Business Process Model and Notation
- Pool and lanes
- Flow objects
- Performing process discovery with Blueworks Live
- The Hiring Requisition Process

Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-3. Topics

3.1. Modeling a process

Modeling a process

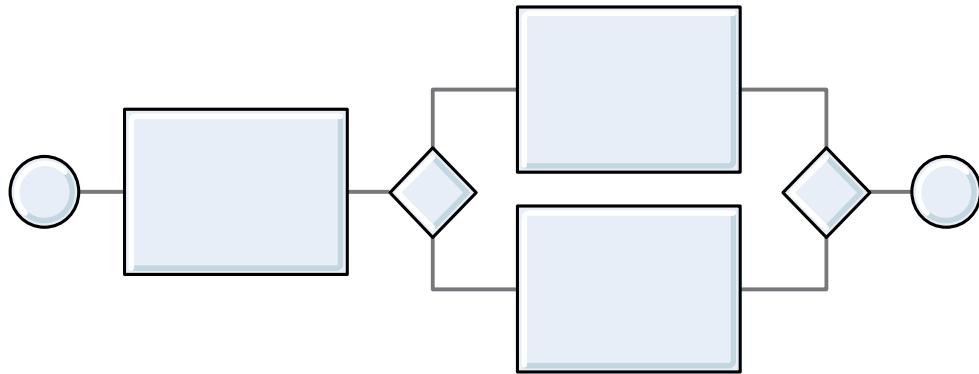
Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-4. Modeling a process

Creating the process is part of the effort within Playback 0 that uses IBM Process Designer. Naturally, modeling does not stop with creation of the process model, but continues on through iterations until the business process reflects the specified improvements and adjustments. For this reason, the shift from descriptive modeling to analytical modeling happens to produce the process model that is deemed worthy of implementation. It begins with the creation of the process model.

Creating the As-Is process model



- When creating a process model during discovery of the business process, the process model should reflect the captured data only
- The process model should ignore solutions for process problems until analytical modeling is completed
- The process model is agile enough for continued adjustments, so the focus is to have the expected order of process tasks that are reflected in the model first

[Playback 0: Modeling the As-Is and To-Be business processes](#)

© Copyright IBM Corporation 2020

Figure 3-5. Creating the As-Is process model

To understand how to create a process model that is incrementally adjusted through the Playback sessions of the business process, follow these guidelines:

- If creating a process model during discovery of the business process, a good practice is to have the process model reflect only the captured data.
- It is a good practice to delay solving process issues until analytical modeling. During discovery, the focus is on capturing the process.
- The process model is agile enough for continued adjustments, so focus on the expected order of process tasks that are reflected in the model first.

To understand how to create a process model that is incrementally adjusted through the Playback sessions of the business process, follow these guidelines:

- Focus on only the current process and the aspects that it directly concerns. Do not model other data that contributes business processes or systems.
- Focus on the as-is model and document its weaknesses also. Save the analysis of how to fix those weaknesses for later.

- At this stage, do not worry too much about getting a perfect model. Focus on documenting your understanding of the activities and how they fit together. Refinements are made during analytical modeling.
- The As-Is model is used to create an optimized To-Be model. The To-Be model is what is used for development.

Automation

- Not the first order of business in process modeling
- Should follow a natural development path from an analysis of the business process issues
- Merely automating a business process makes a bad business process more efficiently bad

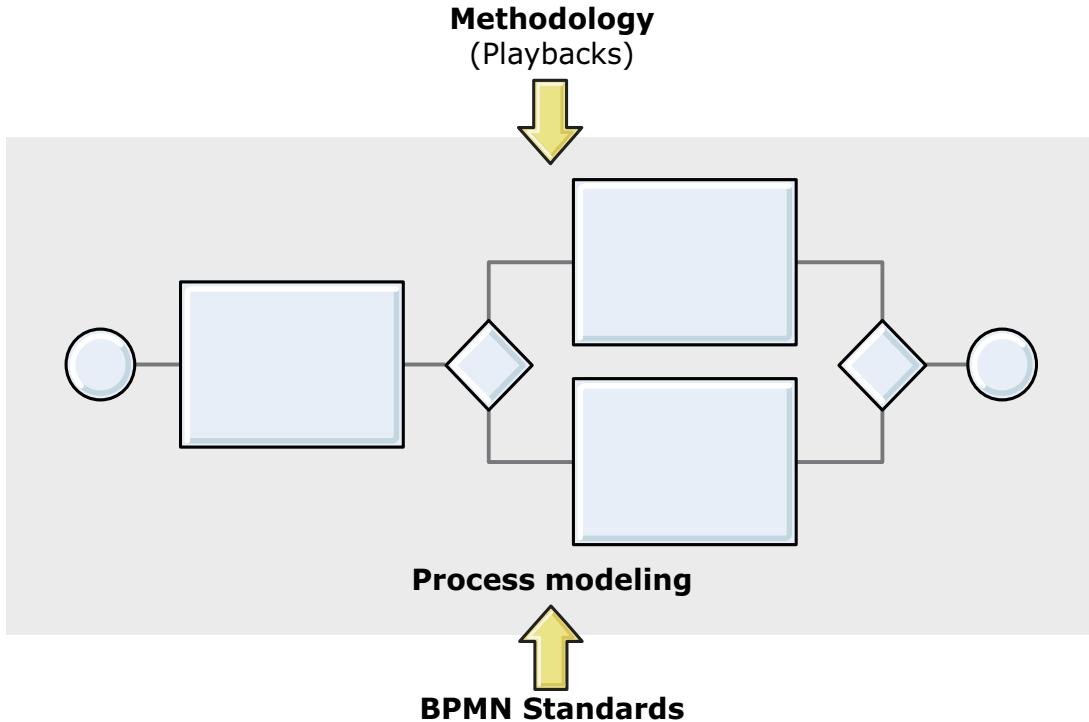
Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-6. Automation

- Give careful attention to the focus of the initial process model. Many times, organizations erroneously believe that BPM is about adding technology solutions to process problems. Soon after the initial sessions to document the business process, the decision to automate process tasks is prematurely made. Automation certainly is a good thing in BPM; however, it is not the first order of business at this early stage. Allow the analytical modeling effort to designate opportunities to automate tasks. It happens naturally as the iterations on the process model provide opportunities to adjust and improve the business process.

Where to start



[Playback 0: Modeling the As-Is and To-Be business processes](#)

© Copyright IBM Corporation 2020

Figure 3-7. Where to start

As described earlier, process modeling captures the ordered sequence of activities within a process along with supporting information from start to end. In modeling, the business process is framed with a workflow model to reflect component activities, the roles that are conducting those activities, conditional branching, and the sequencing of the flow of work between activities. In IBM Process Designer, this model is called a process, but it is also sometimes called a process diagram.

To translate process requirements that are documented in the discovery sessions into a process model, the BPM team must understand how to use the best methods and standards available. Not only is it necessary to translate requirements, they must be translated correctly so that everyone clearly understands the process model.

Where a BPM team starts is by adhering to the standards that are used in process modeling, Business Process Model and Notation (BPMN).

Concurrent to using BPMN to model the business process, a BPM team also uses a development method that works best to collaborate on modeling with business and IT. This development method is called Playbacks. IBM Business Automation Workflow uses the Playback to assist project development and BPMN V2.0 as the model standard.

Modeling task activities

Process documentation provides details on the teams in a business process and also the work they do. It is important to note that the information is primarily captured and in some instances, analyzed for value. However, more change can happen after it is represented in a process model. The information is also work-related and not necessarily conducive to process model needs until translated. It requires more comprehensive work from the process author to model process activities for teams correctly and to create a process model that is not too complicated to communicate the business process effectively.

An important guideline is that in general, it is a good practice to create a process that is as simple as can be modeled. A highly conceptual process is resilient to change.

To create the simple abstraction process model, authors go through a series of changes to the activities. When the initial process model with these refined activities and process flow is in place, BPM analysts continue to analyze the process with the process owner for process improvements. Finally, a stable process model can be implemented.

What is important to remember is that a good process model typically has an activity that represents a single task, which a single process team accomplishes. The next section covers a nested process and how an activity can also represent multiple activities or tasks effectively in a process model.

Process modeling guidelines in IBM Process Designer

- A process diagram or model is called a process in IBM Process Designer
- In general, a process should be as simple as you can make it
 - A highly conceptual process is resilient to change
- Make sure that you use the Documentation area in the Properties tab for each element in IBM Process Designer to include important requirement notes

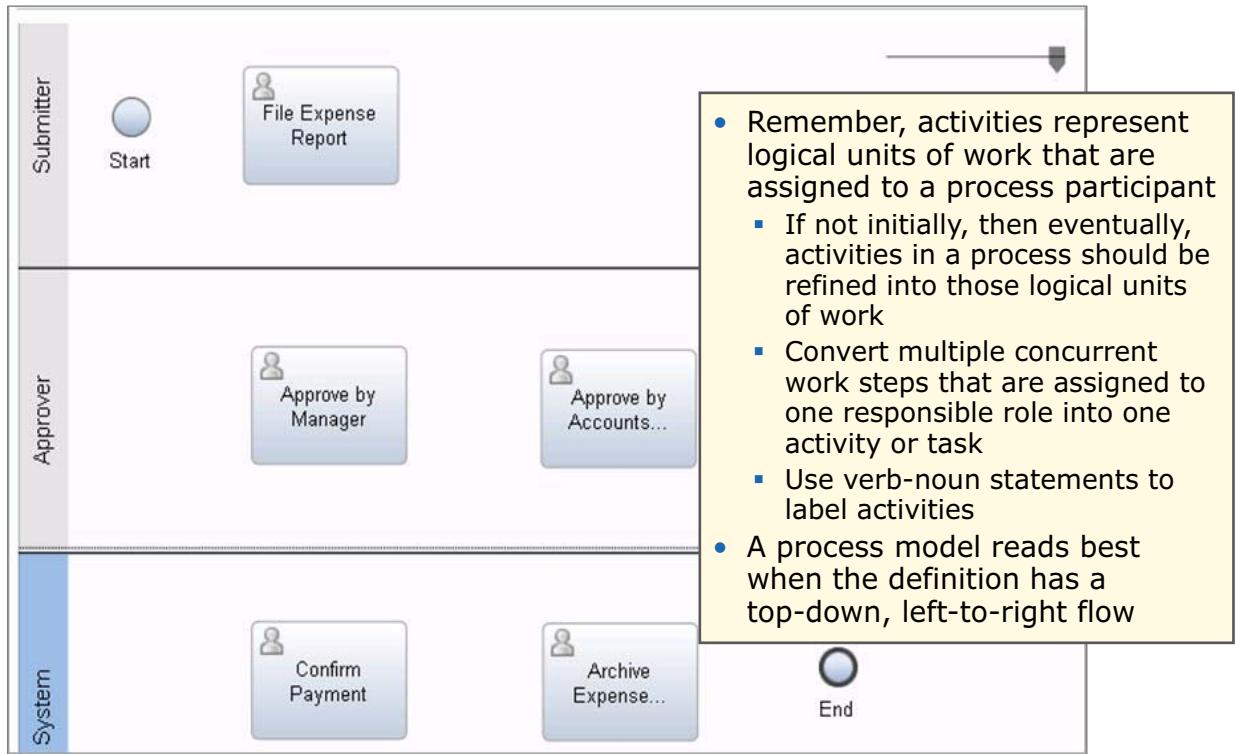
Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-8. Process modeling guidelines in IBM Process Designer

When modeling in IBM Process Designer, remember the guidelines that are shown in the slide.

Guidelines for modeling activities



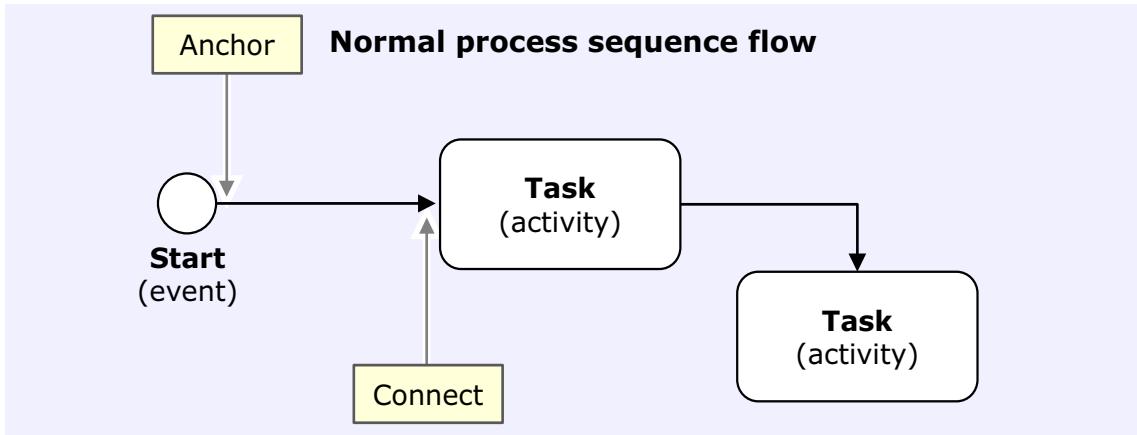
Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-9. Guidelines for modeling activities

When modeling activities in IBM Process Designer, follow the guidelines that are shown on the slide.

Connecting flow objects



- Hover over the flow object to see a flow that emerges out from the control point on the edge of the flow object
- Then, drag the sequence flow to the next flow object and release on the target anchor point
- Using the general rule of top-down, left-to-right flow, connect flow objects from left to right or bottom to top on the flow objects
- This action helps keep the process model simple and the flow lines from crossing

[Playback 0: Modeling the As-Is and To-Be business processes](#)

© Copyright IBM Corporation 2020

Figure 3-10. Connecting flow objects

This type of modeling communicates how the process flows from one activity to another and who does those activities. The method to accomplish this step in IBM Process Designer is as follows:

1. Hover over the flow object to see a flow that emerges out from the control point on the edge of the flow object.
2. Then, drag the flow to the next flow object and release on the target anchor point.
3. Using the guideline of top-down, left-to-right flow, connect flow objects from left to right or from top to bottom on the objects. This rule helps with the simplicity of the process model and helps to keep flow lines from crossing.

Because it is the early stage of process model, it is necessary to communicate only the expected flow of the process from the start event to one activity, to another, and to the end event. As the process model is analyzed and adjusted, the process flow is modified to express the nuances of alternative flow.

Process design considerations

- Models should be easy to understand with little to no explanation
- Models should start in the upper left and flow to the lower right
- Activities should be labeled in a Verb + Noun combination.
- No role, org unit or Systems in process activity naming
- Implementation details should not be depicted in the process model
- Activities should not have “and” depicting two activities in one
- Stay away from vague terms (that is, Process the application)
- Swimlanes depict participants



[Playback 0: Modeling the As-Is and To-Be business processes](#)

© Copyright IBM Corporation 2020

Figure 3-11. Process design considerations

- Process analysis and modeling are initially performed by the Process Analyst. Once the To-Be process model is completed, the developer takes ownership in its implementation. The process model should be easily understood. If the model is sufficiently complex, process decomposition might be necessary. Process decomposition involves creating a hierarchy of nested processes.
- Activities can either represent atomic tasks in which the work cannot be broken down any further, or they might represent nested processes in which the activity itself is a representation of another process. Nested processes are covered in the next topic.

3.2. Nested processes

Nested processes

Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-12. Nested processes

Activities that are gleaned from process documentation are not always going to be tasks that effectively communicate the business process in a simple manner. To judge communication simplicity and effectiveness of the process model, many users use the following guideline:

Clearly and easily communicate processes in 5 minutes or less at any level of granularity. To enable this, the process might need to be decomposed into more than one process. When this is performed, the parent process calls one or more nested processes. Nested processes can be implemented as either subprocesses or linked processes.

Activity: Nested process



The BPMN element representation of an activity that is a nested process is a rectangle with rounded corners and a (+) plus sign in a square at the center

- Decomposition helps provide details for a business process with a series of processes that are connected at a high level to child definitions (nested processes)
- Activities are decomposed into nested processes until single tasks are represented

[Playback 0: Modeling the As-Is and To-Be business processes](#)

© Copyright IBM Corporation 2020

Figure 3-13. Activity: Nested process

Decomposition

To achieve a level of effective communication of the process model, authors use decomposition to judge whether they are at simple abstractions of the model with single logical units of work for activities. Decomposition is basically showing details for a business process with a series of process model definitions that are connected at high level to child definitions.

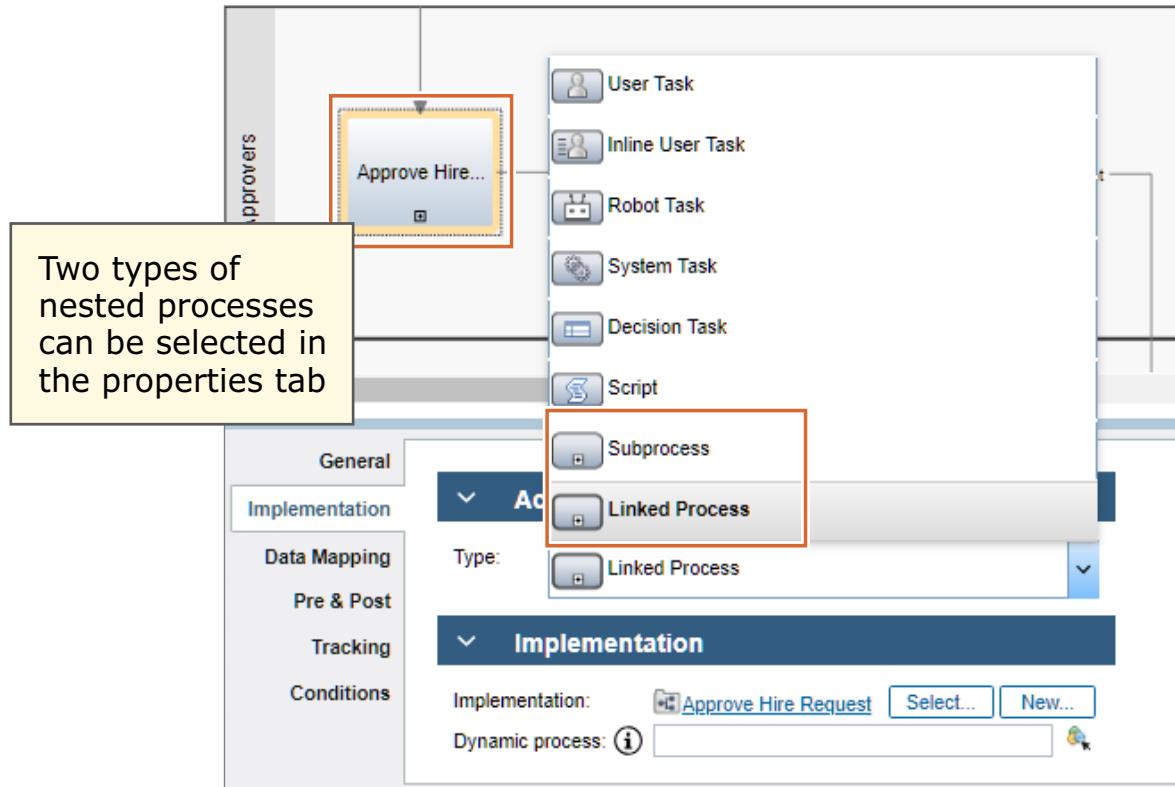
In essence, a flow object activity is also used as a container of a child definition, which in turn can have activities that also contain child definitions. When the activities within the child definition represent logical units for work or tasks for each activity, decomposition is no longer necessary.

Child definitions in process models are called nested processes.

Is decomposition always necessary for process modeling?

Decomposition is necessary only if the process model is too complex to communicate the details of the business process, and the activities do not necessarily equal logical units of work. If the process documentation produced the information that led to a simplified process, it would not be necessary to decompose.

Activity: Nested process types (1 of 3)



Playback 0: Modeling the As-Is and To-Be business processes

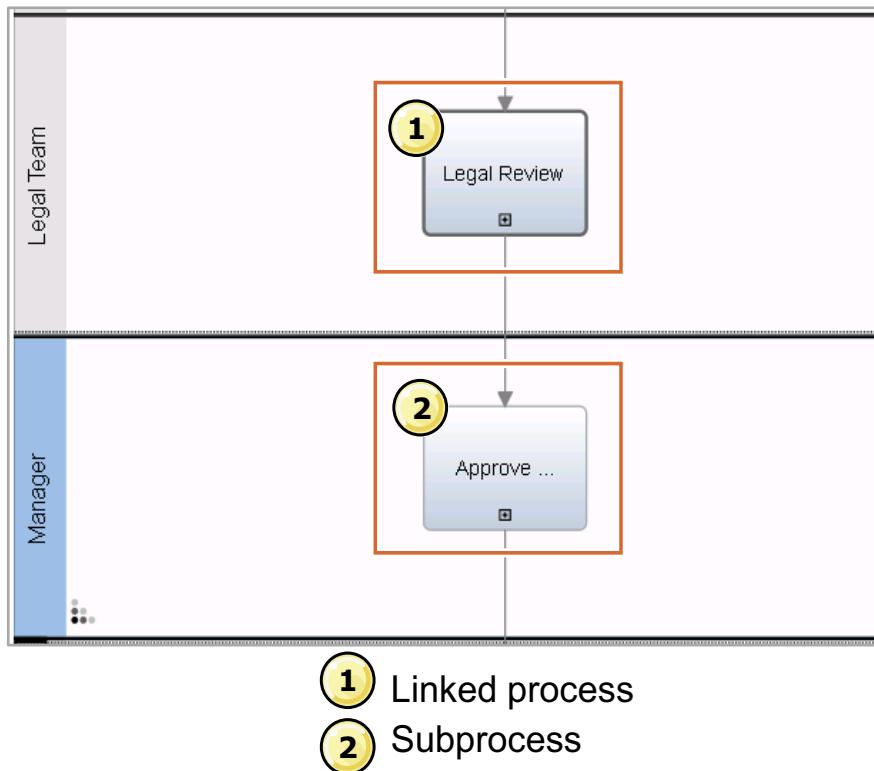
© Copyright IBM Corporation 2020

Figure 3-14. Activity: Nested process types (1 of 3)

All nested process activities are indicated by a plus sign with a square symbol in the center to indicate that they are not a task-type activity.

The two types of nested process activities are subprocess and linked process.

Activity: Nested process types (2 of 3)



- 1 Linked process
- 2 Subprocess

[Playback 0: Modeling the As-Is and To-Be business processes](#)

© Copyright IBM Corporation 2020

Figure 3-15. Activity: Nested process types (2 of 3)

1. Linked process:

A linked process is an activity with a plus sign and square in the center, and a dark bold line encompasses the activity.

This type of process might be thought of more as separate from the original process than as a subprocess. Think of two processes that are linked when using this type of nested process. You can reuse this type of process in many different parent processes. An example might be a legal review that can be used in several different insurance claims processes. If your legal review is the same in an auto insurance claim and in a property insurance claim, you would want to choose a linked process for the legal review.

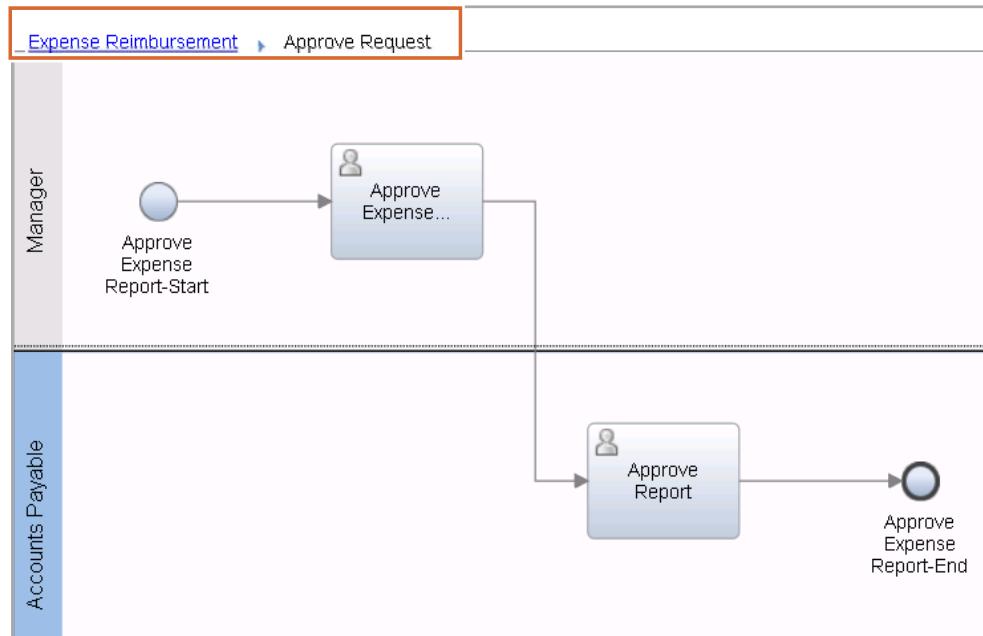
2. Subprocess:

A subprocess is an activity with a (+) plus sign and square in the center, and a normal single line encompasses the activity.

This type of nested process can be thought of as a subset of the original process. Sometimes, you might hear it called an embedded process. You cannot reuse this type of nested process, so think of reuse when choosing this type of nested process. If at any time the nested process might be reused in another process, then choose a linked process instead.

You can think of a subprocess as hiding several activities from view so you can achieve your goal of communicating your process in 5 minutes or less. Subprocesses also work well with the parent data model to pass along data values. Data and data models are explained in more detail in a later unit in this course that deals with implementation of the process model.

Activity: Nested process types (3 of 3)



- A subprocess has breadcrumb navigation back to its parent process
- The start, end, and activities in a subprocess must have unique names

Figure 3-16. Activity: Nested process types (3 of 3)

For a linked process, when you double-click the activity, the linked process is opened (since it is a separate process). For a subprocess, when you double-click the activity, the subprocess shows and a breadcrumb trail is displayed at the top so you can go back to the top-level process. In the screen capture, the teams are narrowed down to two. Notice that the system lane does not exist in the nested process.

Introduction to Business Process Model and Notation

Playback 0: Modeling the As-Is and To-Be business processes

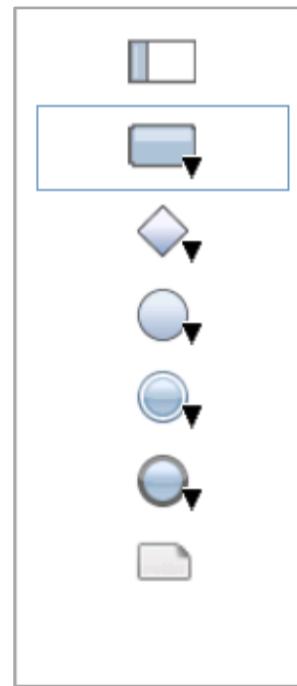
© Copyright IBM Corporation 2020

Figure 3-17. Introduction to Business Process Model and Notation

About BPMN

- The standard flow chart-based notation for defining business processes
- Creates a standardized bridge for the gap between business process design and process implementation
- IBM Business Automation Workflow's Process Designer uses several core elements from BPMN
 - Lane
 - Activity
 - Gateway
 - Event
 - Annotation

IBM Process Designer element palette



Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-18. About BPMN

To communicate the process clearly within your organization, apply a notation standard. This notation is known as Business Process Model and Notation or BPMN. IBM Process Designer supports a subset of the BPMN specification.

BPMN is used so that everyone who is involved can interpret and understand the model. Throughout development, many different parties are involved in modeling. Every Stakeholder, from the least technical to the most technical, understands the model to provide valuable feedback and continuously improve the process.

BPMN also allows a way to compact your process model. Many of the symbols represent ideas, so symbols allow for a more concise and smaller model than drawing a diagram without BPMN.

BPMN has many benefits, but most importantly, BPMN creates a standardized bridge for the gap between the business process design and process implementation. This single notation is agreed upon among multiple BPM vendors for the benefit of the user community.

IBM Process Designer uses several core BPMN elements:

- Pool
- Lane
- Event

- Activity
- Flow
- Gateway

IBM implements and interprets these elements to have specific meanings and terminology in IBM Process Designer. For definitions of the BPMN specification, see the BPMN Specification document version 2.0 from the Object Management Group.

The element palette of IBM Process Designer is shown in the slide image. These items correspond to BPMN elements and are used in modeling a process.

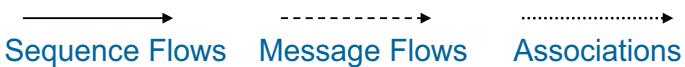
A BPMN standards group, which is called Object Management Group, is an international, open membership, not-for-profit technology standards consortium.

BPMN elements

- Flow Objects



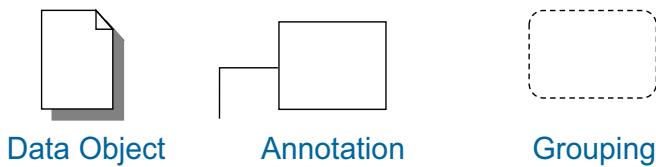
- Connecting Objects



- Pool and lanes



- Artifacts



Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-19. BPMN elements

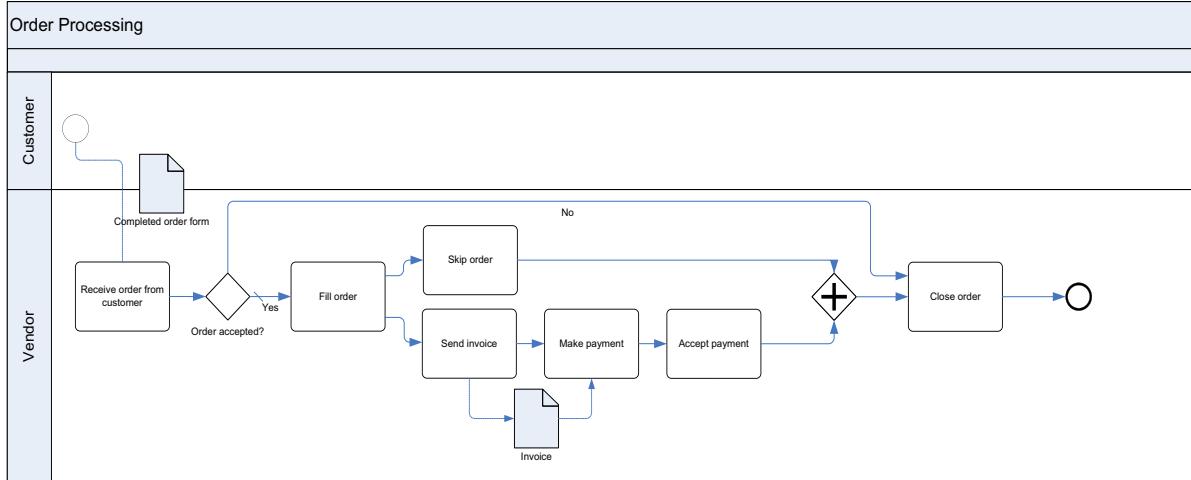
At a high level, focusing on the flow objects helps to understand the specification.

- **Events**: Each process begins and ends with an event. No process exists independently and is normally linked to other processes or services through events.
- **Activities**: Activities are the tasks that are performed by humans and systems.
- **Gateways**: Gateways are similar to decision diamonds in flow chart notation. They direct the sequence of the process based on an evaluation.

The next two slides provide an example of a BPMN-compliant process model.

BPMN example (1 of 2)

- Order processing BPMN model



Playback 0: Modeling the As-Is and To-Be business processes

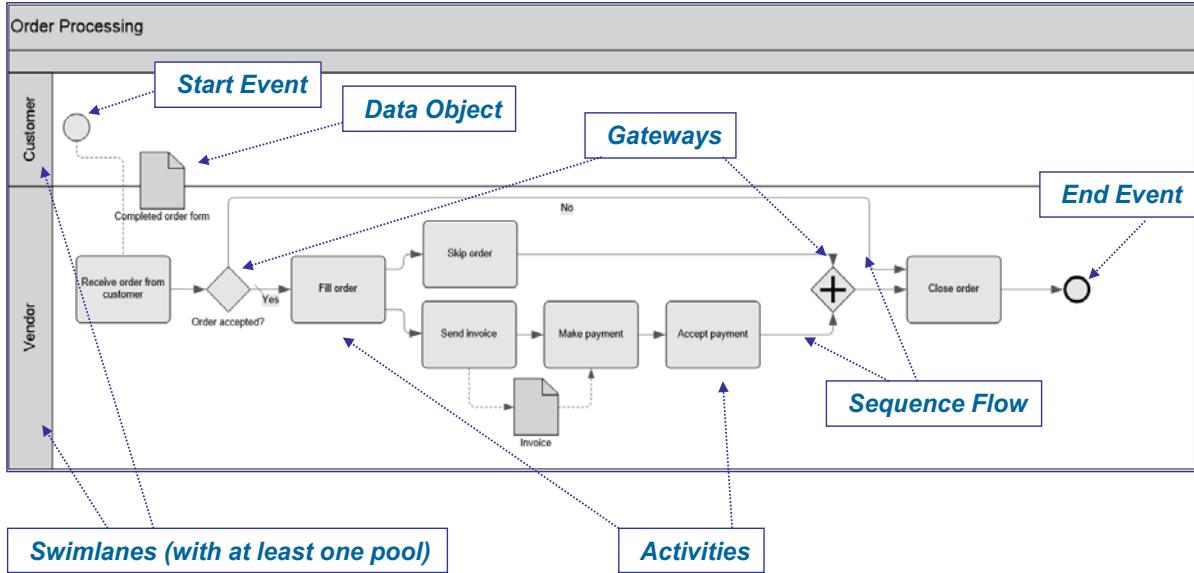
© Copyright IBM Corporation 2020

Figure 3-20. BPMN example (1 of 2)

The screen capture provides an example order processing BPMN model. See whether you can find the BPMN elements.

BPMN example (2 of 2)

- Order processing BPMN model



Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-21. BPMN example (2 of 2)

The screen capture provides an example order processing BPMN model.

The next couple units go into more detail regarding pools, lanes, and flow objects.

3.3. Pool and lanes

Pool and lanes

Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-22. Pool and lanes

The discovery and analysis session provides details about the business process that can be converted into BPMN process model elements. These elements can be used in conjunction as a diagram that describes the business process and later runs the process application. This section deals with two specific elements: pools and lanes.

In IBM Process Designer, the default setup for newly created process is one pool and two lanes. One lane represents a team and the other a system lane.

Pool



- A graphical element that is called a pool contains each process
- The name of the pool is the same as the name of the process

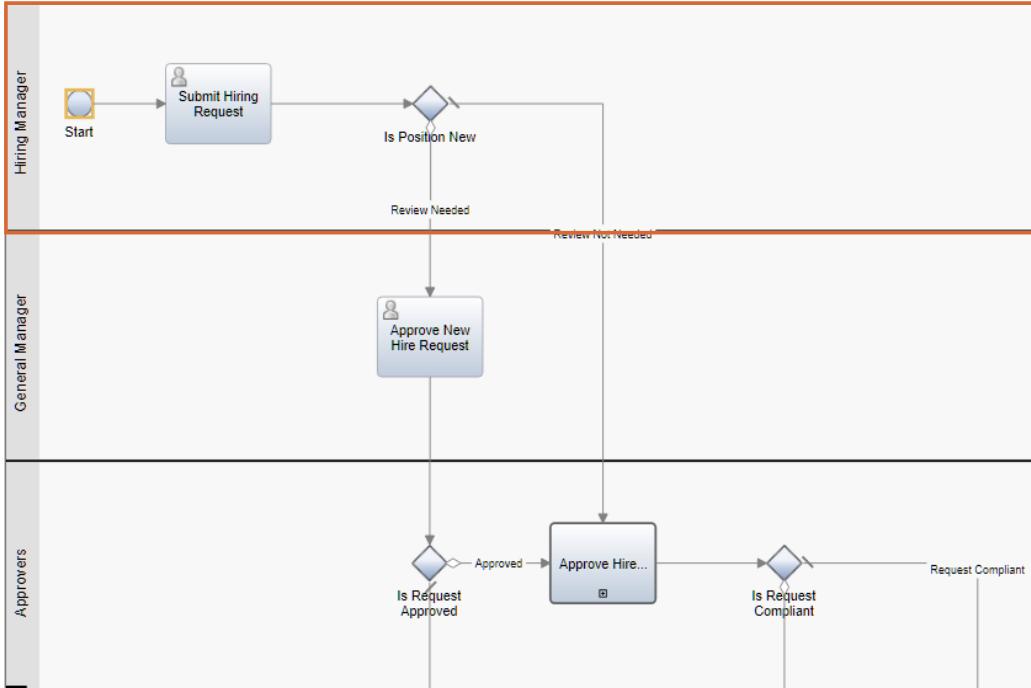
Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-23. Pool

A process that you model in IBM Process Designer includes the default IBM Process Designer pool, which consists of two default lanes (Team and System). In essence, the pool is the BPMN element that represents the entire business process. The pool is the only element that is not found in the element palette and does not have properties, but it is the default setup for all models that are created in IBM Process Designer.

Lanes



- Lanes are contained by the pool
- By default, top lane is a **team** lane and the bottom lane is a **system** lane

Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

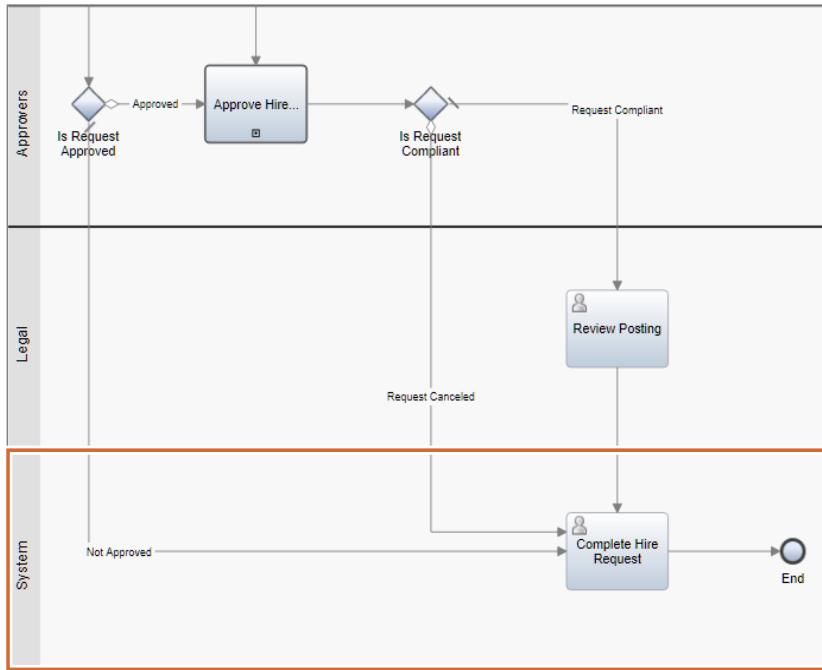
Figure 3-24. Lanes

Each lane represents a team, and the process task responsible role is detailed in the discovery and analysis session. Lanes provide context for a process model as each lane contains a series of activities that are assigned to a specific team member or events that transpire in the process. Activities and events are covered in more detail in the other sections of this unit.

To obtain the details for the team during discovery and analysis, user stories help determine which teams are responsible to conduct specific process tasks. Each of these teams is assigned to a lane when you model the process. It is important to remember that a team is a role, and not a person, in a process model.

The analogy of a swimming pool is a perfect description, as the pool with the associated lanes visually represents swimlanes in a pool. Swimmers are thought of as the teams who are swimming the length of the pool, completing tasks in their lane.

System lane



- Lanes can also be assigned to systems, and automated tasks are often in the designated system lane

Figure 3-25. System lane

- When the time comes to define, process tasks that are automated, the process model needs a way to communicate with automated tasks. A process participant who is assigned to a lane is not always a responsible human role. Process participants who are assigned to lanes can also be systems.
- For example, the discovery and analysis session might find that a system, rather than a human role, completes some process tasks such as conducting a background check on a loan. IBM Process Designer has a specific default lane to contain these sorts of automated tasks: the system lane. During the initial process model build, tasks that are automated are represented as part of the system lane.
- Further automation of process tasks is designed when the process is improved and validated through the iterative Playback project development. This iterative development can mean system lane movement and rearrangement to indicate where efficiency is found for the entire business process.
- Keep in mind, this process has not been implemented yet. There is a user task in the System lane. Before this process is implemented, the task would be converted to a system task.

3.4. Flow objects

Flow objects

Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

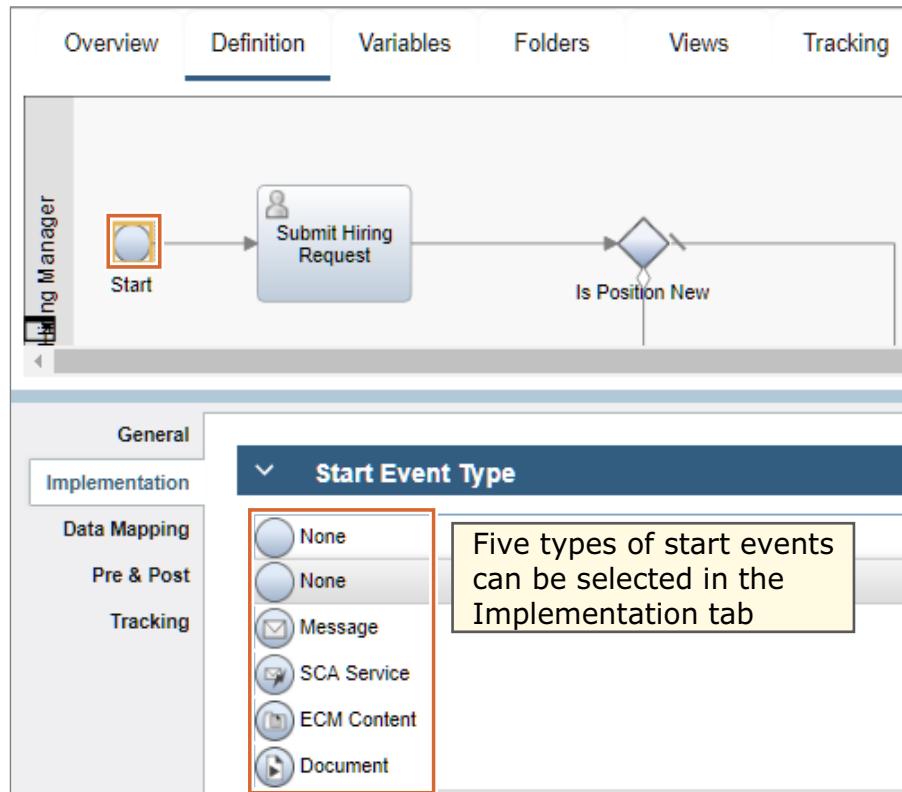
Figure 3-26. Flow objects

At this stage of diagramming a business process, an author considers flow objects for the model. Flow objects in a process model are in the lane for teams because they represent either process task assignments or process controls. In this course, you are not examining every type of event, activity, or gateway available in IBM Process Designer. Instead, you learn the most commonly used types of flow objects and your specific needs for modeling your first process. The remainder of this unit and the upcoming units covers the different types of flow objects as modeled by using IBM Process Designer.



Events: Start events

- Events have three categories: start events, intermediate events, and end events.
- IBM Process Designer has five types of start events: None, Message, SCA service, ECM Content, and Document.



Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-27. Events: Start events

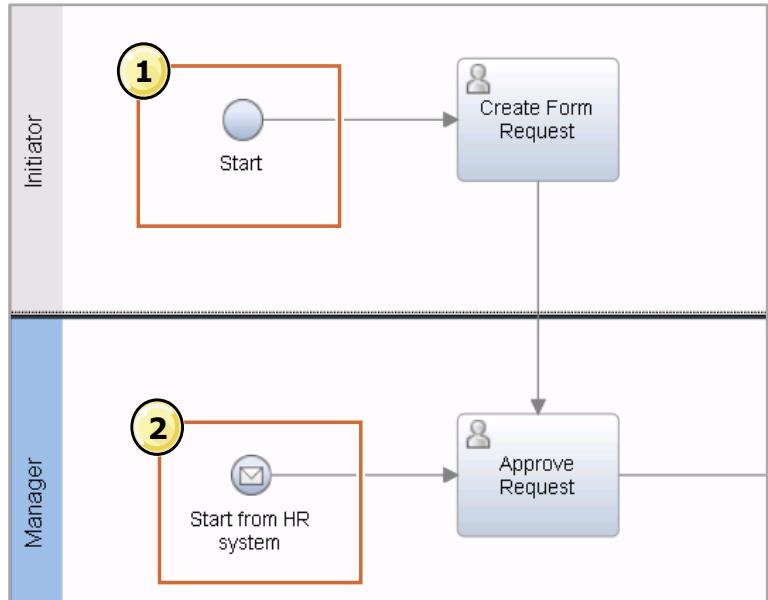
Events are control flow objects for a process model. Just like the definition of an event in everyday life, an event is something that occurs during a process. Events have three categories: start events, intermediate events, and end events. In the initial process model, it is important to have a start event and an end event. You focus on start and end events in this unit and revisit intermediate events in a later unit.

A single line that encompasses a circle represents a **start event**. Start events trigger the initiation of the process through a manual or automatic input. Authors describe the input in the properties tab documentation box that is provided for the element.

IBM Process Designer has five types of start events: None, Message, SCA Service, ECM Content, and Document. This course implements the None, Message, and ECM Content start events.

Events: Multiple start events

- 1** Only one “none” start event is allowed per process



- 2** To add a second start event to a process, you must use a message or content start event
— This example shows a message start event

Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-28. Events: Multiple start events

The two types of start events that are used most often are **none** and **message**.

- None:** The **none** event can be called a standard start event and is represented as a circle that does not have an internal marker. In a process, you can have only a single standard start event. If you try to model more than one standard start event, the process does not run and alerts you to the error. Creation of a process gives you one standard start event by default. When any user that “launches” or creates an instance of this process from the Process Portal, the process starts from the none event type. Other start event types create instances of the process through other methods (messages, events, document, and content uploads).
- Message:** Another type of start event is **message**. This start event is represented as a start event with an internal marker of an envelope. Message start events start a process when an external signal is received.

For example, suppose that your company wants its employee onboarding process to start when a new employee record is created in the HR system. When a new record is created, the HR system sends an event to IBM Business Automation Workflow. IBM captures that event and starts the follow-on events for the process. You can have more than one message start event for a process.

This implementation often happens when existing systems are supplying a newly implemented process with data and a company wants to gradually implement a Business Automation Workflow.

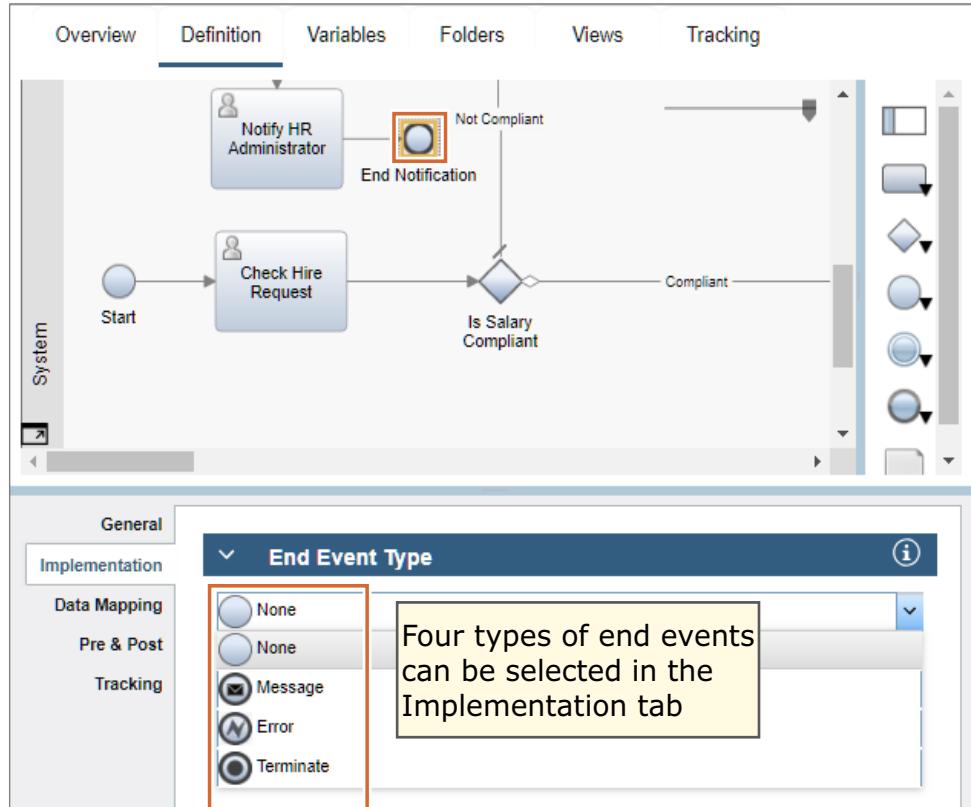
During a transitional period, the process can be started through IBM Business Automation Workflow with the start event or when an existing system inputs similar data to another process activity with the start message event.

For example, 3 or 4 customer service agents are starting to use IBM Business Automation Workflow to input data into an insurance claim process. IBM Business Automation Workflow provides a web form to enter the data to continue the process. The other 10 customer service agents continue to use the old database system and enter information into the database. At an interval, the database sends information to IBM Business Automation Workflow. The start message event starts, and then the data is processed through the rest of the process.



Events: End events

- IBM Process Designer has four types of end events:
 - None
 - Message
 - Error
 - Terminate



Playback 0: Modeling the As-Is and To-Be business processes

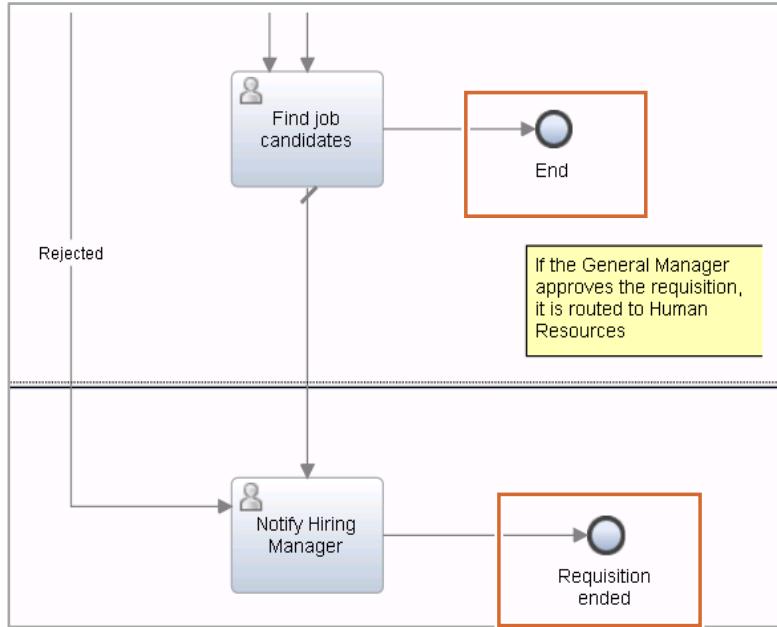
© Copyright IBM Corporation 2020

Figure 3-29. Events: End events

An end event is represented as a circle encompassed by a dark thick single line. End events are reached in a process when a final decision from all activities or a partial set of activities is reached.

End events have four types: none, message, error, and terminate.

Events: Multiple end events



- This process has two none end events: **End** and **Requisition ended**
- When you have more than one none end event, each should have a unique name

Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-30. Events: Multiple end events

The type of end event that is used most often is the **none** end event.

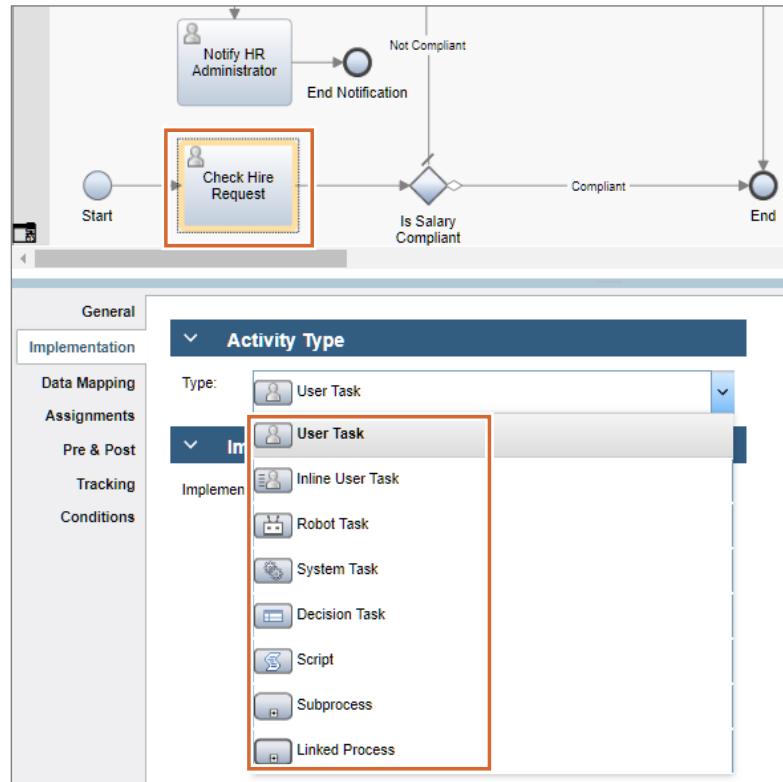
This type of end event can also be called a standard end event, and is represented as a dark circle that does not have an internal image. Unlike standard start events, you can have multiple standard end events. Thus, any process can have a single standard start event, and one or multiple standard end events. Creation of a process gives you one standard end event.

Multiple end events can improve understanding of the business process flow. An end event can signify the end of a process instance. If it is another process that is linked to a process, the end event signifies the end of the subprocess, and the flow continues from the end event. In this regard, you can connect process flow that comes from multiple end events to a parent process that contains this process as a linked process.



Activity: Task types (1 of 2)

- IBM Process Designer supports eight types of activities:
 - Inline User Task
 - Robot Task
 - System Task
 - Decision Task
 - Script
 - Subprocess
 - Linked Process



Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-31. Activity: Task types (1 of 2)

In a process, you can change the activity type by selecting the activity, clicking Implementation > Activity Type in the properties section, and choosing the type of activity from the menu.

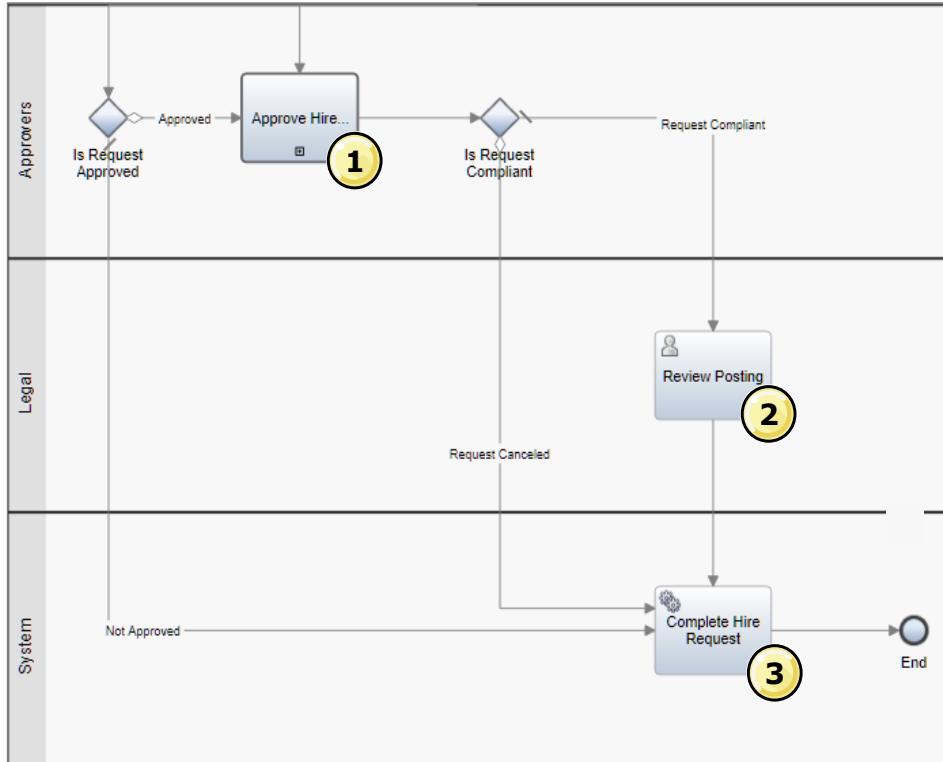
Note some of these are BPMN-supported and some are not. IBM supports a special task proprietary to IBM known as the Robot Task to enable integration with robotic process automation tools.

Inline user tasks are also an IBM proprietary implementation to enable prototyping of a process. The remaining activity types represent BPMN elements.

Inline user tasks are effective when quickly building a working process (prototyping). Inline user task coaches are automatically generated for the variables. If you change a variable in the Variables page, the variable is automatically changed in the Data Mapping properties. Any changes to the mapping cause a regeneration of the affected coach in the backing client-side human service.

Activity: Task types (2 of 2)

- The most common types of activities include
 - linked process
 - user task
 - system task.



Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-32. Activity: Task types (2 of 2)

The developer uses several task types, as follows:

- Task activities:** All task activities have an icon in the upper left corner of the activity to indicate their type.

The eight types of task activities are user task, system task, decision task, and script (task).

User and system tasks are the more common types of task activities that are used for modeling. A decision task is used in another course unit during implementation of the process. Scripts are also used during implementation of the process model.

- Linked process:** Links to another process. It is depicted with a plus at the bottom.
- User tasks:** A user task is represented as an activity with a human or person icon in the upper-left corner.

User tasks are selected when a user or human starts or completes an activity. One example of a user task is that an employee fills out an expense report and submits it for compensation. If you drag an activity from the palette to a non-system team lane, you automatically get a user task activity in your process.

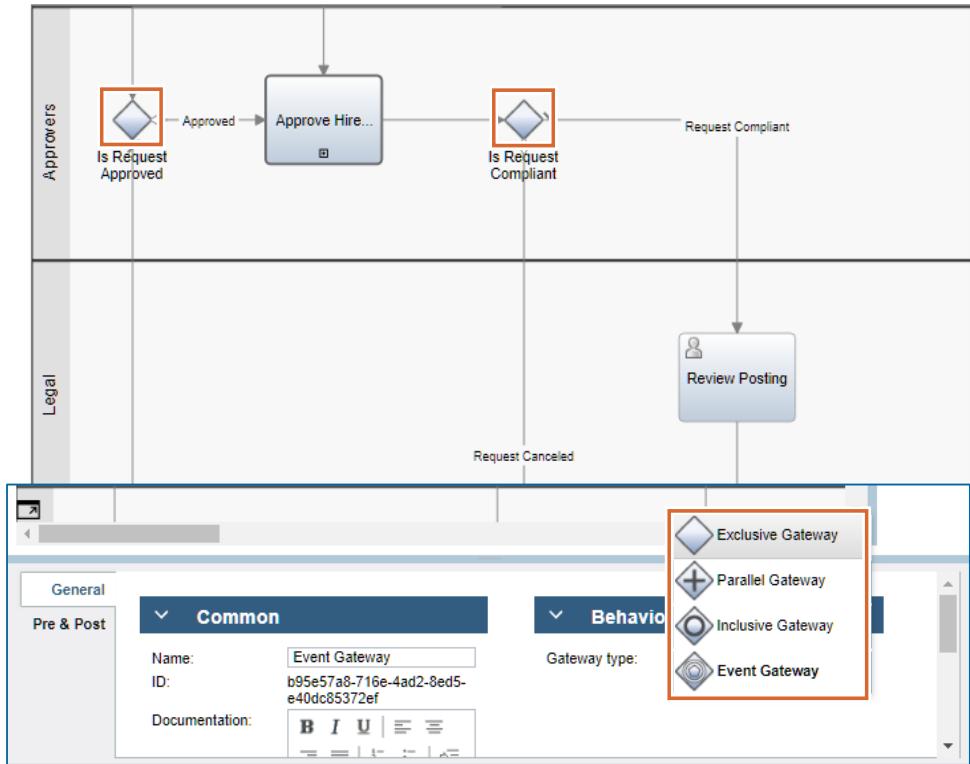
- System tasks:** A system is represented as an activity with two gears in the upper-left corner.

System tasks are selected when an automated system or service completes an activity. One example is a payment of an expense. When the expense is approved, you might want a system to go through steps to process and automatically pay the employee. If you drag an activity from the palette to the system lane, you automatically get a system task activity.



Gateway: types

- The most common type of gateway is the Exclusive Gateway.



Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-33. Gateway: types

When following the Playback methodology, you start with identifying the activities before evaluating how the flow is directed by using gateways. For this reason, in the next exercise you only focus on defining the activities.

The most commonly used gateway type is the exclusive gateway. This gateway is similar to the binary decision diamond in flow charts. Regardless of the number of inbound sequence flows, the gateway only follows one path out of the gateway.

Gateways will be discussed in more detail in the next unit.

3.5. Performing process discovery with Bluetworks Live

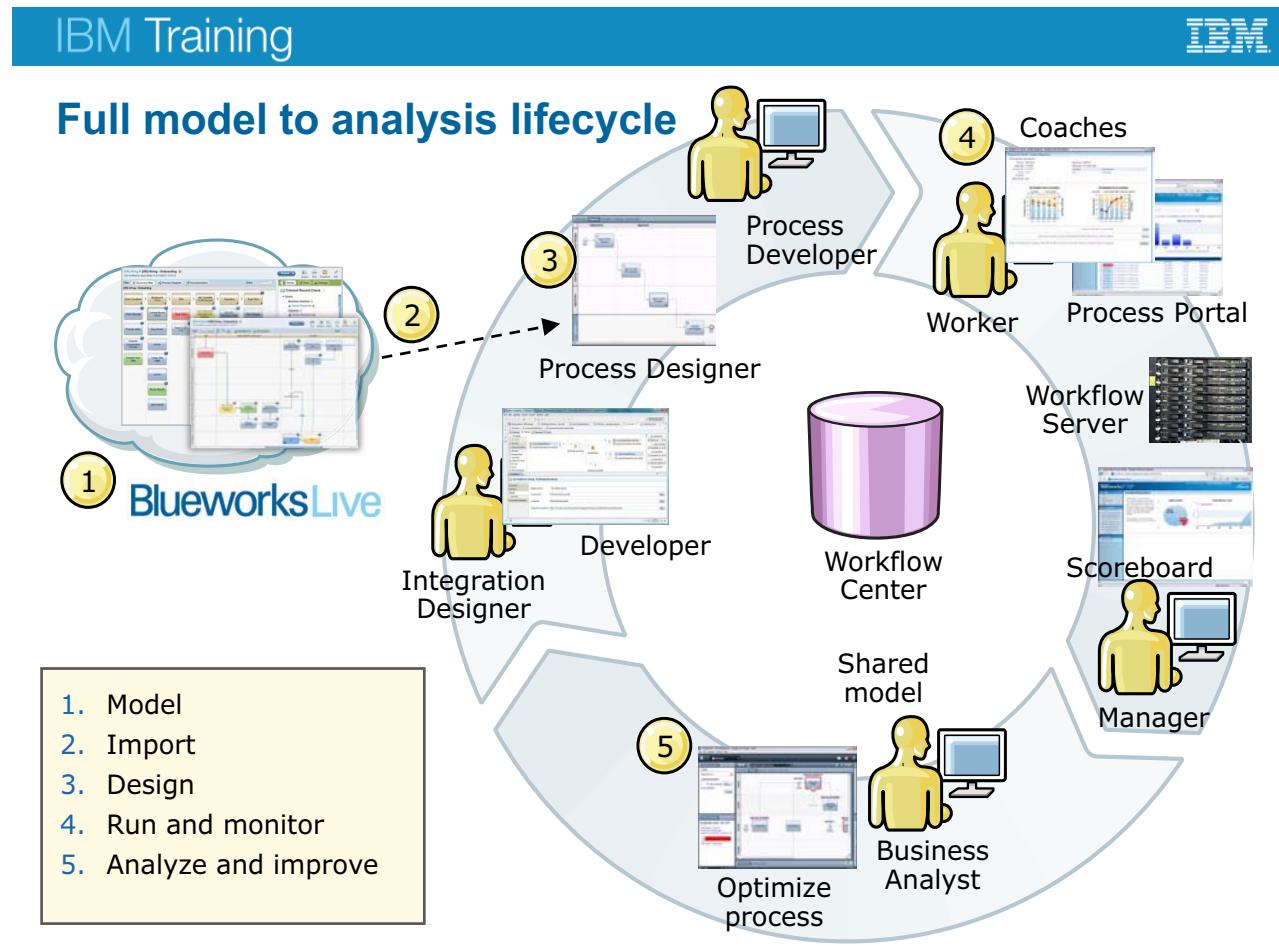
Performing process discovery with Blueworks Live

Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-34. Performing process discovery with Blueworks Live

Blueworks Live is an IBM tool used primarily for modeling and analysis of processes. In this topic, the tool and a few of its capabilities are explored.



Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-35. Full model to analysis lifecycle

The analysis lifecycle is as follows:

Blueworks Live is normally the starting point to a BPM implementation.

Process discovery precedes process implementation. It is the initial effort to determine viable processes to be implemented.

- 1. Model:** Create business processes in either Blueworks Live, a cloud-based, collaboration tool for in-process discovery, modeling, documentation, and process automation, or directly in IBM Business Automation Workflow.
- 2. Import:** As soon as it is created, import the process into Process Designer for detailed design and executions. Subscribe to process changes.
- 3. Design:** Add coaches, services, and other configuration objects.
- 4. Execute and monitor:** As processes execute, the Process Portal provides visibility to the number of processes, what activities are overdue, and how the organization is doing against SLAs.
- 5. Analyze and improve:** Process information is automatically captured, and the Process Designer gives the ability to analyze the information and simulate the impact of changes.

The cornerstone of this technology approach to BPM is the concept of the “shared model”: all parties that are involved in the effort to define, model, implement, measure, and improve the process are working from a common shared platform that encapsulates all of the various components. It involves the analyst who models the process, the developers who construct the detailed implementation of it, the participants who execute the process, and the process owner and analysts who monitor and identify improvements. They are all using the same tool, which uses the same definition of the process. The model of the process that the analysts and developers build is the same one that executes at run time. It also is the same one that is used to create reports on the current performance and status of the process and is the same one that is used to model and simulate potential improvements. The various components are as follows:

- **Blueworks Live**

Blueworks Live is a cloud-based, collaboration tool for process discovery, modeling, documentation, and process automation.

- **Workflow Center**

The Workflow Center includes a repository for all process artifacts and provides the tools that are required to develop, deploy, and manage process applications.

- **Workflow Server**

Workflow Server is an IBM Business Automation Workflow runtime environment that supports running a range of business processes and integrations. By using Workflow Server, you can run processes as you build them.

- **IBM Process Designer**

IBM Process Designer is a design-time tool that is used to develop process applications.

- **IBM Integration Designer**

IBM Integration Designer is a development environment for building end-to-end applications. It includes numerous prepackaged integration adapters to build complex automated processes. This tool is not covered in this course.



Blueworks Live and SaaS

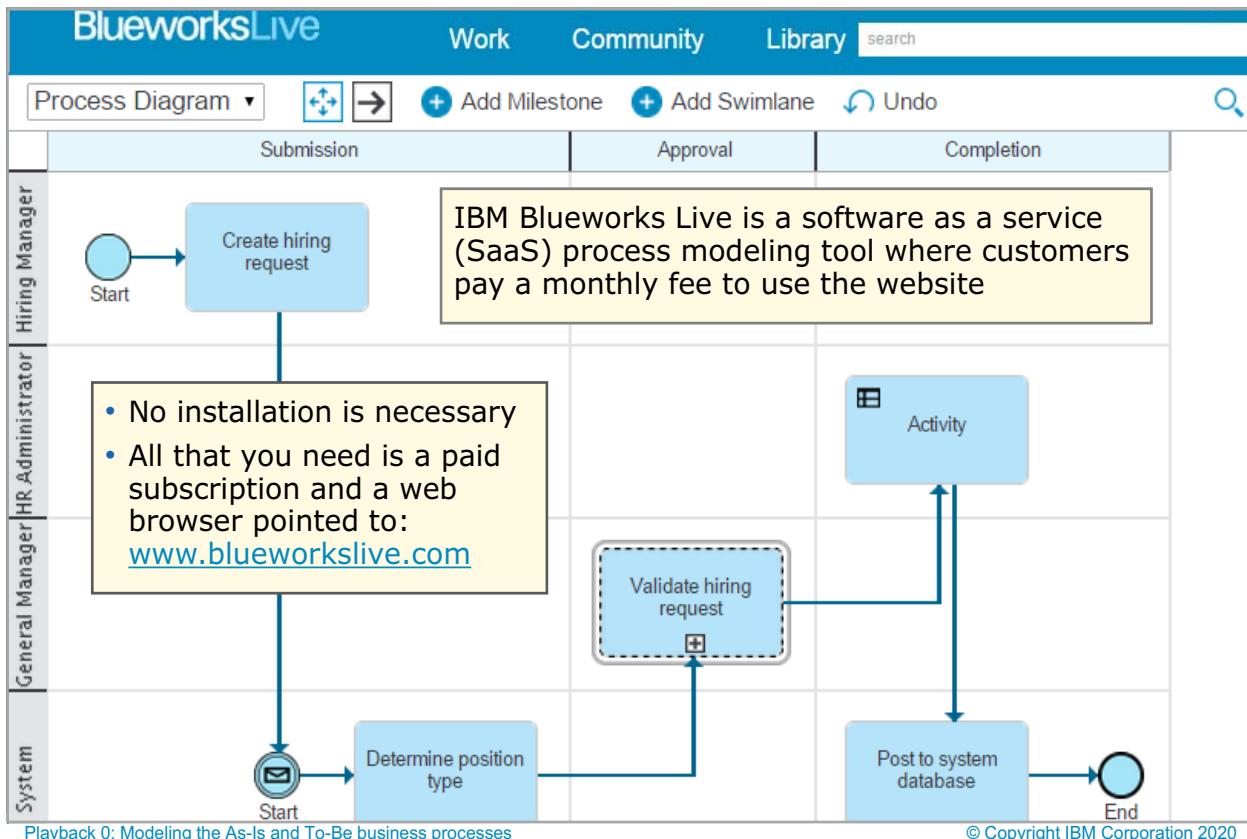


Figure 3-36. Blueworks Live and SaaS

IBM Blueworks Live is a software as a service (SaaS) tool where customers pay a monthly fee to use the website. It is frequently updated and improved based on user comments and requests. No installation is necessary. All that is needed is a paid subscription and web browser pointed to: www.blueworkslive.com

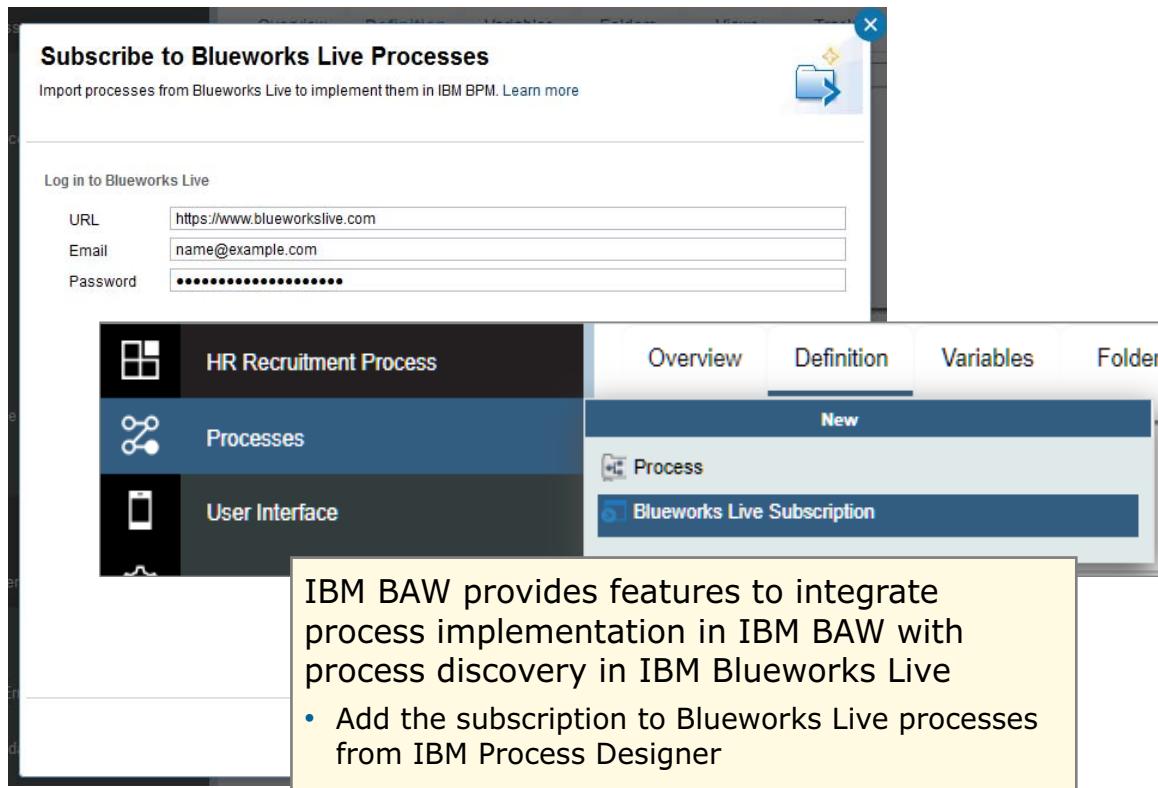
Blueworks Live is different from many other IBM products because it is sold as software as a service, or SaaS. Customers pay a monthly fee to subscribe to the service.

Blueworks Live requires only an internet connection and web browser to get started with modeling. Product installation and maintenance hassles are not an issue. It is a quick way to start mapping your processes and can be used for running small, simple processes.

Although IBM Blueworks Live is used as an example in the unit slides for modeling and analyzing your processes, you can also use other third-party modeling tools for process modeling.

IBM Training

Using IBM Blueworks Live and IBM BAW (1 of 2)



Playback 0: Modeling the As-Is and To-Be business processes

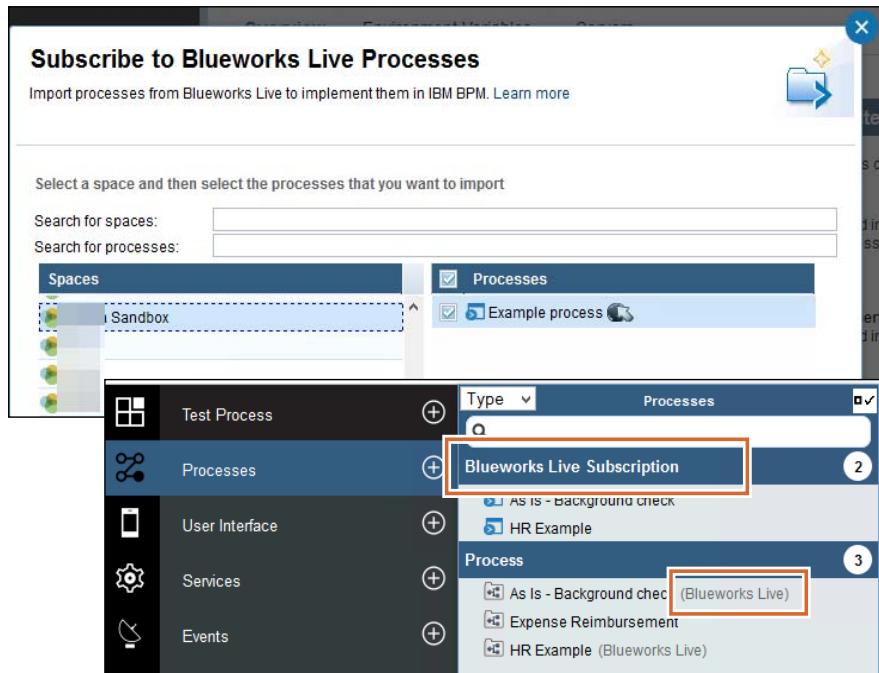
© Copyright IBM Corporation 2020

Figure 3-37. Using IBM Blueworks Live and IBM BAW (1 of 2)

You can add a subscription to IBM Blueworks Live processes by clicking the (+) plus sign next to the IBM Blueworks Live Processes section in IBM Process Designer.

Using IBM Blueworks Live and IBM BAW (2 of 2)

- After subscribing, you can locate your subscription items in the IBM Process Designer library
- After a process is imported, you still need to implement it



Note:

- For this course, you model the process using IBM Process Designer. However, many organizations start with Blueworks Live.

Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-38. Using IBM Blueworks Live and IBM BAW (2 of 2)

A business programmer can use the integration between IBM Blueworks Live and IBM BAW to implement the models that business analysts define. A business programmer already knows that the models accurately reflect the business needs of the user because business analysts established that when they used IBM Blueworks Live for process discovery. So now the business programmer can focus on quickly implementing those models in the Workflow Center and IBM Process Designer, by subscribing to the processes discovered in IBM Blueworks Live.

IBM Business Automation Workflow provides features to integrate process *implementation* in IBM BAW with process *discovery* in IBM Blueworks Live. As soon as a business analyst completes the process discovery phase, the process requirements are quickly delivered to IBM Business Automation Workflow. They can be implemented as a process by a business programmer by using IBM Process Designer.

When you click the artifact in the library that is listed under the Blueworks Live Subscription section, it brings up the settings for the subscription. You can then update your processes to check for newer versions and list out all the imported objects that were part of the Blueworks Live subscription. All the objects in the library with the (Blueworks Live) tag are the items that are part of the Blueworks Live subscription.

In this course, you do not use Blueworks Live. You model your processes by using IBM Process Designer.

To learn about Blueworks Live, you can enroll in one of the IBM Blueworks Live self-paced virtual classes:

- ZB031: *Process Discovery and Modeling in IBM Blueworks Live*
- ZB030: *IBM Blueworks Live Account Administration*

Try the full version of IBM Blueworks Live for 30 days without charge. Sign up at:

<https://www.blueworkslive.com>

When you import a process from Blueworks Live, the tasks are implemented as Inline Tasks. Once imported, the process can be run without having to perform any coding.

3.6. The Hiring Requisition Process

The Hiring Requisition Process

Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-39. The Hiring Requisition Process

The Hiring Requisition Process

- This use case is the example that is used throughout this course in the lab exercises.
- A company is experiencing much growth and must hire many people in a short amount of time. The process that you are going to examine, and model is called the Hiring Requisition process. This process covers a new job position through submission, approval, and completion so applicants can apply for the job position.
- The process requirements already went through an analysis stage and were refined.
- Review the requirements in the next three slides and see if you can find the potential activities in this process

Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-40. The Hiring Requisition Process

Core requirements (1 of 3)

- 1.1 A Hiring Manager submits a hiring requisition to the HR department
- The request contains the following information:
 - Requisition number
 - Date of request
 - Requester
 - Date position available
 - Job title
 - Job description
 - Job level
 - Number of direct reports
 - Division
 - Department
 - Salary to offer
 - Bonus amount
 - Hiring Manager comments
 - New position

[Playback 0: Modeling the As-Is and To-Be business processes](#)

© Copyright IBM Corporation 2020

Figure 3-41. Core requirements (1 of 3)

A company is experiencing much growth and must hire many people in a short amount of time. The process that you are going to examine and model is called the Hiring Requisition process. This process covers a new job position through submission, approval, and completion so applicants can apply for the job position.

The following process requirements already went through an analysis stage and were refined. For more information about how to get to this point in your process and complete the discovery, analysis, and refinement stages, consider taking the courses on process analysis methods.

Core requirements

- 1.1: A Hiring Manager submits a hiring requisition to the HR department. The request contains the following information:
 - Requisition number
 - Date of request
 - Requester
 - Date position available
 - Job title
 - Job description

- Job level
 - Number of direct reports
 - Division
 - Department
 - Salary to offer
 - Bonus amount
 - Hiring Manager comments
 - New position
- 2.1: If the answer to “New position” is yes, the request is forwarded to a General Manager. After the General Manager receives the request, the General Manager indicates approval or disapproval.
 - 2.2: If the request is not approved, the General Manager specifies a reason and the request is closed. If the request is approved, a salary compliance check is conducted.
 - 2.3: The Hiring Manager is notified of the General Manager’s decision after the General Manager’s approval step.
 - 2.4: When the hiring requisition is submitted, an automated system level checks for salary compliance. If the request meets salary compliance, the hiring request is automatically posted to the HR Positions database and made available for dissemination.
 - 2.5: When a request violates the established salary guidelines of the company, the HR Administrator can approve or reject the requested salary override.
 - 2.6: If the salary override is approved, the request is posted to the HR Positions database and made available for dissemination.
 - 2.7: If HR Administrators reject the requested salary, they must provide comments about the violation, add a proposed salary, and send the request back to the Hiring Manager who originated the request.
 - 2.8: When the Hiring Manager gets the request back because of a rejection, the Hiring Manager attempts to negotiate an adjusted salary or can cancel the request. If the negotiation is successful, the request is resubmitted back to the same HR Administrator.
 - 2.9: All hiring requests must be added to the HR Positions database regardless of the disposition at the end of the process during a finalization activity.
 - 2.10: The HR Administrator has 4 hours to complete the review. If the review is not completed within 4 hours, an email is sent to the HR Administrator. The email notifies the HR Administrator of the missed deadline.

Core requirements (2 of 3)

After the Hiring Manager submits a hiring requisition:

- 2.1: If the answer to “New position” is yes, the request is forwarded to a General Manager. After the General Manager receives the request, the General Manager indicates approval or disapproval.
- 2.2: If the request is not approved, the General Manager specifies a reason, and the request is closed. If the request is approved, a salary compliance check is conducted.
- 2.3: The Hiring Manager is notified of the General Manager’s decision after the General Manager’s approval step.
- 2.4: When the hiring requisition is submitted, an automated system level checks for salary compliance. If the request meets salary compliance, the hiring request is automatically posted to the HR Positions database and made available for dissemination.
- 2.5: When a request violates the established salary guidelines of the company, the HR Administrator can approve or reject the requested salary override.

[Playback 0: Modeling the As-Is and To-Be business processes](#)

© Copyright IBM Corporation 2020

Figure 3-42. Core requirements (2 of 3)

Review the requirements and see if you can find the potential activities in this process.

2.1 Points to an activity by the General Manager to approve the request.

2.2 Indicates a salary compliance check

2.3 This is an example of a notification, which is not the same as an activity

2.4 Indicates an activity that posts to the HR Positions database

2.5 Points to an activity where the HR Administrator approves the request (after the salary override)

Core requirements (3 of 3)

After the Hiring Manager submits a hiring requisition:

- 2.6: If the salary override is approved, the request is posted to the HR Positions database and made available for dissemination.
- 2.7: If HR Administrators reject the requested salary, they must provide comments about the violation, add a proposed salary, and send the request back to the Hiring Manager who originated the request.
- 2.8: When the Hiring Manager gets the request back because of a rejection, the Hiring Manager attempts to negotiate an adjusted salary or can cancel the request. If the negotiation is successful, the request is resubmitted back to the same HR Administrator.
- 2.9: All hiring requests must be added to the HR Positions database regardless of the disposition at the end of the process during a finalization activity.
- 2.10: The HR Administrator has 4 hours to complete the review. If the review is not completed within 4 hours, an email is sent to the HR Administrator. The email notifies the HR Administrator of the missed deadline.

[Playback 0: Modeling the As-Is and To-Be business processes](#)

© Copyright IBM Corporation 2020

Figure 3-43. Core requirements (3 of 3)

2.6 This further elucidates when the post to the HR Positions database occurs

2.7 This requirement states the Hiring Manager has an added activity if the HR Administrator rejects the requested salary.

2.8 This indicates a negotiation activity for the Hiring Manager.

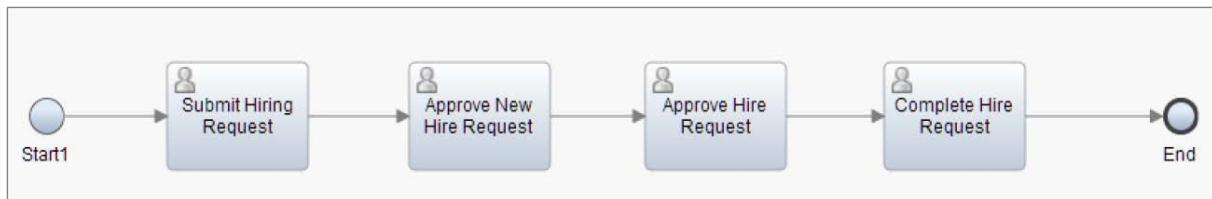
2.9 Another reference to the activity involving posting to the HR Positions database

2.10 This is an example of an escalation requirement.

For the purposes of this training, you partially implement this solution.

Translating business process work steps into activities

- Capturing the process information does not necessarily mean that the work steps that are captured are filtered into logical units of work.
- It depends on the depth of work that captures and documents the data. The better the process discovery, the more easily the work steps get translated into the process model activities.
- After reviewing the requirements, activities can be determined. For now, you concentrate on activities. Later, the logic behind decision making is incorporated by using Gateways.



Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-44. Translating business process work steps into activities

Capturing the process information does not necessarily mean that the work steps that are captured are filtered into logical units of work. It depends on the depth of work that captures and documents the data. The better the process discovery, the more easily the work steps get translated into the process model activities. Some of the steps must be conducted together to enable the entire unit of work to be completed. After reviewing the requirements, start by putting together a set of activities. Later, the logic behind decision making is incorporated by using Gateways.

1.1 The Hiring Manager submits a Hiring Request

2.1 The General Manager Approves the New Hire Requests

2.2 Salary compliance check performed by system

2.4 Post to HR Positions database performed by system

Other requirements point to Approval activities performed by the HR Administrator and Hiring Manager

Given there are other activities that are performed after the process is completed, a final activity labeled "Complete Hire Request" will serve to perform all these functions.

The following activities have been identified:

- Submit Hiring Request

- Approve New Hire Request
- Approve Hire Request
- Complete Hire Request

You build this model in the exercise at the end of this unit.

Unit summary

- Describe how to model a process
- Decompose activities into a nested process
- List and describe the core notation elements that are used in IBM Process Designer
- Describe the purpose and function of Blueworks Live
- Examine a defined workflow from detailed process requirements and identify the interrelated process activities and the roles that are responsible for completing them
- Describe the requirements of the Hiring Request Process use case

[Playback 0: Modeling the As-Is and To-Be business processes](#)

© Copyright IBM Corporation 2020

Figure 3-45. Unit summary

Review questions

1. True or False:

Lanes can also be assigned to systems, and automated tasks are often in the designated system lane.

2. _____ is a container for process models and their supporting implementations, and it is stored in the repository.

- A. Process application
- B. Pool
- C. Activity
- D. Team

Figure 3-46. Review questions

Write your answers here:

1.

2.

Review answers

1. True
2. A: Process application

Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2020

Figure 3-47. Review answers

Exercise: Playback 0: Creating the To-Be process

Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2019

Figure 3-48. Exercise: Playback 0: Creating the To-Be process

Exercise introduction

- Translate business process workflow steps that are documented in the process discovery and analysis into process model tasks
- Create the foundation for a process by adding the appropriate lanes to the default pool
- Model the expected process flow for the initial process model
- Decompose business process workflow steps that are documented in the process discovery and analysis into process model tasks
- Create a linked process



Playback 0: Modeling the As-Is and To-Be business processes

© Copyright IBM Corporation 2019

Figure 3-49. Exercise introduction

Unit 4. Playback 0: Controlling process flow

Estimated time

01:30

Overview

This unit covers the modeling of process flow, sequence flow, tokens, gateways, and intermediate events.

How you will check your progress

- Review
- Exercise

Unit objectives

- Describe process sequence flow and the runtime use of process tokens
- List and describe gateways as they are used in IBM Process Designer
- Explain how to evaluate conditions for a process gateway
- Model gateways in a process
- List and describe intermediate event types that are used in IBM Process Designer
- Model a business process escalation path with an attached timer intermediate event

[Playback 0: Controlling process flow](#)

© Copyright IBM Corporation 2020

Figure 4-1. Unit objectives

Topics

- Process sequence flow
- Introduction to tokens
- Directing sequence flow by using gateways
- Interrupting the process by using intermediate events

Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-2. Topics

Key concepts in this unit

- **Process flow:** Encompasses both the normal, expected process path to completion, and alternative process paths that might occur with different process conditions or business rules
- **Tokens:** Describe how the process flows when the process is run
- **Gateways:** Used to control the flow of a process
 - Split, join, exclusive, inclusive, and parallel
- **Intermediate events:** An event that takes place between a start and an end event in the process
 - Sequence flow intermediate events: Message, content, timer, and tracking
 - Boundary (attached) intermediate events: Error, message, content, and timer

Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-3. Key concepts in this unit

4.1. Process sequence flow

Process sequence flow

Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-4. Process sequence flow

Comprehensive process models have one thing in common: they communicate process flow well. Process flow encompasses both the normal, expected process path to completion, and alternative process paths that might occur with different process conditions or business rules. To understand how to communicate both kinds of process flows in the process model, it is important to understand what types of sequence flow exist in process modeling and how to implement gateways.

Normal sequence flow



Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-5. Normal sequence flow

What is sequence flow?

Sequence flow is considered to be part of a category of connecting objects. These objects connect each element on the diagram to indicate the order in which elements are conducted.

Normal sequence flow is the simplest example of sequence flow: the connection of two flow objects. A plain arrow represents a normal sequence flow. It is the type of flow that was used when you connected elements in the last unit.

With normal sequence flow, the business process progresses to the next step in the process as soon as the first step is completed.

Conditional sequence flow



- BPMN standards use a small diamond with an arrow to represent the conditional sequence flow
- The difference between normal and conditional sequence flows is that conditional flows are not automatically followed. The condition must be met first.
- The number of conditional flows that are followed is determined according to the type of element that is used

Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-6. Conditional sequence flow

Conditional sequence flow is associated with gateways. Conditional expressions are evaluated to determine which path the flow is going to take. The difference between normal and conditional sequence flows is that conditional flows are not automatically followed. The condition must be met first.

The number of conditional flows that are followed is determined according to the type of element that is used and the requirements of the instance that is being processed.

An arrow with a diamond at the start of the arrow represents a conditional sequence flow.

Default sequence flow



- A plain arrow with a backslash at the beginning of the flow line
- A default sequence flow is required and is automatically created when a conditional sequence flow exists
- The default sequence flow indicates a processing path to follow when none of the conditions on the conditional sequence flows are true
- The default sequence flow ensures that there is at least one processing path (the default path) for the business process to follow

Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-7. Default sequence flow

When using conditional sequence flow in IBM Process Designer, a default flow is required. The default flow indicates a processing path to follow when none of the conditions on the conditional flows are true. It allows at least one path of processing (the default path) for the business process to follow.

This type of sequence flow has a slash that is added to the beginning of the flow line. You notice when you draw sequence flows to and from certain elements that the slash immediately shows.

4.2. Introduction to tokens

Introduction to tokens

Playback 0: Controlling process flow

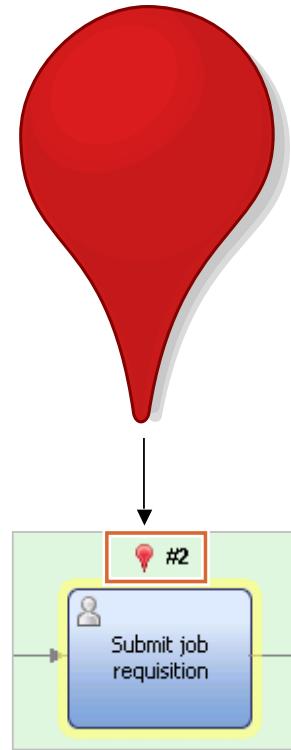
© Copyright IBM Corporation 2020

Figure 4-8. Introduction to tokens

Tokens are used to describe how the process flows when the process is run.

The indication of a token on a step identifies the location of active processing steps of that business process.

What is a token?



Playback 0: Controlling process flow

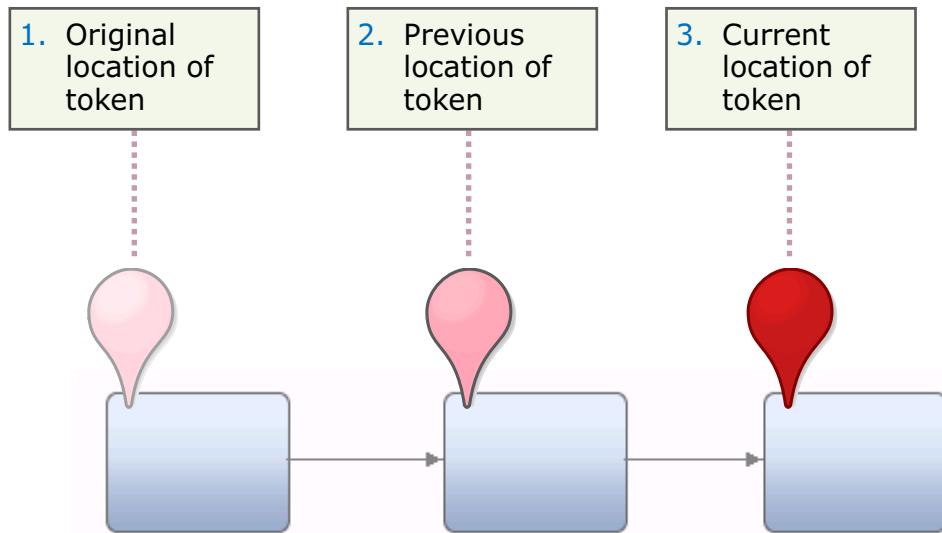
© Copyright IBM Corporation 2020

Figure 4-9. What is a token?

A token is used in two ways:

- To display to developers, designers, and business the task it is completing.
- To help in understanding the flow through a business process, which is based on the type of gateway that is chosen at the design stage and different use case scenarios. Tokens can help to determine whether the correct gateway is selected based on the business requirements.

Traverses the flow objects

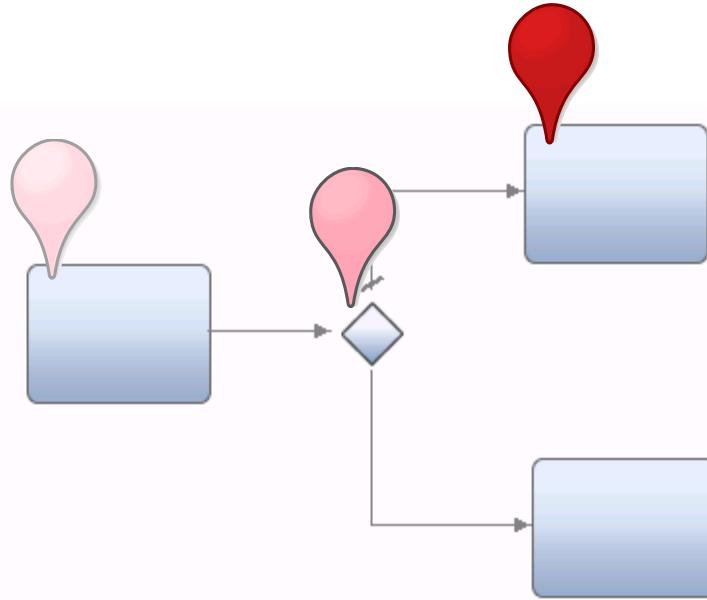


Note: The different shades of the token that you see in this course are for class illustration purpose, and only the current location of the token displays in the process in the tool

Figure 4-10. Traverses the flow objects

An example of serial processing is shown, along with the explanation for the placement of the token. As each step is completed, a token identifies the next step that is required until the path of processing reaches an end event. In this case, the third step is active, which shows the red token on that step. The two previous steps are completed in order, and the lighter colored tokens are emphasizing that fact.

Diverted through alternative paths



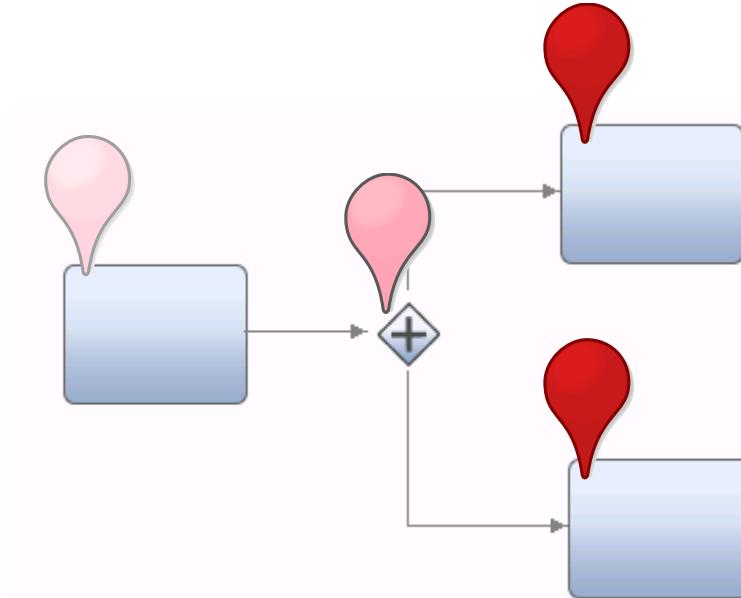
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-11. Diverted through alternative paths

Tokens can help in the understanding and identification of a particular path of processing that an instance takes. In this slide, you see that by tracking how the token gets diverted through alternative paths, the sequence flow that the use case requires is definable.

Split into parallel paths



- A parallel gateway is used here, and is covered later in this unit

Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-12. Split into parallel paths

In many situations, more than one step within a business process can be active. The token allows the identification of all active steps of a business process. Each active step within the business process is highlighted or shown with a token. Here you can see that after the token passed through the gateway, the business conditions required that two steps of the business process must be run concurrently.

4.3. Directing sequence flow by using gateways

Directing sequence flow by using gateways

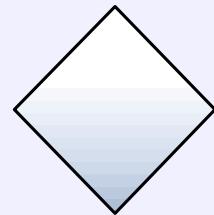
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-13. Directing sequence flow by using gateways

Often, a gateway is used to control the flow of a process.

Represented as a diamond



Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-14. Represented as a diamond

A gateway is represented as a diamond.

Can be thought of as a question



Is loan approved

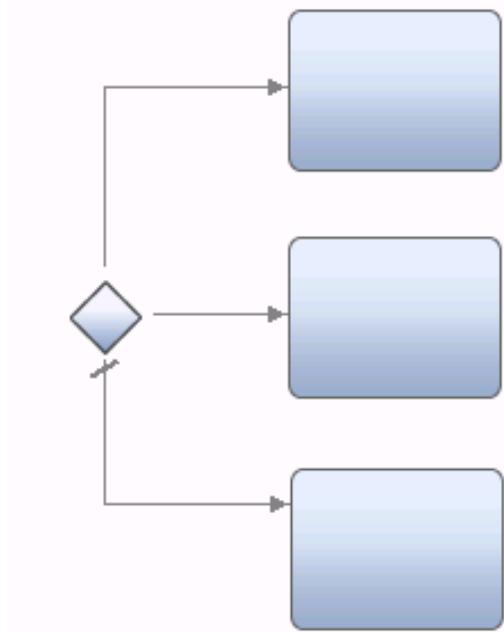
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-15. Can be thought of as a question

Often, a gateway is representative of a question that is asked at a particular point in a process. When you label the gateway, you automatically assume that the label is a question, so the question mark at the end of the question is implied.

Has a defined set of alternative answers



Playback 0: Controlling process flow

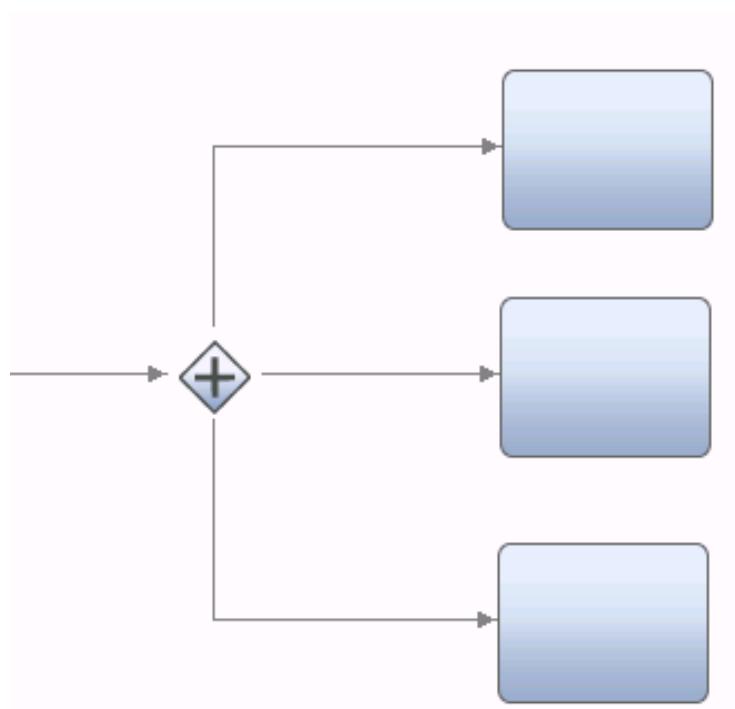
© Copyright IBM Corporation 2020

Figure 4-16. Has a defined set of alternative answers

The question has a defined set of alternative answers. All of the answers can be thought of as gates that are keeping the process from continuing until a valid answer is provided for the question.

Gateways control the divergence and convergence of sequence lines, determining branching and merging of the paths that a process can take.

Two distinct modes: Split



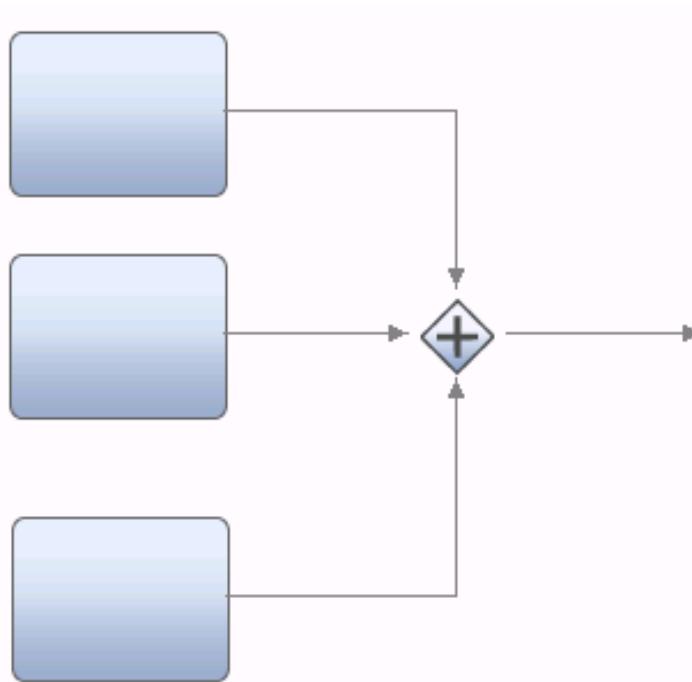
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-17. Two distinct modes: Split

Typically, gateways have two distinct modes. One mode is that a gateway can split an incoming path into multiple outgoing paths. It is known as a split.

Two distinct modes: Join



Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-18. Two distinct modes: Join

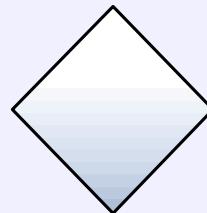
The second mode for a gateway is that it can merge multiple incoming paths into one outgoing path. It is known as a join.

In IBM Process Designer, an exclusive gateway is not used to join multiple tokens. Inclusive and parallel gateways allow for joins to be used.

When using gateway splits and joins, keep in mind the following information:

- Gateway splits allow for activities to occur at the same time.
- In some situations, an action or actions must not proceed until a set of previous activities are completed. An example of this gateway is a summary task of the results of previous actions.
- For this example situation, you need an accompanying join to make the process work in a simple, sensible manner. It also makes the process diagram easy to understand by various audiences.
- A good rule to remember is that when modeling splits and joins, you have one token into the process and one token out of the process.

Exclusive: Diamond shape with no internal marker



Exclusive
gateway

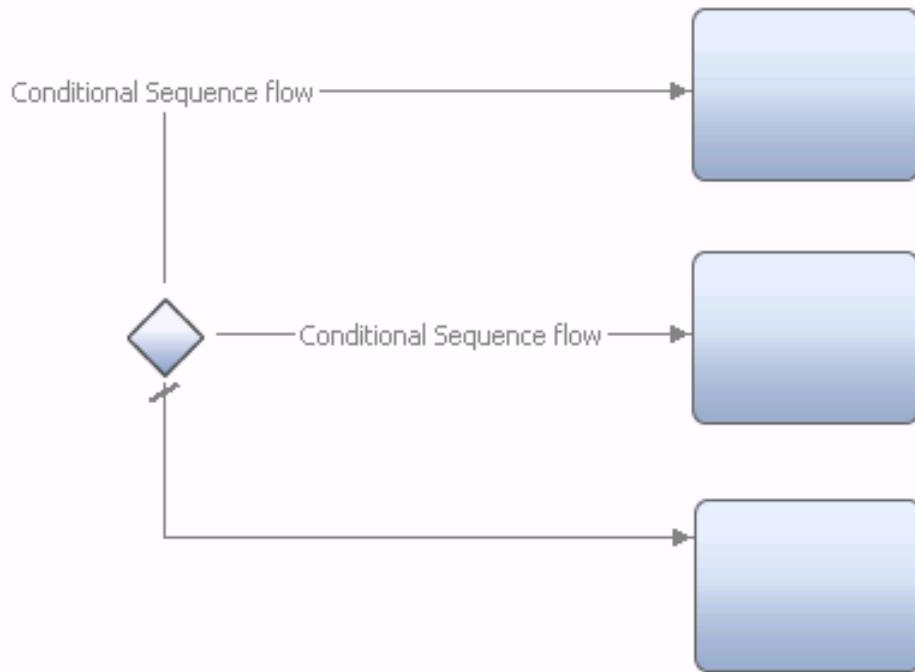
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-19. Exclusive: Diamond shape with no internal marker

The first type of gateway you examine is the exclusive gateway. This gateway is sometimes called an XOR gateway. If you want to send the process flow along only one of the available sequence flows, use an exclusive gateway.

Exclusive: One or more outgoing conditional sequence flows



Playback 0: Controlling process flow

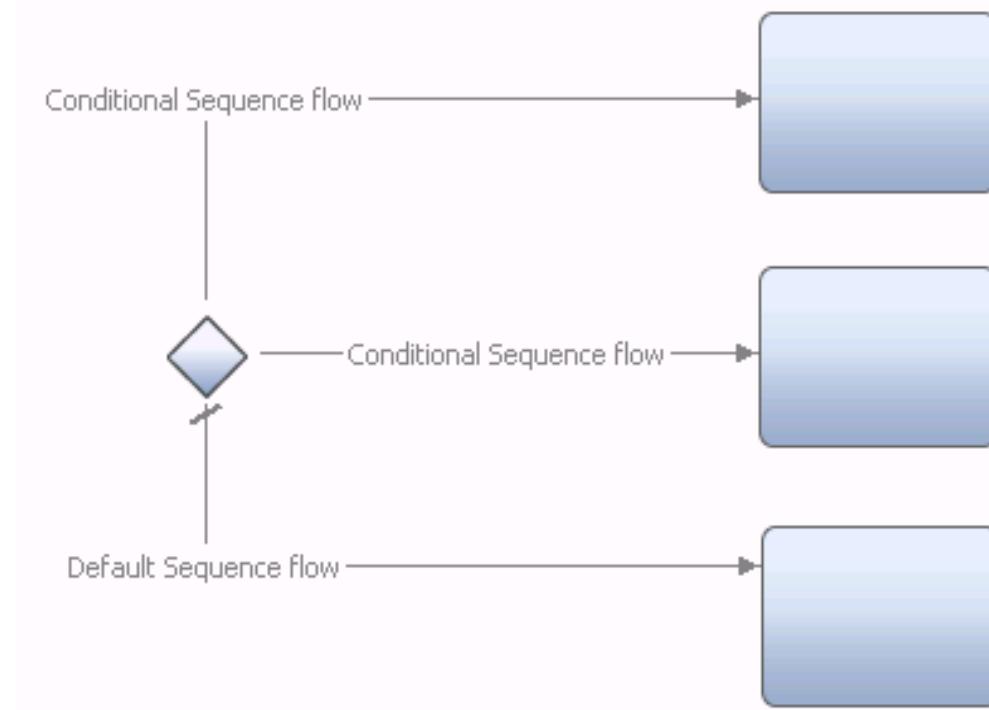
© Copyright IBM Corporation 2020

Figure 4-20. Exclusive: One or more outgoing conditional sequence flows

Outgoing sequence flow conditions are evaluated from top to bottom as defined in the properties tab of the gateway.

- It is a good practice to model exclusive gateways so that only one outgoing sequence flow condition can be true.
- After a business process condition is met, that conditional sequence path is followed, and evaluation of subsequent outgoing sequence flow conditions stops.
- If no conditions are met, the default processing path is followed.

Exclusive: Default sequence flow



Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-21. Exclusive: Default sequence flow

Because this gateway uses conditional sequence flow, an outgoing default sequence flow (a line with no condition) must be modeled with exclusive gateways. It allows the process to proceed even if none of the conditions on the conditional processing paths evaluate to true.

Process narrative

Submit auto damage claim:

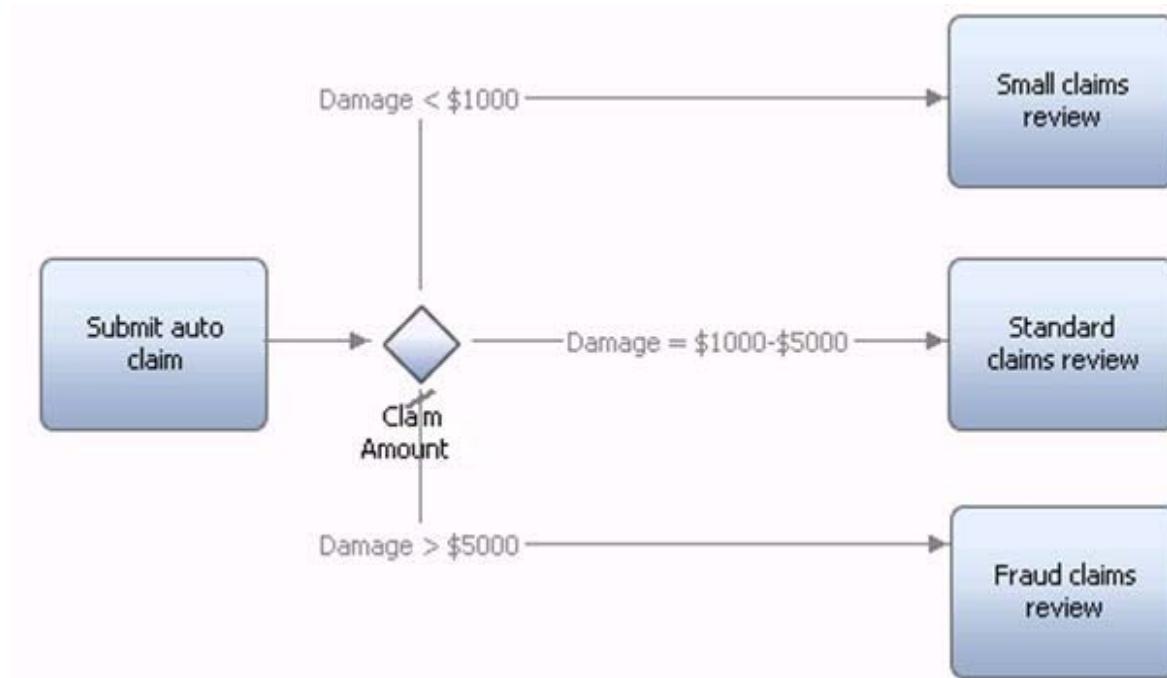
- If claim amount is less than \$1000, conduct a small claims review
- If claim amount is \$1000 to \$5000, conduct a standard claims review
- If claim amount is greater than \$5000, conduct a fraud claims review

Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-22. Process narrative

Submit auto claim example (1 of 6)



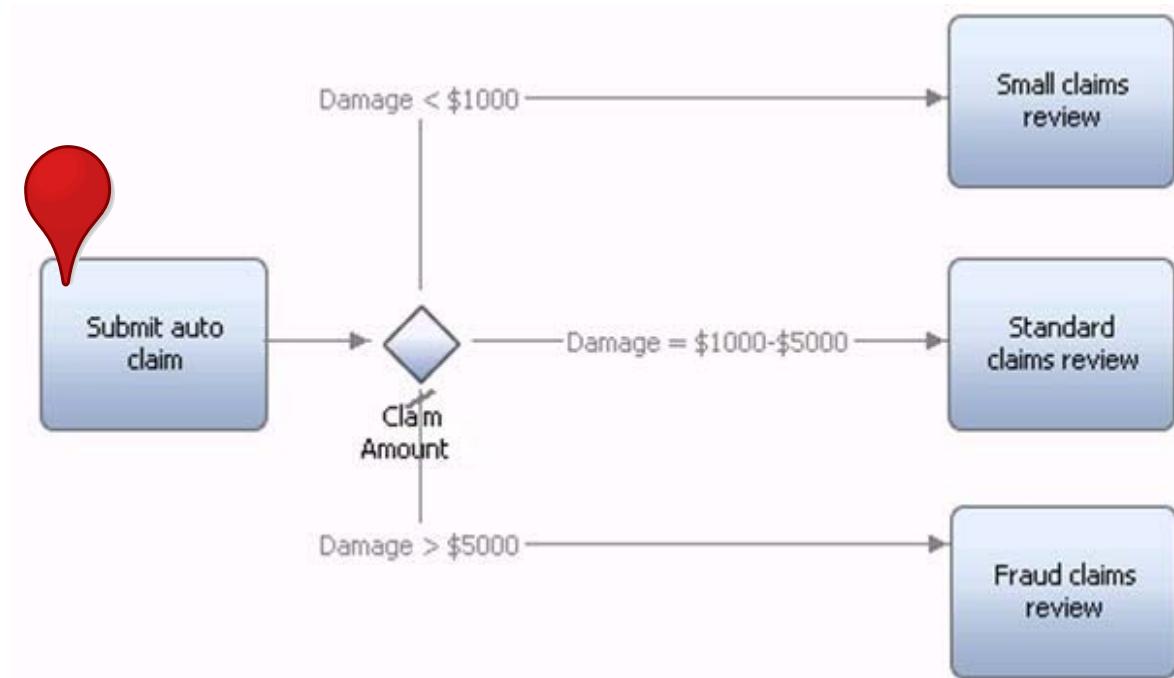
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-23. Submit auto claim example (1 of 6)

This example has an exclusive gateway. The activity labels are not all verb-noun pairings. Sometimes the available space is not enough for a full name. If it happens this way during your modeling, the key is to make sure that your labels communicate the process clearly.

Submit auto claim example (2 of 6)



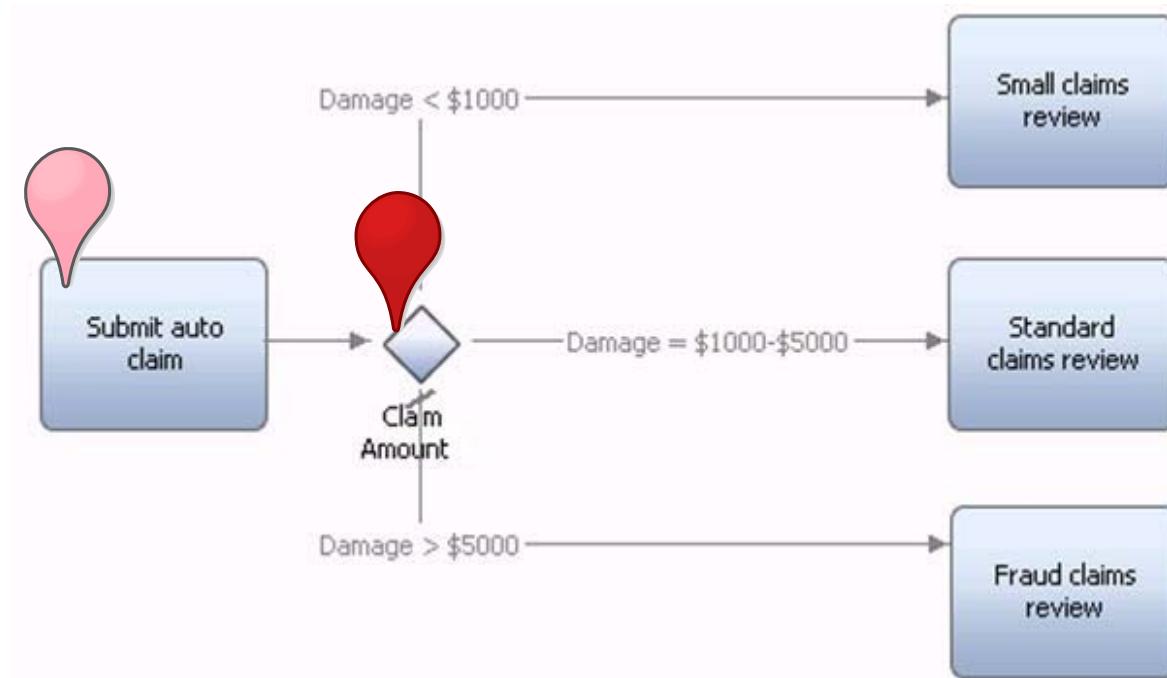
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-24. Submit auto claim example (2 of 6)

This diagram shows an example with tokens.

Submit auto claim example (3 of 6)



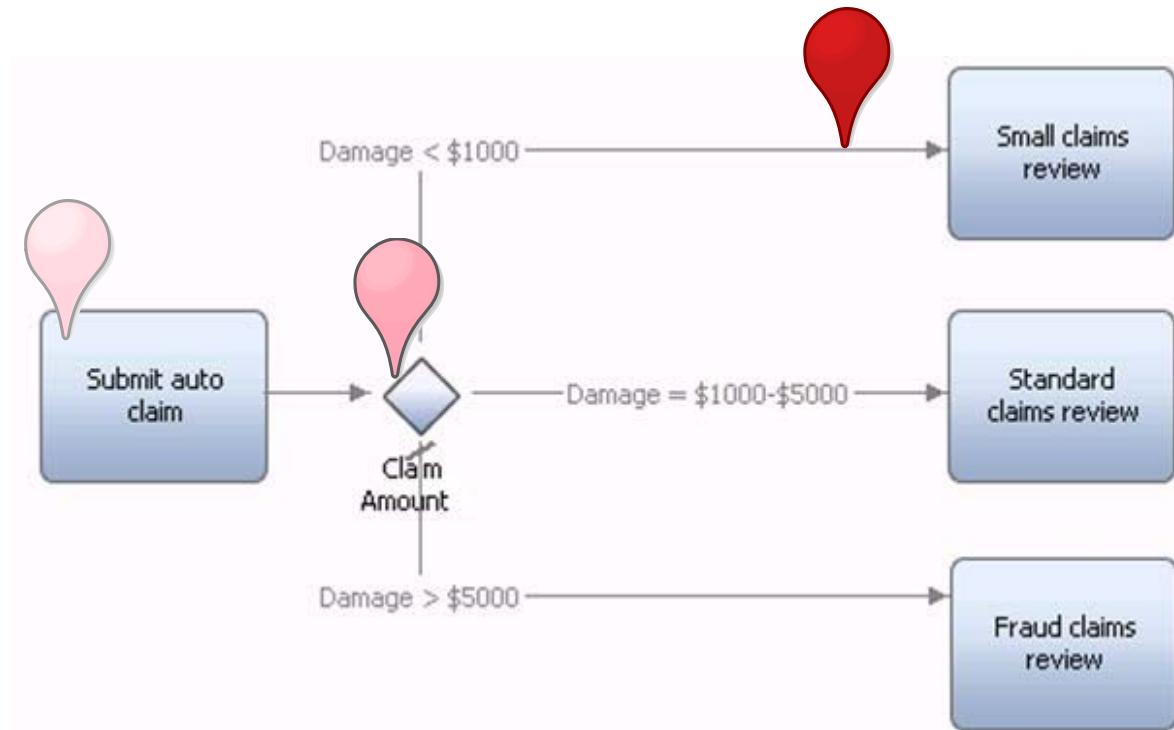
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-25. Submit auto claim example (3 of 6)

The lighter tokens show where the token was. These lighter tokens are shown to further understanding and are not part of the software. The red token indicates where the token is. When the user submits the auto claim, the token flows to the decision gateway. The logic in the decision gateway examines the claim amount, and the token moves to the appropriate flow, depending on the rules that are embedded in the gateway.

Submit auto claim example (4 of 6)



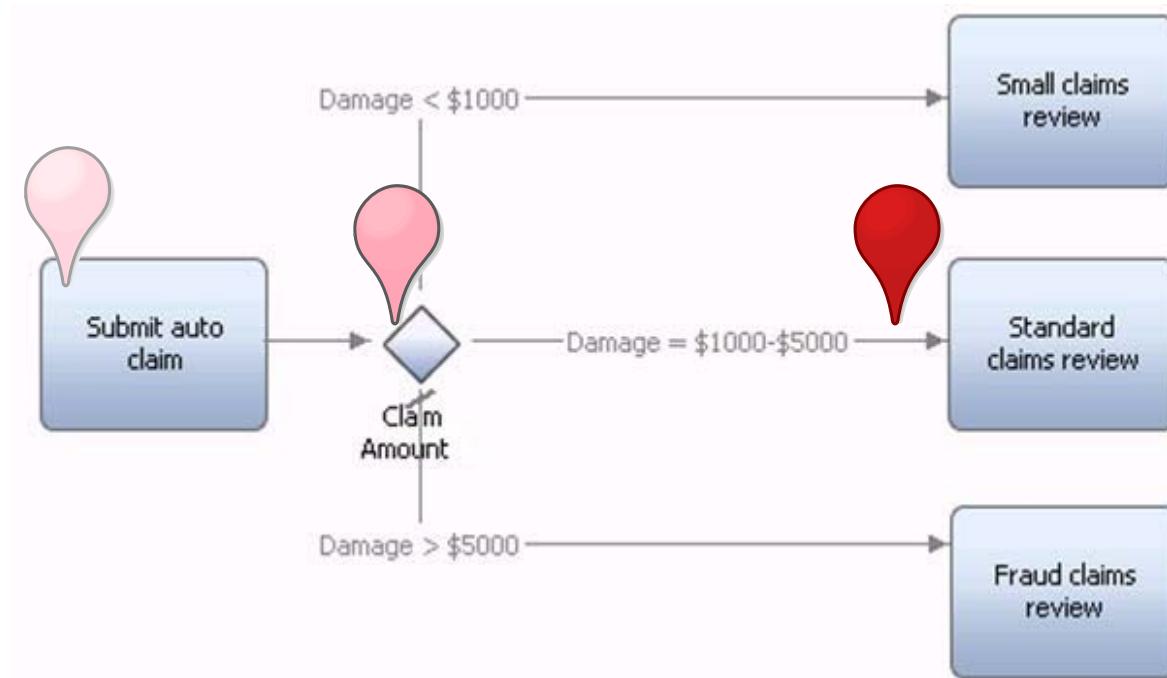
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-26. Submit auto claim example (4 of 6)

This path is taken if the claim is under \$1000.

Submit auto claim example (5 of 6)



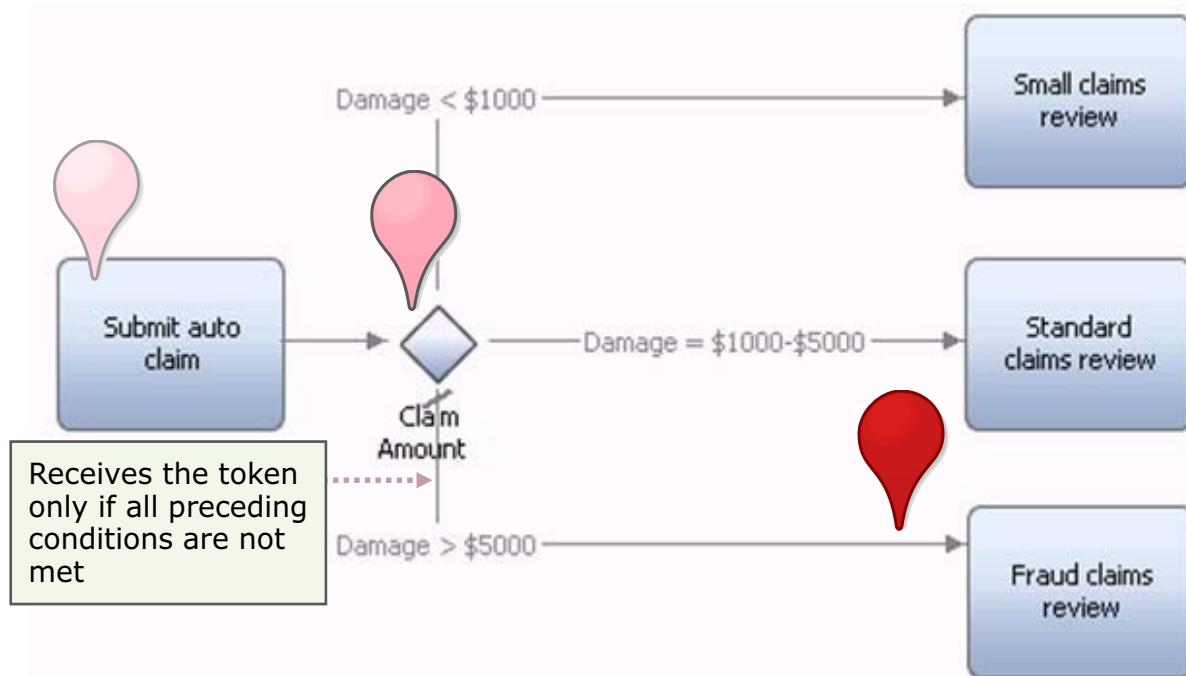
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-27. Submit auto claim example (5 of 6)

This path is taken if the claim is between \$1000 and \$5000.

Submit auto claim example (6 of 6)



Playback 0: Controlling process flow

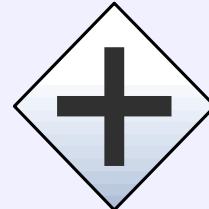
© Copyright IBM Corporation 2020

Figure 4-28. Submit auto claim example (6 of 6)

If the rest of the conditions are not met, the default sequence flow is taken. Even though the flow is labeled as `Damage > $5000`, the gateway evaluates only the top two conditions. When none of the conditions evaluate to true, the default sequence flow is followed.

If the damage is under \$5000, notice that it is not meeting the condition. It is taking that line only because none of the others were met. You always label the default line with the appropriate label, but the label is not functional; the implementer applies the logic to the gateway to determine when the token flows down the default flow.

Parallel gateway: Diamond with an internal plus



Parallel
gateway

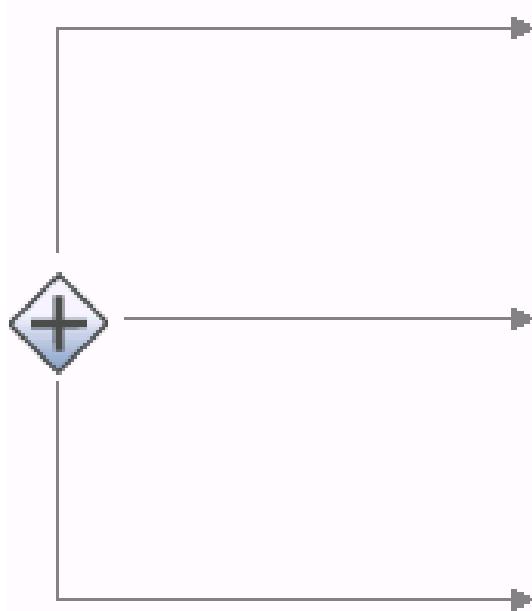
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-29. Parallel gateway: Diamond with an internal plus

The next type of gateway is a parallel gateway (AND). The split mode is called a parallel split, and the join is a parallel join.

Parallel split: No conditional or default flows



Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-30. Parallel split: No conditional or default flows

Parallel split gateways are used to direct the process flow along every sequence flow in parallel. Parallel split gateways have no conditional or default flows. Every exiting sequence path is followed.

A parallel gateway also has a join capability, and it is called a **parallel join gateway**. Situations might occasionally occur in which the business criteria caused multiple sequence flows to be followed in parallel, and the business criteria now needs the sequence flow to be joined.

The parallel join has the following capabilities:

- Any number of incoming sequence flows can be modeled.
- After all incoming sequence flows reach the parallel join, the outgoing sequence flow is followed.
- The parallel join is unable to determine whether an incoming sequence flow is no longer live. Care must be taken when designing the parallel join to ensure that all sequence flows reach it. If you cannot be sure that all incoming sequence flows are followed for every instance likely to be processed, do not use a parallel join.

Process narrative

New hire onboarding process:

- On the first day of employment, employees must complete the HR new hire forms

Then, they must:

- Apply for a security badge
- Requisition a computer
- Apply for a network ID and email address

[Playback 0: Controlling process flow](#)

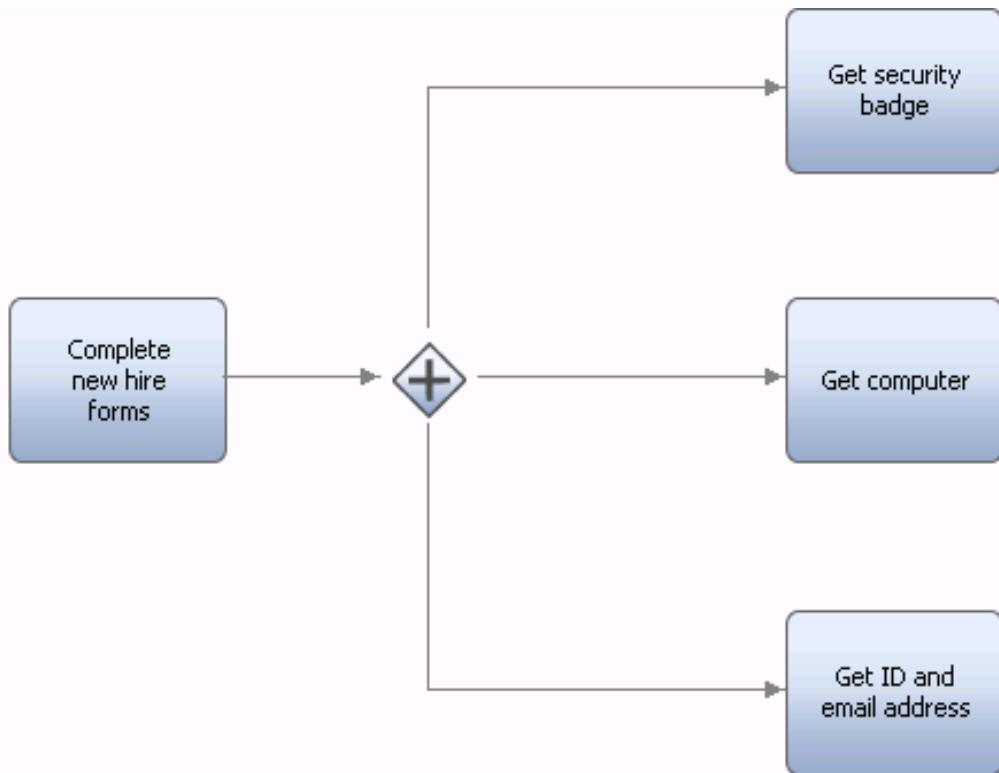
© Copyright IBM Corporation 2020

Figure 4-31. Process narrative

You can try creating the model by using this process narrative.

Next, you see a simple example of a parallel split that uses this scenario.

New hire onboarding example (1 of 4)



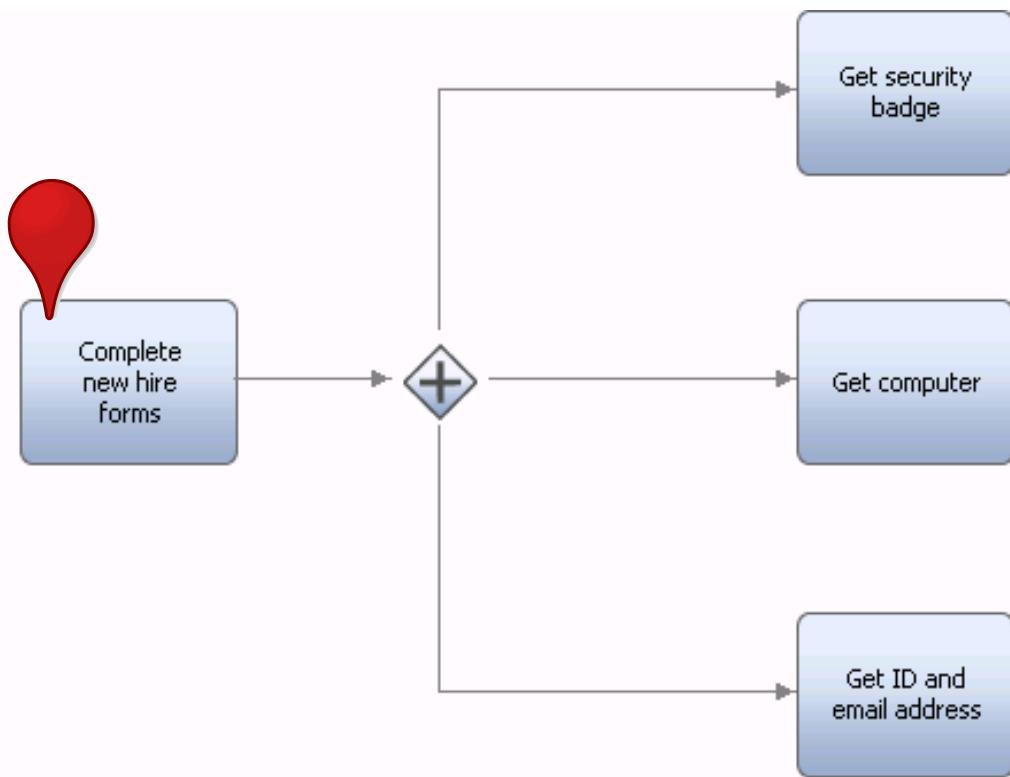
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-32. New hire onboarding example (1 of 4)

This example has a parallel split gateway.

New hire onboarding example (2 of 4)



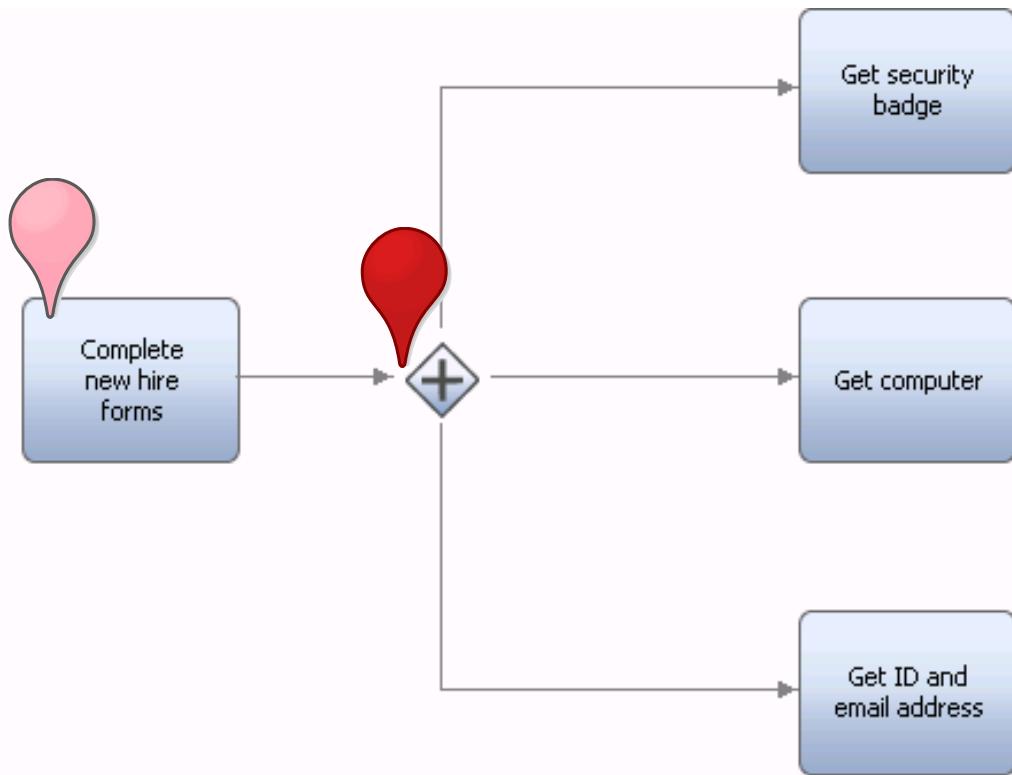
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-33. New hire onboarding example (2 of 4)

This example uses tokens.

New hire onboarding example (3 of 4)



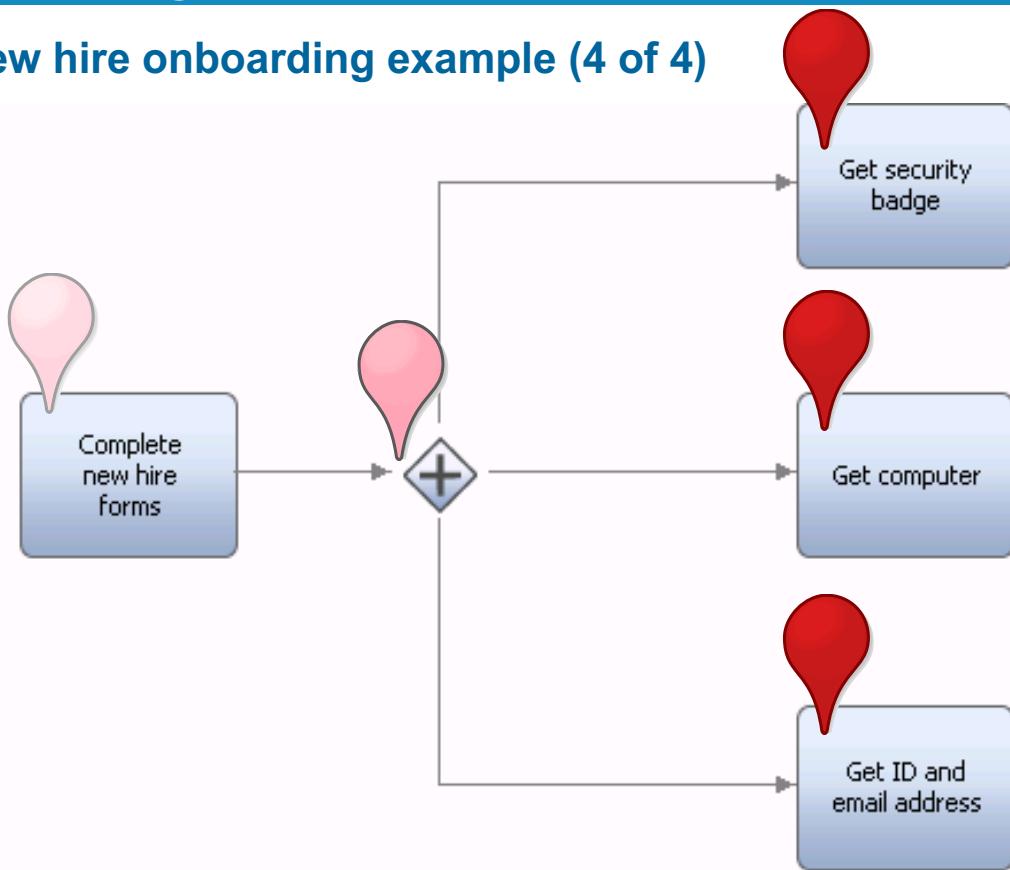
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-34. New hire onboarding example (3 of 4)

The token is now on the gateway.

New hire onboarding example (4 of 4)



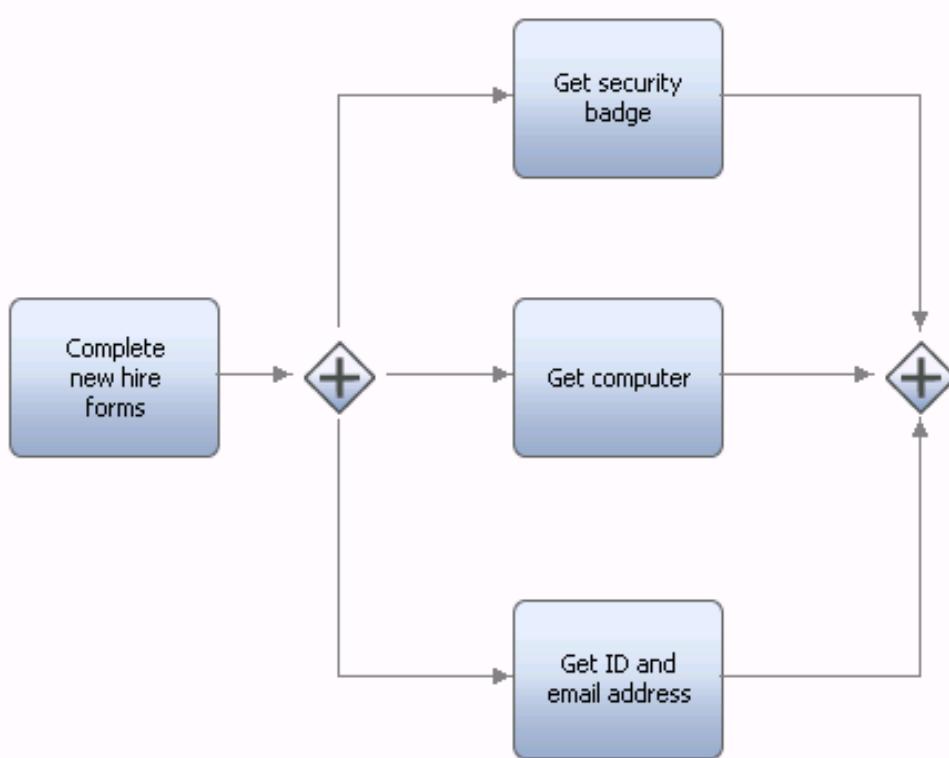
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-35. New hire onboarding example (4 of 4)

No conditions exist on the lines of a parallel split, so all paths are taken at the same time.

Parallel join



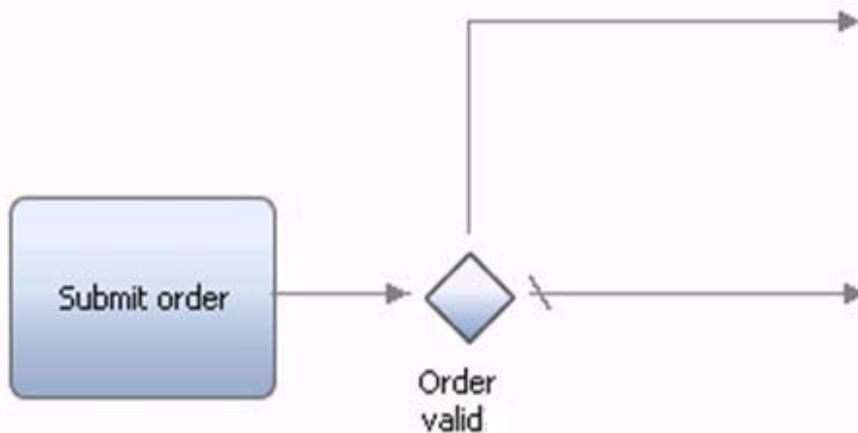
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-36. Parallel join

After all the tasks are completed, you must consider how the rest of the flow is going to occur. Adding a join to your diagram is a good practice. In this case, a parallel join is used to “collect” all the tokens before moving further down the process.

Evaluating conditions: Decision logic in the outgoing sequence flow



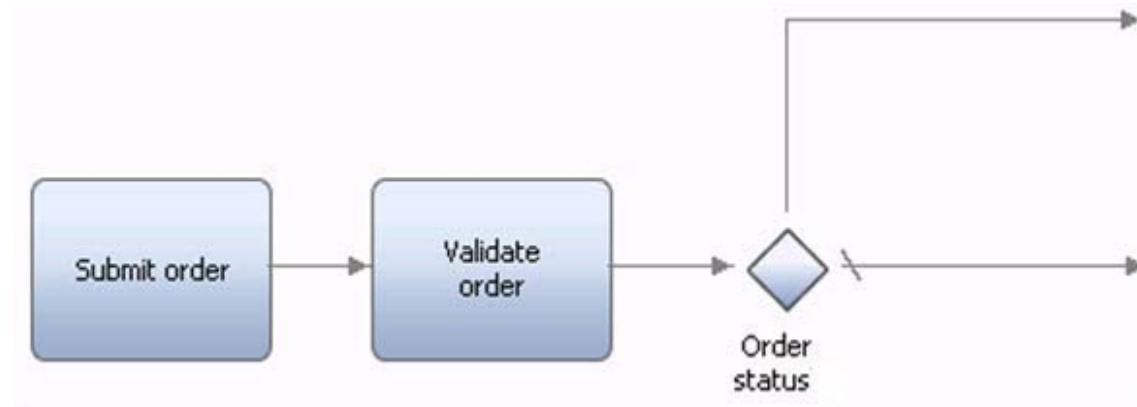
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-37. Evaluating conditions: Decision logic in the outgoing sequence flow

Exclusive gateways use conditions. These gateways allow the evaluation of these conditions to determine whether they are true or false. If the conditions are simple expressions of process data, you can put the decision logic in the outgoing sequence flows of the gateway.

Evaluating conditions: Externalized decision logic



[Playback 0: Controlling process flow](#)

© Copyright IBM Corporation 2020

Figure 4-38. Evaluating conditions: Externalized decision logic

If the conditions are not simple expressions of process data, the good practice is to externalize the decision logic to make it independent of the process model.

Use an activity before the gateway to decide what path to take. Then, use the outgoing sequence flows from the gateway to route the flow based on the decision, as shown in the diagram. This method is the most explicit way to model decision logic used by a gateway.

Gateways can also use a rule service or decision service as part of the implementation of the gateway. This approach has the advantage that it eliminates the creation of a system lane activity on the process to provide the logic for the gateway. The disadvantage is that the logic is hidden from implementers who maintain the code. Both approaches are effective, but it is up to the implementer to decide which approach works best for the organization.

Many consultants prefer to be explicit with an activity on a process instead of hiding the implementation details for the decision service inside the configuration for the decision gateway. Now that the configuration option exists, some consultants prefer to keep their processes clean and implement the logic inside the gateway configuration.

4.4. Interrupting the process by using intermediate events

Interrupting the process by using intermediate events

Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-39. Interrupting the process by using intermediate events

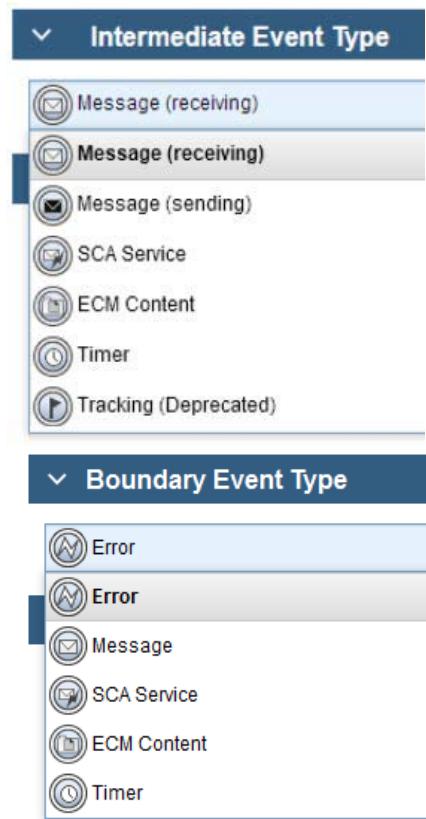
If an event takes place between a start and an end event in the process, it is called an intermediate event.

The intermediate event is designated by a double-lined circle, and an internal marker specifies the type of intermediate event that is taking place.



Intermediate events

- Five types of sequence flow intermediate events:
 - Message
 - SCA service
 - ECM content
 - Timer
 - Tracking
- Five types of boundary (attached) intermediate events:
 - Error
 - Message
 - SCA service
 - ECM content
 - Timer



Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

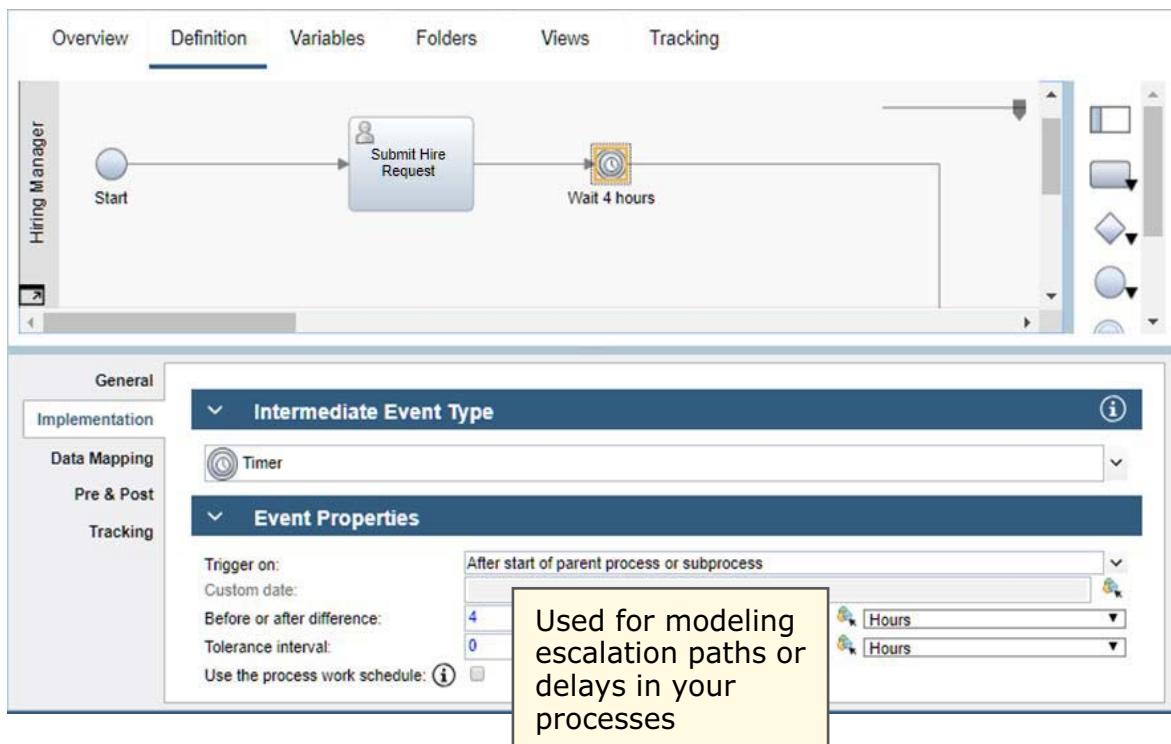
Figure 4-40. Intermediate events

Intermediate events have the following characteristics:

- The five main types of sequence flow intermediate events are message, SCA service, ECM content, timer, and tracking.
- The five main types of boundary (attached) intermediate events are error, message, SCA service, ECM content, and timer.
- Intermediate events are drawn as a double circle with an internal marker that identifies the type of event.
- All intermediate events behave the same way; they respond to a specific event. However, the implementation of each intermediate event, whether attached or sequence flow, differs.



Intermediate events: Timer



Playback 0: Controlling process flow

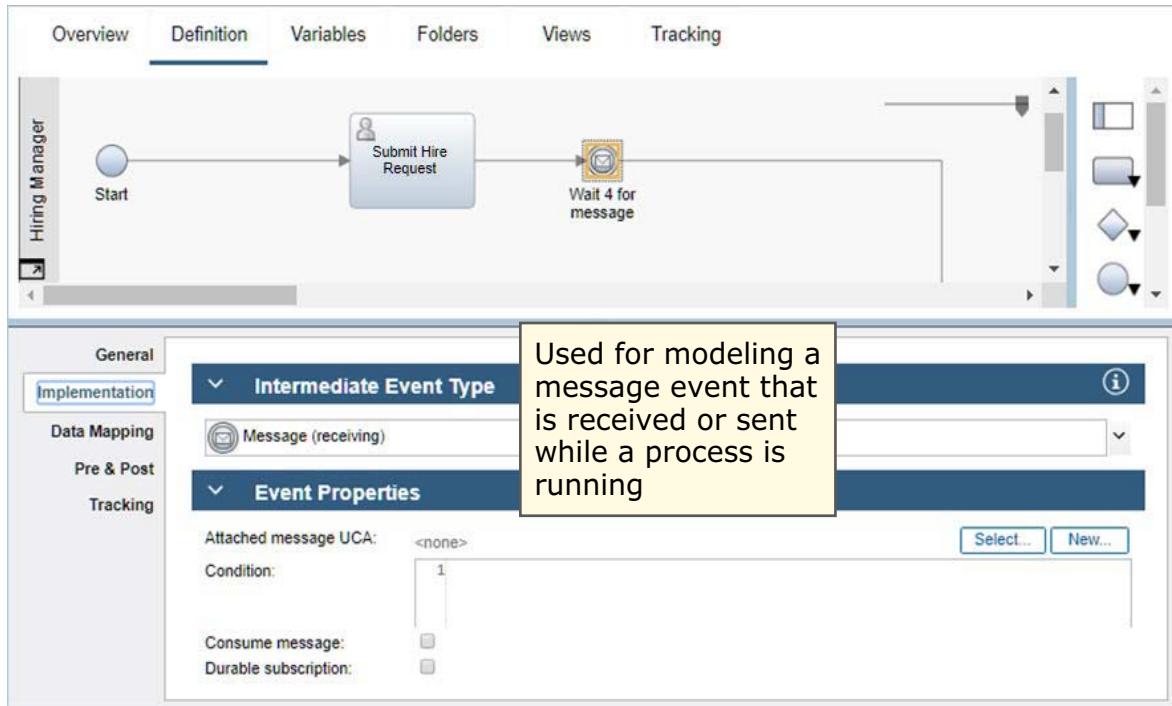
© Copyright IBM Corporation 2020

Figure 4-41. Intermediate events: Timer

A timer intermediate event is used to model escalation paths or delays in a process. By using a timer intermediate event, developers can specify a time interval after or before which some activity is conducted.



Intermediate events: Message



Playback 0: Controlling process flow

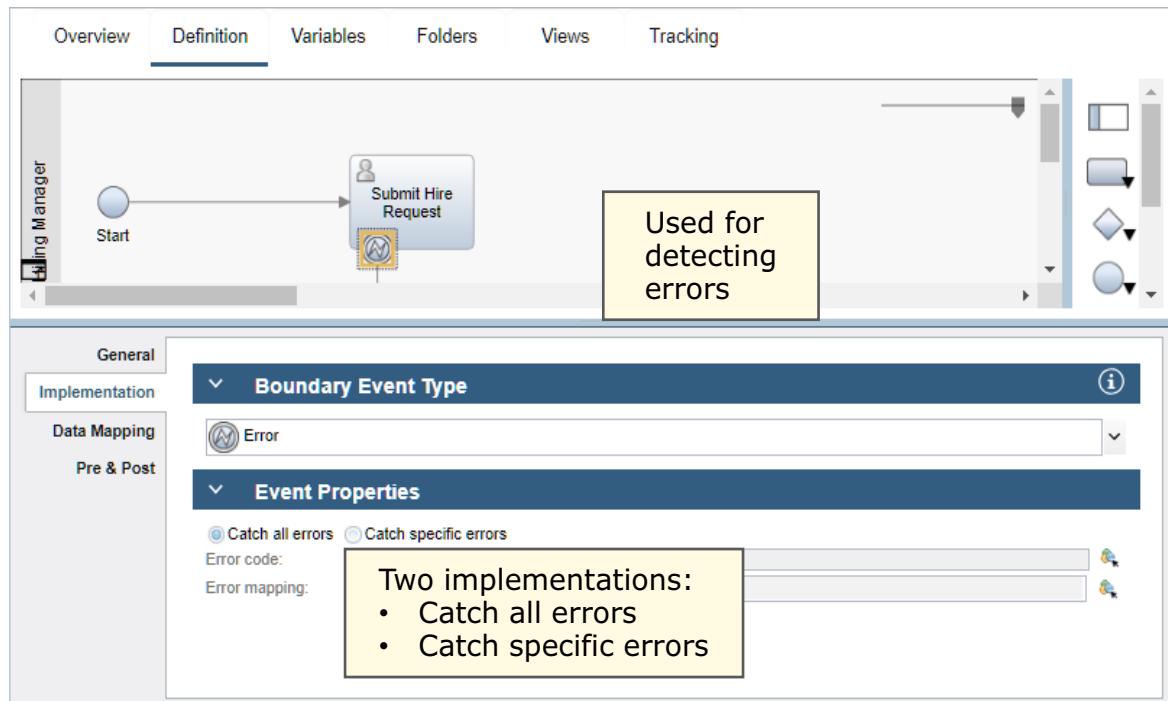
© Copyright IBM Corporation 2020

Figure 4-42. [No title]

A message intermediate event is used to model a message event that is received or sent while a process is running. For the sequence flow message intermediate event, a light envelope receives a message, and a dark envelope sends a message. When the message intermediate event is attached to an activity, the event receives messages but does not send messages.

IBM Training

Intermediate events: Error



Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

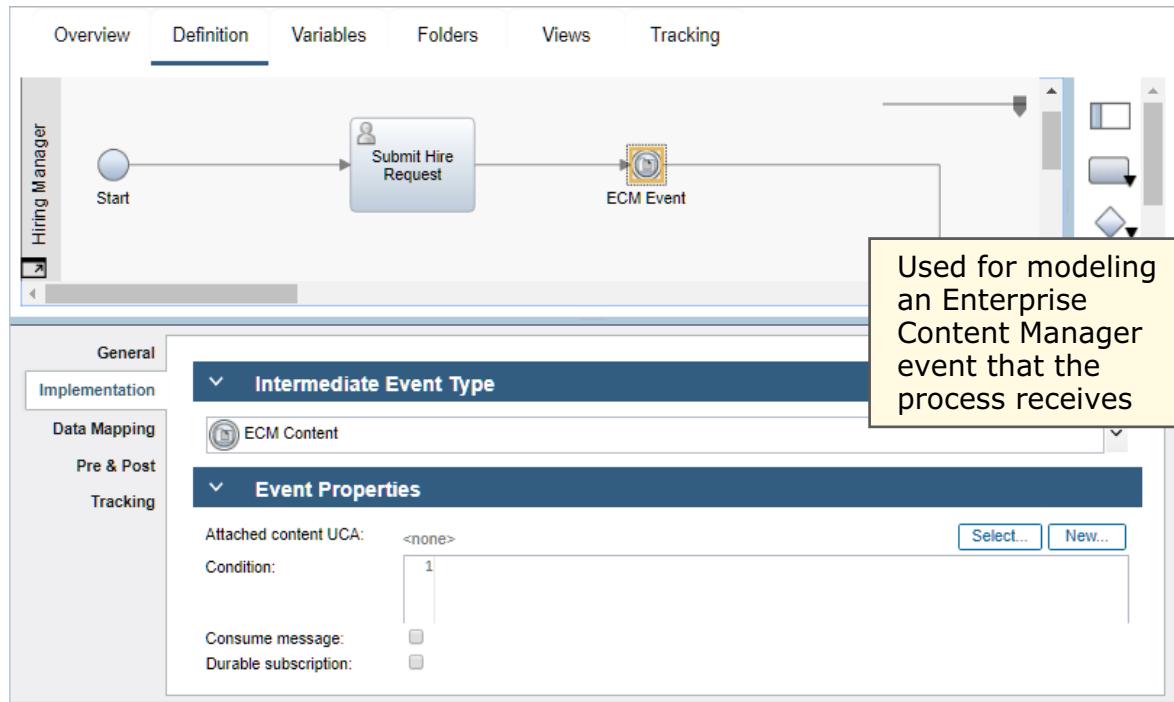
Figure 4-43. [No title]

Use the intermediate event to detect errors and to handle errors with login in the process flow.

IMPORTANT: The Error implementation option is available for events that are attached to activities. Error intermediate events are not used in sequence flow.



Intermediate events: Content



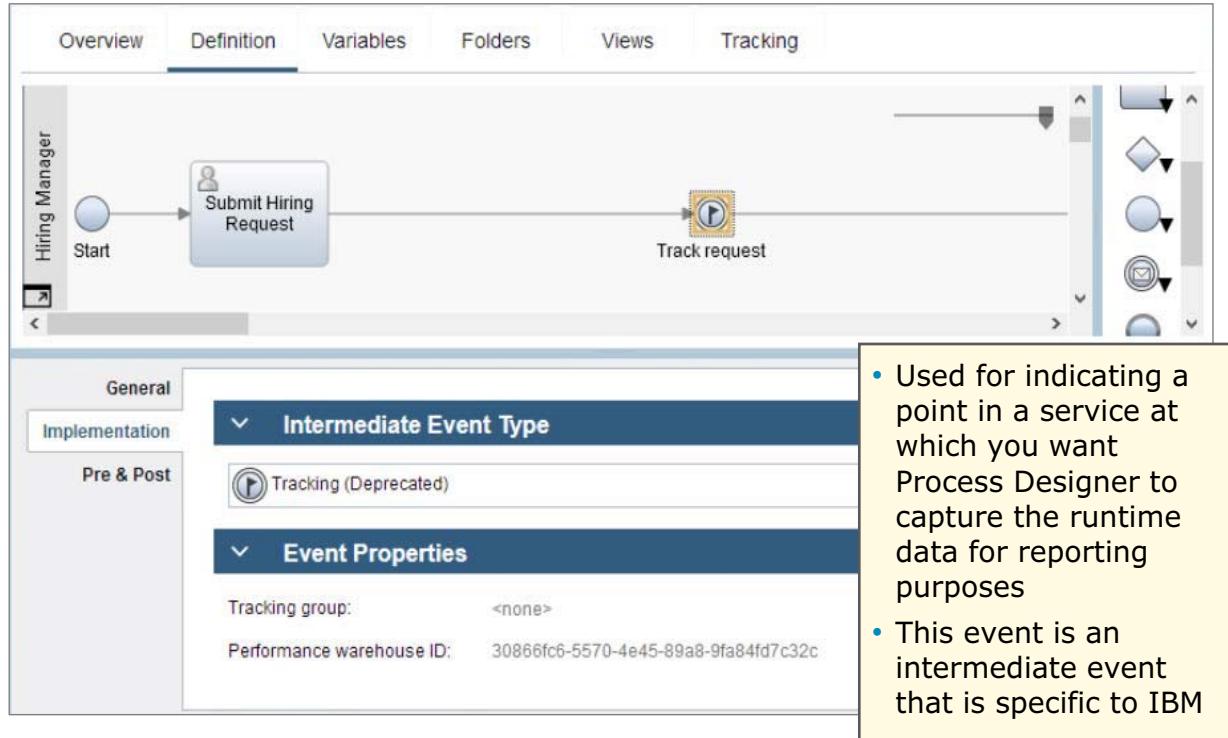
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-44. Intermediate events: Content

Use the Content implementation option to model an Enterprise Content Manager event that is received. The Content implementation option is available for events that are included in the process flow and events that are attached to an activity.

Intermediate events: Tracking



Playback 0: Controlling process flow

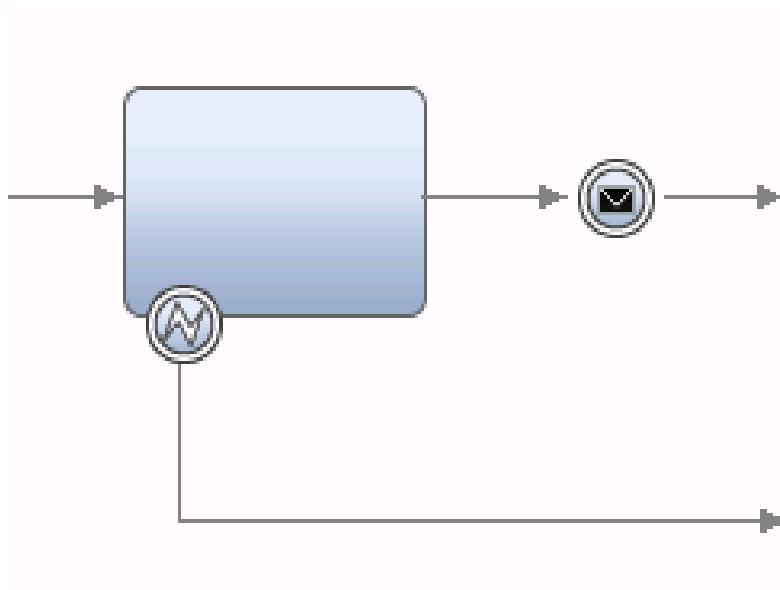
© Copyright IBM Corporation 2020

Figure 4-45. [No title]

The tracking intermediate event is used to indicate a point in a service at which you want IBM Process Designer to capture the runtime data for reporting purposes.

A tracking intermediate event is a sequence flow implementation. This event is an intermediate event that is specific to IBM Business Automation Workflow.

Attached and sequence flow intermediate events



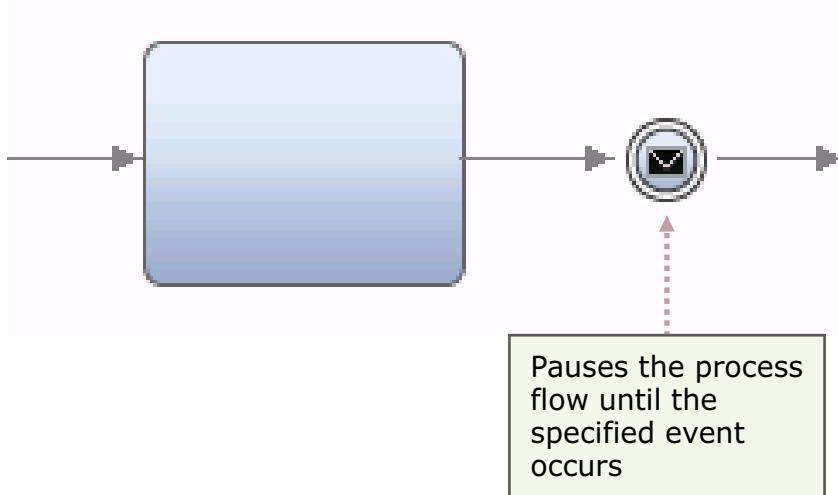
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-46. Attached and sequence flow intermediate events

Intermediate events can be in sequence flow or attached to the boundary of an activity. All intermediate events, except for tracking intermediate events, are processed the same way in IBM Process Designer.

Sequence flow intermediate event



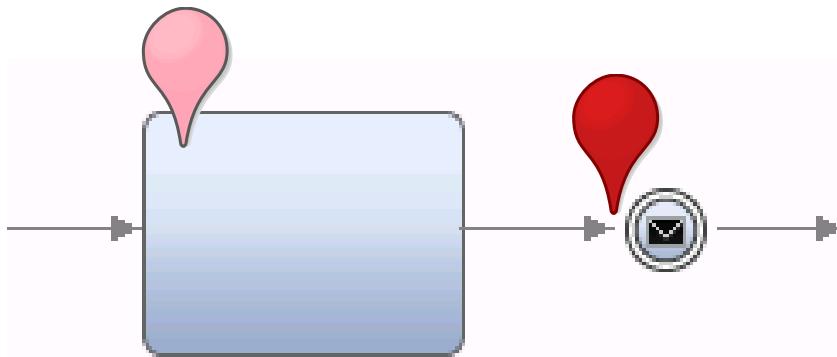
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-47. Sequence flow intermediate event

An intermediate event in the sequence flow pauses the process until the specified event takes place.

Process stops until an intermediate event occurs



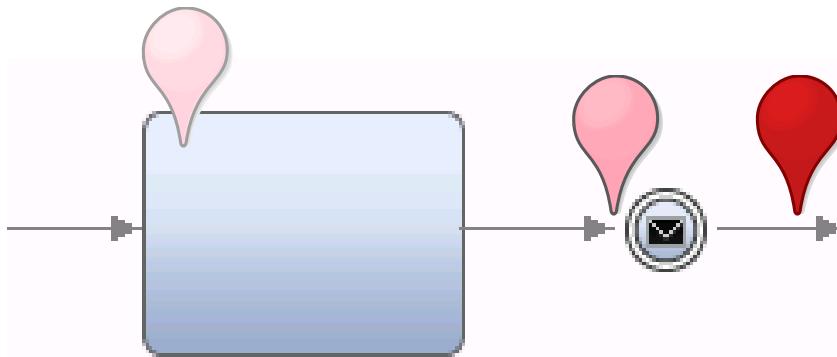
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-48. Process stops until an intermediate event occurs

The token stops at the intermediate message event.

Process continues on sequence flow when an intermediate event completes



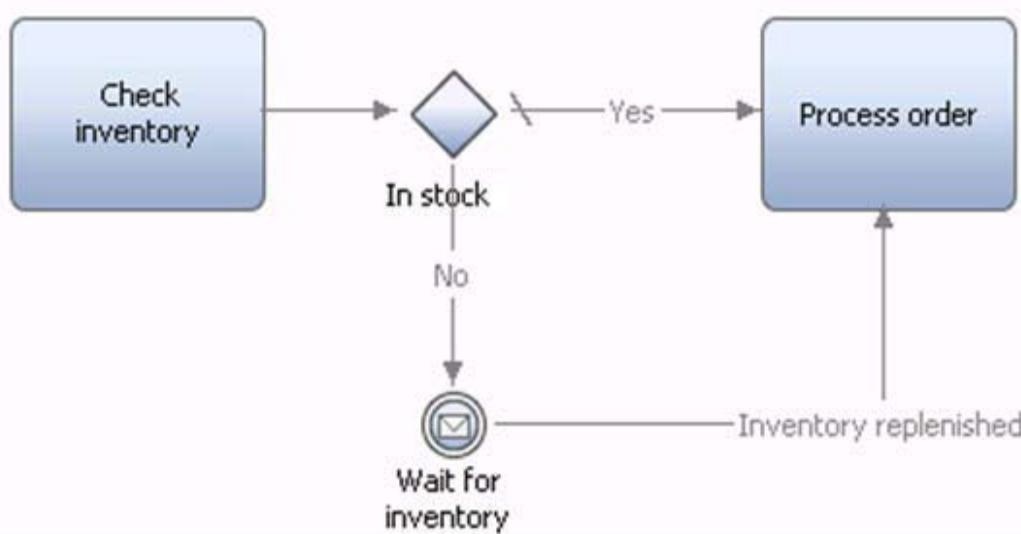
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-49. Process continues on sequence flow when an intermediate event completes

When the specified event occurs, the process flow continues along the normal sequence flow.

A common requirement (1 of 5)



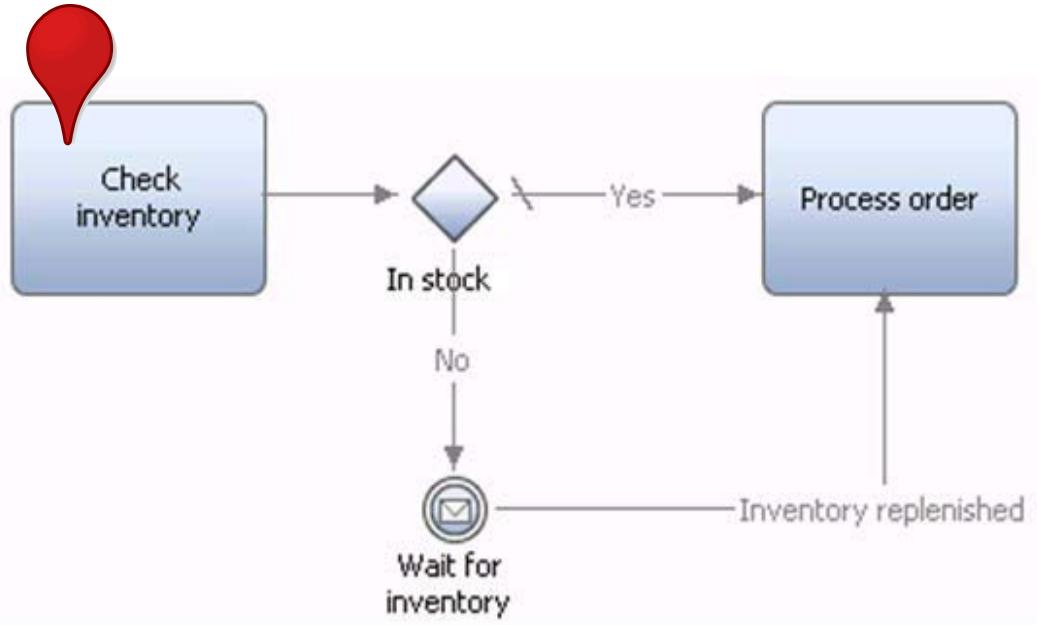
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-50. A common requirement (1 of 5)

This example shows the inventory requirement model. When someone places an order, check inventory and see whether it is in stock or out of stock. If it is in stock, process the order. If it is not in stock, wait for replenishing of the inventory before processing the order.

A common requirement (2 of 5)



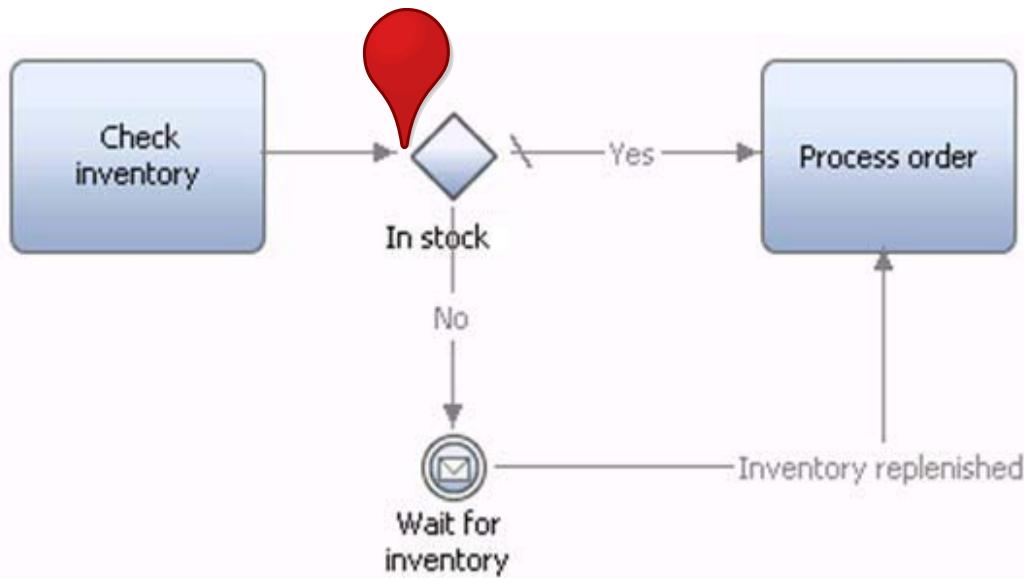
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-51. A common requirement (2 of 5)

Follow the token again.

A common requirement (3 of 5)



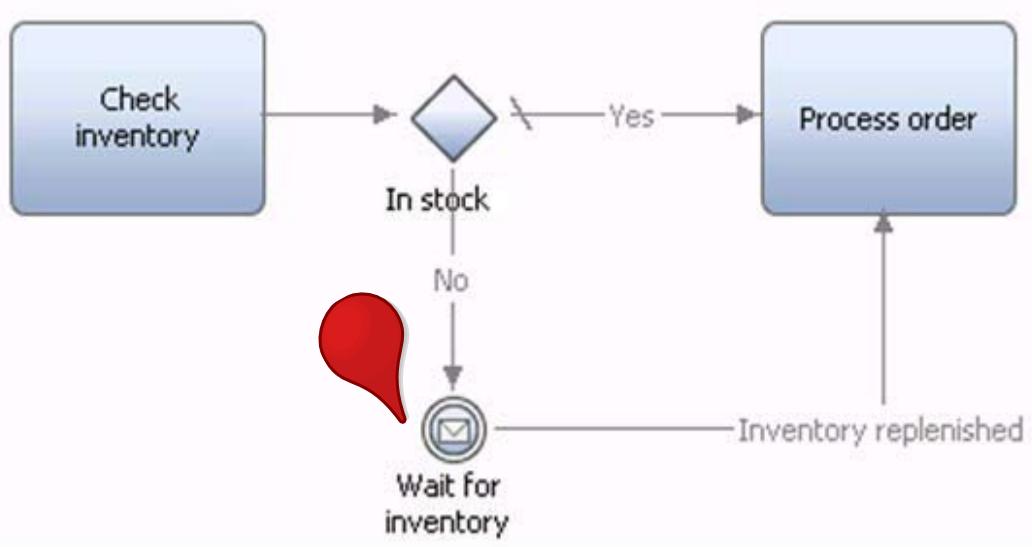
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-52. A common requirement (3 of 5)

The token is on the gateway, and the inventory is not in stock.

A common requirement (4 of 5)



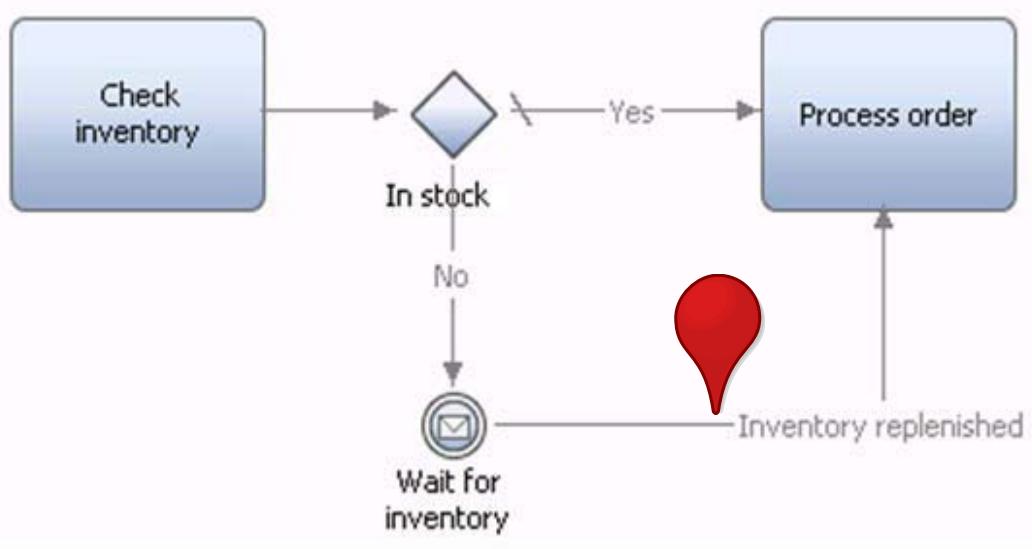
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-53. A common requirement (4 of 5)

The token stops on the intermediate event.

A common requirement (5 of 5)



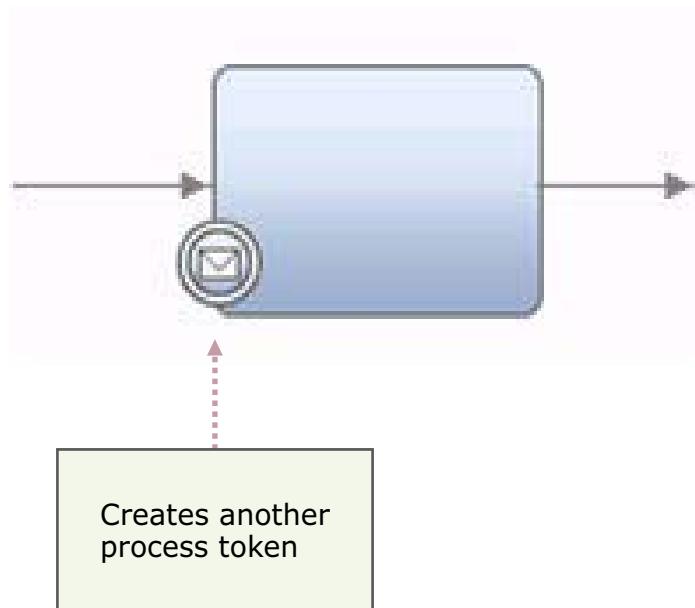
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-54. A common requirement (5 of 5)

The process is paused while the token is on the event, and then the process continues again after the **Wait for inventory** event takes place.

Attached intermediate event



Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

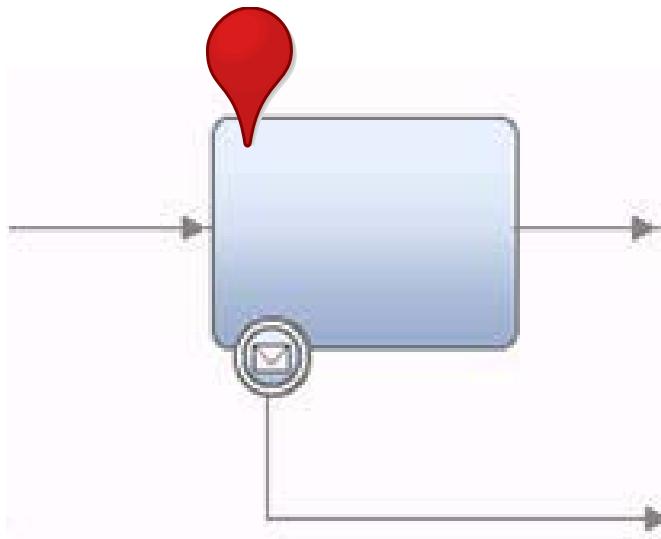
Figure 4-55. Attached intermediate event

An intermediate event that is attached to the boundary of an activity produces a separate token. If the specified event takes place while the activity is active, the intermediate event distributes the token along the outgoing sequence flow. After it occurs, you can specify whether to create parallel or an alternative process flow.

IMPORTANT:

An intermediate tracking event has a different behavior and does not conform to these behaviors.

Process flow reaches an activity



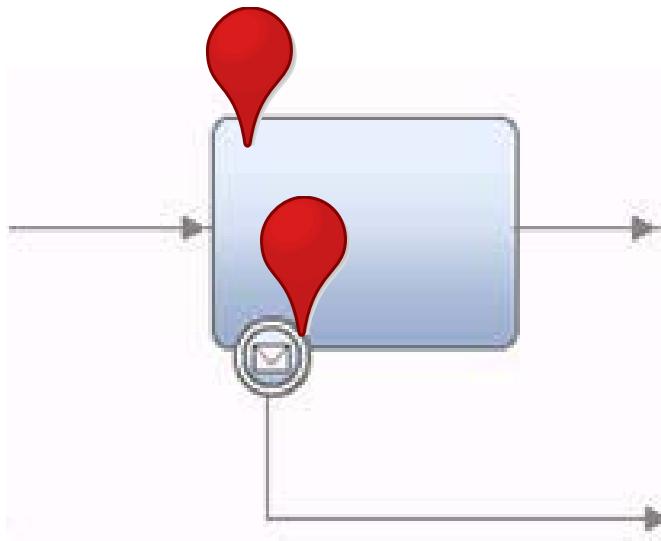
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-56. Process flow reaches an activity

Follow the token. Here the process flow reaches the activity.

Attached intermediate event generates a separate token



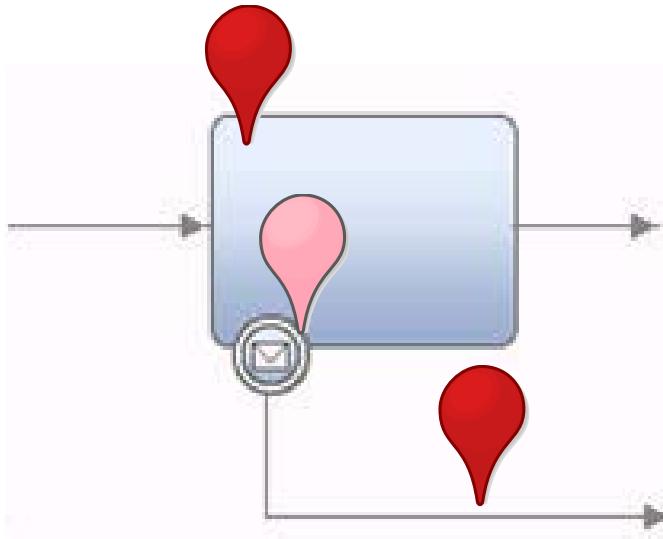
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-57. Attached intermediate event generates a separate token

The attached intermediate event generates a separate token.

Attached intermediate event can create a parallel flow



Playback 0: Controlling process flow

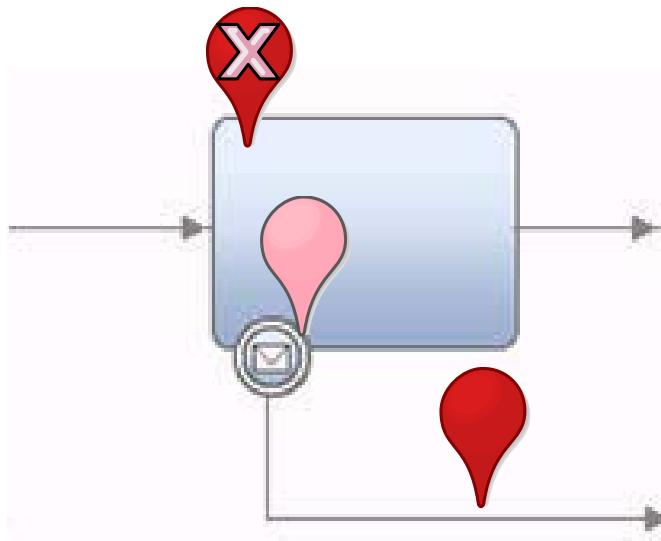
© Copyright IBM Corporation 2020

Figure 4-58. Attached intermediate event can create a parallel flow

Two things might happen:

- One option is that the attached intermediate event can create a parallel flow, and both tokens continue.
- The second option is shown on the next slide (closes activity).
- The third option is that the token on the activity completes before the message is received, and the activity consumes the event token before moving down the process flow.

Attached intermediate event can close an activity



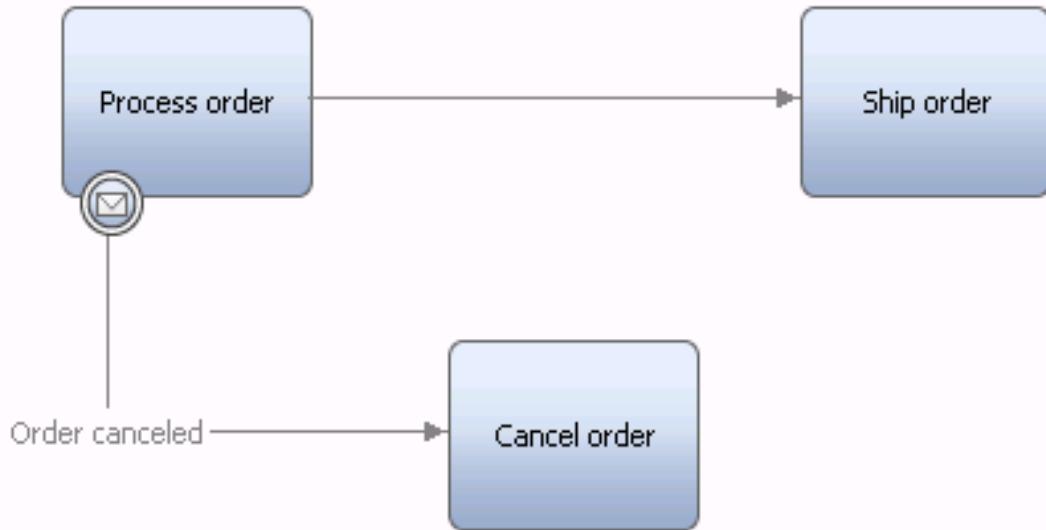
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-59. Attached intermediate event can close an activity

Another option is that the attached intermediate event can be defined to close an activity. The token on the activity is consumed, and one token continues down an alternative path.

Attached intermediate event example (1 of 4)



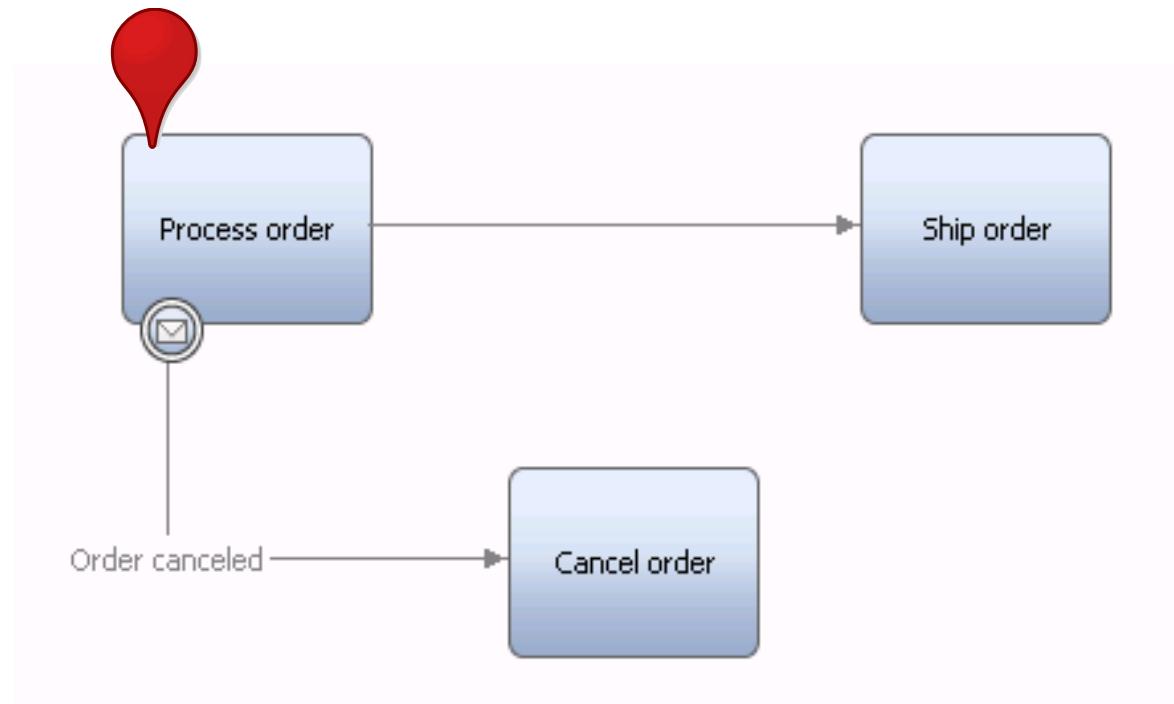
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-60. Attached intermediate event example (1 of 4)

This slide is an example process for an attached intermediate event.

Attached intermediate event example (2 of 4)



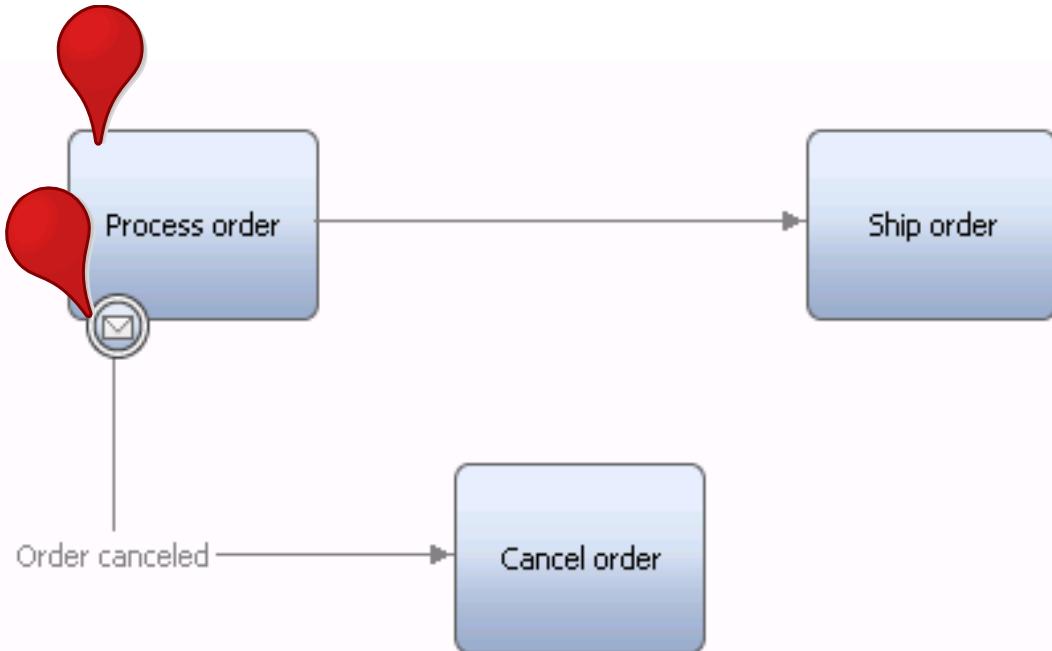
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-61. Attached intermediate event example (2 of 4)

The token starts on the **Process order** activity.

Attached intermediate event example (3 of 4)



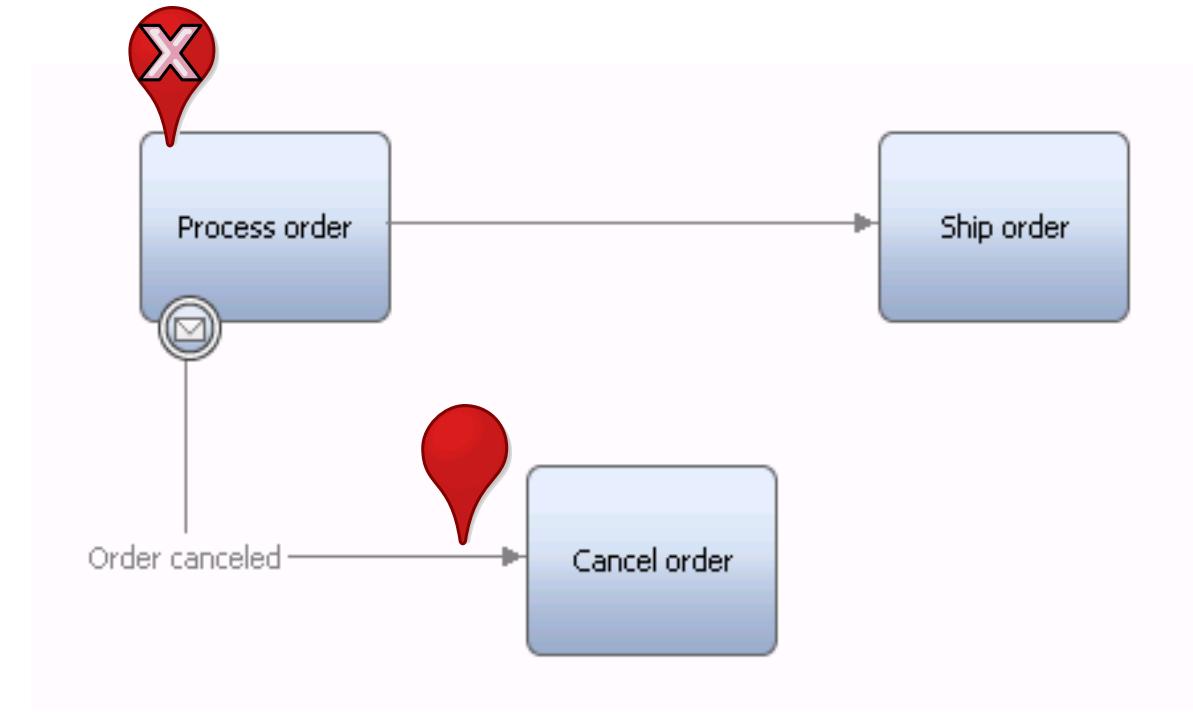
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-62. Attached intermediate event example (3 of 4)

Intermediate events can also be attached to the boundary of an activity.

Attached intermediate event example (4 of 4)

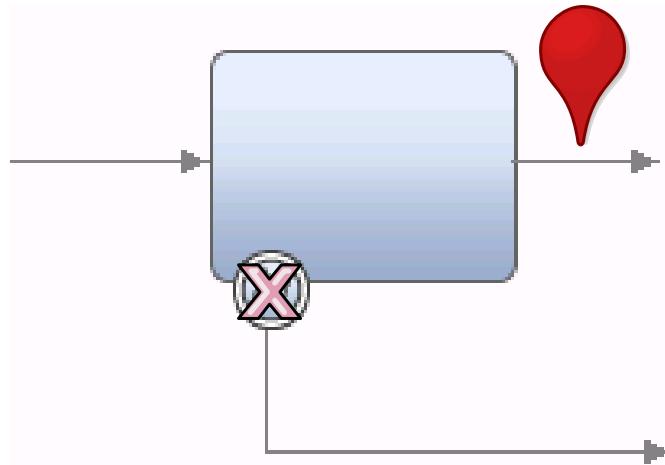


Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-63. Attached intermediate event example (4 of 4)

The intermediate event is defined to close the activity, so only one token continues to **Cancel order**.

Attached intermediate event is active only when the activity it is attached to is active

Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

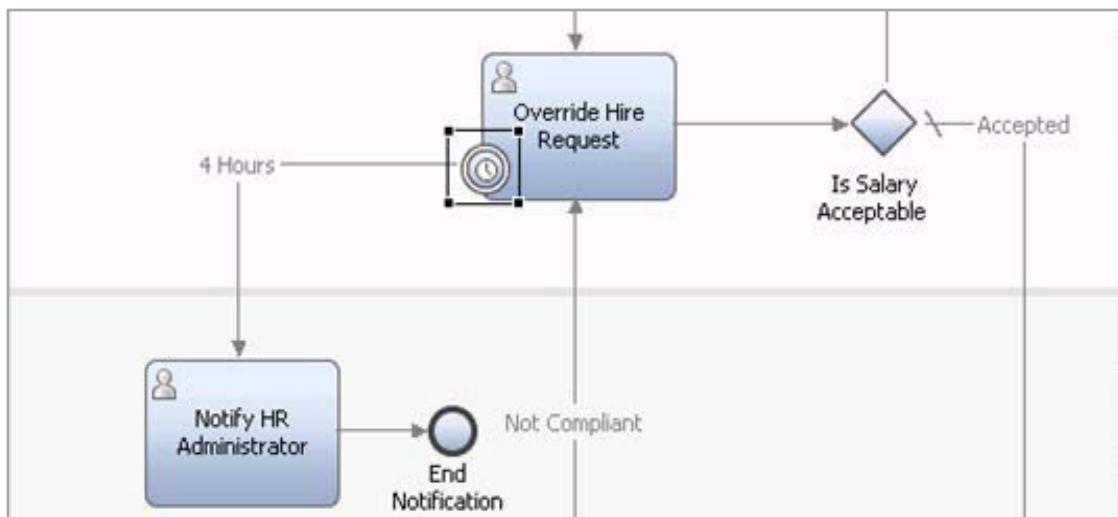
Figure 4-64. Attached intermediate event is active only when the activity it is attached to is active

The event is no longer active when the process activity is not active.

The tracking event has a particular way of working. As already mentioned, it is used to capture runtime data for reporting purposes. Tracking events can be placed only on process flows and not attached to activities. After the flow reaches a tracking event, it does not “pause” the process, but instead causes the capturing of runtime data at the event. Multiple tracking points can be placed within a process to capture different data at different points and different “snapshots” of data at various points.

A way to model an escalation

- Use an attached intermediate event
- Activity takes longer to complete than a defined amount of time
- The attached intermediate event triggers an escalation, which follows the path from the attached intermediate event to an escalation activity



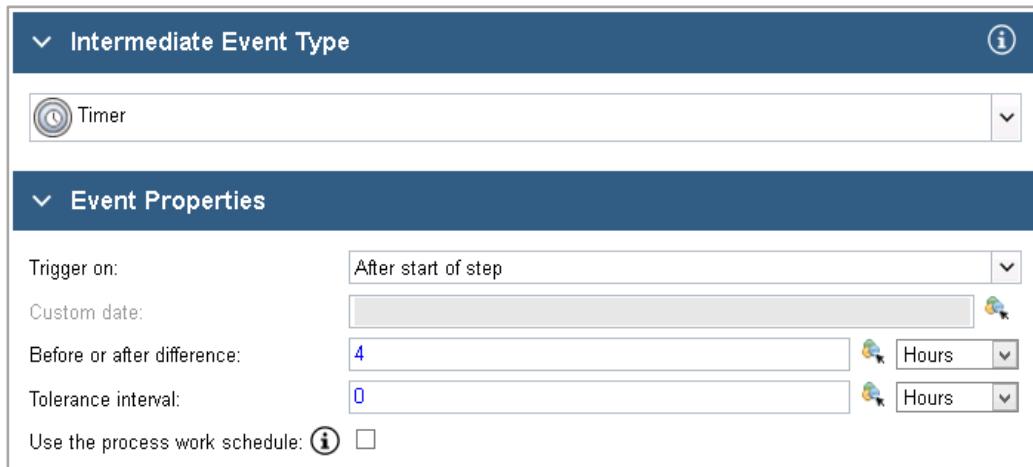
Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-65. A way to model an escalation

Timer intermediate event

- Allows a process to wait explicitly, or react to the passing of time
- Time interval can be based on system time, a due date, or a custom time



Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-66. Timer intermediate event

A timer intermediate event has the following characteristics:

- Allows a process to wait explicitly, or react to the passing of time
- Time interval can be based on system time, a due date, or a custom time
- Example: A traffic citation process (pay the fine activity) where a citation is issued when someone does not pay a traffic ticket within 30 days

Unit summary

- Describe process sequence flow and the runtime use of process tokens
- List and describe gateways as they are used in IBM Process Designer
- Explain how to evaluate conditions for a process gateway
- Model gateways in a process
- List and describe intermediate event types that are used in IBM Process Designer
- Model a business process escalation path with an attached timer intermediate event

[Playback 0: Controlling process flow](#)

© Copyright IBM Corporation 2020

Figure 4-67. Unit summary

Review questions (1 of 2)

1. True or False:

A gateway that merges multiple incoming paths into one outgoing path is known as a split.

2. Which of the following sequence flows indicates a processing path to follow when none of the conditions on the conditional flows are true?

- A. Default
 - B. Normal
 - C. Conditional
- 3.** Which of the following can be used to model escalation paths or delays in your business process definition?
- A. Message intermediate events
 - B. Timer intermediate events
 - C. Error intermediate events
 - D. Tracking intermediate events

4. True or False:

Exclusive gateways should be modeled so that only one outgoing sequence flow condition can be true.

Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-68. Review questions (1 of 2)

Write your answers here:

1.

2.

3.

Review answers

1. **False.** The gateway that merges multiple incoming paths is known as a join.
2. **A:** Default.
3. **B:** Timer intermediate events.
4. True.

Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-69. Review answers

Exercise: Playback 0: Controlling process flow

Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-70. Exercise: Playback 0: Controlling process flow

Exercise introduction

- Add gateways to a process
- Model the appropriate sequence flows for each gateway
- Add a timer intermediate event to a process based on business requirements
- Model an escalation path in a process with IBM Process Designer
- Add a new swimlane and activity for legal review to meet additional requirements.
- Validate the process application and create a snapshot



Playback 0: Controlling process flow

© Copyright IBM Corporation 2020

Figure 4-71. Exercise introduction

Unit 5. Playback 1: Controlling process flow with business data

Estimated time

01:30

Overview

This unit describes how to manage the variables and data flow. It covers the implementation of the intermediate timer event, gateways, and routing tasks.

How you will check your progress

- Review
- Exercise

Unit objectives

- Describe the differences between process flow data and business flow data
- Add variables to a process
- Implement gateways to control process flow
- Describe teams and process lanes
- Implement routing for tasks
- Assign an expert group to an activity
- Expose a process application to a team

Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-1. Unit objectives

Topics

- Managing variables and data flow
- Implementing the intermediate event: Timer
- Implementing gateways
- Routing tasks

Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-2. Topics

Key concepts in this unit

- **Playback 1: Controlling process flow:** To demonstrate that the process is following the correct flow out of the gateways, the developer sets the value of the process flow variables
- **Flow data:** Data elements that decision points use on process and service diagrams, and are used to determine the next paths to take
- **Business data:** Provides the context of the activity to all participants and what it is that they are working on
- **Variables:** Capture the business data that the activities use in a process
- **Timer:** Intermediate event that is implemented per the business requirements
- **Team:** Represents the groups of users in your enterprise that can be assigned a task in a swimlane or assigned directly to an activity

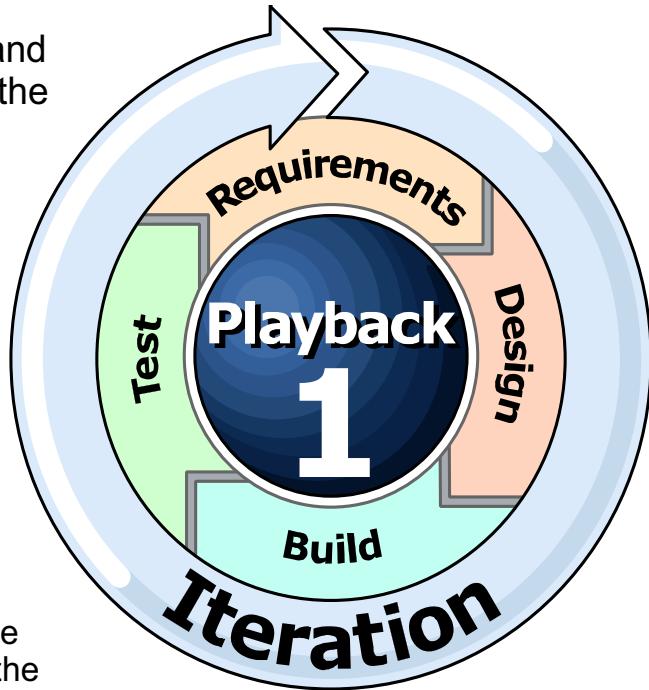
Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-3. Key concepts in this unit

Playback 1: Controlling process flow

- Set the process flow variables and watch the process follow along the correct path on your decision gateways
- Gather all the process stakeholders
- Validate the process that was created
 - Demonstrate that the process is following the different paths that are flowing from the exclusive gateways on the processes
 - Demonstrate that the tasks are being assigned and created in the Portal inbox corresponding with the correct lane and assignment settings



[Playback 1: Controlling process flow with business data](#)

© Copyright IBM Corporation 2020

Figure 5-4. *Playback 1: Controlling process flow*

To demonstrate that the process is following the correct flow out of the gateways, the developer sets the value of the process flow variables.

It is important to gather all the process stakeholders to validate the process that was created, and to verify that it meets the business needs outlined in Playback 0. Verify that the process works as modeled and as expected. Set the process flow variables and watch the process follow along the correct path on the decision gateways.

- Demonstrate that the process is following the different paths that are flowing from the exclusive gateways in the processes.
- Demonstrate that the tasks are being assigned and created in the Process Portal inbox that corresponds with the correct swimlane and assignment settings.

If it is necessary to change the high-level process, the project development team can move back and redo Playback 0. After they receive buy-in that the model meets the goals of the current release, the project development team is ready to move on to next stage of Playback 1.

5.1. Managing variables and data flow

Managing variables and data flow

Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

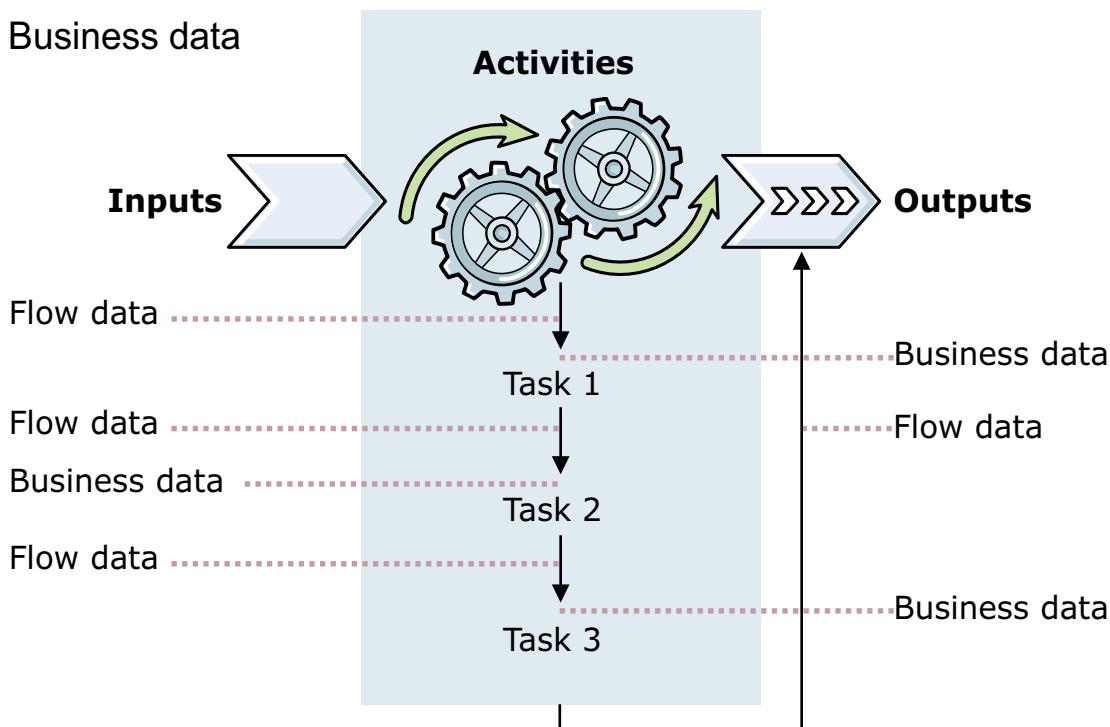
Figure 5-5. Managing variables and data flow

Data flow and management is an essential part of working with services and processes. As part of Playback 1, developers create process data variables to control and demonstrate the process flow. Later in the next stage of Playback 1, developers create business data variables for their processes.

Variables represent the data that provides the business and process context to a running process. Using variables, the business process passes the data from one process step to another. This passage is called the data flow.

Two types of process data

- Flow data
- Business data



Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-6. Two types of process data

To distinguish the different types of data that flow through the process, you can categorize data into two different types of process data: business data and flow data.

What is flow data?

- Data that is used to determine:
 - Which activities to complete
 - Who completes each activity
 - The time an activity is due or when an activity is to be escalated

Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-7. What is flow data?

Flow data moves the process along. The most obvious examples of flow data are the data elements that decision points use on process and service diagrams. When a token is at a decision gateway, the value of each of the data elements is used to determine the next paths to take.

Flow data elements go beyond just the data that is needed to drive process decision points. Flow data includes all of the following categories:

- Data that is used to determine which activities to complete
- Data that is used to determine who completes each activity
- Data that is used to determine when an activity is due or when an activity is escalated

Flow data must be identified early on in the implementation process. Generally, by the end of this stage of Playback 1, your flow data is identified and implemented. Flow data gets the right activities to the right participants at the right time. Without flow data, the process cannot work.

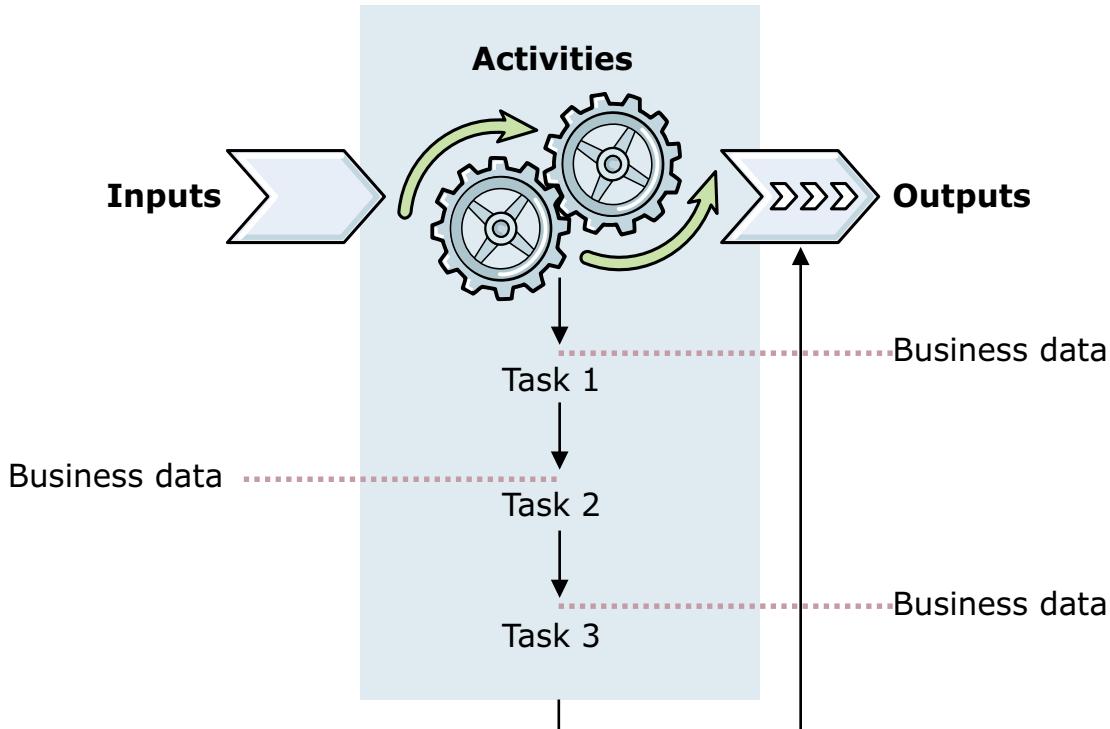
The set of flow data elements often overlaps the set of business data elements, but as developers you want to break them out. Changes to the set of flow data elements are changes to the process itself. Any changes to the business data variables are insulated from the process flow data variables.

Many flow data elements are gathered directly from the participants, but others must be retrieved from external sources. Common sources for flow data elements include LDAP repositories, SQL data sources, and web services. Flow data variables map and share data with the variables from external sources, but the flow data variables themselves are separate and distinct.

Flow data elements are initialized as soon in a process instance as possible. You can use this action to somewhat predict the future path of a process instance.

What is business data?

- Business data provides the context of the activity to each participant



Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-8. What is business data?

The business data is a bit more problematic to define than the flow data. In general, the business data provides the context of the activity to each participant. The data is used to make it clear to participants what it is that they are working on.

For example, customer service representatives know (based on the activity that was assigned) that they are working on an insurance claim. From the business data, the representatives can tell which claim they are working on by claim type, claim number, customer, and claim description.

Business objects, variables, and data mapping

- Business objects
 - In Process Designer, business objects are complex objects that are created from simple business objects or other complex business objects
 - Define the business data
- Variables
 - Instantiate business objects that activities use in a process or by steps in services, such as service flows or human services
 - Capture and store the business data
 - Each has its own type and scope
 - Must be declared before you can start to use them
- Data mapping
 - In Process Designer, you map the input and output data to pass variables to an activity or a step
 - You must set the input and output mapping for each activity in a process
 - Map variables in the parent process to the variables received and generated by nested processes and services

Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

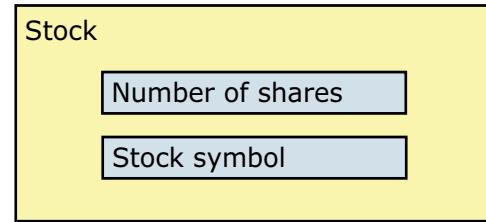
Figure 5-9. Business objects, variables, and data mapping

More on data mapping is covered in the next unit.

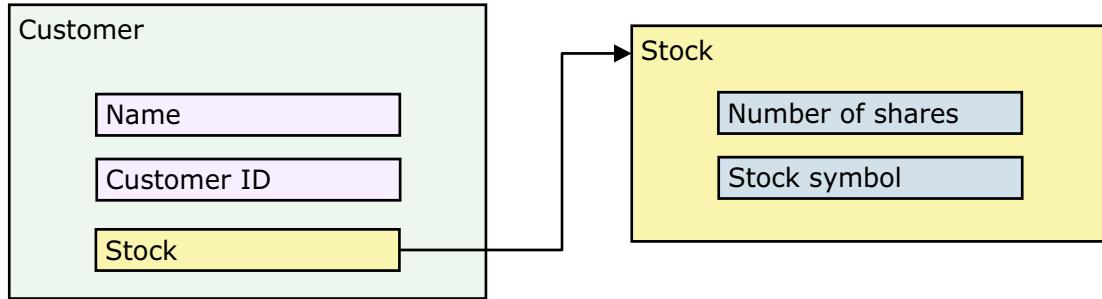
Business objects

- Supports two types of business objects: simple and complex

- Simple business objects
 - Composed only of scalar properties



- Hierarchical business objects
 - Composed of attributes that reference nested business object definitions



[Playback 1: Controlling process flow with business data](#)

© Copyright IBM Corporation 2020

Figure 5-10. Business objects

In the Process Designer, business objects are complex objects that are created from simple business objects or other complex business objects.



IBM

Example of business objects and variables

The screenshot shows the IBM Business Object Designer interface. On the left, a sidebar lists business objects: Candidate, Person (selected and highlighted with a red box), Position, Qualifications, and Requisition. The main area is titled 'Data' and shows the 'Common' tab for the selected 'Person' object. It displays the Name as 'Person', Modified by 'author1' on 'Mar 15, 2016, 7:03 PM', and a Documentation section with a rich text editor. A callout box points to the 'Name' field with the text: 'Business object defines the type'. Below the Common tab is a 'Parameters' tab containing a list of parameters: lastName (String), firstName (String), supervisor (String), startDate (Date), payLevel (String), payType (String), and notes (String). To the right is a 'Variables' panel divided into 'Input', 'Output', 'Private', and 'Exposed Process Variables' sections. The 'Private' section contains variables like isNewPosition (String), isApproved (String), isCompliant (String), requisitionDetails (HiringRequisition), isWeekend (Boolean), and cancellationId (String). The 'Exposed Process Variables' section is currently empty.

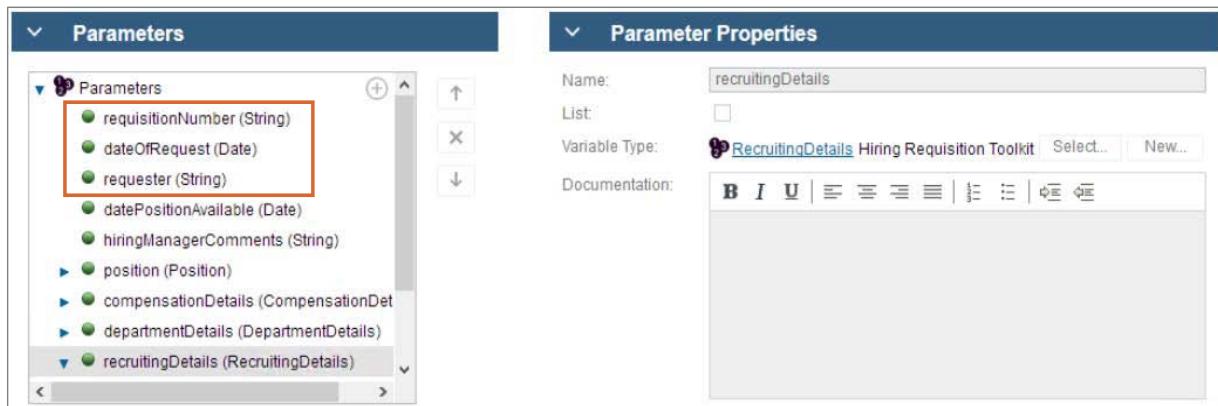
Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-11. Example of business objects and variables

Create the business objects in the lab exercise

- You create a few business objects during the lab
- Example of business object
 - HiringRequisition
- Hiring Requisition business object has some parameters
- For example:
 - requisitionNumber (String)
 - dateOfRequest (Date)
 - requester (String)



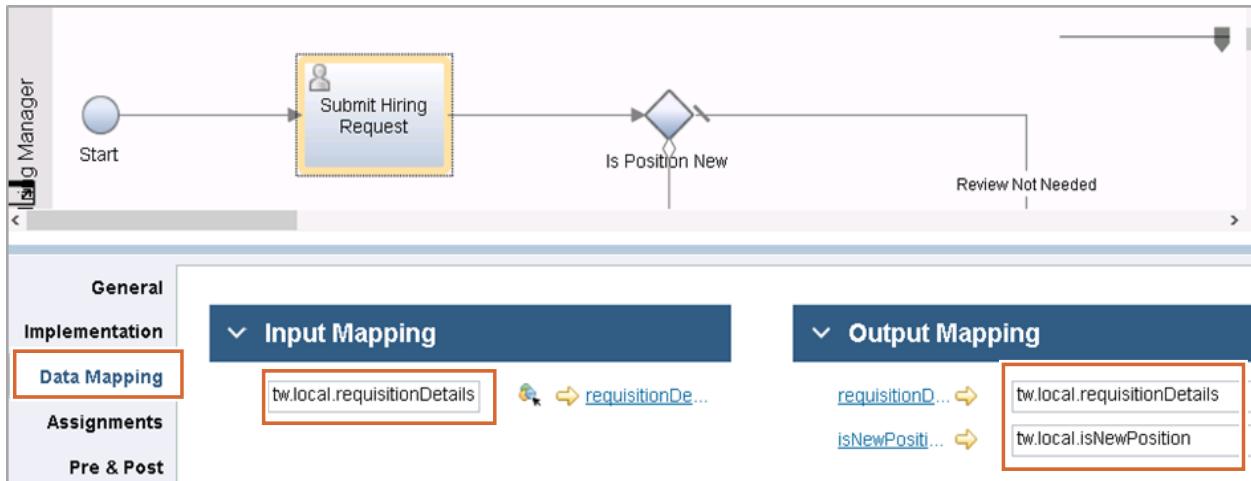
Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-12. Create the business objects in the lab exercise

Example of data mapping

- Variables define the inputs and outputs of an activity
- Mapping data moves the data that is stored in variables that are defined at the process-level into and out of the activity



Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-13. Example of data mapping

In this example, the Submit Hiring Request activity has one input and two output variables. At the process level, you map the `tw.local.requisitionDetails` variable to the input variable for the human service and map the two output variables from the service to the process variables.

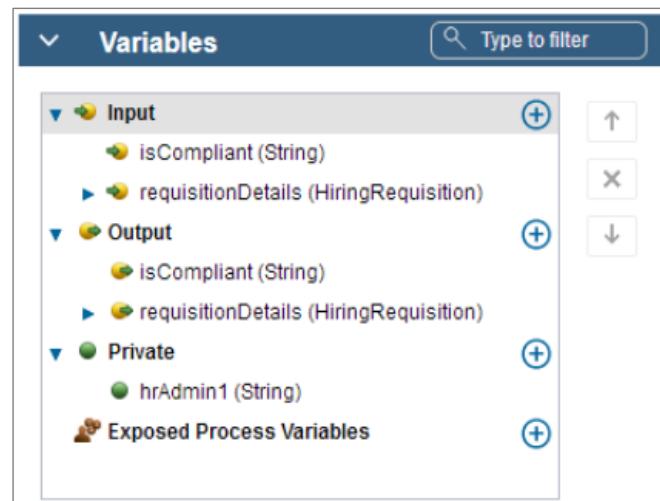
More on data mapping is covered in the next unit.

Declaring three kinds of variables

- **Input:** Defines the variables that are passed into the current process or service

- **Output:** Passes variables out of the process or service to a parent process or service

- **Private:** Any variables that are not passed in or out of the process or service



Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-14. Declaring three kinds of variables

Variables in a process or service can be declared as three kinds:

- Private variables refer to values that the current process or service needs, but which the parent process or service does not know or need. The value of a private variable can still be of interest to any nested processes or nested services.
- Input variables refer to values that you can pass into the current process or service.
- Output variables store values that are passed out from a process or service to a parent process or service.

Standardizing variable names

- Create variable names that begin with a lowercase letter
 - Capitalize the first letter when creating a variable type (business objects), but use camel case for the instantiation of the variable (for example, `employeeId`)
 - Camel case refers to a word or string of letters that has no space and has an uppercase letter in a position other than the first letter
- If the variable name you choose consists of only one word, spell that word in all lowercase letters
- If the variable name consists of more than one word, capitalize the first letter of each subsequent word
- Variable names are case-sensitive

Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-15. Standardizing variable names

To standardize variable names, create variable names that begin with a lowercase letter. This good practice makes it easier to distinguish between a variable and its variable type, which begins with an uppercase letter as a good practice.

Capitalize the first letter when creating a business object but use camel case for the variable instantiation. For example, the variable `employeeId` uses a lowercase first letter to signify that it is an instance of a business object type `Integer`, which starts with an uppercase letter. The standard business object types (`Date`, `String`, `Integer`, and other types) all follow this same naming convention.

If the variable name you choose consists of only one word, spell that word in all lowercase letters. If the variable name consists of more than one word, capitalize the first letter of each subsequent word. For example, a variable that contains an employee ID number has the variable name `employeeId`.

Variable names are case-sensitive.

Understanding namespaces

Namespace	Description
tw	The top-level namespace
tw.local	Access and update process-local and service-local variables
tw.system	System features and functions
tw.object	Used to initialize complex IBM Business Automation Workflow objects

Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-16. Understanding namespaces

All variables in IBM Business Automation Workflow are JavaScript objects. IBM Business Automation Workflow uses namespaces to organize these objects, and their functions and methods. An example of some of the methods is shown in the chart.

5.2. Implementing the intermediate event: Timer

Implementing the intermediate event: Timer

Playback 1: Controlling process flow with business data

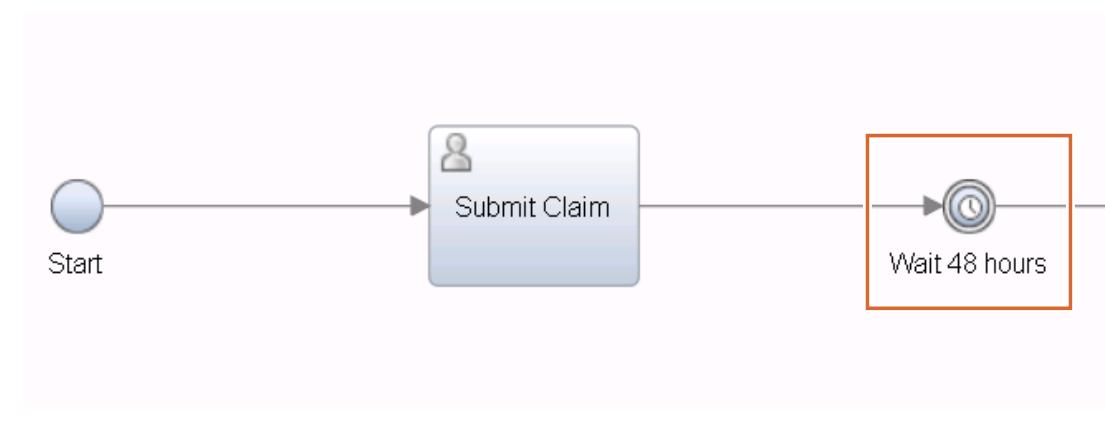
© Copyright IBM Corporation 2020

Figure 5-17. Implementing the intermediate event: Timer

Although you modeled timer events in Playback 0, you have no control over the functions of the timer. In this topic, the timer is implemented to fire according to the business requirements.

Timer intermediate event

- Use a timer intermediate event to specify a time before or after an activity occurs
- Timer intermediate events, which are used to model escalation paths or delays in the process, can occur either between activities or attached to an activity



Playback 1: Controlling process flow with business data

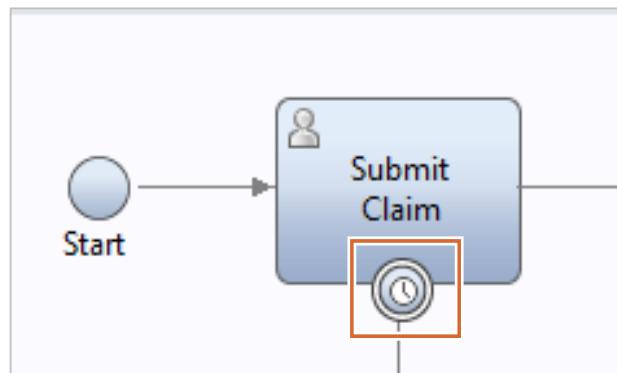
© Copyright IBM Corporation 2020

Figure 5-18. Timer intermediate event

Use a timer intermediate event to specify a time before or after an activity occurs. Timer intermediate events are used to model escalation paths or delays in the process.

Attached timer intermediate event

- When a running process instance reaches an activity with an attached timer intermediate event, a timer starts
- The interval for the timer is calculated according to the configuration that you specify in the implementation properties for the timer intermediate event
- When the specified interval elapses, the process follows the path from the attached timer intermediate event to the subsequent activity



Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-19. Attached timer intermediate event

Timer intermediate events, which are used to model escalation paths or delays in the process, can occur either between activities or attached to an activity.

Timer intermediate event implementation (1 of 2)

Implementation details:

- **Trigger On:** Specifies when a timer event starts
- **Custom Date:** Use JavaScript to calculate and specify a date
- **Before/After Difference:** Amount of time
- **Tolerance Interval:** If work is in progress, this setting specifies an extra delay
 - Measures only one time
- **Use the activity work schedule:** Limits the timer activity to the period specified

Two options are available only when configuring an attached timer event:

- **Interrupt activity:** Closes the attached activity after time elapses
- **Repeatable:** Resets the timer to countdown again after time elapses

A developer implements an inline, or sequence, timer intermediate event through the Implementation tab in the properties section of the component. The implementation details are:

- **Trigger On:** Specifies when timer event should start.
- **Custom Date:** Use JavaScript to calculate and specify a date.
- **Before/After Difference:** Amount of time to wait before allowing the token to continue on the flow line.
- **Tolerance Interval:** If work is in progress, this setting specifies an extra delay that is measured one time. For example, if users accept a task during the delay, they are allowed the tolerance time to complete the task before the token continues on the sequence flow line.
- **Use the process work schedule:** Limits the timer activity to the period specified.

Timer intermediate event implementation (2 of 2)

Intermediate Event Type: Timer

Event Properties:

Trigger on:	After start of step
Custom date:	
Before or after difference:	0 Hours
Tolerance interval:	0 Hours
Use the process work schedule:	<input type="checkbox"/>

Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-21. Timer intermediate event implementation (2 of 2)

5.3. Implementing gateways

Implementing gateways

Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-22. Implementing gateways

Any gateway that is modeled in Playback 0 is functional (that is, processes would flow down a path). To control the path that is taken, the developer implements the decision logic that the gateway uses.

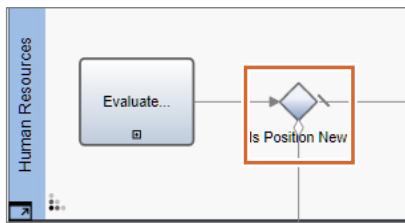
Implementing a gateway

Decisions

Review Needed: tw.local.isNewPosition == "1"

Default flow: Review Not Needed

- Define conditions that control whether a path is followed during the running process
- Test the newly functioning gateways with the process inspector



- Once the gateway is implemented, the conditional flow is flagged with a diamond at the point at which it exits the gateway. The default sequence flow is depicted with a slash.

Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-23. Implementing a gateway

During the Playback process, it is necessary to demonstrate each path that can be taken, but the logic on how that decision is reached is implemented in a later Playback.

When you specify the implementation for a gateway, you define conditions that control whether a path is followed during the running process. After creating the rules and integrating the data, make sure that you test your newly functioning gateways with the process inspector.

A good practice for Playback 1 is to avoid the use of Boolean (true or false) variables to implement exclusive gateways to accommodate more flows that might be added in the future. Use simple variables (String, Integer) to drive all your exclusive gateways. Developers can then set the default value of the process variables and show the different paths that are taken during Playback 1. Document the gateway logic to help in troubleshooting the process later.

5.4. Routing tasks

Routing tasks

Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

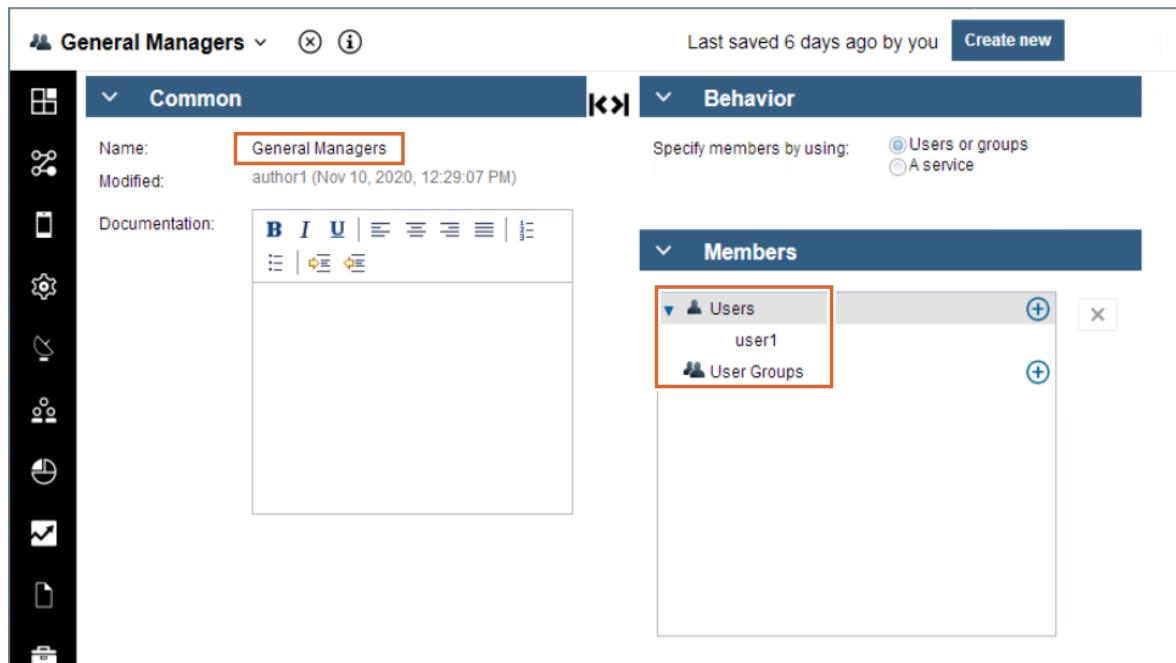
Figure 5-24. Routing tasks

This stage of Playback 1 completes the goal of getting the right tasks to the right people at the right time. Variables were built, and those variables were used to drive decision gateways. When a task is created for a process participant to complete an activity, the system assigns the task to the right individual who can complete the work.

In this topic, teams are introduced and then process routing or assignment is examined.

IBM Training

Creating a team



Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

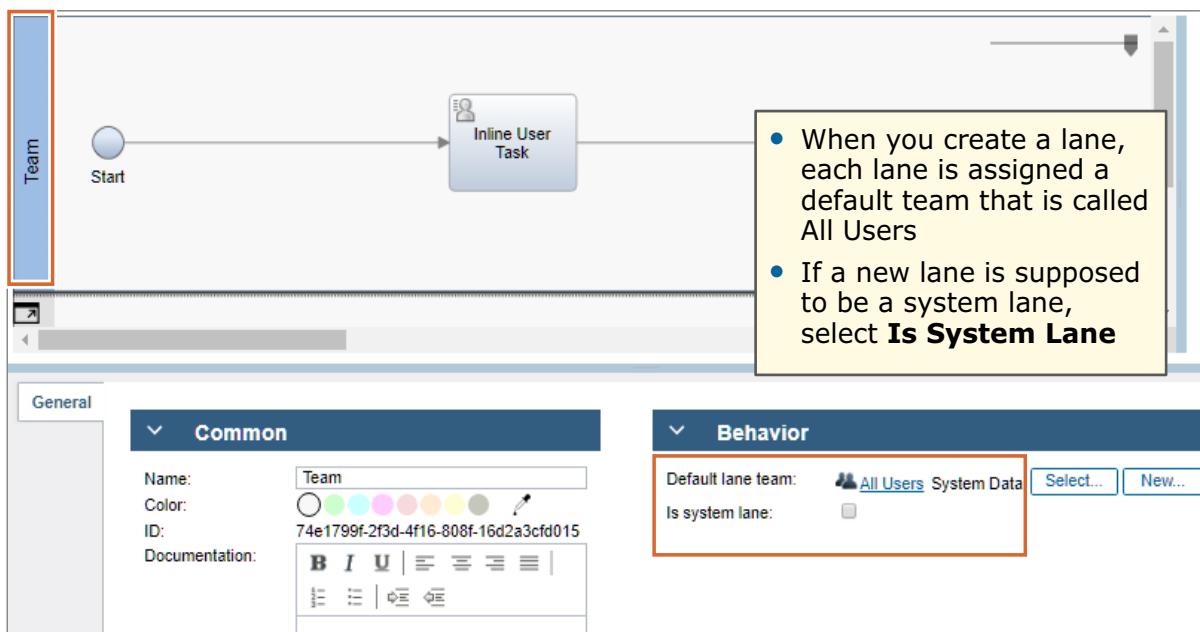
Figure 5-25. Creating a team

A team represents the groups of users in your enterprise that can be assigned a task in a swimlane or assigned directly to an activity. To create a team and add users to that group:

- In the Designer view, click the **(+)** plus sign next to Processes and select **Team** from the list of components.
- In the new Team window, enter a name and click **Finish**.
- IBM Process Designer displays the property settings for the team.
- Enter the required information in the Team property interface.

Team and team members

- A team contains the users or participants who complete the runtime activities that are modeled in each lane



Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-26. Team and team members

A team contains the users who complete the runtime activities that are modeled in each lane. Team lane assignments ensure that any activities that are not routed to a specific user have an automatic default assignment.

Creating lanes:

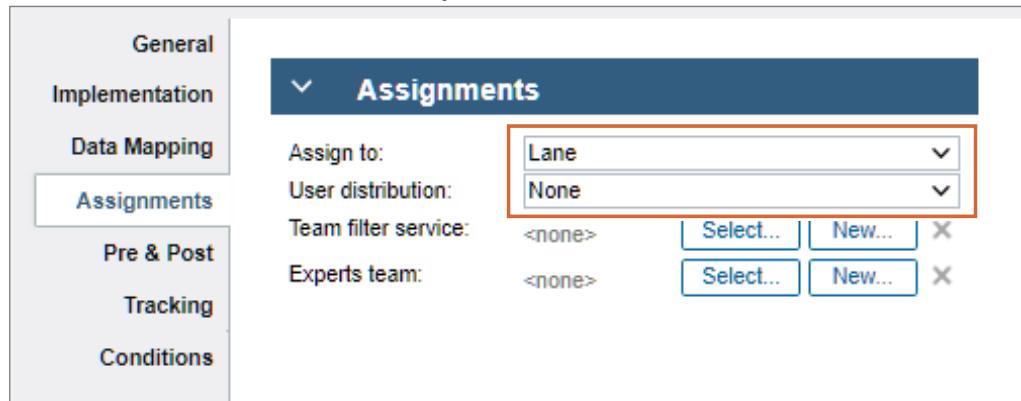
When you create a lane, each lane is assigned a default team that is called All Users. This default group contains all of the users of IBM Business Automation Workflow to allow for testing of your processes. If a new lane is a system lane, select **Is System Lane** and add the System team in the Behavior section. System lanes are shaded a different color so that they can be easily recognized.

Subscribing to a Blueworks Live process:

If you subscribe to a Blueworks Live Process with preassigned team lanes and no matching teams exist, IBM Business Automation Workflow creates a team for each lane. IBM Business Automation Workflow automatically attaches the team to the corresponding lane.

Routing activities

- For any activity with a service (task) implementation, you can designate the users who receive the runtime task by using the Assignments option in the property tab of the activity
 - This implementation designates whom to route or assign the activity to and how it is to be distributed
- By default, **Assign to** is set to Lane
- The two lane assignment selections are Lane and Team
- User distribution** is most commonly set to None or Last User



Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-27. Routing activities

Routing activities:

Developers might not want an activity to go to the default team, or they might need a more dynamic solution.

For any activity with a service (task) implementation, developers use the Assignments option in the property tab of the activity to designate the users who receive the runtime task. Developers control how the runtime task is distributed to designated teams and members.

To assign a task, developers select the activity that they want, and then select the assignments section to display the assignment properties for that activity.

Routing options:

The options available in routing allow the developer to designate whom to route the activity to (**Assign To** menu) and how it is distributed (**User Distribution** menu).

Assign to:

By default, **Assign to**: is set to `Lane`, indicating that tasks are assigned to the team for that lane. The two most commonly used lane assignment selections are:

- Lane**: Routes the runtime task to the team that is assigned to the lane in which the selected activity is placed.

- **Team:** Routes the runtime task to any team that the developer wants to designate the task to. This assignment overrides the default team that is bound to the lane. This function provides a dynamic layer of routing for a particular task in a process.

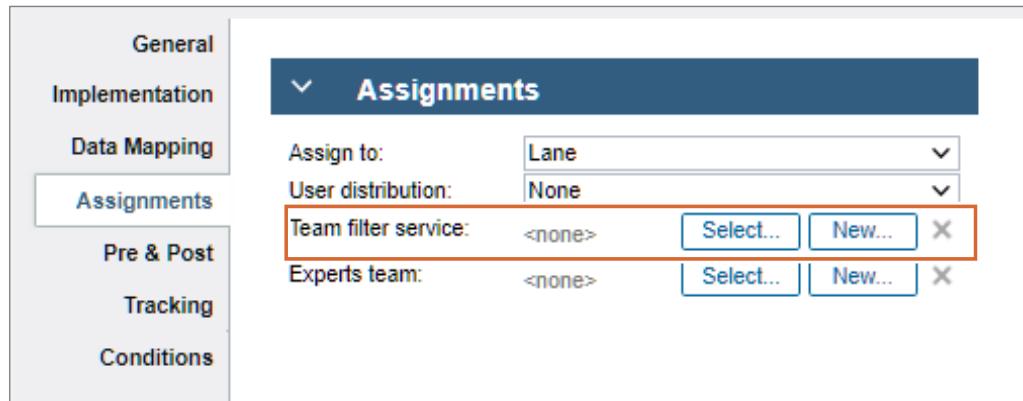
User distribution:

By default, **User Distribution** is set to **None**, meaning that no specific team member distribution is applied to the routed task. The task is assigned to the pool of potential team members, allowing individual members to acquire tasks that are assigned to the pool. The **Last User** setting routes the runtime task to the team member who completes the activity that immediately precedes the selected activity in the lane. Set this option for the first activity in the lane when the process is started from the process portal and the activity is the first activity after the start event. In this case, the runtime task is routed to the user who started the process.

Other advanced user distributions available in IBM Business Automation Workflow allow the system to assign tasks to individual team members. In practice, **User Distribution** is most commonly set to **None** or **Last User**.

Team filter service

- Are used to filter down a team to a subset team
- The new filtered down subset team is returned as a Team object



Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-28. Team filter service

Routing activities:

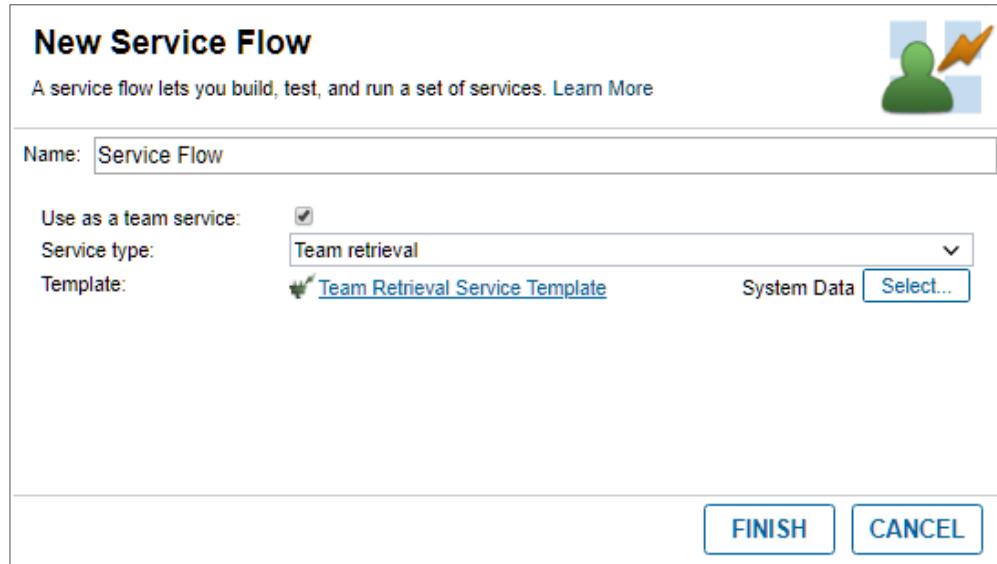
Sometimes a developer does not want the whole team to be assigned to a task, but rather a subset of the team. A developer creates team filter services to implement assignment policies. The team filter service takes the initially resolved team as a parameter and then returns the filtered team as a Team object. If necessary, the developer can add extra input parameters that are required to filter the team.

For example, to implement a separation of duties policy, the developer must remove the user who completed the previous activity from the list of users who can complete the next activity. In that case, the filter service needs an input parameter for the user ID in the service that is set to be removed from the input team.

For example, the developer creates a “High claim value” team filter for insurance claims that are above a certain threshold amount. These claims are for certain types of insurance claim representatives. This filter uses an input parameter `claimValue` to filter out any users who are not qualified to work high-value claims. These types of dynamic routing techniques are covered in more detail in the Implementation II course.

Team retrieval and filter service templates

- Team retrieval and filter services are used to dynamically determine who is eligible to complete a task
- The templates ensure that the minimum required parameters are met
- They are implemented as service flows



New Service Flow

A service flow lets you build, test, and run a set of services. [Learn More](#)

Name:

Use as a team service:

Service type:

Template: [Select...](#)

FINISH **CANCEL**

Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-29. Team retrieval and filter service templates

Developers use the team retrieval service and the team filter service to dynamically determine who is eligible to complete activities. These services take parameters from environment variables to influence the team selection. Developers select a template for the team retrieval and team filter service when they create a service flow. Using the templates ensures that the minimum required parameters for the services are met. Service flows are covered in more detail later in this course.

You get a chance to work with team filter services later in this training.

Identifying expert users for an activity

- Business users who are working with your process applications can collaborate or request assistance from a set of expert users who are associated with a particular task or activity
- This list of experts is displayed in the Experts panel in the Process Portal environment
- An expert group can be explicitly identified by using the **Experts Team** selection in the Assignments menu in Process Designer



Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-30. Identifying expert users for an activity

Business users who are working with the process applications can collaborate or request assistance from a set of expert users who are associated with a particular task or activity. This list of experts is displayed in the Experts panel in the Process Portal environment.

An activity must be associated with a human service before it can be assigned experts.

The experts who are listed for an activity are defined in two ways:

- Users who completed this activity in the past, which is based on historical analysis. This list is limited to a small group of users who completed the activity most frequently.
- Users belonging to a team, which is explicitly specified as an expert group for this activity.

To explicitly specify a group of experts for an activity:

- Open the business process diagram in IBM Process Designer and select the activity.
- Go to the **Assignments** tab in the Properties view.
- Specify the relevant team in the **Expert Team** field. If developers did not already create a team to identify the experts for this task, they can create a team to use.

Your server administrator can configure the teams at run time to ensure that the correct set of users is identified as experts for the activity in the Process Portal.

Unit summary

- Describe the differences between process flow data and business flow data
- Add variables to a process
- Implement gateways to control process flow
- Describe teams and process lanes
- Implement routing for tasks
- Assign an expert group to an activity
- Expose a process application to a team

Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-31. Unit summary

Review questions

1. _____ are the three types of variables that are used in Process Designer.
 - A. Private, Input, Output
 - B. Shared, Input, Output
 - C. Input, Output, Business Objects
 - D. Business Objects, Parameters, Variables

2. _____ define conditions that control whether a path is followed during the running process.
 - A. Activities
 - B. Events
 - C. Pools
 - D. Gateways

3. True or False: To model an escalation, the intermediate event must be attached to an activity.

Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-32. Review questions

Write your answers here:

- 1.

- 2.

- 3.

Review answers

1. A: Private, input, and output.
2. D: Gateways define conditions that control whether a path is followed during the running process.
3. True.

Exercise: Playback 1: Controlling process flow with business data

Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-34. Exercise: Playback 1: Controlling process flow with business data

Exercise introduction

- Create simple variables in a process
- Implement timer intermediate events in a process
- Implement gateways for a process
- Implement routing for an activity



Playback 1: Controlling process flow with business data

© Copyright IBM Corporation 2020

Figure 5-35. Exercise introduction

Unit 6. Playback 1: Business data, services, and coaches

Estimated time

02:00

Overview

This unit explains the goal of this Playback: to demonstrate that data flows from one coach to another and from one task to another inside the process. All data is bound on the coaches, and if the same data is shared across multiple tasks, Playback participants can track the data when it moves through the various activities.

How you will check your progress

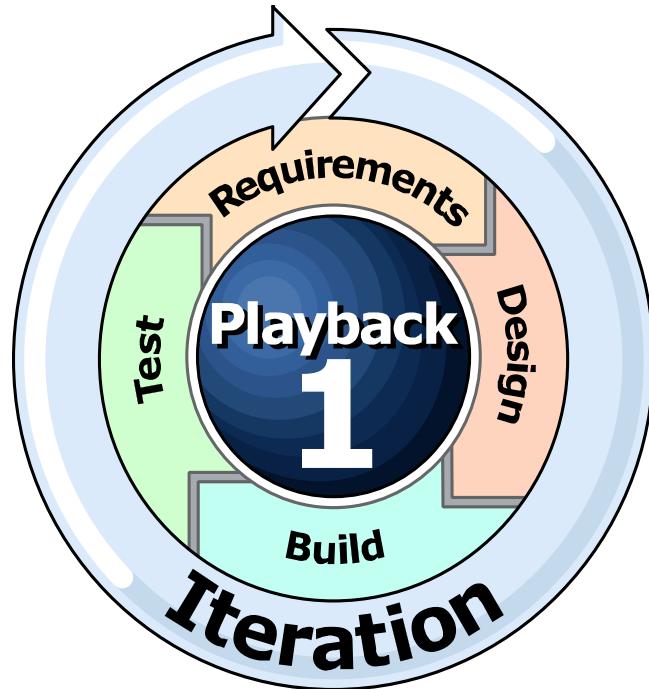
- Review
- Exercise

Unit objectives

- Build a business object
- Initialize a complex object and a list
- Build a service
- Use coaches to define and implement guided user interactions
- Implement a service for an activity in a process
- Map variables between a nested service and an activity in the overlying process
- Describe the object methods

Playback 1: Business data and services

- The goal is to demonstrate data flows from one coach to another, and from one task to another inside the process
- All data is bound on the coaches, and if the same data is shared across multiple tasks, Playback participants can see the data move through the different activities



Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-2. Playback 1: Business data and services

The goal is to demonstrate data flows from one coach to another, and from one task to another inside of the process. All data is bound on the coaches, and if the same data is shared across multiple tasks, Playback participants can track the data when it moves through the different activities.

Topics

- Creating a data model
- Building services
- Building coaches
- Implementing services in a process

Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-3. Topics

Key concepts in this unit

- **Playback 1: Business data and services:** To demonstrate data flows from one coach to another, and from one task to another inside of the process
- **Business objects:** Are used to represent the business data that is relevant in the context of your business process
- **Coach:** The user interfaces for human services
- **Services:** Made up of steps that define what happens when service (task) activities are triggered in a process
- **Data mapping:** Is used to map the input and output variables of the service to variables in the process so that runtime variable values can be passed to and from the service

Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-4. Key concepts in this unit

6.1. Creating a data model

Creating a data model

Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-5. Creating a data model

Until now, simple variables are implemented to drive process flow. Process flow data is only part of the data necessary for the process to function correctly. Much of the process flow data is derived from or dependent upon business data. In this Playback, the project development team defines the business data model for the process.

Defining the business object data model

- Business objects are used to represent the business data that is relevant in the context of your business process
- Create a data model that accurately reflects the business data and its structure
 - Do not be influenced by existing logical or physical data models

Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-6. Defining the business object data model

Business objects are used to represent the business data that is relevant in the context of your business process. Create a data model that accurately reflects the business data and its structure. Do not allow existing web pages, existing systems, coaches, web services, database tables, or other influences to affect the development of your process business objects that belong in your data model.



Building a complex business object

HiringRequisition

Common

Name: HiringRequisition
Modified: author1 (Nov 17, 2020, 10:39:10 AM)

Documentation:

Behavior

Definition type: Complex type
Shared object:

Parameters

Parameters

- requisitionNumber (String)
- dateOfRequest (Date)
- requester (String)
- datePositionAvailable (Date)
- hiringManagerComments (String)
- position (Position)
- compensationDetails (CompensationDetails)
- departmentDetails (DepartmentDetails)
- recruitingDetails (RecruitingDetails)

Advanced Properties

- When the system data toolkit or other business objects do not match the specifications, create a custom business object
- Create rules about complex data that is nested or hierarchical

Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

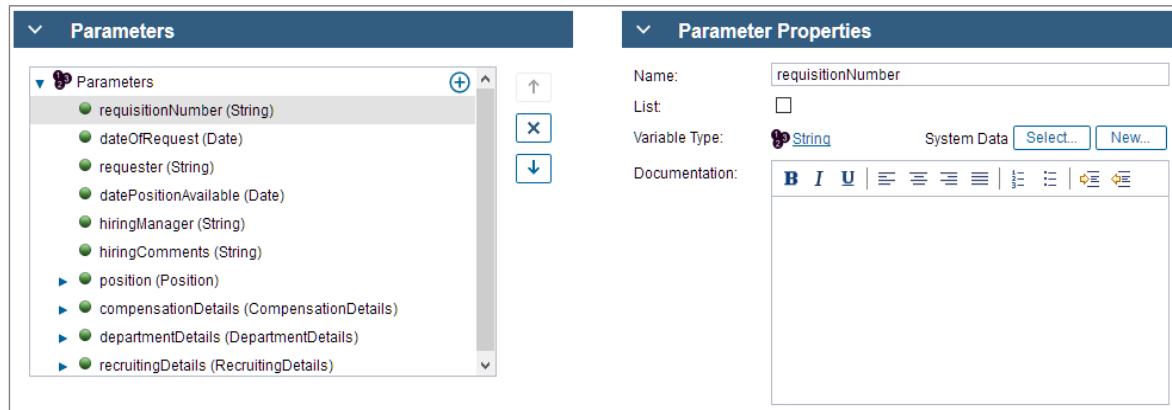
Figure 6-7. Building a complex business object

When the System Data toolkit types or the other business objects do not match the project specifications, you can create custom business object types. In IBM Business Automation Workflow, you can use a base business object type or define a new complex structure to create a custom business object type. You can create rules about complex data that is nested, or hierarchical. Data that is referenced within the text of a rule is not limited to simple object types such as String, Integer, or Date. You can also create complicated rules with nested object structure.

Building the data model

After you define the data, organize your data into different logical units:

- Some organization is obvious, but other data is more difficult to organize
- Look to existing data models in your company for ideas
- Example of a business object: A structure that contains multiple elements, all pertaining to the same “subject,” which is what the structure represents



Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-8. Building the data model

When data is defined, organize the data into different logical units. An example might be a street address, city, and state that become a unit that is called **address**. Some organization is obvious, but other data is more difficult to organize. Look to existing data models in the organization for ideas. However, it is not necessary for these models to match what is specified. These models can be starting points for a robust data model solution.

An example of a business object is a structure that contains multiple elements all pertaining to the same subject. This subject is what the structure represents (for example, a name and phone number would be relevant to a customer identification).



Standardizing the variable naming convention

- Business objects begin with an uppercase letter
- If the business object consists of more than one word, capitalize the first letter of each word
- Variables and business objects are case-sensitive
- In this example, the **position** variable uses the business object type Position

The screenshot shows a software interface with a sidebar titled "Parameters". Under "Parameters", there is a list of variables:

- requisitionNumber (String)
- dateOfRequest (Date)
- requester (String)
- datePositionAvailable (Date)
- hiringManager (String)
- hiringComments (String)
- position (Position)** (highlighted with a red box)
 - jobTitle (String)
 - jobDescription (String)
 - jobLevel (String)
 - numDirectReports (Integer)
- compensationDetails (CompensationDetails)
 - salaryToOffer (Decimal)
 - bonusAmount (Decimal)
- departmentDetails (DepartmentDetails)
- recruitingDetails (RecruitingDetails)

Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-9. Standardizing the variable naming convention

Business objects begin with an uppercase letter. Adopt this convention to differentiate variables from business object types. If the business object consists of more than one word, capitalize the first letter of each word. In the example that is provided on screen, the `position` variable uses the business object type `Position`. Variables and business objects are case-sensitive.

Refactoring

- Limited refactoring support for the following cases:
 - Business object rename in processes and services
 - Business object attribute rename in processes and services
 - Local variable rename in the specific process and related services



[Playback 1: Business data, services, and coaches](#)

© Copyright IBM Corporation 2020

Figure 6-10. Refactoring

With time, applications change and business objects, their attributes, and variables might be renamed. However, many parts of a business process might reference or have a dependency on a business object, an attribute, or a variable so that renaming can produce unexpected results. Refactoring can help to solve this problem, and is described as follows:

- **Refactoring business objects:**

To rename a business object, you right-click the business object that is in the process application library in the **Data** category list. You then rename the business object.

- **Refactoring business object attributes:**

Business objects are themselves composed of other variables called attributes. To rename a business object attribute, select the attribute in the **Parameters** list for the business object. Change the name in the **Name** field.

- **Refactoring limitations:**

- JavaScript code in coaches is not updated.

- **Rename variables:**

Variables are found within a business process or a service. In other words, renaming a variable does not affect another business process or service. However, renaming a variable can affect a

reference to it within the same business process or service. To rename a variable, you click the **Variable** tab and select the variable to rename. Change the name in the **Name** field.

Complex objects and lists

- You can declare any variable to be a list, or an array, of a business object type
 - Instead of containing only one instance of the type declared (String, Date, or Integer), your variable holds many of the same business object types
- All complex business objects and all lists (arrays) must be initialized before you use them in a process or service
 - If your business object type includes elements that are themselves complex business object types other than the simple types, then you must initialize them before you use them
 - You must also initialize lists before you use them, by using `.listOf`



Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-11. Complex objects and lists

When declaring lists, developers can declare any variable to be a list, or an array, of a business object type. This statement means that instead of containing only one of the types declared (String, Date, Integer), the business object holds many of the same business object types. Therefore, if developers want to create a business object that contains multiple integers, they create a private list variable of type `Integer`.

To create a variable list, select the **Is List:** checkbox, and the list object adds the `(List)` designator after the variable type in the variable.

Initializing complex objects and lists

- All complex business objects and all lists (arrays) must be initialized before you use them in a process or service
- Before using a complex business object, initialize it by using a script like the following script:

```
tw.local.requisition=new tw.object.Requisition();
```

- You must also initialize lists before you use them, by using `.listOf`

```
tw.local.yourStringList = new tw.object.listOf.String();
```

- Then, you add elements to the list:

```
tw.local.yourStringList[0] = "First Element";
```

or

```
tw.local.yourStringList.insertIntoList(
    tw.local.yourStringList.listLength, "First Element");
```

Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-12. Initializing complex objects and lists

In IBM Business Automation Workflow, all complex business objects and all lists (arrays) must be initialized before they are used in a process or service. If the complex business object or list is not initialized, the process receives runtime errors, or notices that the coach controls to which the business objects are bound do not behave as expected.

Before implementing a complex business object, create a script like the following to initialize it:

```
tw.local.requisition=new tw.object.Requisition();
```

In the preceding script, the name of the variable that is being initialized is `tw.local.requisition`. The name of the complex business object type is `Requisition`.

If a business object includes nested complex business objects, then you must initialize the nested business objects before they are used.

You also initialize lists before you use them with `.listOf`. This command works for simple and complex business object types. If a variable is a list of `Strings`, initialize it by creating a script:

```
tw.local.yourStringList = new tw.object.listOf.String();
```

Then, add elements to the list:

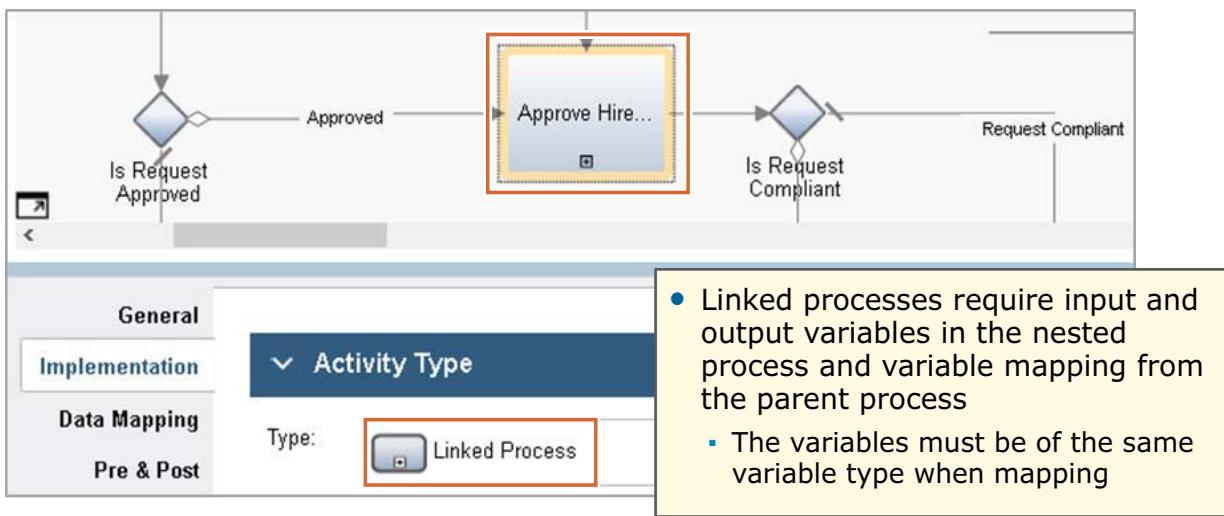
```
tw.local.yourStringList[0] = "First Element";
```

Or:

```
tw.local.yourStringList.insertIntoList(tw.local.yourStringList.listLength, "First  
Element");
```

Nested processes and variables

- A subprocess has access to the data of the parent process
 - Data mapping is not required to pass data into or out of the subprocess
 - You can also declare private variables within the subprocess that are not visible to the parent process



Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

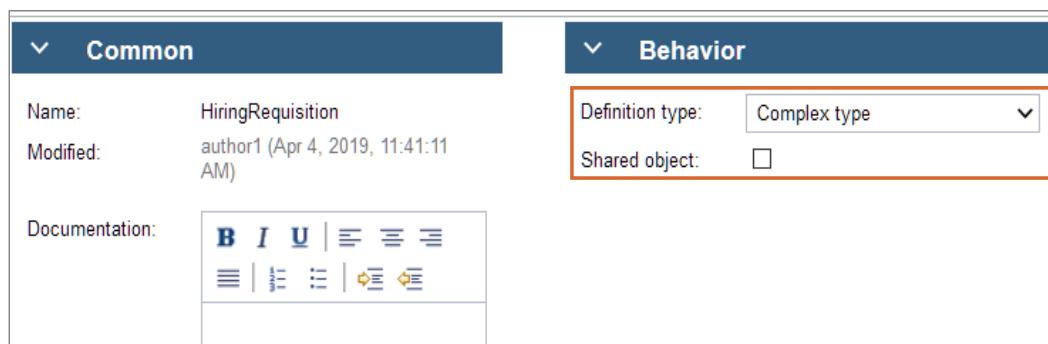
Figure 6-13. Nested processes and variables

Carefully consider how variables are passed from the parent process to other processes when using nested processes. Subprocesses and linked processes handle variables differently.

- A **subprocess** has access to the data of the parent process, and data mapping is not required to pass data into or out of the subprocess. However, you can also declare private variables within the subprocess that are not visible to the parent process.
- **Linked processes** require developers to create input and output variables in the nested process and to map variables from the parent process to the linked process. The variables must be of the same variable type when mapping. This input and output variable declaration allows the data value to pass successfully from one process to another.

Sharing a complex business object

- Custom business objects in a process application are available for all processes and services that are included in the process application
- Share custom business objects across process applications through toolkits
- If the business object and its values must be accessible to other instances at run time, check the shared object checkbox in the Behavior section of the Business Object interface
 - Shared business objects apply only to a complex type
 - The data within a shared business object is shared between business processes and tasks



Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-14. Sharing a complex business object

When a custom business object is created in a process application, that object is available for all processes and services included in the process application. To share a custom business object across process applications, create or store the custom object in a toolkit. Then, create a dependency on that toolkit from the process applications that require the variable.

The **Shared Object** checkbox of the business object and its values must be accessible to other instances at run time. The business object becomes a shared object. Shared business objects apply only to a complex structure type. The data within a shared business object is shared between business processes and tasks.

A shared business object uses database resources. The data within a shared object is persisted to the database when the shared object is created. The business process or task is persisted to the database when the JavaScript method `save()` is applied to the shared business object.

6.2. Building services

Building services

Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

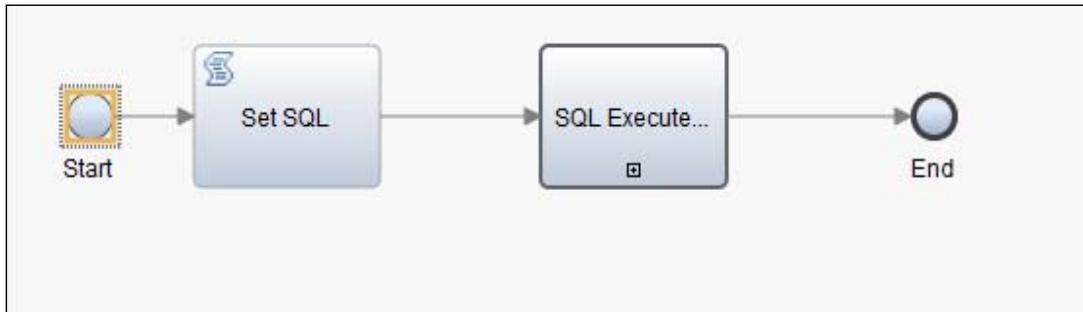
Figure 6-15. Building services

After modeling the business process, it is time to think about these processes as sets of interactions. The goal of this Playback involves building interactions.

These interactions represent a critical piece of Business Process Management (BPM). Instead of focusing solely on the chains of activities with their sets of inputs and outputs, think about the protocols and agreements that are made between the users.

Services provide a strategy for choreographing these protocols and agreements. Choreography is an abstract notion of process. It is used to describe the interactions of collaborating entities, each of which can have its own internal orchestration (modeling) processes.

What are services?



Playback 1: Business data, services, and coaches

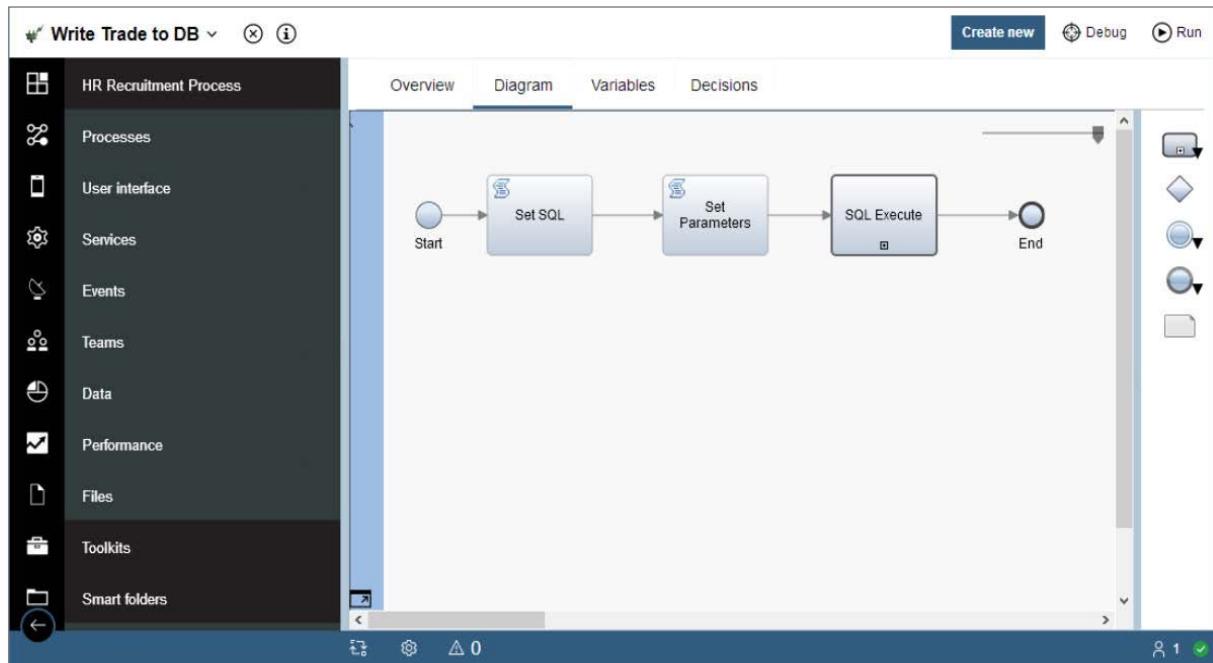
© Copyright IBM Corporation 2020

Figure 6-16. What are services?

Services are made up of steps that define what happens when service (task) activities are triggered in a process. Each step enables the service to do a different job, from integrating with an external data source to generating HTML forms that are called coaches.

IBM Training

Creating services



Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-17. Creating services

The IBM Process Designer is where you build reusable services that implement the activities in your processes. Services differ from processes in that they have no lanes. The service palette is customized with what items you can use to build the service. For example, in this Playback, you are examining human services. A coach can be added only to a human service. Also, for a web service, a service modeler and palette are not required.

When creating a service, developers give them human-readable names. They use spaces between the words so other developers can understand what this service does. Name the service after the activity that the service does so developers can identify its function without having to open the service in the IBM Process Designer application.

Process data in services: Services use business objects and variables just as processes use business objects and variables. Remember, these objects (variables) can be declared as private, input, or output. Private variables are values that are seen only within the service; they are also available to any nested services. Input variables are mapped to values that you can pass into the current service. Output variables are mapped to values that you can pass out from this service to a parent process or service.

6.3. Building coaches

Building coaches

[Playback 1: Business data, services, and coaches](#)

© Copyright IBM Corporation 2020

Figure 6-18. Building coaches

A human service presents a web form for a process participant to interact with. This web form in IBM Business Automation Workflow is an HTML page that is called a coach. This topic covers the fundamentals of building coaches, and enables developers to create coaches in a short amount of time.

During this part of the Playback, developers do not look to alter the appearance or add any extra functions to their coaches. Developers concentrate on creating forms that can be filled out and ensuring that data is posted back to the server. The process gathers all the necessary data from the users in the process to complete the overall process.

Coach overview

- Coaches are the web-based user interfaces that provide process-related data to Process Portal users and collect input from those users
- Coaches are implemented in the client-side human services, which manage the flow from one coach to another
- Two types of user interface are available for human services:
 - Task completion
 - Stand-alone services (dashboard, a startable service, or a URL service)
- Coaches provide a way for team members to input business and process data into the process in a dashboard or task completion
- Coaches provide a wizard-like approach for team members to complete their tasks
- The coach is rendered in a browser for the business user when the coach service (the process activity of the participant) is run
- Similar to building processes and services, developers use a WYSIWYG interface to build coaches by dragging objects from a widget palette onto the coach designer canvas

[Playback 1: Business data, services, and coaches](#)

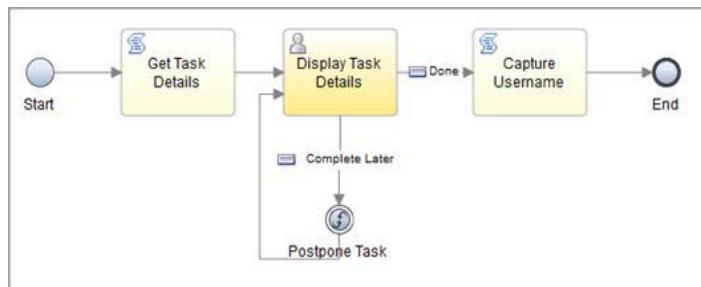
© Copyright IBM Corporation 2020

Figure 6-19. Coach overview

Coaches are the user interfaces for human services. You have two types of user interfaces for human services: task completion and stand-alone services. A task completion user interface implements a specific activity within a process instance. It has access to the details of that process instance. A stand-alone service can be a dashboard, a service, or a URL service. A dashboard is a stand-alone user interface that users can run at any time. Users can access dashboards through the Process Portal. A service can be started in Process Portal at any time, whereas a URL service can be started directly through the URL.

Build user interface for human services by using coaches

- To build the user interface for human services in the Process Designer, you use coaches
- Coaches can contain coach views, also known as views
 - A view is a reusable unit that you use in coaches and other coach views
- A view generally defines the user interface for a particular type of data, giving you the potential to customize the user interface by specifying the configuration options



Playback 1: Business data, services, and coaches

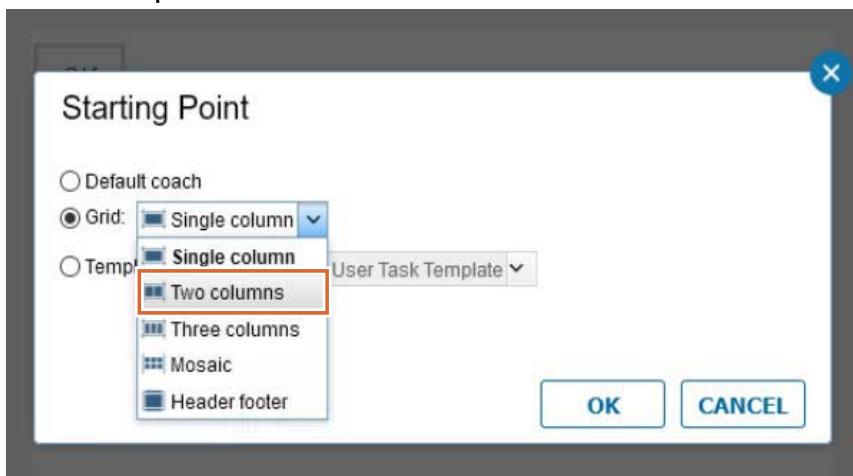
© Copyright IBM Corporation 2020

Figure 6-20. Build user interface for human services by using coaches

- The UI toolkit contains a set of (coach) views for designing applications that can run on multiple device types, such as mobile and desktop devices. All the views in the UI toolkit are suitable for use on both desktop and mobile devices.
 - Each view points to a specific set of files: A JavaScript file that provides the behavior of the view
 - Image files to represent the view on the palette and on the canvas
 - A preview JavaScript file that shows a preliminary image of the view when it is dropped on the coach editor palette
- The view can also have other files such as CSS and LESS files for styling, localization files, and other control-specific resource files, but these additional files are optional.
- Views are covered in more detail in the next Unit.

Laying out a coach by using the grid layout

- You can lay out a coach by using cells in a grid
- A grid is a container that is 12 units wide
 - The actual width of each unit is variable and depends on the screen size
- Within a grid, you can have containers and cells
 - A container groups related cells
 - Cells are placeholders for content



Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-21. Laying out a coach by using the grid layout



The coach designer interface

Coaches are web-based forms or dashboards that a developer builds for business users to complete tasks that are assigned to them at run time.

The screenshot shows the IBM Coach Designer interface. At the top, there are tabs for 'DESIGNER' and 'INSPECTOR'. Below the tabs, the title 'Hiring Form' is displayed, along with buttons for 'Create new', 'Debug', and 'Run'. The main workspace contains a navigation bar with tabs: 'Overview', 'Diagram', 'Variables', and 'Coach: Hiring Form'. The 'Coach: Hiring Form' tab is selected. The workspace itself contains several sections and controls. On the left, a sidebar provides navigation icons. On the right, a panel titled 'Building blocks' lists various components like 'Alerts', 'Area chart SDS', 'Badge', 'Bar chart SDS', 'Breadc...', 'Button', 'Caption box', and 'Check box', each with a preview icon and a 'Search for a building block' input field.

Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-22. The coach designer interface

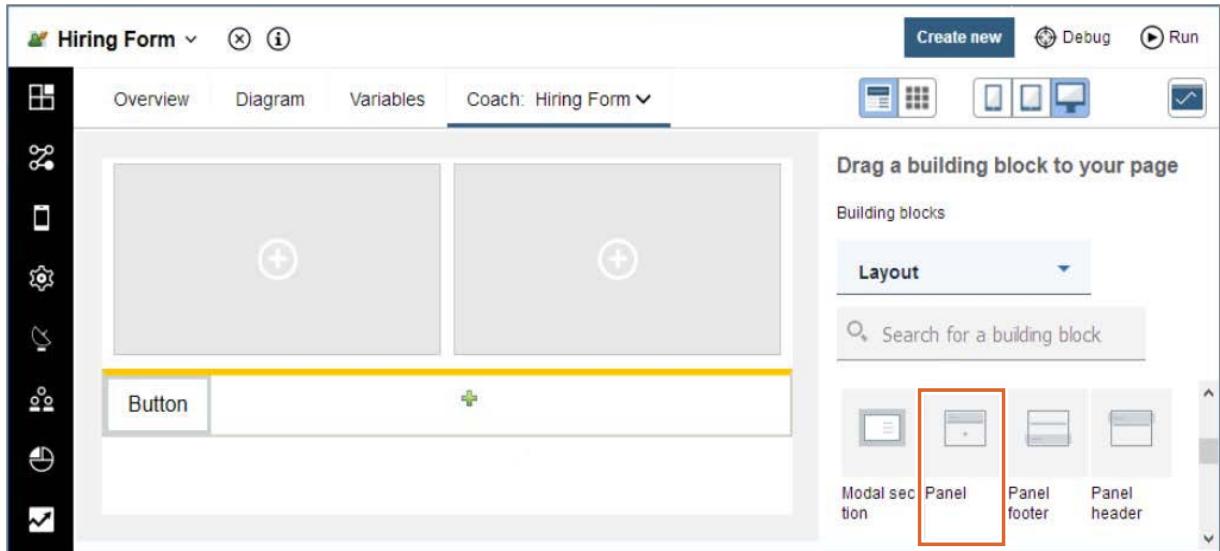
Coaches are web-based forms or dashboards that a developer builds for business users to complete the tasks or follow process or task performance metrics that are assigned to them at run time. All development occurs inside a web browser, so developers have a “what you see is what you get” (WYSIWYG) view of the rendering of the coach.

Coaches are made up of views and controls. The view provides the user interface elements and layout for the coach through sections and controls. Each view can contain one or more views, which creates a parent-child relationship between these views. At run time, the parent view is rendered as a `<div></div>` tag that contains a nested `<div></div>` tag for each child view. Each view also has a binding function to use data values from a business object, CSS code to control its visual layout, and JavaScript to define its behavior. More information on views is covered later in the next unit.



Panels

- Views contain a group of views or controls, or both (other views).
- Views that are panels contain a group of panels or controls or both.
- Panels control the layout of your controls after they are rendered in the browser.



Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

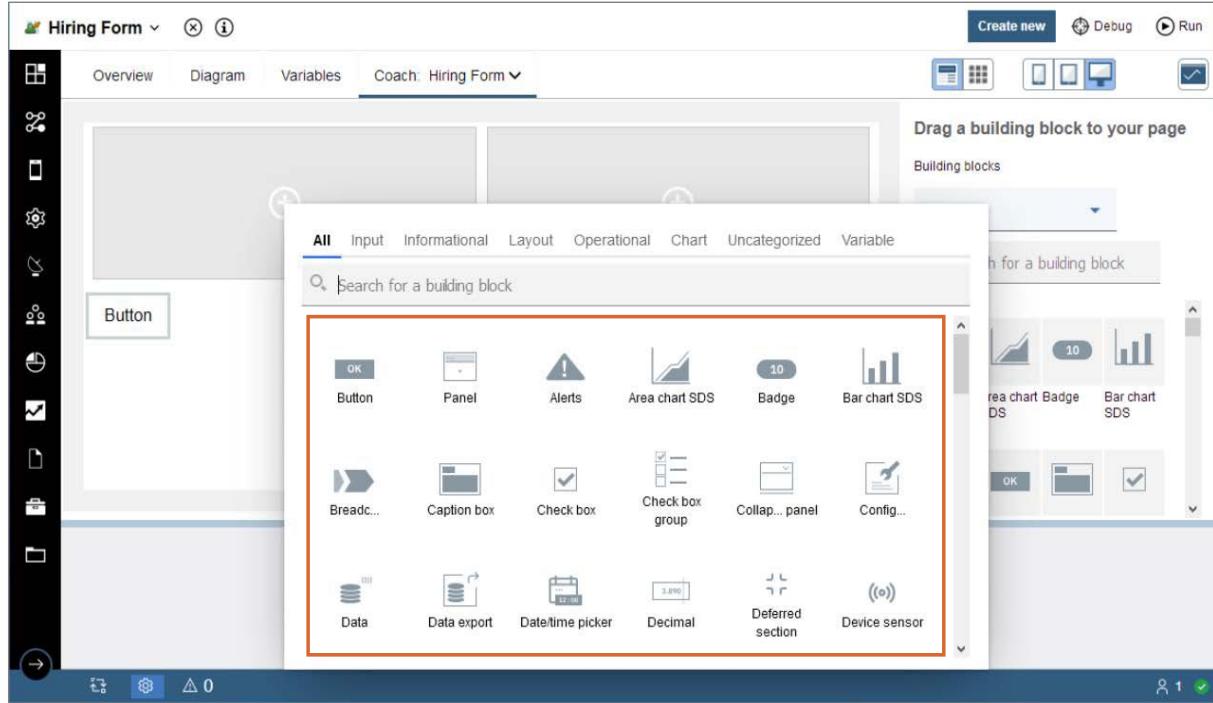
Figure 6-23. Panels

Views that are panels contain a group of panels or controls or both (other views). Panels control the layout of your controls after they are rendered in the browser. The controls are made of HTML `div` and `span` sections. To add a panel, select Layout in the Building blocks section and drag the Panel to a spot on the canvas. When the mouse cursor turns to a green plus (+) sign, you can drop the panel. Gold highlights of other objects on the canvas highlight where the object is placed. A tabs control is also used to group similar sections. All the panels can be nested within each other.



Controls

Controls are Views that include buttons and other standard HTML objects.



Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-24. Controls

Alternatively, you can click the plus sign (+) in a grid to add Controls or objects (or widgets) to the canvas. Controls are Views that include buttons, check boxes, input boxes, date/time pickers, select controls, and other standard HTML objects. Although the standard objects cover most of a developer's needs, IBM Business Automation Workflow allows developers to customize the controls. Controls are customized by using a custom HTML block on the coach designer that includes JavaScript, CSS, or custom HTML elements. When the controls are laid out, it is easy to copy and paste them into a view to share with other coaches across your process application or other installations.



View object settings

View property settings are separated into five categories

Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

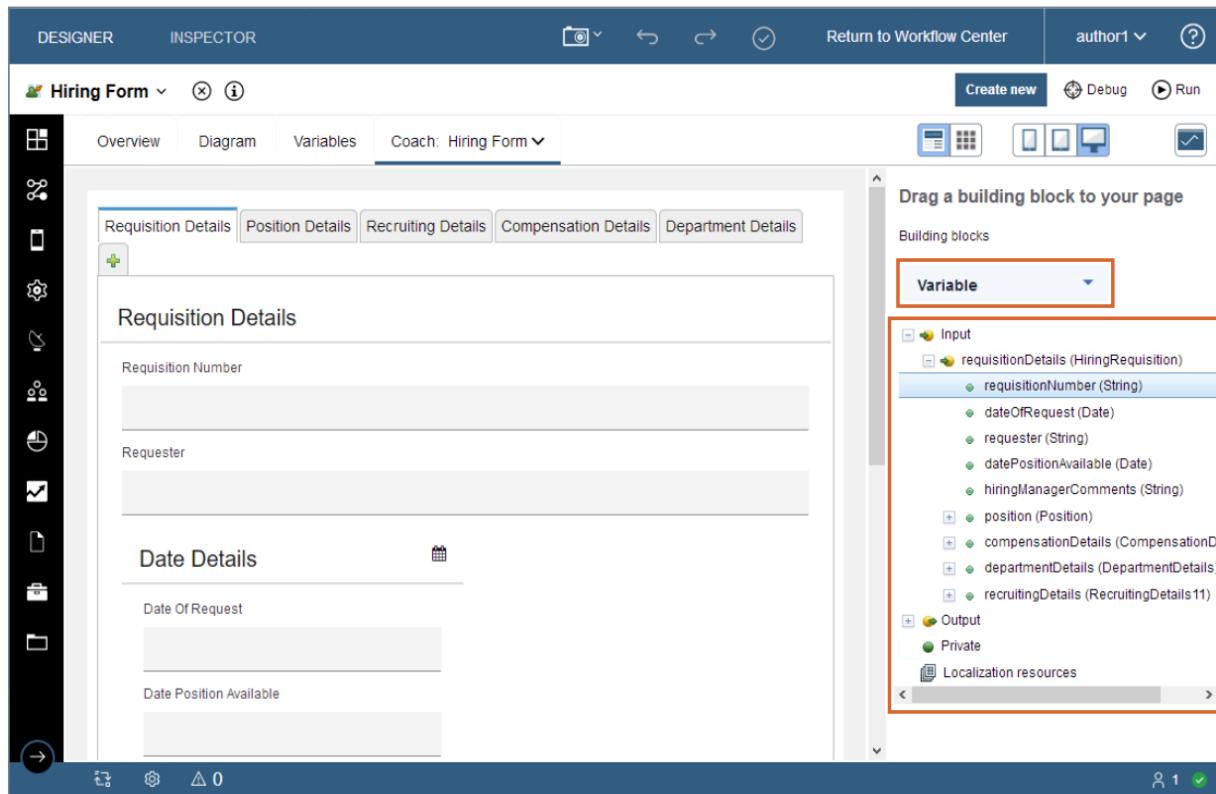
Figure 6-25. View object settings

Views, whether sections or controls, contain settings for use in the coach. The property settings are separated into five categories:

- **General:** Describes the common properties of the object, such as the control ID, and the behavior, such as binding and label visibility
- **Positioning:** Sets the padding, margins, height, width, and overflow of the objects on the canvas
- **Configuration:** Allows developers to modify the configuration attributes of an object
- **Events:** Define logic to be called based on events such as on load, on change, on focus, and on blur.
- **Visibility:** Sets the visibility of the object through a variable value, a rule, or a script
- **HTML Attributes:** Allows developers to add classes and attributes to the HTML objects that are rendered in the browser

IBM Training

Adding controls from server-side variables



Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

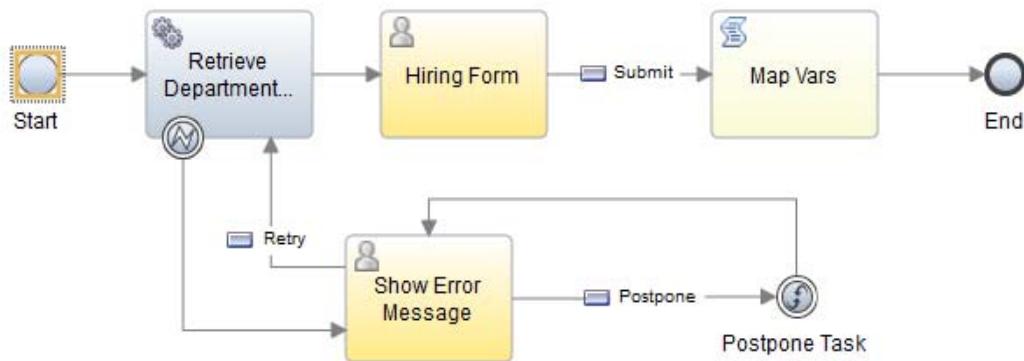
Figure 6-26. Adding controls from server-side variables

Developers can use a quick and easy way to add controls to a coach bound to server-side variables. The first step is to make sure that the simple and complex variables (input, private, or output) are defined in the service.

You add variables in the same manner as adding views. Select Variables in the Building blocks section. A developer can drag the individual elements of a variable or the entire variable.

If the output of the coach controls the process flow, add the process flow variables necessary to control the process directly on the coach. They look like input boxes when running the coach, and when doing a Playback, developers enter the process flow data into that input box to control the gateways.

Buttons and sequence flow



Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-27. Buttons and sequence flow

Next, the developer needs the ability to submit the coach to the server and move the service along. Buttons allow the form to post the data back to the server from the client. Developers model services, such as saving progress or retrieving external data, after the user fills out some form data.

After developers add the number of buttons that they want to the coach, sequence flow is connected in the service diagram. Each button requires drawing a new line in the diagram as each button item corresponds to one line or flow.



Setting screen size

The screenshot shows a form titled 'Requisition Details' with several fields: 'Requisition Number', 'Requester', 'Date Details', 'Date Of Request', 'Date Position Available', and 'Hiring Comments'. On the right, a properties panel is open with tabs for General, Positioning, Configuration, Events, and Visibility. The 'Positioning' tab is selected, showing options for 'Width' (set to 50%), 'Margin', 'Padding', 'Content' (with a dashed border), 'Minimum height', 'Overflow Content' (set to 'Show all content (visible)'), and 'Minimum width'.

- The controls can respond dynamically to three different screen sizes.

Size	Resolution
	Small 640 pixels or less
	Medium 641 - 1024 pixels
	Large More than 1024 pixels

Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-28. Setting screen size

Process Designer supports three size layouts for user interfaces.

Use the Positioning tab in the properties for the control to set the size of the control. The settings resize the coach to match three different formats: small, medium, and large. These resolutions approximate the resolutions for mobile devices, tablets, and desktop displays.

Screen size settings are as follows:

- Small:** 640 pixels or less
- Medium:** 641 – 1024 pixels
- Large (default):** More than 1024 pixels

6.4. Implementing services in a process

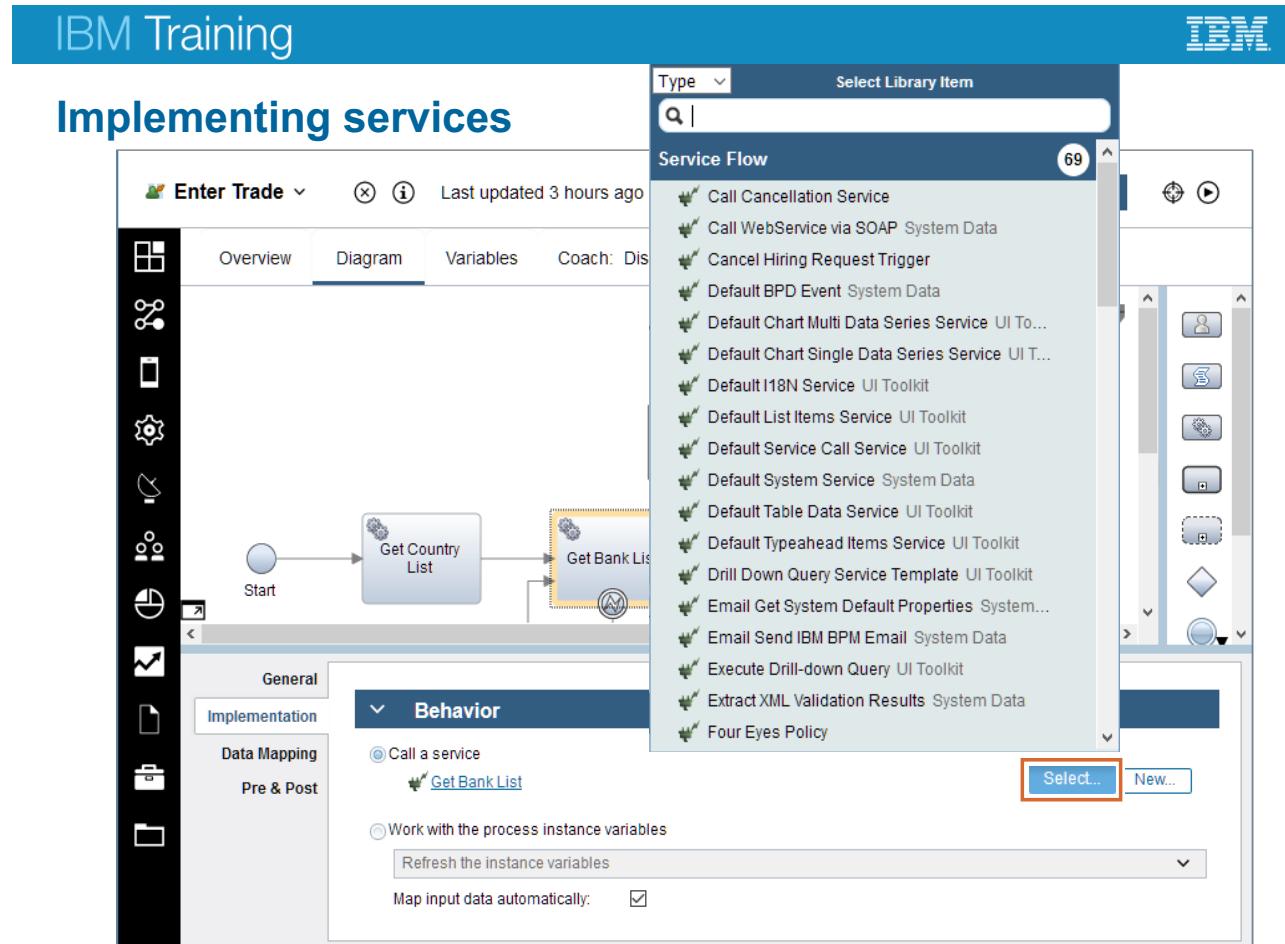
Implementing services in a process

Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-29. Implementing services in a process

In this topic, the activities are implemented with the artifacts that were previously built.



Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-30. Implementing services

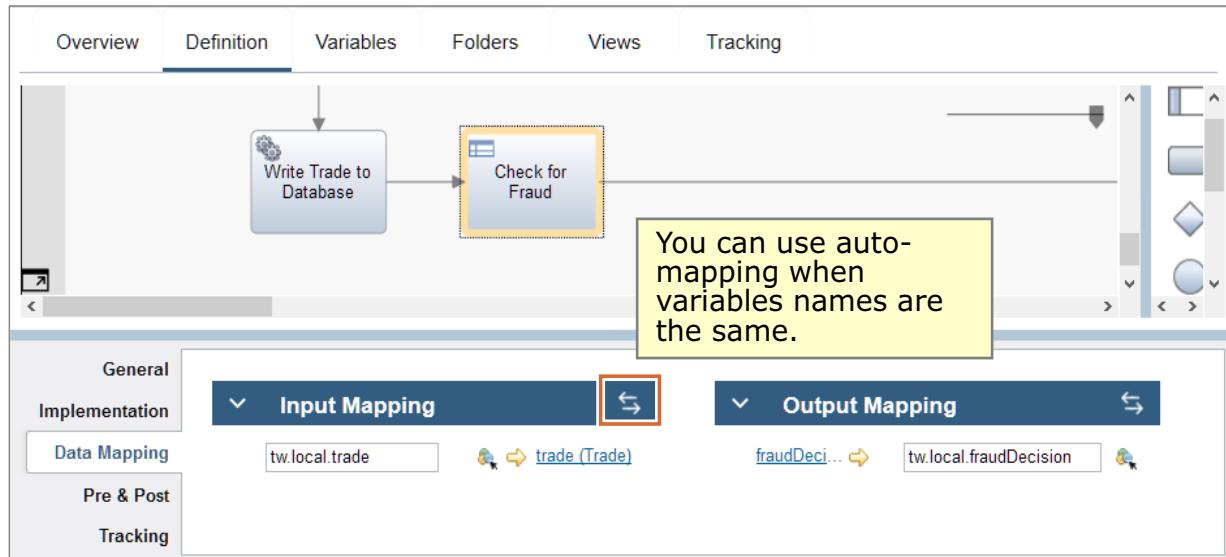
When developers first add an activity to a lane in a process, it has default functions. To specify what happens when an activity is triggered at run time, you must attach a service. Services are attached in the **Implementation** section in the **Properties** tab of an activity. Services can be run from the process, from a coach, or from other services.

Business objects, variables, and data mapping

- Business objects
 - In the Process Designer, business objects are complex objects that are created from simple business objects or other complex business objects
 - Define the business data
- Variables
 - Instantiate business objects that activities use in a process or by steps in services, such as service flows or human services
 - Capture and store the business data
 - Each has its own type and scope
 - Must be declared before you can start to use them
- Data mapping
 - In the Process Designer, you set the input and output data map to pass variables to an activity or a step
 - You must set the input and output mapping for each activity in a process
 - Map variables in the parent process to the variables received and generated by nested processes and services

Data mapping

- Data mapping is used to pass the values of variables between an activity (task) in a process and a service.
- When you attach a service to an activity, the **Data Mapping** section must be populated with the input and output variables of that service.



Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-32. Data mapping

Data mapping is used to pass the values of variables between an activity (task) in a process and a service. When you attach a service to an activity, the **Data Mapping** section must be populated with the input and output variables of that service. The **Data Mapping** section is used to map the input and output variables of the service to variables in the process so that runtime variable values can be passed to and from the service. Variables are automatically mapped to the process variables, or they are manually specified.

Mapping input and output data for an activity or step

- Data mapping sets the input and output map to pass variables to an activity or a step
- Requires a set of declared variables and an activity, or a service that contains a set of declared variables and a step
- The activity or step must implement a service or linked process, which also contains a set of declared variables
- Set the input and output mapping for each activity in a process
- The subprocesses and services that implement the activities generate variable values
 - These variable values map to the variables from the main process
- Auto-mapping works only when variable names and types match exactly
- Always use an identical name and data type for a set of input and output variables that are passed in, processed, and then passed back

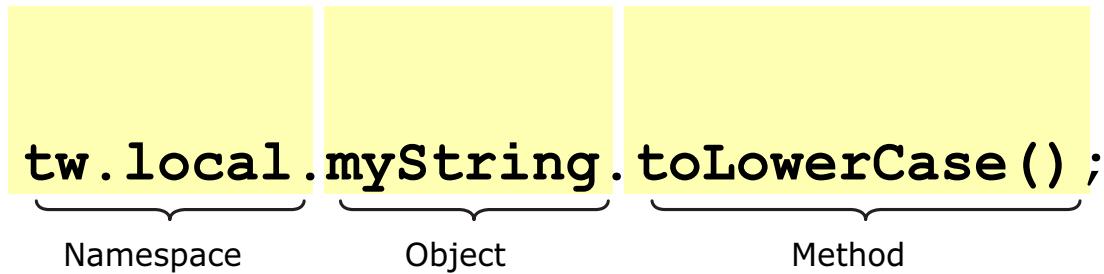
Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-33. Mapping input and output data for an activity or step

Mapping input and output data are applicable only if the client-side human service is used within a process. If the client-side human service is used for a dashboard, data mapping is not applicable.

Understanding object methods



Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-34. Understanding object methods

JavaScript methods corresponding to the object type are available with the standard syntax.

For example, the JavaScript method `toLowerCase` can be called on an object of type String. In this example, the results of the call are used to convert the value of `tw.local.myString` to all lowercase.

Unit summary

- Build a business object
- Initialize a complex object and a list
- Build a service
- Use coaches to define and implement guided user interactions
- Implement a service for an activity in a process
- Map variables between a nested service and an activity in the overlying process
- Describe the object methods

Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-35. Unit summary

Review questions

1. True or false:

Capitalize the first letter of business objects. Use a camel case data type for a variable when creating an instance of the object (for example, employeeId).

2. Coaches are made up of what two types of components?

3. How do you define what should occur in order for a participant to complete an activity?

- A. Services
- B. Gateways
- C. Lanes

Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-36. Review questions

Write your answers here:

- 1.
- 2.
- 3.

Review answers

1. **True.** This naming convention makes it easier to identify whether a business object is the object type or an instance of the object.
2. **Sections and controls.** Sections control the layout of your controls after they are rendered in the browser. Controls include buttons, check boxes, input boxes, date time pickers, select controls, and other standard HTML objects.
3. **A.** Services define what happens when users try to complete their activity or when the system must run an activity in a process. Services are attached to an activity from the Implementation section in the Properties view.

Exercise: Playback 1: Business data, services, and coaches

Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-38. Exercise: Playback 1: Business data, services, and coaches

Exercise introduction

- Determine and organize data when provided with a written process
- Add business objects and object types
- Create a client-side human service
- Add variables and business objects to a process application
- Create and configure a coach to obtain process participant input
- Model a coach by using the concept of grids
- Add coach controls to control process flow
- Create a client-side human service and coach for the General Manager review activity
- Implement an activity by attaching a service and mapping data



Playback 1: Business data, services, and coaches

© Copyright IBM Corporation 2020

Figure 6-39. Exercise introduction

Unit 7. Playback 1: Enhancing coaches

Estimated time

01:30

Overview

This unit covers how to build coaches and then enhance them. Views are reusable assets that can be shared with multiple coaches or even multiple process applications. The unit covers how to validate the process flow and use toolkits.

How you will check your progress

- Review
- Exercise

Unit objectives

- Describe how to create tabs on a coach
- Enhance coaches and apply a theme
- Explain how to create a reusable View
- Describe how to create a snapshot
- Export your process application
- Share your assets by using a toolkit
- Conduct a Playback session

Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-1. Unit objectives

Playback 1: Enhancing coaches

- It is your chance to show off all the new features that you built into your coaches
- Demonstrate how each coach looks to the different participants in your process
- This Playback garners intense scrutiny
- Before you enter this Playback session, make sure that no regression errors occur
- Ensure that all the coaches are functioning correctly, and that this Playback shows off the good results

Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-2. Playback 1: Enhancing coaches

This stage of the Playback is your chance to show off all the new features that you built into your coaches. Log on to the process portal and demonstrate how each coach looks to the different participants in your process. This Playback garners intense scrutiny. Before you enter this Playback session, make sure that no regression errors occur; ensure that all the coaches are functioning correctly and this Playback shows off the good results.

Business users always demand heavy user interface (UI) requirements. Traditionally, it is a struggle between developers and project management. By debating the compromises to the UI, consensus can be built around what is necessary to help participants complete their tasks in the least amount of time in the process. Compare this approach to spending a tremendous amount of time to create a complex UI screen, which might hold little business value.

Key concepts in this unit

- **Coaches:** The user interfaces for human services, and composed of HTML, JavaScript, and CSS
 - The three elements control the content, functions, and presentation of every web page
- **Theme:** Theme determines the global appearance of a page
 - The purpose of theme is to ensure visual consistency
 - Themes affect the navigational structure, the banner, the colors, fonts, and other visual elements of a page
- **Views:** Reusable assets that can be shared with multiple coaches or even multiple process applications

Topics

- Enhancing coaches
- Views
- Theme support
- Conducting the Playback session

Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-4. Topics

7.1. Enhancing coaches

Enhancing coaches

[Playback 1: Enhancing coaches](#)

© Copyright IBM Corporation 2020

Figure 7-5. Enhancing coaches

After modeling the basic coaches and ensuring that the data is flowing in the process, it is time to revisit the coaches that were created during this Playback. Now is the opportunity to add functions, interaction, stylization, menus to select from, and all the other user interface enhancements necessary for users to complete their tasks.

This portion of the Playback usually receives the highest scrutiny, so take enough time to meet the business requirements, but balance this time with the overall project timelines. Avoid adding extra functions when it does not provide value. This unit focuses on using controls that are included in the IBM Business Automation Workflow coach toolkits. HTML, JavaScript, and CSS are used to enhance the stock coaches; but if these stock controls do not meet your needs, feel free to create your own by using Views.

Now is not the time to cause a regression error. Because you verified the data flow, make sure that you do not break something that was working previously.

Basics of coach enhancement

- Coaches are made of three things:
 - HTML controls the content of the page (divs, spans, inputs, and other tags)
 - JavaScript controls the interactive functions of the page
 - Cascading stylesheets (CSS) control the visual style of the page
- All of the following tasks can be accomplished:
 - Build custom HTML elements and controls
 - Match a corporate color scheme or existing website layout through custom CSS
 - Alter control or page functions with JavaScript
- Any page requirement can be fulfilled
 - If it can be built and shown in a browser, it can be done with a coach

[Playback 1: Enhancing coaches](#)

© Copyright IBM Corporation 2020

Figure 7-6. Basics of coach enhancement

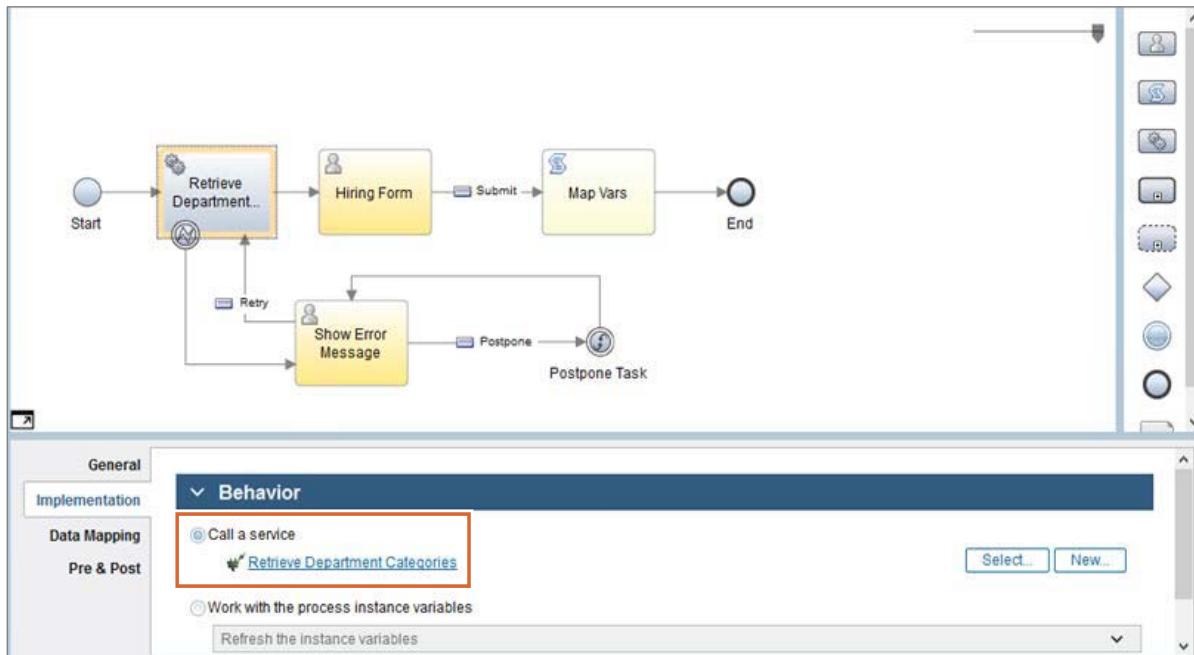
Coaches use the same technology as any web development environment to serve web content to users. Coaches are composed of HTML, JavaScript, and CSS. The three elements control the content, functions, and presentation of every web page. All three elements are contained in Views, the widget components that are used in coach designer to create a final web form or dashboard. Views are covered in more detail later in the unit.

Coaches are made of HTML divisions (divs) and spans, which control the structure of the page. Controls and labels are inside these elements. The stock JavaScript and CSS files are linked through <include> elements in the HTML. Although the Views in coaches come preinstalled with CSS and JavaScript to control the stock functions, any modification from the stock can be accomplished. That includes creating custom HTML controls from scratch, creating custom stylesheets, and changing or adding JavaScript functions.

Some organizations might be hesitant to use coaches because they are unaware of the capabilities, but the use of coaches to create a web page has no limitations. If it can be built and shown in a browser, it can be done with a coach. The advantage is that the rapid application development tools allow developers to create the web page quickly and connecting data to the fields on the page is much quicker than using external web pages. Other considerations like security, sessions, cookies, and the back-end considerations are taken care of for the coach developer.

Implementing a select control (1 of 2)

- You can use a select control to limit input values in a specific control.



Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-7. Implementing a select control (1 of 2)

You can use a select control to limit input values in a specific control. Until now, a user was able to insert any type of data into an input box. One way to standardize the input data is to provide options for a user to choose from. A user can choose among many different options from a list of choices on a coach. The intent is to limit the values that a user can input for a certain control. The solution that is provided in IBM Business Automation Workflow is to use the select control that is provided in the UI Toolkit. This default control is a setup that uses an Ajax service with specific inputs and outputs to feed the list in the control object.

Another common approach is to provide a selection of options for the user to choose from. The control itself is bound to a variable, similar to any input. The selection data can come from a list that is retrieved from a database.

The easiest way to accomplish a select control in a coach is to store the dynamic data in a list of String or a list of name-value pair. In the previous step, the coach retrieves the data with an SQL integration service that was created in Playback and saves the data in a list object.

Drag the integration service onto the palette and hook up the flows. Map the output of the integration service to a local variable. The variable feeds the control on the coach.



Implementing a select control (2 of 2)

The screenshot shows the IBM BPM Designer interface. On the left is a toolbar with various icons. The main workspace displays a form titled 'Hiring Form'. Inside the form, there are several input fields: 'Department' (highlighted with a red box), 'Division', and a 'Submit' button. Below the workspace is a palette with sections for 'General', 'Positioning', 'Configuration', 'Events', and 'Visibility'. Under 'Configuration', the 'Common' section shows 'Label: Department' and 'Help:'. The 'Behavior' section shows 'Binding:' (set to 'requis'), 'View:' (set to 'Single Select (ANY)'), and 'Label visibility: Show'. To the right of the workspace is a 'Select Library Item' palette. The 'Type' dropdown is set to 'UI Toolkit'. The palette lists various toolkits, with 'Single Select UI Toolkit' highlighted with a blue box.

Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-8. Implementing a select control (2 of 2)

The next step is to identify the input control that was created in the Playback that must be changed into a select control. At times, by using a toolkit that comes from other developers, the implementation is quicker. The default select control is easily configured to show the list of options to a user. As a developer, you have plenty of options to choose from, including building your own View select control. Views are explained in more detail later in this unit.

To accomplish this change, in the **View:** option, the developer clicks **Select** to choose the **Single Select** control from the system toolkit to replace the default **Text** view. The image on the palette changes to reflect the new look of the select box.



Configuring the select list

The screenshot shows the 'Department Details' tab selected in a top navigation bar. Below it, a 'Department' control is displayed. On the left, a sidebar lists tabs: General, Positioning, Configuration (which is selected), Events, Visibility, and HTML Attributes. The main panel contains sections for 'Behavior' and 'Items'. Under 'Items', the 'Item lookup mode' is set to 'Items From Config Option'. The 'List items service' is 'Default List Items Service'. The 'Service input data' field contains 'departmentCategoriesList (NameValuePair)' with a red box around it. Buttons for 'Select...', 'New...', 'Clear', and two more 'Select...' and 'Clear' buttons are visible. A vertical scrollbar is on the right.

Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-9. Configuring the select list

The final step is to set what values are displayed to the user in the control, and what values are stored in the bound variable when the user makes the selection. The binding on the **Properties > Configuration** tab sets the list that users can select from. When users select their choice, the selection is a member of the type of variable that is configured as the binding variable. The variable is stored in the variable that is bound to the control on the **Properties > General** tab.

If the binding is a list of name-value pair, the selected item must be a simple name-value pair variable. If the binding is a list of String, the selected item must be a simple String variable.

7.2. Views

Views

Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-10. Views

Views are reusable sets of user interface (widgets or controls) that users can use to interact with a business object or service. Views consist of one or more other Views, data bindings, layout instructions, and behaviors. Views are stock (included, not customized) or custom controls or dashboards. Custom Views are controls or dashboards that a developer creates or that other programs or companies provide. In terms of use, IBM Business Automation Workflow treats stock and custom Views identically.

This topic covers how to create custom Views.

Views

- Views are reusable assets that can be shared with multiple coaches or even multiple process applications
- Views and coaches can share parts of their user interface with other Views and coaches
- In general, create highly reusable Views in toolkits and more specialized Views in process applications
 - If the View is in a toolkit and then someone edits it, the changes apply to all instances of the View in all applications that use that toolkit
 - You cannot directly edit the definition of the View from within the parent coach or View
 - Instead, you must first open the View definition before you can change it

[Playback 1: Enhancing coaches](#)

© Copyright IBM Corporation 2020

Figure 7-11. Views

Beginning with version 19, Coach Views are known as Views.

Views are reusable assets (or coach controls) that can be shared with multiple coaches or even multiple process applications. For example, after developers apply the CSS classes and achieve the style that they want for a coach, they can create a custom View. By moving the class into a View, the class can now be a shared asset.

Because Views are reusable, Views and coaches can share parts of their user interface with other Views and coaches. For example, suppose that you create a coach with a View that contains a set of address fields. If you create a second coach that needs address fields, you can reuse the View from the first coach. In both cases, the coach is using an instance of the View. You can edit the properties of each instance independently. For example, changing the label of one View instance does not change the label of the other. Both instances of the View use a reference to point to the View definition. This approach means that if the View definition changes, you can see that the change is reflected in the instances of the View.

You can create a View in the process application or in a toolkit. In general, you create highly reusable Views in toolkits and more specialized Views in process applications. Choosing the process application means that you can reuse it only within the process application. However, it also means that if someone edits the View, the changes apply to the instances of the View in the process application. If the View is in a toolkit, when someone edits it, the changes apply to all

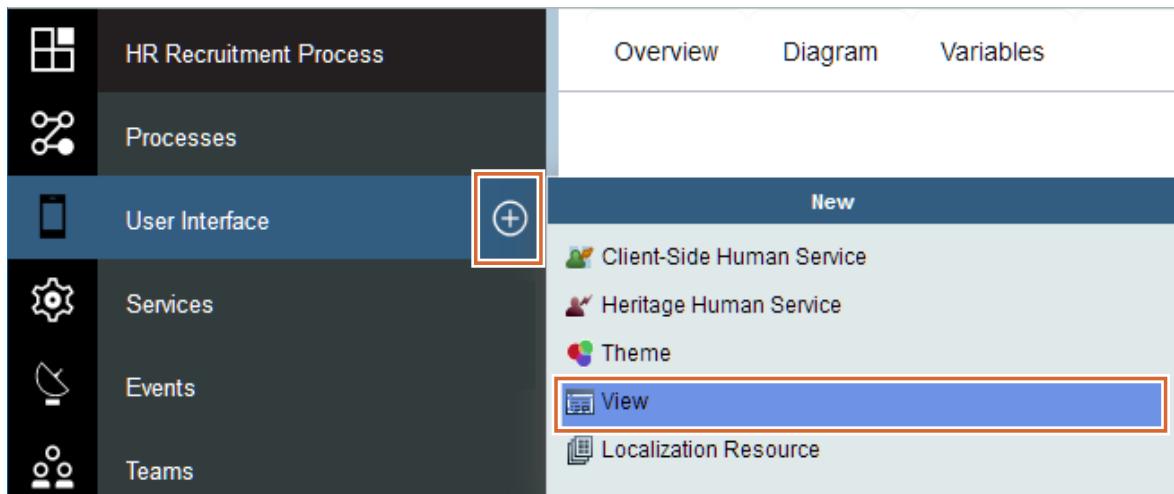
instances of the View in all applications that use that toolkit. Because editing a coach definition can affect many instances, be careful in your changes. For example, deleting a content box in the View definition means that coaches or Views that contain instances of that View cannot display the content.

You cannot directly edit the definition of the View from within the parent coach or View. Instead, you must open the View definition first before you can change it.



Create a View

- Create a View through the User Interface category in the library.
- To create a custom View, the developer clicks the **(+)** plus icon next to the User Interface category in the library in Process Designer. After naming the View, the developer is directed to the View definition page.



Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-12. Create a View

To create a custom View, the developer clicks the **(+)** plus icon next to the User Interface category in the library in IBM Process Designer. After naming the View, the developer is directed to the View definition page.

IBM Training 

Create a View

New View

Views are reusable user interface components, which can be simple widgets or represent complex business objects.



Name:

Start with blank view

Start with a grid:

Select template:

Intended for use on multiple devices

FINISH **CANCEL**

Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-13. Create a View

Templates are an ideal way to create a standardized look across multiple Views.

A template is a View that someone marks as being usable as a template in its Overview page. Users can then select the template when they are creating Views. The new Views have the content of the template as base content to which the users can then add content. For example, you create a View that has the company logo and name in a banner area and a content box as a placeholder for other content. When you use this View as a template, you can then select it when you are creating another View. In the new View, the banner area is defined in the template along with an area for content. Other users can also use the template when they are creating Views, and the template provides a consistent look across the new Views. Because templates are Views, you can also drop them onto coaches. For example, if you have a template that has a common banner, you can drop it onto a coach so that the coach has the common banner.

The screenshot shows the 'View definition page' interface in IBM Business Automation Workflow. At the top, there's a blue header bar with the 'IBM Training' logo on the left and the 'IBM' logo on the right. Below the header, the title 'View definition page' is displayed in a large, bold, dark blue font. The main content area is a form with several tabs at the top: 'Overview' (which is selected and highlighted with an orange border), 'Behavior', 'Variables', and 'Layout'. On the left side, there's a vertical toolbar with various icons for file operations like create, delete, and save. The 'Overview' tab contains sections for 'Common' (Name: View, Modified: author1 (Nov 17, 2020, 4:10:48 PM), Tags: <none>, Documentation: with rich text editor), 'Usage' (checkboxes for Can Fire Boundary Event, Use as a Template, Supports a Label, Prototype-level event handlers, Intended for use on multiple devices), and 'Preview' (options for Palette Icon, Layout Image, Use URL binding, Preview Label Position: Top). Below these tabs, there's a section labeled 'Advanced Preview' with a small arrow icon. At the bottom of the page, there are footer links for 'Playback 1: Enhancing coaches' and 'Copyright IBM Corporation 2020', along with a user count indicator.

Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-14. View definition page

When developers create a View or open an existing View to edit it, they see the following sections available:

- **Overview**

This page displays the View name, information about the View, the images that are used to represent the View during design time, and how the View is used. You can also tag your View to make it easier to find in the library and on the palette.

- **Behavior**

This page displays the scripts and CSS files that are contained in the View. The Behavior page is also where you define event handler code. The event handlers are the entry points for the code of the View. While the View might reference supporting JavaScript files, the event handlers contain the functions that the IBM Business Automation Workflow framework calls.

- **Variables**

This page displays the business data binding, configuration options (which includes Ajax services), and localization resources that are available to the View or what the View uses.

- **Layout**

This page displays the Views and controls that are contained within the View and their relative positions. The layout page also displays the palette, which contains items that you can add to the View. These items consist of Views, which are categorized with tags, advanced items, and variables.



Defining View behavior

- In the Behavior section of the View definition, you include reusable scripts or add inline JavaScript or CSS code.

A screenshot of a software interface titled "View". The "Behavior" tab is selected. On the left, there's a sidebar with various icons. The main area shows a list of behavior components under "Behavior" and "Event Handlers". Under "Behavior", there are "Dynamic Style", "Included Scripts", and "CSS Inline CSS" (which is highlighted). Under "Event Handlers", there are options like "load", "unload", "view", "change", "collaboration (deprecated)", and "validate". To the right, there's a panel titled "Code: Inline CSS" containing the number "1". At the bottom, there are navigation icons and a status bar showing "Playback 1: Enhancing coaches" and "© Copyright IBM Corporation 2020".

Figure 7-15. Defining View behavior

In the Behavior section of the View definition, you include reusable scripts or add inline JavaScript or CSS code to the View. You can also define events.

To define the behavior for the View in the Behavior page, you add existing script files from the library through the Included Scripts section. You add CSS code and JavaScript to the View definition through the Inline CSS and Inline JavaScript sections. In general, you include scripts for reusable code and the inline options for JavaScript and CSS code that is not reusable.

UI Toolkit

- A set of controls for designing applications that can be run on multiple device types, such as mobile and desktop devices
- All the Views in the UI Toolkit are suitable for use on both desktop and mobile devices
- Each control in the UI Toolkit consists of a View that points to a specific set of files:
 - A JavaScript file that provides the behavior of the control
 - Image files to represent the control on the palette and on the canvas
 - A preview JavaScript file that shows a preliminary view of the View when it is dropped on the coach editor palette

Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-16. UI Toolkit

7.3. Theme support

Theme support

Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-17. Theme support



Theme support

- A theme consists of a set of theme definitions, dynamic stylesheets, and generated CSS that you can apply to a process application or toolkit
- A new process application, by default, uses the **Carbon Theme**
- The Classic Theme and the Carbon Theme contains all of the definitions that the IBM Business Automation Workflow responsive controls use
- Theme support is based on an open source CSS precompiler called **LESS**
- An Admin command can be used to update the appearance of a deployed process app without changing and redeploying the process app
- Three themes come with the product:
 - Carbon (System Data)
 - Classic (System Data)
 - SPARK UI Theme (UI Toolkit)



Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

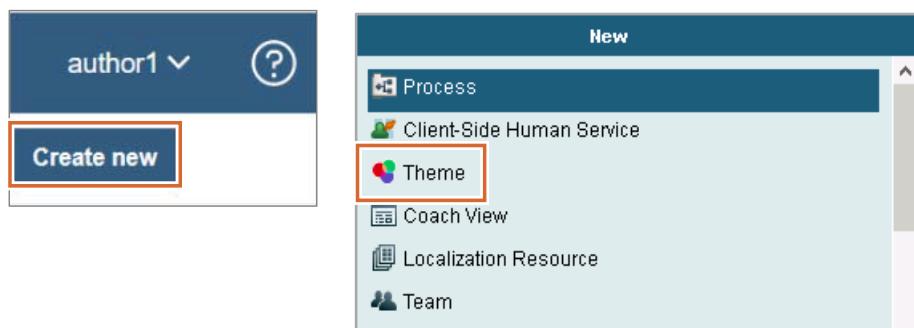
Figure 7-18. Theme support



Create custom theme

To create a theme:

1. In the web-based Process Designer, click **Create New** and select **Theme**.
2. In the New Theme window, type the name of the new theme.
3. Select whether you want to copy an existing theme from the current project or dependent toolkit or import a theme.
4. In the theme editor, assign values to theme variables. The value can be a specific value, a formula, another variable, or a combination of these types.



Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-19. Create custom theme

To create a theme:

1. In the web-based Process Designer, click **Create New** and select **Theme**, or in the library, click the (+) plus sign in the User Interface category and then select **Theme**.
2. In the New Theme window, type the name of the new theme.
3. Select whether you want to copy an existing theme from the current project or dependent toolkit or import a theme. After you click **Finish**, the editor opens the new theme.
4. In the theme editor, assign values to theme variables. The value can be a specific value, a formula, another variable, or a combination of these types. For example, `@bpm-neutral: #586464` defines the value to a specific color while `@bpm-link-color: @bpm-color-primary;` defines the value with the value of another variable. In the Design page, change the value for one or more variables.



Create custom theme

- Use the Classic Theme as your starting point to create a custom theme and to use any of the system controls in your process application
- Extend your theme by adding custom variables and modifying the system variables

New Theme

A theme sets the visual style of the coaches within a process application or toolkit. A theme can also apply to a portal in Process Portal.



Name:

Copy an existing theme
Classic (System Data)
 Import a theme file
Browse... No file selected.

FINISH **CANCEL**

Playback 1: Enhancing coaches

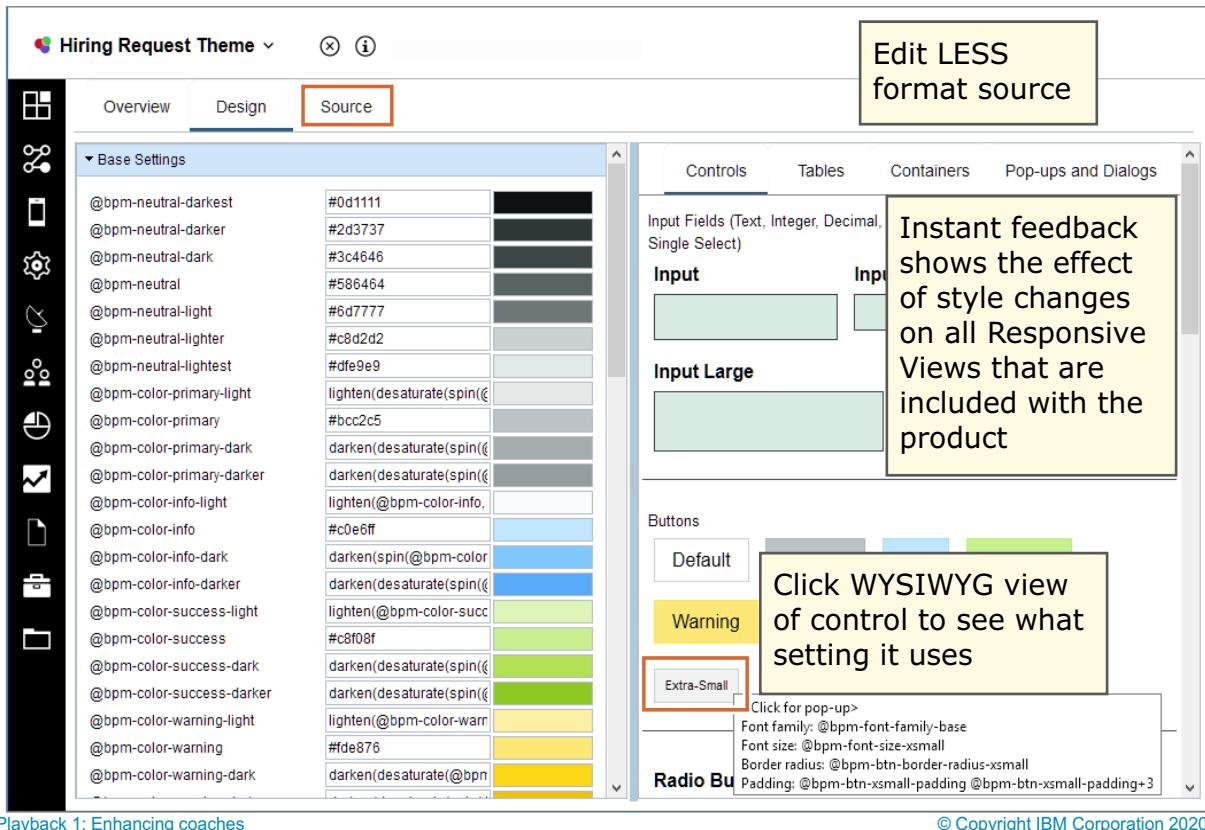
© Copyright IBM Corporation 2020

Figure 7-20. Create custom theme

IBM Training



WYSIWYG Theme Editor: Create and edit theme (1 of 3)



Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-21. WYSIWYG Theme Editor: Create and edit theme (1 of 3)

Many variables have a swatch that you can click and then choose a value from a picker. The example controls update to display the new values. If you hover over a control, you can see the specific variables that affect that control and its current values.



WYSIWYG Theme Editor: Create and edit theme (2 of 3)

Hiring Request Theme ▾ × ⓘ

Overview Design Source

Base Settings

@bpm-neutral-darkest	#0d1111
@bpm-neutral-darker	#2d3737
@bpm-neutral-dark	#3c4646
@bpm-neutral	#586464
@bpm-neutral-light	#6d7777
@bpm-neutral-lighter	#8d82d2
@bpm-neutral-lightest	#dfc9e9
@bpm-color-primary-light	lighten(desaturate(spin(@bpm-color-primary, 0, 100), 10), 10)
@bpm-color-primary	#bcc2c5
@bpm-color-primary-dark	darker(desaturate(spin(@bpm-color-primary, 0, 100), 10), 10)
@bpm-color-primary-darker	darker(desaturate(spin(@bpm-color-primary, 0, 100), 10), 10)
@bpm-color-info-light	lighten(@bpm-color-info, 10)
@bpm-color-info	#c0e6ff
@bpm-color-info-dark	darker(spin(@bpm-color-info, 0, 100), 10)
@bpm-color-info-darker	darker(desaturate(spin(@bpm-color-info, 0, 100), 10), 10)
@bpm-color-success-light	lighten(@bpm-color-success, 10)
@bpm-color-success	#c8f08f
@bpm-color-success-dark	darker(desaturate(spin(@bpm-color-success, 0, 100), 10), 10)
@bpm-color-success-darker	darker(desaturate(spin(@bpm-color-success, 0, 100), 10), 10)
@bpm-color-warning-light	lighten(@bpm-color-warning, 10)
@bpm-color-warning	#fde876
@bpm-color-warning-dark	darker(desaturate(@bpm-color-warning, 10), 10)

Many variables also have a swatch that you can click and then choose a value directly from a picker

Input Fields (Text, Integer, Decimal, Text Area, Type Ahead Text, Multi Select, Single Select)

Input

Color

Basic colors:

Red	Yellow	Green	Cyan	Blue	Magenta	Pink	Light Gray
Dark Red	Dark Yellow	Dark Green	Dark Cyan	Dark Blue	Dark Magenta	Dark Pink	Gray
Maroon	Orange	Light Green	Light Cyan	Light Blue	Light Magenta	Light Pink	White
Dark Maroon	Dark Orange	Dark Light Green	Dark Light Cyan	Dark Light Blue	Dark Light Magenta	Dark Light Pink	Black
Dark Green	Dark Yellow-Green	Dark Cyan-Green	Dark Cyan-Blue	Dark Blue-Black	Dark Magenta-Black	Dark Pink-Black	
Dark Gray	Dark Gray-Yellow	Dark Gray-Green	Dark Gray-Cyan	Dark Gray-Blue	Dark Gray-Magenta	Dark Gray-Pink	

Custom colors:

Black	Dark Gray	Gray	Light Gray	White
Dark Green	Dark Yellow-Green	Dark Cyan-Green	Dark Cyan-Blue	Dark Blue-Black
Dark Maroon	Dark Orange	Dark Light Green	Dark Light Cyan	Dark Light Blue
Dark Red	Dark Yellow	Dark Green	Dark Cyan	Dark Blue
Dark Purple	Dark Magenta	Dark Pink		

Define Custom Colors >>

OK Cancel Add to Custom Colors

Use color palette to select the preferred color

Hue: 133 Red: 188
Sat: 17 Green: 194
Lum: 181 Blue: 197

© Copyright IBM Corporation 2020

Playback 1: Enhancing coaches

Figure 7-22. WYSIWYG Theme Editor: Create and edit theme (2 of 3)

IBM Training

WYSIWYG Theme Editor: Create and edit theme (3 of 3)

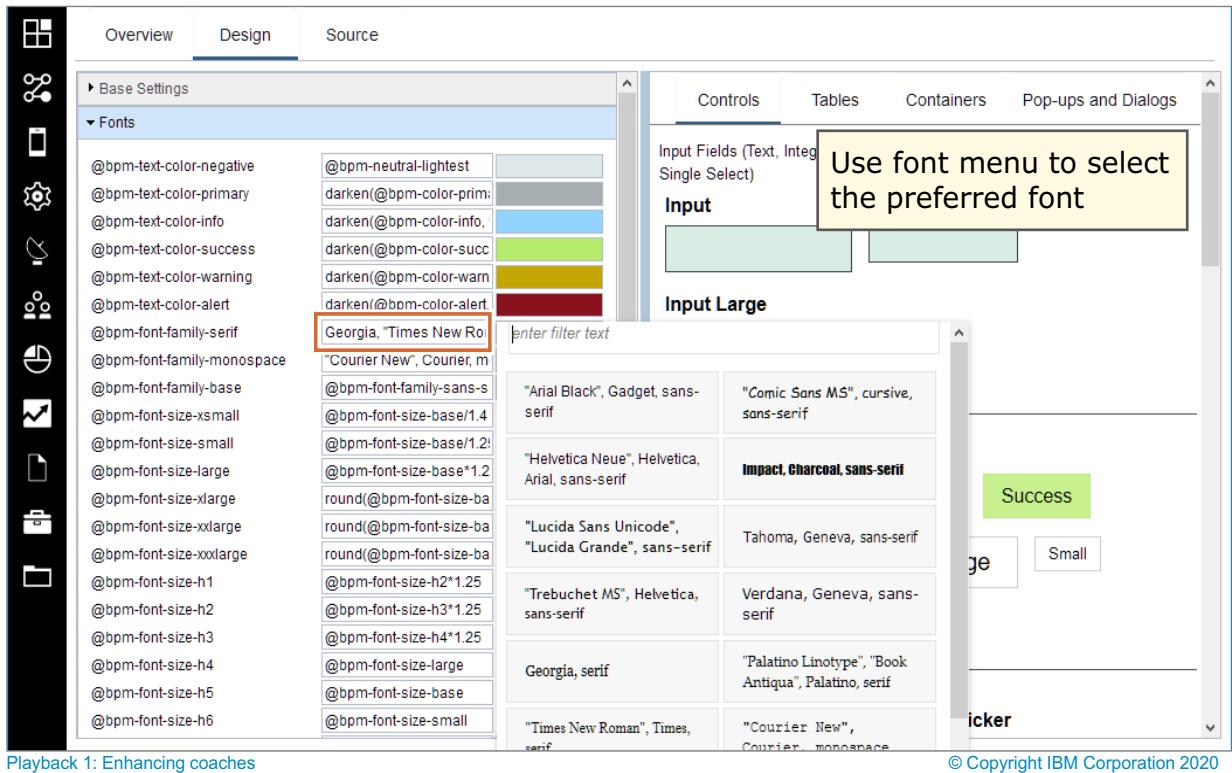


Figure 7-23. WYSIWYG Theme Editor: Create and edit theme (3 of 3)



Applying custom theme to process apps

- Theme is set in the Process App Settings of a Process App or Toolkit
- The theme setting affects all Views, including the Views that come from included toolkits activities

The screenshot shows the 'Process App Settings' interface for an 'HR Recruitment Process'. In the center, a 'Select Library Item' dialog box is open, specifically the 'Theme' section. The 'Type' dropdown is set to 'Theme'. Three options are listed: 'Carbon System Data' (selected and highlighted with a red box), 'Classic System Data', and 'SPARK UI Theme UI Toolkit'. The background shows various settings sections like 'Common', 'Target Environment', 'Authorization Settings', 'Monitor Settings', and 'XML Settings'.

Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-24. Applying custom theme to process apps

The Carbon theme is the default. You have a chance to build a custom theme based on the Classic theme at the end of this unit.

7.4. Conducting the Playback session

Conducting the Playback session

Playback 1: Enhancing coaches

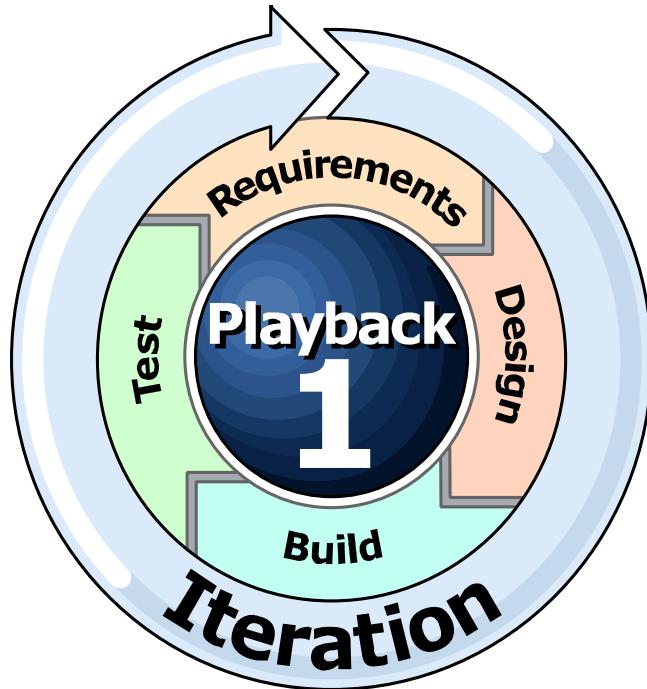
© Copyright IBM Corporation 2020

Figure 7-25. Conducting the Playback session

This Playback is your chance to demonstrate everything that was created thus far: process flow variables that drive decision gateways, tasks that are created and assigned to the right teams, and correct task routing. This Playback also demonstrates that the process is following the correct path.

Playback 1: Business data and services

- The goal is to demonstrate data flows from one coach to another, and from one task to another inside the process
- All data is bound on the coaches, and if the same data is shared across multiple tasks, Playback participants can see the data move through the different activities



Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-26. Playback 1: Business data and services

A Playback is a focused demonstration of a partially implemented process application, which is delivered to the business and IT communities for discussion, consensus-building and approval.

As you recall, the goal of Playback 1 is to demonstrate data flows from one coach to another, and from one task to another inside of the process. All data is bound on the coaches, and if the same data is shared across multiple tasks, Playback participants can track the data when it moves through the different activities.

Playback 1: Focus on user interface design and implementation

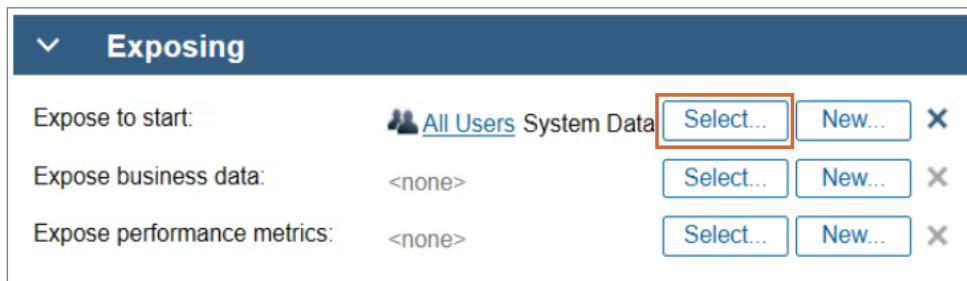
- Data model and process flow implementation
- Human service or coach design, and data mapping
- Business rules, process flow control implementation

At times, the project team might make specific requests for demonstrations. For instance, if the project team is tasked with demonstrating the tracking of work while the process is running, you would highlight the priority and status of the work items as they are active in the process. You demonstrate this in Exercise 7.



Expose a process to a team

- Create an instance of the process before you demonstrate it
- Create an instance of a process application to a team (All Users) by using **Expose to start**
 - Click the Overview Tab on the Process App
 - For **Expose to start**, choose a team



Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-27. Expose a process to a team

Before developers demonstrate the process, they create an instance of the process. This task is accomplished when they expose the process application to a team. A developer verifies the exposure settings on the Process App Settings Overview page. This menu has other functions that pertain to process applications and the processes and services that are contained within.

- Go to the **Overview** tab of the process, and next to the **Expose to start** option, click **Select** to select a team that has access to start this process.
- The quickest way to demonstrate the process is to expose the process to the **All Users** team so any participant with a valid account can start the process.



Creating a snapshot

- A snapshot captures the state of the library items within a process application or toolkit at a specific point in time.
- To create a snapshot in the Workflow Center:
 1. Click **Details** on the process app tile
 2. On the Snapshots tab, open the list of Options by clicking the three dots on the right side of the Current tip.
 3. Select **Create a snapshot**

Back to process apps /

HR Recruitment Process

Name	Status	Created	Last updated	Target environment
→ Current	author1	9/22/2020, 12:51:50 PM	author1 9/22/2020, 12:51:50 PM	Traditional

Snapshots (0) ⓘ

1 Details →

2 ⋮

3 Create a snapshot

Open in Designer

Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-28. Creating a snapshot

Recall that a snapshot captures the state of the library items within a process application or toolkit at a specific point in time. Snapshots usually represent a milestone or are used for Playbacks or for installation. As you recall, you took a snapshot at the end of Playback 0 in Process Designer and in the first exercise.



Guidelines for using snapshots

- Snapshots can take up large amounts of space in a database
 - Agree on intervals in your organization and take snapshots at agreed-upon milestones
- Define a meaningful naming convention for snapshots and use that naming convention for all projects
- Work closely with IBM Business Automation Workflow administrators to come up with a snapshot deployment and activation plan

Name	Status	Created	Last updated	Target environment
→ Current		author1 9/22/2020, 12:51:50 PM	author1 9/22/2020, 12:51:50 PM	Traditional
→ Exercise 1 solution (E1S)	New	author1 9/22/2020, 3:35:42 PM	author1 9/22/2020, 3:35:42 PM	Traditional

Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-29. Guidelines for using snapshots

The following list contains some guidelines for using snapshots for a process application:

- Snapshots can take up large amounts of space in a database, so create snapshots that are targeted to important phases in development. Agree on intervals in your organization and take snapshots at agreed-upon milestones.
- Define a meaningful naming convention for snapshots and use that naming convention for all projects.
- Work closely with IBM Business Automation Workflow administrators to come up with a snapshot deployment and activation plan. Administrators and developers with administration privileges are the team members who can deploy snapshots to other environments.

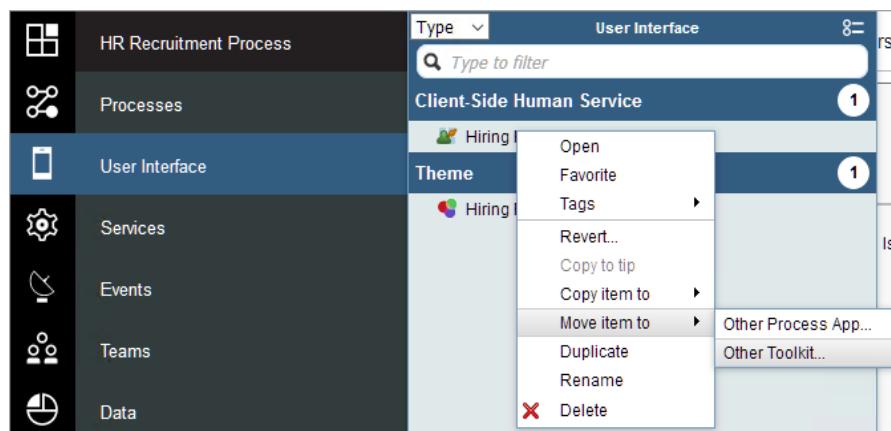
Once you create a snapshot, you can export your process application. You cannot export the tip so each time you want to export your process application, you need to create a snapshot.

In the screen capture, there is one snapshot (Exercise 1 solution) and the Tip.



Toolkits

- A toolkit is a container where artifacts are stored for reuse by process applications or other toolkits.
- You can create toolkits to enable Process Designer users to share library items across process applications.
- Process applications can share library items from one or more toolkits, and toolkits can share library items from other toolkits.
- Example user-defined toolkit:
 - Hiring Requisition Toolkit (HRT)
- Example system-defined toolkits:
 - UI Toolkit
 - System Data



Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-30. Toolkits

As development continues, it makes sense to share assets across developers. This is where Toolkits fit in.

- A toolkit is a container where artifacts can be stored for reuse by process applications or other toolkits.
- You can create toolkits to enable Process Designer users to share library items across process applications.
- Process applications can share library items from one or more toolkits, and toolkits can share library items from other toolkits.
- Users who have access to a toolkit can create a dependency on the toolkit and use the library items within it for their process development efforts.

When a user creates a toolkit and moves items into it, all the dependent assets are also moved into the toolkit

Unit summary

- Describe how to create tabs on a coach
- Enhance coaches and apply a theme
- Explain how to create a reusable View
- Describe how to create a snapshot
- Export your process application
- Share your assets by using a toolkit
- Conduct a Playback session

Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-31. Unit summary

Review questions

1. Coaches consist of what three things?
2. A new process application, by default, uses the _____ Theme.
3. Theme support is based on open source CSS precompiler that is called _____.
4. True or False: A deployed process app can get an updated appearance without changing and redeploying the process app.

Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-32. Review questions

Write your answers here:

- 1.
- 2.
- 3.
- 4.

Review answers

1. HTML, JavaScript, and CSS
2. Classic
3. LESS
4. True

Playback 1: Enhancing coaches

© Copyright IBM Corporation 2020

Figure 7-33. Review answers

Exercise: Playback 1: User interface design and implementation

Playback 1: Enhancing coaches

© Copyright IBM Corporation 2019

Figure 7-34. Exercise: Playback 1: User interface design and implementation

Exercise introduction

- Create tabs on a coach
- Change the appearance of a coach by applying a custom theme
- Change the coach layout for a mobile format
- Configure controls to respond to different screen sizes
- Debug the coach by using a responsive sensor



Playback 1: Enhancing coaches

© Copyright IBM Corporation 2019

Figure 7-35. Exercise introduction

Exercise: Playback 1: Conducting the Playback session

Playback 1: Enhancing coaches

© Copyright IBM Corporation 2019

Figure 7-36. Exercise: Playback 1: Conducting the Playback session

Exercise introduction

- Log on to the Process Portal and create an instance of a process
- Demonstrate that the process follows the various paths modeled
- Use Process Portal to view the state of activities in a process
- Create a toolkit
- Create a snapshot in the Workflow Center
- Export the process application



Playback 1: Enhancing coaches

© Copyright IBM Corporation 2019

Figure 7-37. Exercise introduction

Unit 8. Playback 2: Integrations

Estimated time

01:30

Overview

This unit covers how to create a decision service, implement message events, apply asset tagging, and access and manipulate external data. You learn about exposed process variables (EPVs) and environment variables (ENVs).

How you will check your progress

- Review
- Exercise

Unit objectives

- Explain how to create a decision service
- Define a message start event
- Explain how an enabling service is used with the message start event
- Describe how to create and configure an undercover agent (UCA)
- Describe how to start a process with a message start event
- Define the basic function of an integration service flow
- Identify the components of the IBM Business Automation Workflow integration architecture
- Describe how integration components interact with services
- Configure and define integration service flows for outbound integration
- Describe the differences between an environment variable and an exposed process variable
- Organize assets with favorites, tagging, and smart folders

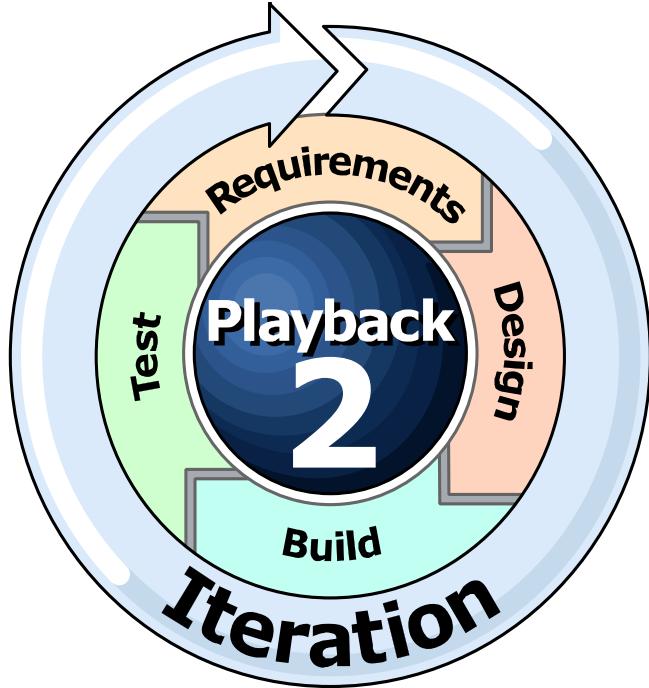
Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-1. Unit objectives

Playback 2: Integrations

- The goal is to demonstrate “real data, real time”
- Show the business rule logic and show tasks that are being created as a result of the decision service
- Demonstrate message events in the process that affect process flow
- Implement the integrations that are created in Playback 1: Building coaches to enable coach functions



Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-2. Playback 2: Integrations

The goal is to demonstrate “real data, real time.” Log on to the process portal and demonstrate any integrations that your process depends on. Show the business rule logic and show tasks that are being created as a result of the decision service. Demonstrate message events in the process that affect process flow. Some of the integrations that are created in previous Playbacks enable coach functions. Those integrations are demonstrated in the next Playback.

When this development phase is complete, some external services for this Playback might be incomplete. Continue to use mock data services to enable your processes and services, but focus the Playback presentation on those integration endpoint services that are still not available. All artifacts to enable the integration are already created in the library, and a quick switch of flow is all that is necessary when the new integration is enabled.

Key concepts in this unit

- **Decision service:** Specify a decision or condition in a business rule to determine which process implementation is started
- **Message event:** A listener that the incoming message triggers, and it runs a task or set of tasks
- **Undercover agent:** An interface to accept incoming messages, and it triggers and runs a task or set of tasks
- **Exposed process variable (EPV):** Allows business users to modify the value of a variable

Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-3. Key concepts in this unit

Topics

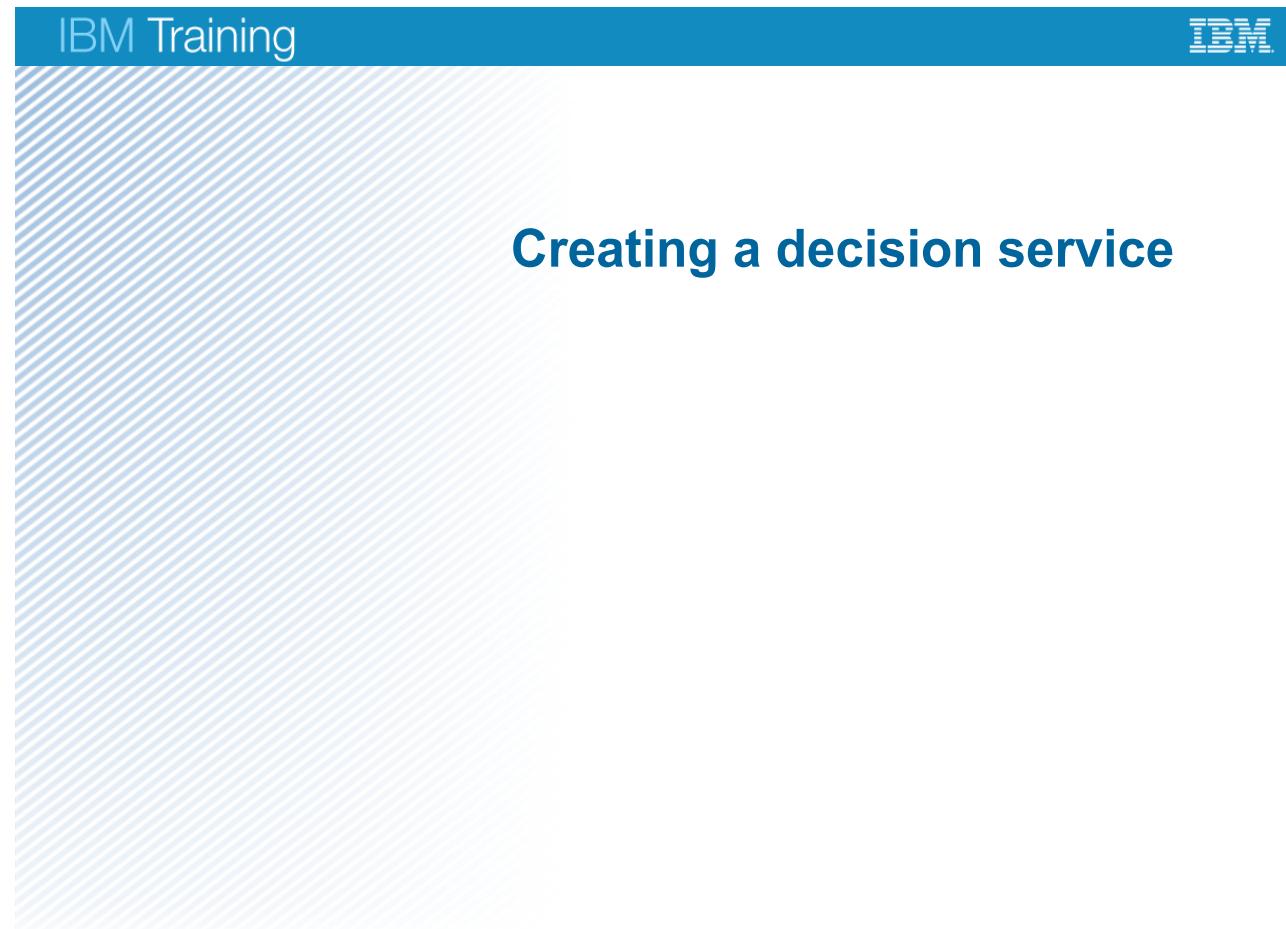
- Creating a decision service
- Implementing message events
- Accessing and manipulating external data
- Exposed process variables (EPVs) and environment variables (ENVs)
- Applying asset tagging

Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-4. Topics

8.1. Creating a decision service



Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-5. Creating a decision service

Developers build a decision service when they want a decision or condition in a business rule to determine which process implementation is started. For example, when a certain condition evaluates to true, IBM Process Designer implements the associated activity or action.

IBM Process Designer supports business analysts and business users who create business rule authoring tasks, who are rule designers rather than programmers. Business rule designers can express business logic with rule syntax that resembles natural human language. This rule syntax is called **Business Action Language (BAL)**, which is a declarative language that relates business concepts to business data and actions.

Business rules are an expression of business policy in a form that business users can understand and a rules engine can interpret. Business rules formalize a business policy into a series of If-Then statements.

Building a decision service (1 of 2)

- Business rules are included in a process by adding a decision service to the process
- Add a decision service to a process application when the actions that should take place in your process depend upon one or more conditions
- Build your rule hierarchy so that rule conditions are ordered from most complex to least complex
- Create a final condition that is a catch-all rule
 - This rule is necessary if you cannot verify that the variable you want to modify in the rule is going to be set before running the process that triggers the decision service
- Consider encapsulating your rules in a single-function decision service
 - Encapsulating your rules makes the service available to any other part of the process application that needs the same rule logic

[Playback 2: Integrations](#)

© Copyright IBM Corporation 2020

Figure 8-6. Building a decision service (1 of 2)

You can create rules by using the Business Action Language (BAL) rule editor. The rule editor uses natural language technology to express business decisions in a form that is readable by humans but can also be run by a rule service runtime such as the IBM Operational Decision Manager Rule Execution Server.

Business rules are included in a process by adding a decision service to the process. Add a decision service to a process application when the actions that take place in the process depend upon one or more conditions. For example, if an employee holds the position of director and submits a meal expense for more than \$250, then a rule is created, and a variable is set in the rule. A rule such as `approvalRequired` routes the process sequence flow into a specific approval activity.

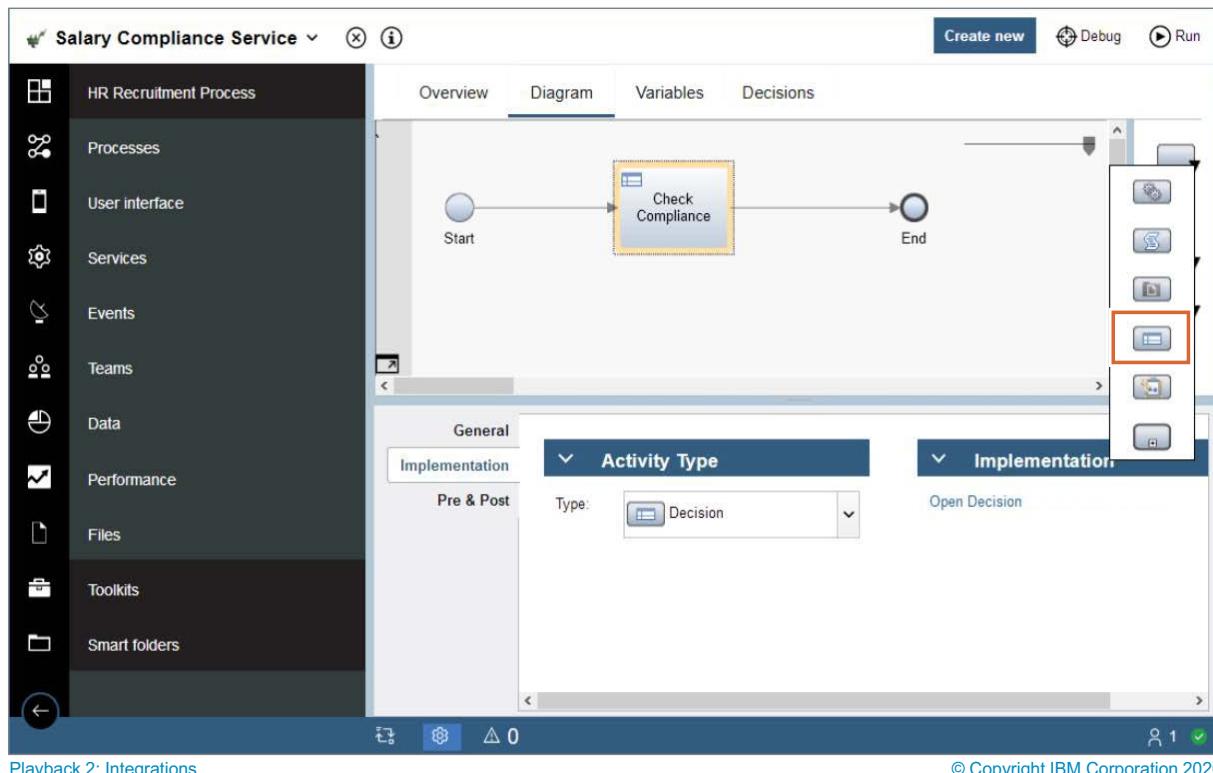
When building a decision service, follow these guidelines:

- Build the rule hierarchy so that the rule conditions are ordered from most complex to least complex.
- Create a final condition that is a catch-all rule. This rule is necessary if the team cannot verify that the variable that they want to modify in the rule is set before running the process that triggers the decision service.

- Consider encapsulating the rules in a single-function decision service that allows the service to be available to any other part of the process application that needs the same rule logic.



Building a decision service (2 of 2)



Playback 2: Integrations

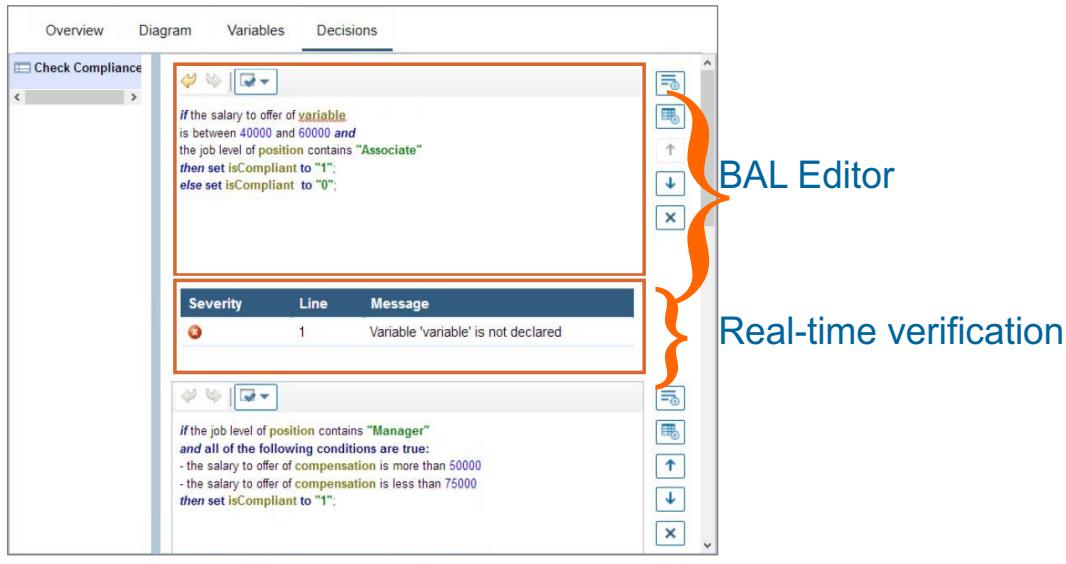
© Copyright IBM Corporation 2020

Figure 8-7. Building a decision service (2 of 2)

In the library pane, expand the task menu and drag a Decision from the library to create a decision service. A decision service contains one or more components. You can create a Business Action Language (BAL) Rule that uses this type of decision service. Use the rule editor in this component to author business rules with BAL, a natural language technology.

Adding a BAL rule

- The Business Action Language (BAL) rule component:
 - Provides a rule editor that rule designers can use to author business rules with natural language technology
 - Requires no programming expertise to create business rules
 - Is easier for people to read and understand



Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-8. Adding a BAL rule

The Business Action Language (BAL) rule component provides a rule editor that allows rule designers to author business rules with natural language technology. Using natural language instead of JavaScript to author rules means that no programming expertise is required to create business rules, and the rules are easier for people to read and understand.

Create the service with the necessary input, private, and output variables. The Decisions tab at the top is where the author creates the rule with the service variables.

Add rules with the (+) plus sign at the top of the screen and remove rules with the X next to the condition to delete. Move the rules up and down with the arrows to the right of the corresponding rule. The BAL editor verifies the rules in real time.

Building a BAL rule

The parts must be defined in the following order:

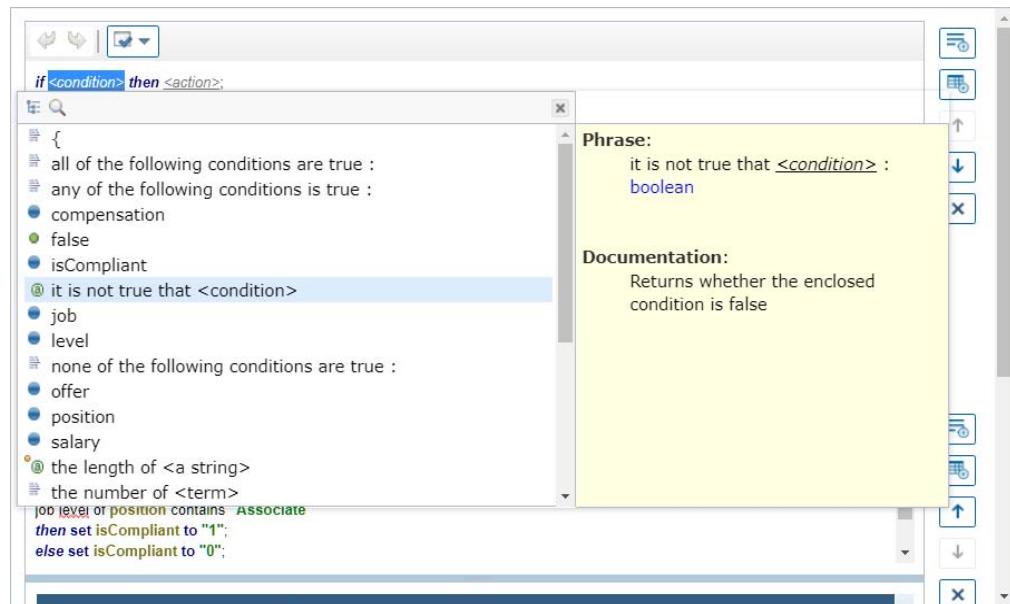
1. Definitions (optional)

2. If

3. Then

4. Else

(optional)



Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-9. Building a BAL rule

The developer uses the BAL rule editor to build rules, add rule parts, statements, and fragments, and replace placeholders with variables and values. Use the completion menu in the editor to insert or edit constants, values, parts, or fragments of rule statements. While creating or editing rules, the editor highlights errors to help you identify and resolve problems in the rules.

A business rule is composed of some or all of the following parts. The parts must be defined in the following order:

1. Definitions part (optional)
2. If part
3. Then part
4. Else part (optional)

Use the **Content Assist** box whenever possible. It provides a developer multiple options to choose from to build the rules. If developers get stuck, they press Ctrl+Space to open the menu and read through the suggestions and the descriptions of the options available.

8.2. Implementing message events

Implementing message events

Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-10. Implementing message events

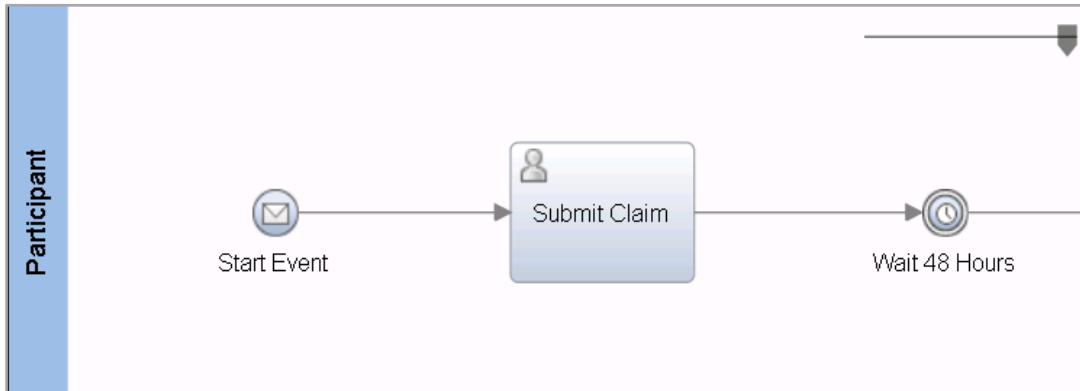
Events can occur at all points of a business process, and they affect the business process flow. Thus, a major part of implementing business processes is knowing how to handle and react to a dynamic event like a message event. These dynamic message events are internal or external to the process. IBM Business Automation Workflow provides two components, message events and undercover agents (UCAs) to model and complete these dynamic interactions at run time. The result is that a process application has an interface to accept incoming messages (undercover agents) and a listener (message event) that the incoming message triggers to run a task or set of tasks.

In Business Process Model and Notation (BPMN), a message generally signifies any signal from outside the process. Like all intermediate events, the message intermediate event can be used in a sequence flow or attached to an activity.

UCAs and messaging are topics that take time to understand. This topic is intended to introduce you to messaging. To obtain a deeper understanding of messaging, consider taking the *Implementing with IBM Business Automation Workflow Level II* course.

Message event introduction

- Would a timer event, similar to the Wait 48 Hours event that is shown here, create instances of a process?
- How do you create instances of a process on a regular timed interval?



[Playback 2: Integrations](#)

© Copyright IBM Corporation 2020

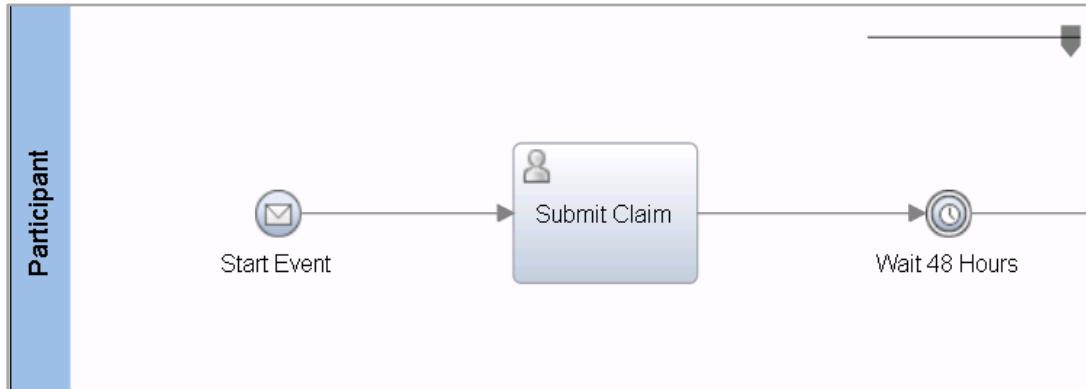
Figure 8-11. Message event introduction

The project team already implemented a timer event, but what happens when a certain process requires creation of instances on a regular time interval? Because a timer event cannot create instances (it is an intermediate event, not a start event), they must look to another solution. That need can be met with a start message event and a time-elapsed undercover agent (UCA).

In this example, a participant (a manager) is required to submit a monthly claim for the group expenses. The system generates a generic claim shell for the manager to complete, and the manager must complete the claim before it is submitted. Developers cannot use the intermediate timer event to create instances, so they must turn to the time-elapsed undercover agent.

Message start events

- When a message is received, a new instance of the process is started



Playback 2: Integrations

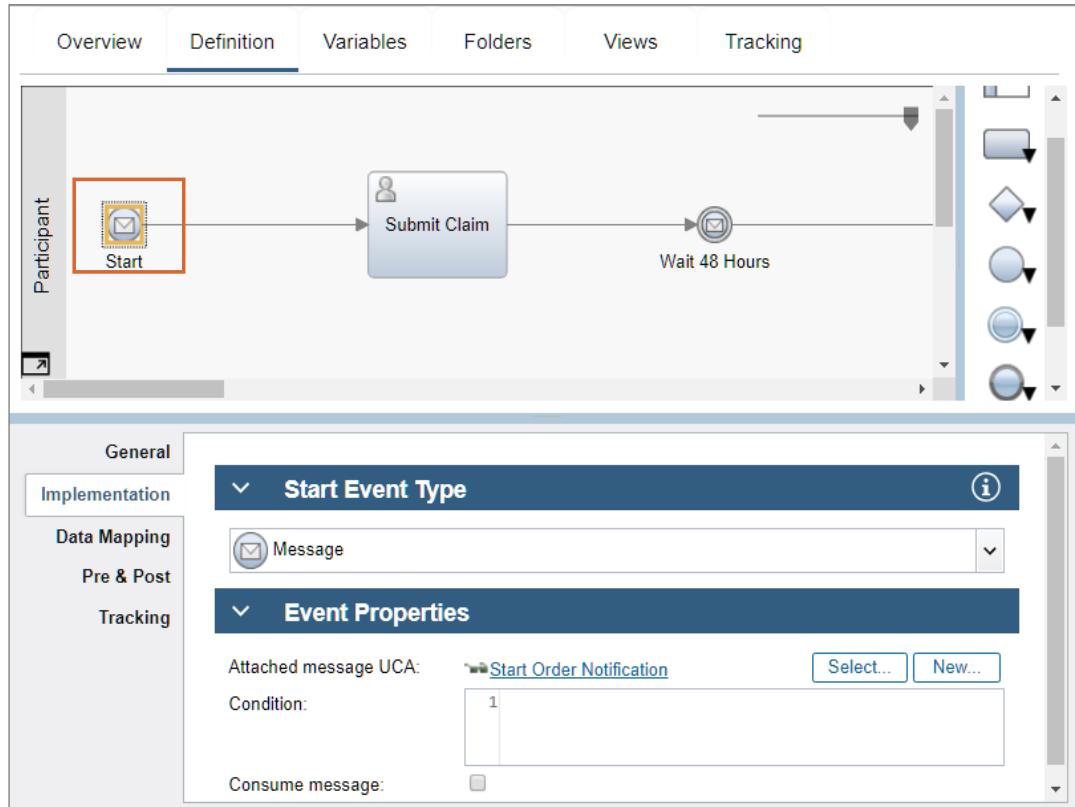
© Copyright IBM Corporation 2020

Figure 8-12. Message start events

When a message start event receives a message (specifying that an incoming message is to start a process at run time), an instance of the process is created. A unique process instance ID is assigned to it. The message start event functions the same as a regular start event, but a user does not create the process instance through the Process Portal. A message event creates the instance.



Implementing message start events



Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-13. Implementing message start events

The final step is to implement the start event on a process.

When implementing message start events, the following configuration settings are configured for the start message event to work properly:

- **Attached UCA**

Undercover agents (UCA) are used to send and receive messages.

- **Condition**

Occasionally, developers must set conditions on the processing of incoming messages. If the condition that the developers specify evaluates to true, the message is accepted and processing continues; otherwise, it is stopped. Because the message condition is evaluated before the message values can be passed to the input variables of the process, the message values are passed to the condition in a special namespace, `tw.message`. If the message condition evaluates to true, the values are passed from the `tw.message` namespace to the process input variables.

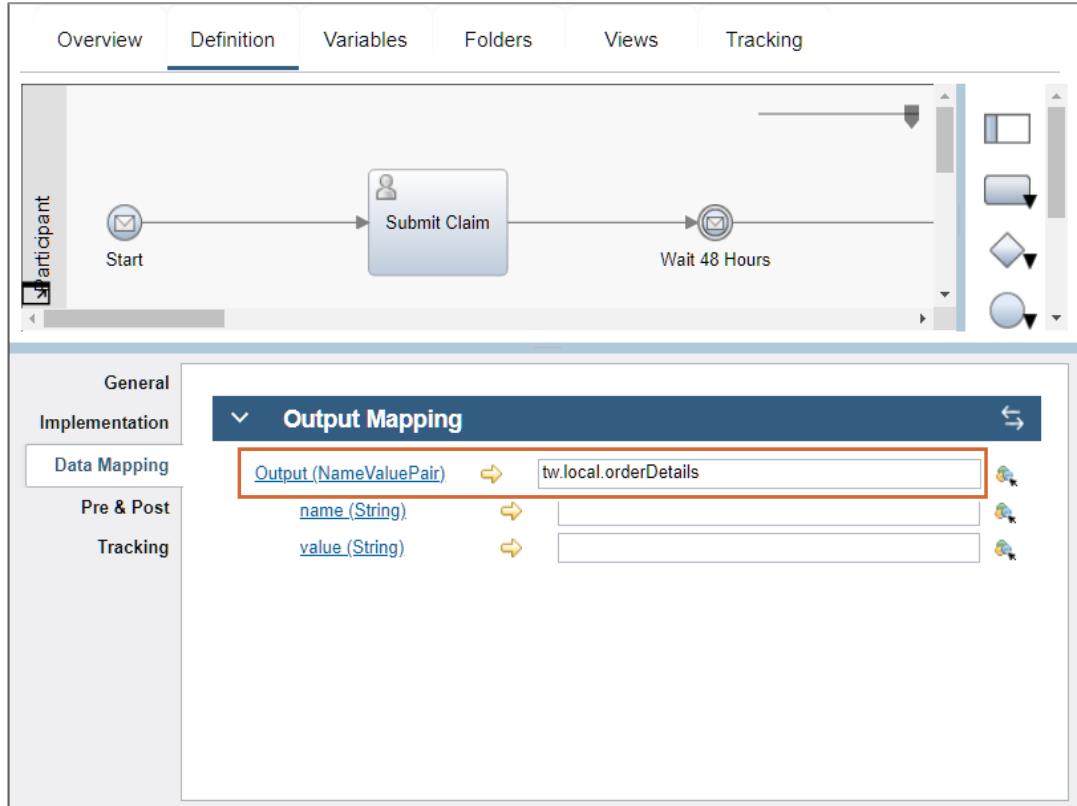
- **Consume message**

Developers configure message events to consume messages. If they do, when a message is delivered to a running process, the message consumes the first message event in the process

that can accept it. The UCA that is attached to the message event determines the consumption. If the execution of the process instance loops back and reaches the same message event listener or listeners, the message cannot be processed again when a message is consumed. If a new instance of the message is delivered to the process instance, this message is available for consumption again, and the message event accepts it.



Mapping message start event variables



Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-14. Mapping message start event variables

The final step is to map the outputs of the start message event to the variables in the process. Again, the variables that come from the start event are defined as outputs in the enabling service as part of the UCA message.

8.3. Accessing and manipulating external data

Accessing and manipulating external data

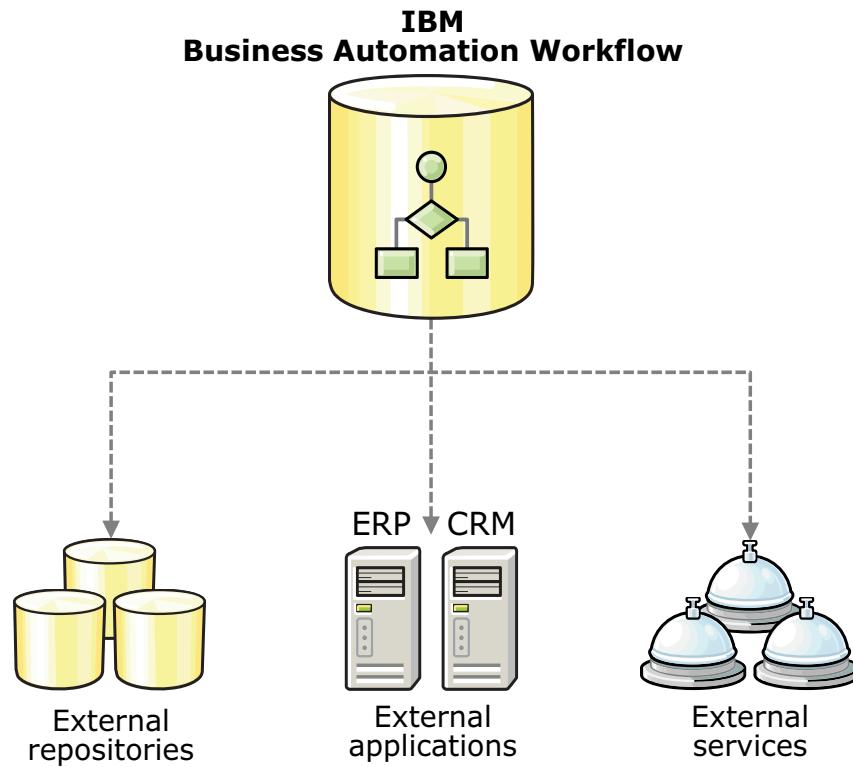
Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-15. Accessing and manipulating external data

A major concern of any enterprise system is its integration capability with other systems. For example, the project team wants users to choose from a list of products available from a web service. Focus on integrating with other systems in this Playback. The integration framework allows interaction with existing applications, Enterprise Content Managers, ERPs, CRMs, web services, and external data sources with a number of protocols.

Integrating with other systems



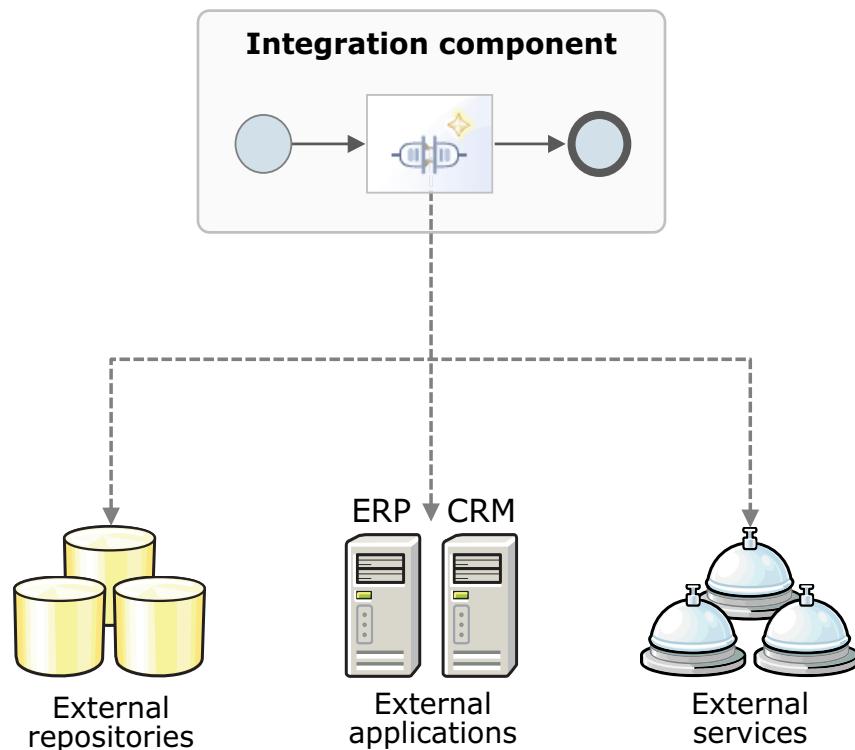
Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-16. Integrating with other systems

When integrating with other systems, consider the fact that IBM Business Automation Workflow supports outbound and inbound integration. When communicating with an outside system like an enterprise resource planning (ERP) or customer relationship management (CRM) system to retrieve, update, or insert data, it is known as an outbound integration.

Integration service flows



Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-17. Integration service flows

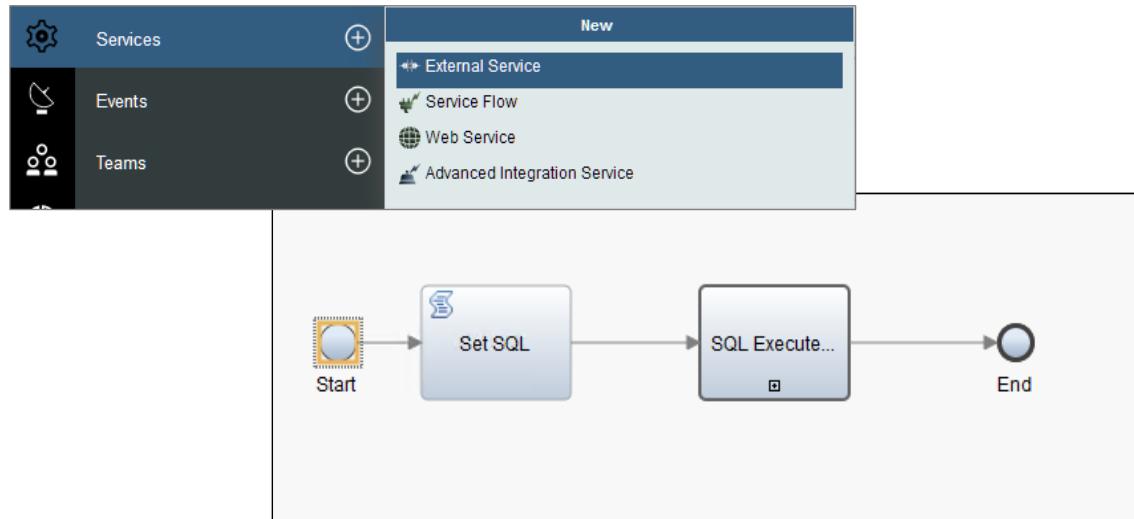
Most outbound integrations involve calling out to data sources and web services. In considering what type of integration component to build, think about the available integration methods and protocols:

- Web service integrations tend to be easy to build and are useful, especially when you are not passing volumes of information.
- Java integrations are robust because they are built in Java, and sometimes you can find existing Java connections to existing systems.

When you message other systems, you do so in a service flow or a Java or web service component. The service components handle the lowest level of communication to the outside application or data source.

Using service components

- A service flow calls the web service or Java integration component, which in turn calls the external application
- The output data (usually XML) comes from the integration component, not the raw external application output data



Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-18. Using service components

You can add a service to multiple service flows, or even use it in the same service more than one time to produce different data from different sources.

When you run a service flow at run time, it calls the web service or Java integration component, which in turn calls the external application.

The screenshot shows the IBM Business Automation Workflow interface under the 'Prebuilt Service Flows' section. On the left, there's a sidebar with icons for HR Recruitment Process, Processes, User Interface, Services, Events, Teams, Data, Performance, Files, Toolkits, Content Management, Hiring Requisition Toolkit, and System Data. The 'Toolkits' item is selected and highlighted in blue. To the right, a list of services is displayed with a search bar at the top. The list includes various integration services like 'Isw Get Conditional Activity Template Names', 'Read email via IMAP (Mail)', and 'SQL Execute Statement (Database)'. The 'SQL Execute Statement (Database)' service is highlighted with a red box.

Playback 2: Integrations

Figure 8-19. Prebuilt Service Flows

© Copyright IBM Corporation 2020

IBM Business Automation Workflow contains many prebuilt services that already contain integration components. These services support common database interactions, including support for parameterized queries. In addition, these services can automatically map query results directly into a specified variable type.

One example is the service that is named SQL Execute Statement.

Use the SQL Execute Statement service flow to develop implementations to:

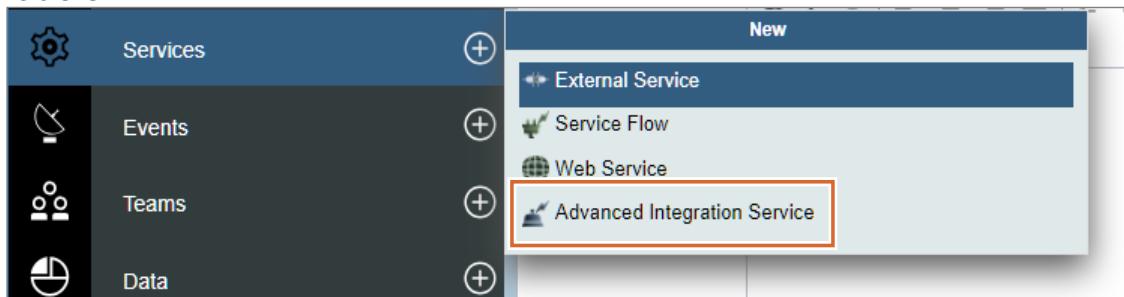
- Read existing data from a database
- Update existing data in a database
- Write new data to a database

To use an SQL service flow in an implementation, you can:

- Select an SQL service flow as the implementation for an activity
- Nest an SQL service flow in another service by dragging it from the library to the diagram of the parent service

Advanced Integration service

- An Advanced Integration service (AIS) is used to call a service that is implemented in IBM Integration Designer from a process (a system task) or another service (a nested service)
- A collaboration must exist between the developer who is working with IBM Process Designer and an Integration Developer who is working with IBM Integration Designer
- Use the information in authoring services in IBM Integration Designer to continue developing your Advanced Integration service
- Add services, service-related functions, BPEL processes, or monitor models



Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-20. Advanced Integration service

An Advanced Integration service is a collaboration between a business user who works with IBM Process Designer and an Integration Developer who works with IBM Integration Designer.

For example, your business process might need a list of computer parts in your warehouses in Canada. Checking with an Integration Developer, you realize that a service is being built in Integration Designer to query the Canadian warehouses and return an inventory list of the computer parts available. You might create an Advanced Integration service that would use this Integration Designer service as an activity in your business process.

To create services, you must have access to a process application or toolkit in the workflow center repository. Users who have administrative rights to the repository control access to process applications and toolkits.

Advanced Integration Services are not covered in this course.

8.4. Exposed process variables (EPVs) and environment variables (ENVs)

Exposed process variables (EPVs) and environment variables (ENVs)

Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-21. Exposed process variables (EPVs) and environment variables (ENVs)

Business objects and variables were covered earlier in this course in terms of the data model for the process application. Two other kinds of data elements are available: the exposed process variable (EPV) and the environment variable (ENV).

IBM Training

Environment variables (ENVs)

The screenshot shows the IBM Business Automation Workflow interface. At the top, there are tabs for 'DESIGNER' and 'INSPECTOR'. Below them is a header with a user icon, a search bar, and buttons for 'Return to Workflow Center' and 'author1'. A 'Create new' button is also visible. On the left, a sidebar titled 'Process App Settings' shows an 'HR Recruitment Process' selected. The main area has tabs for 'Overview', 'Environment Variables', and 'Servers', with 'Environment Variables' being the active tab. This tab displays a table with columns: Key, Default, Development, Test, Stage, and Production. One row is shown: 'TrainingDB' with 'jdbc/TrainingDB' in the Default column. There are also '+' and '-' buttons for adding or removing rows.

- Each process application and toolkit that is created in the workflow center repository includes environment variables
- The project team sets these environment variables to ensure that the process implementations are using correct values, whatever environment they deploy to or whatever change occurs at run time in the environment

Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-22. Environment variables (ENVs)

IBM Business Automation Workflow has the capacity to define environment variables. Each process application and toolkit that is created in the workflow center repository includes environment variables. The project team sets these environment variables to ensure that the process implementations are using correct values, whatever environment they deploy to or whatever change occurs at run time in the environment. For example, suppose that your process includes an implementation that requires the port number for an external application. With an environment variable, you can set the port number for each environment in which the process runs. If the process is running on a test environment, it might have a different port number than if the same application were running on a production environment. Developers can specify a default value and a value for each Workflow Server. Environment variables have a limitation of four environments (plus a default value). If you need more than four environments, use an exposed process variable for each environment needed.

Name environment variables by starting with lowercase letters and separating different words with periods. Thus, `connector.version` is an acceptable name for an environment variable.

IBM Training

Exposed process variables (EPVs)

The screenshot shows the 'DailyAllowance' EPV configuration page. A callout box highlights the text: 'With EPVs, business users can modify the value of a variable that is exposed in the Process Portal without providing access to IBM Process Designer'. The interface includes sections for Common (Name: DailyAllowance, Modified: author1 (Nov 19, 2020, 12:39:58 PM), Documentation: 'This EPV holds the value that Supervisors can change to limit daily expenditure allowances.'), Details (Feedback email contact: admin@email.com), Exposed Process Value Variables (dailyDollarAmount (String)), Variable Details (External name: dailyDollarAmount, Variable name: dailyDollarAmount, External description: 'Use this value to limit daily expenditure allowances.'), and Exposing (Exposed to: <none>, Select... button).

Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-23. Exposed process variables (EPVs)

In IBM Process Designer, you can create Exposed Process Values (EPVs) to define a set of variables you want to expose to specific users. These variables can be modified by the users while instances of a process are running. For example, if you create a process to handle expense reimbursement, you might want to enable supervisors to change the allowed amounts for daily expenditures, or the dollar amount that coincides with various levels of approvers. By creating EPVs, you can provide this type of flexibility, allowing users to adjust specific variable values as constants, thereby affecting the flow of all running process instances, task assignments, and so on.

EPV versus environment variable comparison (1 of 2)

	EPV	Environment variable
Value can be changed in the Portal by business users	Yes	No
Can be scheduled to change value in the future	Yes	No
Environment-specific values	Same value for all environments initially, but it can be set independently with the Process Admin console	Different values for each environment, set from the IBM Process Designer

Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-24. EPV versus environment variable comparison (1 of 2)

This chart is a quick comparison of the EPV and the environment variable.

EPV versus environment variable comparison (2 of 2)

	EPV	Environment variable
Environment cap	None	4 + 1 (default)
Casting	Must always be cast to a type (String, Integer, or other types)	Always String type, but can be cast again
Scope	EPVs must be "linked" everywhere they are used	Can be referenced anywhere in the same process app without a local declaration

Figure 8-25. EPV versus environment variable comparison (2 of 2)

This chart is a continuation of the comparison of the EPV and the environment variable.

8.5. Applying asset tagging

Applying asset tagging

Playback 2: Integrations

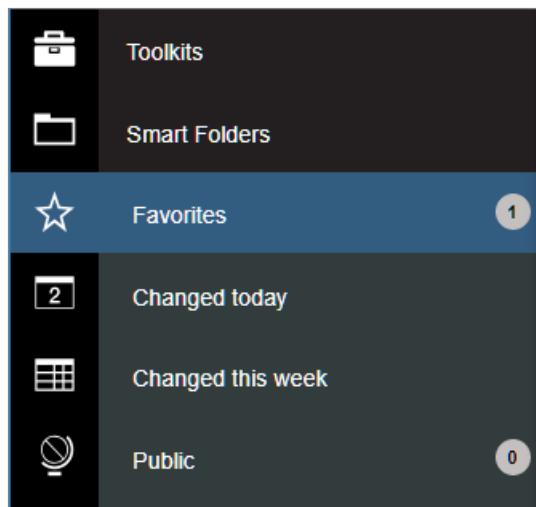
© Copyright IBM Corporation 2020

Figure 8-26. Applying asset tagging

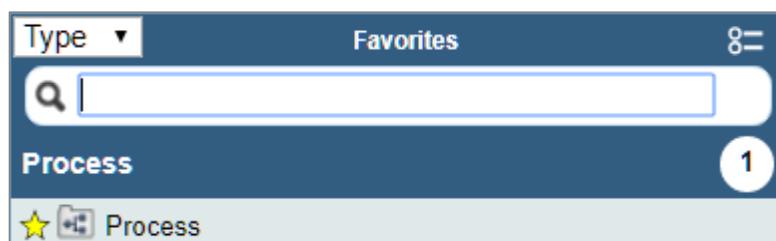
When library artifacts are created, you have a constant necessity to organize assets throughout the development cycle. Favorites, tagging, and smart folders can be used to provide quick access to all of your assets.



Creating favorites



- When organizing assets, IBM Business Automation Workflow offers several ways to organize the library assets.
- Any asset can be marked for quick access by marking it as a favorite.
- All of these assets automatically show up in the smart folders category favorites in the lower-left corner.



Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-27. Creating favorites

When organizing assets, IBM Business Automation Workflow offers several ways to organize the library assets. Many of these concepts are already familiar to developers.

Any asset can be marked for quick access by marking it as a favorite. All of these assets automatically show up in the smart folders category favorites in the lower-left corner. Select the star next to any asset to mark it as a favorite.

IBM Training

The screenshot shows the IBM Training interface with a sidebar on the left containing categories such as Processes, User Interface, Services, Events, Teams, Data, Performance, Files, Toolkits, Smart Folders, Favorites, and Changed today. A context menu is open over a process item named 'Hiring Request Process'. The menu includes options like Open, Favorite, Tags, Revert..., Copy to tip, Copy item to, Move item to, Duplicate, Rename, and Delete.

Playback 2: Integrations

© Copyright IBM Corporation 2020

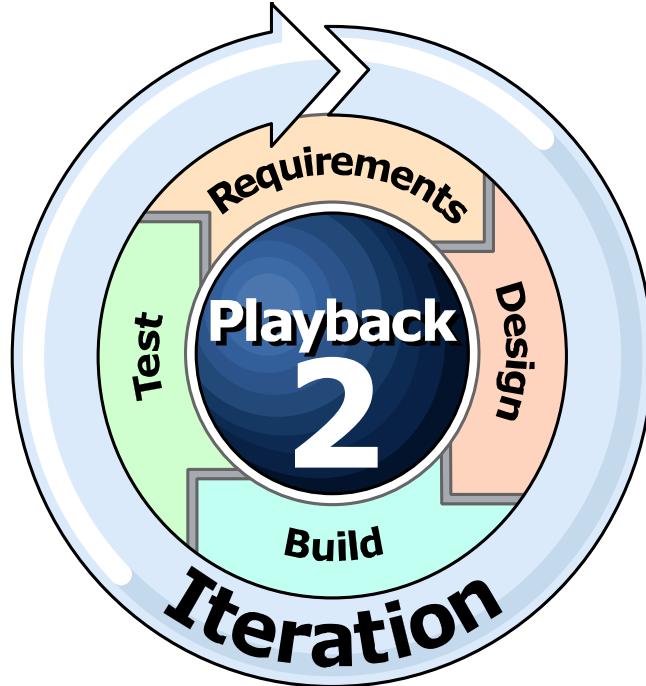
Figure 8-28. Tagging library items

When tagging library items, many of the library assets fall into categories. Developers can use a predefined set of tags or create their own tags for groups of items. After developers tag a set of items, they can view them by asset type or by tagged items.

Right-click an asset and click **Tag** to tag an item. To view by Tag, click the **Type** menu and then click **Tag**.

Playback 2: Integrations

- Playback 2: Integration is now ready
- The goal is to demonstrate “real data, real time”
- Show the business rule logic and show tasks that are being created as a result of the decision service
- Demonstrate message events in the process that affect process flow
- Some of the integrations are created in Playback 1: Building coaches to enable coach functions



Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-29. Playback 2: Integrations

Playback 2: Integrations are now ready. The goal is to demonstrate “real data, real time.” Log on to the portal and demonstrate any integrations that your process depends on. Show the business rule logic and show tasks that are being created as a result of the decision service. Demonstrate message events in the process that affect process flow. Some of the integrations are created in Playback 1 to enable coach functions. Those integrations are demonstrated in the next Playback.

When this development phase is complete, some external services for this Playback might be incomplete. Continue to use mock data services to enable your processes and services, but focus the Playback presentation on those integration endpoint services that are still not available. All artifacts to enable the integration are already created in the library, and a quick switch of flow is all that is necessary when the new integration is enabled.

To ensure that you are ready for Playback 2: Integrations:

- **Log on to the portal:**
 - Log on to the Process Portal. The URL is: `http://<server URL:port>/portal`
- **Complete a human activity:**
 - Complete the coach.
 - Click **OK** when the coach is displayed.

- **Demonstrate the result of a decision service:**

- Show tasks that are being created and process flow changes depending on the result of a decision service.

Unit summary

- Explain how to create a decision service
- Define a message start event
- Explain how an enabling service is used with the message start event
- Describe how to create and configure an undercover agent (UCA)
- Describe how to start a process with a message start event
- Define the basic function of an integration service flow
- Identify the components of the IBM Business Automation Workflow integration architecture
- Describe how integration components interact with services
- Configure and define integration service flows for outbound integration
- Describe the differences between an environment variable and an exposed process variable
- Organize assets with favorites, tagging, and smart folders

Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-30. Unit summary

Review questions

1. What does the acronym “BAL” stand for?
2. True or False: The three main steps that are required to implement a start message event are:
 - Create the enabling service (which defines the payload of the message)
 - Create the UCA
 - Attach the UCA to the start message event.
3. What smart folder automatically contains all the artifacts with stars that are attached to them?
4. If you want to create an outbound integration to an SQL database, where is the first place you look?
5. Which variable type is designed to allow business users to modify the value of a variable without providing them access to the IBM Process Designer?

Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-31. Review questions

Write your answers here:

- 1.
- 2.
- 3.
- 4.

Review answers

1. **Business Action Language.** Using natural language to author rules, instead of JavaScript, means that no programming expertise is required to create business rules, and the rules are easier for people to read and understand.
2. **True**
3. **Attaching a star** to an artifact makes it a favorite and puts it in the favorites smart folder.
4. **The system data toolkit** contains numerous service flows that you can drag easily into your service. For example, the system data toolkit has almost 20 different SQL service flows.
5. **The exposed process variable (EPV)** is designed for business users to change values of business data in real time and without providing a login to the IBM Process Designer.

Playback 2: Integrations

© Copyright IBM Corporation 2020

Figure 8-32. Review answers

Exercise: Playback 2: Integrations

Playback 2: Integrations

© Copyright IBM Corporation 2019

Figure 8-33. Exercise: Playback 2: Integrations

Exercise introduction

- Create a decision service
- Create and configure a UCA
- Start a process with a message start event
- Use tagging to organize assets
- Query a database to obtain information and populate a list variable
- Create environment variables (ENVs) and exposed process variables (EPVs)
- Change a text control to a single select control



Playback 2: Integrations

© Copyright IBM Corporation 2019

Figure 8-34. Exercise introduction

Unit 9. Playback 3: Error handling and deployment

Estimated time

01:00

Overview

Playback 3 is the final Playback. The goal of this Playback is to demonstrate what happens when your process encounters an error, and how that error is handled. This unit covers error handling patterns that are used in your process application, conveying general principles for error handling without having to demonstrate every instance.

How you will check your progress

- Review
- Exercise

Unit objectives

- Catch an error in a process or service

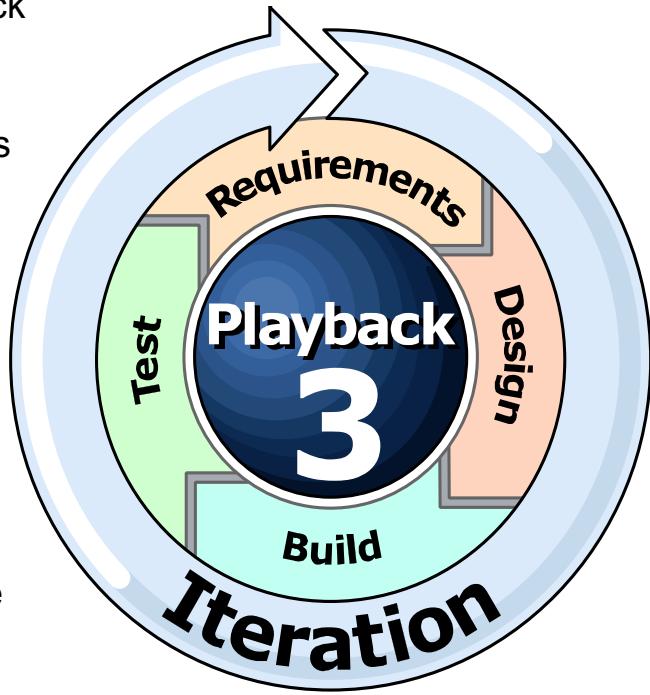
Playback 3: Error handling and deployment

© Copyright IBM Corporation 2020

Figure 9-1. Unit objectives

Playback 3: Hardening processes and services

- Playback 3 is the final Playback
- The goal of this Playback is to demonstrate what happens when your process encounters an error, and to demonstrate how that error is handled
- Consider demonstrating the different error handling patterns that are used in your process application
- Conveys the general principles that are used for error handling without having to demonstrate every instance of error handling in your process app



[Playback 3: Error handling and deployment](#)

© Copyright IBM Corporation 2020

Figure 9-2. Playback 3: Hardening processes and services

The goal of this Playback is to demonstrate what happens when your process encounters an error, and to demonstrate how that error is handled. Demonstrating error handling might take some test harnesses to generate the errors internally.

Because your processes and the sheer number of artifacts that are created might be large, demonstrating all of the error handling can be overwhelming. Consider demonstrating the different error handling patterns that are used in your process application so stakeholders have a good idea of the general way errors are handled when they occur. It conveys the general principles that are used for error handling without having to demonstrate every instance of error handling in your process app.

Topics

- Handling errors in a process
- Handling errors in services

[Playback 3: Error handling and deployment](#)

© Copyright IBM Corporation 2020

Figure 9-3. Topics

Key concepts in this unit

- **Playback 3: Hardening processes and services:** This stage of the Playback demonstrates what happens when your process encounters an error, and how that error is handled
- **Error handling in process:** Attach an error intermediate event to an activity and connect that event to an error handling flow or activity
- **Error handling in services:** Use error intermediate events to catch errors, and use error end events to throw errors

[Playback 3: Error handling and deployment](#)

© Copyright IBM Corporation 2020

Figure 9-4. Key concepts in this unit

9.1. Handling errors in a process

Handling errors in a process

Playback 3: Error handling and deployment

© Copyright IBM Corporation 2020

Figure 9-5. Handling errors in a process

When modeling error handling as part of your business processes, developers discover errors with error intermediate events or event subprocesses, and they can cause errors with error end events.

Using error events to handle errors

- When you develop an application in IBM Business Automation Workflow, build error handling into processes and services as follows:
 - To detect errors
 - To specify how errors are thrown and caught in your runtime environment
 - To recover in a predictable manner
- Three types of error events are possible:
 - Error end events in processes and services that throw errors
 - Error intermediate events in processes and services that catch errors
 - Error start events in process event subprocesses that catch errors
- Assign error codes and error data to thrown errors from the error end events

[Playback 3: Error handling and deployment](#)

© Copyright IBM Corporation 2020

Figure 9-6. Using error events to handle errors

To catch errors by using error intermediate events, select an error code from a list of previously defined errors and map the error data to a variable. The error intermediate events are boundary events, which are intermediate events that are attached to the boundary of an activity. Each boundary event can be triggered only while the activity is running, interrupting the activity. From the IBM Process Designer, you can use an error intermediate event that is attached to the boundary of an activity. This error intermediate event catches specific errors and error data from a linked process, a subprocess, or a service.

Another way to catch errors is by using error intermediate events in services that catch all errors. When building services, you can attach an error intermediate event to the boundary of a step to catch all errors for the step. You can use an error intermediate event as part of the service flow. You use it to catch all errors that steps of the service flow raise that are not handled through an error intermediate event at the boundary of the step.

You also can catch errors by using error event subprocesses in processes. In the subprocess, you use an error start event that catches errors if the start event is triggered.

However you decide to catch errors, designate the error behavior for the events on the Properties tab in your diagram. Under Implementation, go to the Error Properties section to designate the following error handling behavior:

- Catch all errors or specific errors. To catch specific errors, you can select the error code, map the error data, or both, as described in the following bullets.
 - Filter the specific errors that are caught by selecting an error code from a list of all thrown errors for the linked process, subprocess, or service.
 - Map the error data into a variable by selecting an error-mapping variable that was previously defined on the Variables tab.

Important: If the error code changed, make sure to select the variable again so that it is mapped properly.

If multiple error events are defined to catch errors for an error that is thrown in a linked process, subprocess, or service, the precedence rules determine the catching event. The precedence is in the order that they are listed in the Error event components table.

Errors are caught in the following order in your runtime environment:

1. The boundary events catch errors that the attached activity raises, as described in the following table.
2. If no error boundary event handles the error, and a subprocess is in a process or in an unattached intermediate error event in a service, errors are caught in the error event subprocesses. This process is described in the following table.
3. If no error event subprocess handles the error in an event subprocess, linked process, or service, errors are propagated to the next level.

Handling errors in a process

- **Catching errors with error intermediate events:** For processes, you can attach an error intermediate event to an activity and connect that event to an error handling flow or activity.
- **Throwing errors:** You can use an error end event in your process to specify an error code and map to an error type on errors that are thrown from the flow of a process or a service.

Process events	Description
 Error intermediate event at the boundary of an activity	<ul style="list-style-type: none"> • Catches specified errors or all errors • Provides error handling logic for errors that the attached activity raises
 Error end event	<ul style="list-style-type: none"> • Use to throw an error to parent processes

[Playback 3: Error handling and deployment](#)

© Copyright IBM Corporation 2020

Figure 9-7. Handling errors in a process

The following considerations apply when you handle errors in a process:

- **Catching errors with error intermediate events**

For processes, you can attach an error intermediate event to an activity and connect that event to an error handling flow or activity.

To determine whether to use error immediate events, consider the following points:

- If an error occurs while a process is running an activity with an attached error event at the boundary, the process flows along the sequence line that is attached to the error event. Errors are handled in the flow and then proceed with the normal processing.
- Error intermediate events must be attached to an activity.
- You can have multiple error events for an activity, but only one catches the error.
- Consider specifying the error data to catch specific errors, filtering on the error code for the types of errors that are caught, and mapping to a variable after the errors are caught. When all errors are caught, or if only an error code is specified, the error data is captured in an XMLElement in the `tw.system.step.error` variable.

- **Throwing errors**

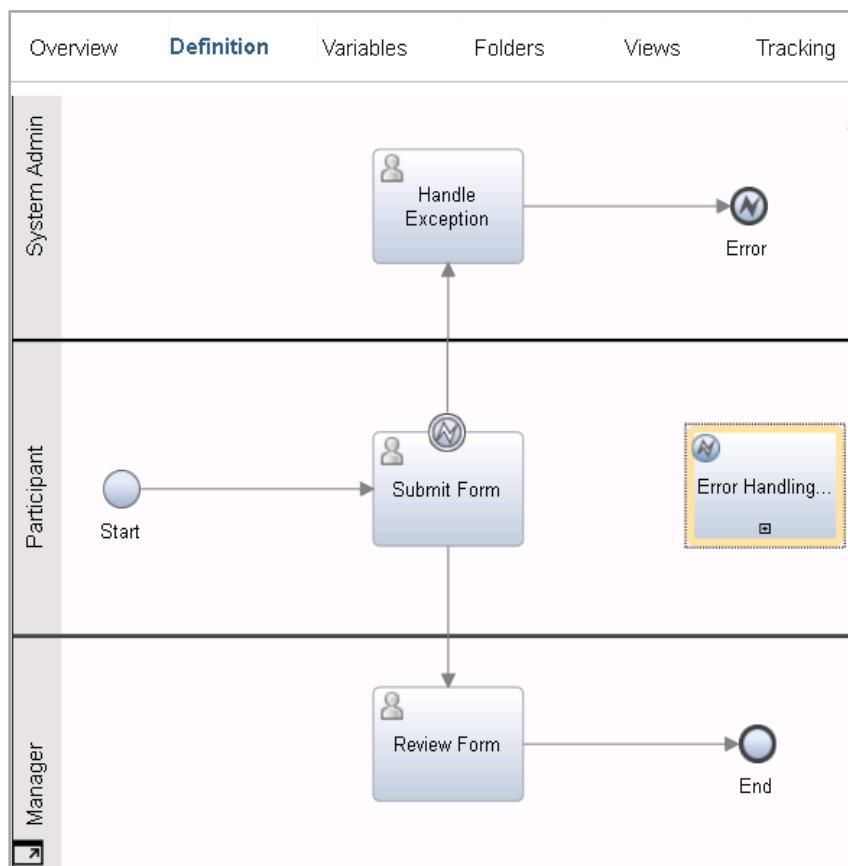
You can use an error end event in your process to specify an error code and map to an error type on errors that are thrown from the flow of a process or a service.

When working with either error events or event subprocesses, think about whether errors can be handled immediately, and normal processing can continue, or if another error can be thrown at another level. Then, implement error handling from the bottom up. In other cases, it might be more efficient and readable if a subprocess can be reused. Build each linked process and service so that errors can be captured and corrected. If a correction is not possible at the lowest level of the implementation, you can allow the error to move up a level. Do not include an error event to rethrow the error to the calling service or process, as shown in the following section.

Catching errors

Errors are caught in the following order in your runtime environment:

1. The boundary events catch errors that the attached activity raises
2. If the subprocess is in a process or in an unattached intermediate error event in a service, errors are caught in the error event subprocesses
3. Errors are propagated to the next level



Playback 3: Error handling and deployment

© Copyright IBM Corporation 2020

Figure 9-8. Catching errors

Use error events to specify how errors are thrown and caught in your runtime environment.

You can assign error codes and error data to errors that the error end event throws.

Errors are caught in the following order in your runtime environment:

- The boundary events catch errors that the attached activity raises.
- If the subprocess is in a process or in an unattached intermediate error event in a service, errors are caught in the error event subprocesses.
- Errors are propagated to the next level.
- Specifying the variable name in the mapping controls filtering by error data type. If a mismatch occurs between the type of variable and the error data type displayed on the **Properties** tab, the variable and its type determine the behavior.

9.2. Handling errors in services

Handling errors in services

Playback 3: Error handling and deployment

© Copyright IBM Corporation 2020

Figure 9-9. Handling errors in services

Processes that you develop in IBM Business Automation Workflow include integrations with external systems, server scripts, and other complex implementations. Anticipate potential system exceptions and create the components that are required to handle those exceptions when they occur. For example, if a process includes an integration with a database system, that database might not be available when each new instance of the process runs. So, when you develop the integration in IBM Business Automation Workflow, you must build in exception handling to detect errors and recover in a predictable manner.

You can build error handling capabilities into both processes and services with the available service components.

Exception handling is a topic that is explored with system administrators, developers, and other interested stakeholders to plan what actions are taken when dealing with errors.

Catching errors in services

Service component	Description
 Error intermediate event that is attached to the boundary of a step	<ul style="list-style-type: none"> • Listens for exceptions from the service component to which it is attached
 Error intermediate event as part of the service flow	<ul style="list-style-type: none"> • Errors in the steps of the service flow and errors that are missed in the error intermediate event that is attached to an activity are caught • This event can have only outbound links
 Throw exception	<ul style="list-style-type: none"> • Use to purposely throw an error and end processing • You might, for example, use a Throw Exception component if you return too many rows from a database (over a limit that is normal and would bog down the server)

[Playback 3: Error handling and deployment](#)

© Copyright IBM Corporation 2020

Figure 9-10. Catching errors in services

For services, you can use error intermediate events to catch errors, and you can use error end events to throw errors.

To determine whether to use error events in your services, consider the following points:

- You must attach error intermediate events to steps in your service.
- Include error intermediate events in the service flow so that they can act as global error handlers in the service.
- Determine whether errors can be handled immediately, and normal processing can continue, or if another error can be thrown at another level. Then, implement error handling from the bottom up.
- Use an error end event to throw a specific error. You can specify an error code and error data for the error.
- Consider specifying the error data to catch specific errors. For example, you might filter on the error code for the types of errors that are caught and map the error code to a variable after the errors are caught. When all errors are caught, or if only an error code is specified, the error data is captured in an `XMLElement` in the `tw.system.error` variable.

When building services that include integrations with external systems or other implementations, use error intermediate events when it is possible to handle errors in the steps in the service. Be sure to include the appropriate logic and an error end event in your service to throw errors when they still exist after attempts to handle them. Throwing the errors ensures that the errors are passed to parent processes and services and can be handled at a higher level.

Client-side human services: Throwing errors by using error end events (1 of 2)

- For errors that are thrown from the flow of a client-side human service, use error end events to end the processing of a service flow at a specified step
- To throw a specific error, do the following steps:
 - Set the properties of the error in the Implementation tab of the error end event
 - Specify the error code and map the error data to a specified variable

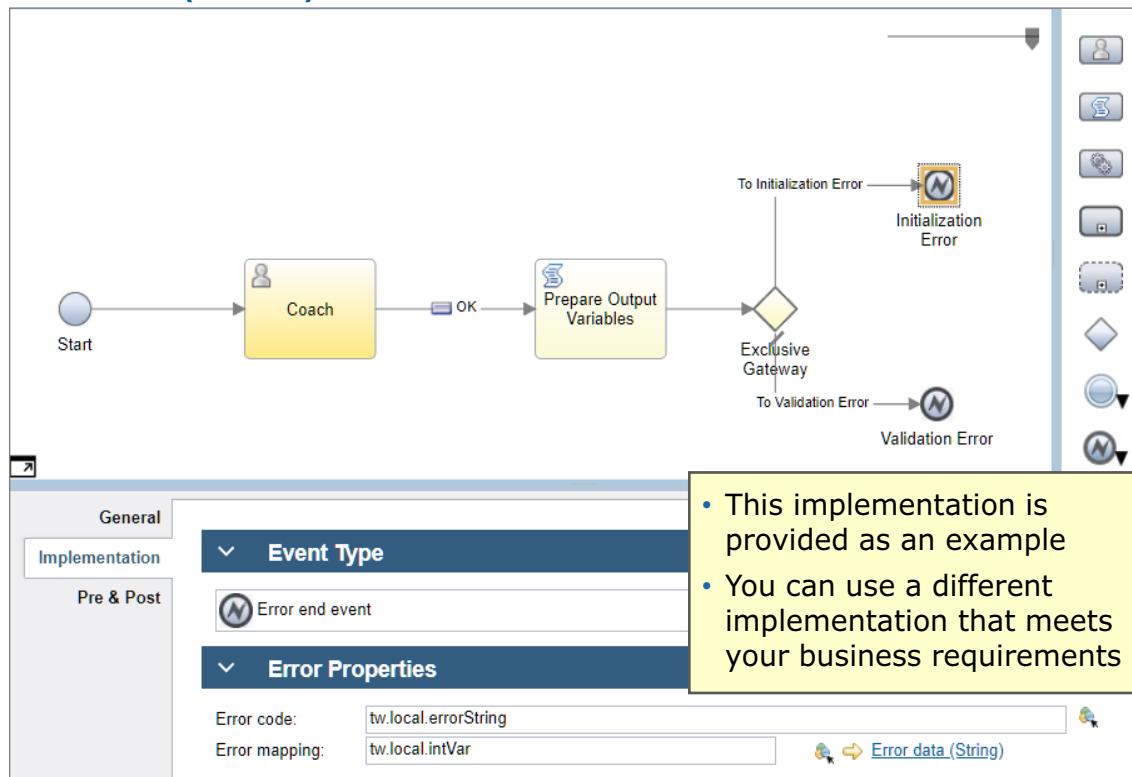
Playback 3: Error handling and deployment

© Copyright IBM Corporation 2020

Figure 9-11. Client-side human services: Throwing errors by using error end events (1 of 2)

IBM Training

Client-side human services: Throwing errors by using error end events (2 of 2)



Playback 3: Error handling and deployment

© Copyright IBM Corporation 2020

Figure 9-12. Client-side human services: Throwing errors by using error end events (2 of 2)

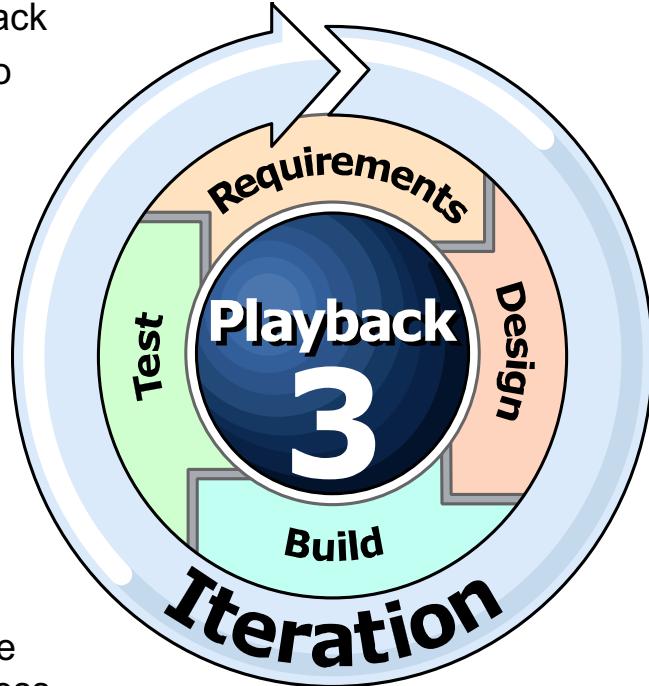
To add an error end event to the client-side human service, do the following steps:

1. Open IBM Process Designer.
2. Open the client-side human service that you want to work with.
3. In the Diagram view, drag an end event onto the canvas.
4. Select the end event, and in the Implementation tab, under Event Type, select **Error end event**. The end event changes into an error end event.
5. In the Implementation tab, under Event Properties, click the Error code picker to select a local variable and specify the error code for the error to be thrown. The error code expression must evaluate to a string value at run time. The error code might be defined as a string literal (for example, "ErrorCode1"). If so, the code is included in the list of defined errors that can be caught when creating an error intermediate event to catch the error in a process. Click the Error-mapping picker to map the error data to an error-mapping variable that was previously defined on the Variables tab.
6. Optional: To define multiple error end events that can have different error handling logic for different errors, iterate through steps 3 – 5. For each error end event, use different error codes or error data to differentiate between the different kinds of errors.

7. In the diagram, connect each error end event to the logic that you want to run when the error occurs. If you have multiple error end events, connect each one to the error handling logic that applies. The following example shows a client-side human service that uses a coach to display an error message to the user. The coach uses a variable validation script, which is connected to two different error end events with an exclusive gateway.

Playback 3: Hardening processes and services

- You arrived at the final Playback
- The goal of this Playback is to demonstrate what happens when your process encounters an error, and to demonstrate how that error is handled
- Consider demonstrating the different error handling patterns that are used in your process application
- Conveys the general principles that are used for error handling without having to demonstrate every instance of error handling in your process application



Playback 3: Error handling and deployment

© Copyright IBM Corporation 2020

Figure 9-13. Playback 3: Hardening processes and services

You arrive at the final Playback. The goal of this Playback is to demonstrate what happens when your process encounters an error, and to demonstrate how that error is handled. Demonstrating error handling might take some test harnesses to generate the errors internally.

Because your processes and the sheer number of artifacts that are created might be large, demonstrating all of the error handling can be overwhelming. Consider demonstrating the different error handling patterns that are used in your process application so stakeholders have a good idea of the general way errors are handled when they occur. It conveys the general principles that are used for error handling without having to demonstrate every instance of error handling in your process app.

To ensure that you are ready for Playback 3:

1. Log on to the portal
 - Log on to the portal. The URL is: `http://<server URL:port>/portal`
2. Show error handling
3. Run the service.
4. Demonstrate the error handling capabilities that are built into the service.

Unit summary

- Catch an error in a process or service

Playback 3: Error handling and deployment

© Copyright IBM Corporation 2020

Figure 9-14. Unit summary

Review questions

1. When an error occurs in an activity in a process, which takes the highest precedence in catching the error?
 - A. An event subprocess in the same process as the activity
 - B. A flow that is exiting the activity and going to an error end event
 - C. An attached error intermediate event

2. What are the types of error events that can be modeled in services?
 - A. Error message event
 - B. Error intermediate event
 - C. Error end event
 - D. Error start event

[Playback 3: Error handling and deployment](#)

© Copyright IBM Corporation 2020

Figure 9-15. Review questions

Write your answers here:

1.

2.

Review answers

1. C. Boundary events take first precedence.
2. **Error intermediate event and error end event.**

The error intermediate event can be attached to a step or can be placed on the palette to catch errors globally inside the service.

Exercise: Playback 3: Handling errors and deploying your process application

Playback 3: Error handling and deployment

© Copyright IBM Corporation 2019

Figure 9-17. Exercise: Playback 3: Handling errors and deploying your process application

Exercise introduction

- Harden a service with a catch exception component
- Create a snapshot for deployment



Playback 3: Error handling and deployment

© Copyright IBM Corporation 2019

Figure 9-18. Exercise introduction

Unit 10. Advanced routing

Estimated time

01:00

Overview

This unit covers routing tasks to process participants by using the advanced routing capabilities in IBM Business Automation Workflow. It focuses on routing by using a team filter service and dynamically generated teams.

How you will check your progress

- Review
- Exercise

Unit objectives

- Explain user distribution in a process application
- Determine the best source for team data
- Model decision authority for a process
- Determine when to implement various types of routing
- Explain how to create a routing design by using a team filter service
- Explain when to use team filter services to support business policy

Advanced routing

© Copyright IBM Corporation 2020

Figure 10-1. Unit objectives

Topics

- Team data and lanes
- Custom routing
- Team filters
- Experts team
- Four Eyes

[Advanced routing](#)

© Copyright IBM Corporation 2020

Figure 10-2. Topics

Key concepts in this unit

- **SOR:** System of Record
- **LDAP:** Lightweight Directory Access Protocol: LDAP is a protocol for locating organizations, individuals, and other resources in a network
- **Team Filter:** You can use team retrieval services and team filter services to dynamically determine who is eligible to perform activities. Instead of using a statically defined team, you can use a team retrieval service that returns a list of team members and the name of a team of managers. For example, you can implement a separation of duties policy, or remove members who are not available. To improve performance, these services or service flows can use results caching.
- **Four Eyes:** Four eyes is an example of a business policy relevant to implementing Team Filter services. The four eyes principle requires that two people approve some activity.

10.1. Team data and lanes

Team data and lanes

Advanced routing

© Copyright IBM Corporation 2020

Figure 10-4. Team data and lanes

Team data and lanes

- Team members come from two sources:
 - System of record (SOR)
 - User list
- System administrators have governance over SOR change control procedures
- User lists are more dynamic, and they are set up so business people can administer the group
- Lanes communicate how advanced routing works with a compact number of lanes

[Advanced routing](#)

© Copyright IBM Corporation 2020

Figure 10-5. Team data and lanes

Team member data can come from two sources: a system of record or a user list. When team groups come from a system of record, such as LDAP or a relational database, then system administrators or external data administrators have governance over the change control procedures. Team members that come from a user list are more dynamic and are set up so business users can administer the group.

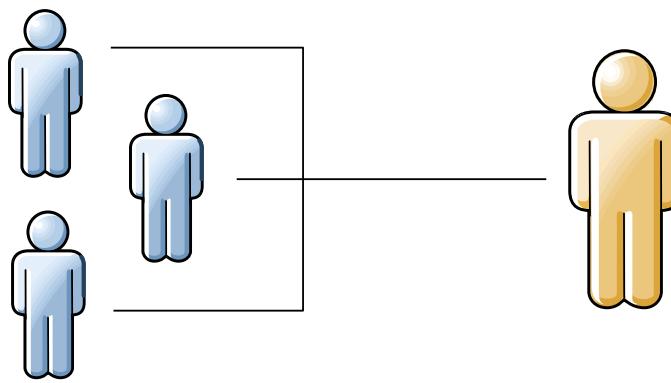
You should consider these things when examining team data from either source:

- The system of record data source
 - Does your company currently use an external security provider, such as LDAP or Active Directory?
 - Is this security provider already integrated with IBM Business Automation Workflow?
 - If not already established, must users be stored and administered internally or externally?
- User list
 - Do developers have access to the Process Admin Console to create groups?
 - If you have limited permissions to the Process Admin Console, is there a dedicated IBM Business Automation Workflow system administrator to create groups?
- Lanes

- Developers want a diagram to be clear for the business audience, but do not want to confuse the audience with a massive process diagram that has numerous lanes and activities.
- With advanced routing, you can avoid adding lanes by setting the routing on an activity, but the diagram must still be readable and answer the question, "Who does what?"
- It is important to remember to communicate how advanced routing works with a compact number of lanes and without getting too technical in the explanation.

Organizational data

- The requirements of everyday business often demand complex routing rules because many organizations have complex process requirements
- One goal of Business Process Management is process improvement
 - The first release of a project is not the end of the software development project
 - It represents the beginning of a process improvement cycle
- Simplifying a process might be the end goal, but the journey to achieve that goal might take a long time
- Integrate your company organizational chart into your process model



Advanced routing

© Copyright IBM Corporation 2020

Figure 10-6. Organizational data

Modeling organizational data can be difficult, as most organizational charts constantly change and become web-like, instead of being rigid. However, be sure to integrate the company organizational chart into your process model.

The requirements of everyday business often demand complex routing rules because many organizations have complex process requirements. Remember, a goal of BPM is process improvement, so the first release of a project is not the end of the software development project. It represents the beginning of a process improvement cycle. Simplifying a process might be the end goal, but the journey to achieve that goal might be a long one.

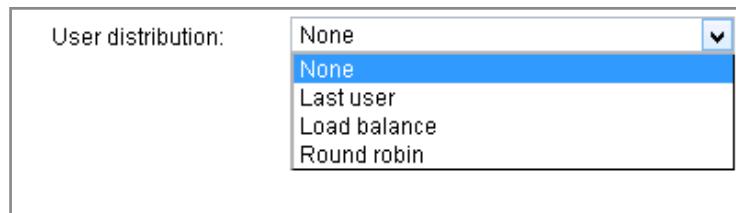
Types of routing

The different types of routing and distribution types

- Types:
 - Lane
 - Team
 - Custom



- Distribution:
 - None
 - Last user
 - Load balance
 - Round robin



[Advanced routing](#)

© Copyright IBM Corporation 2020

Figure 10-7. Types of routing

Several routing and distribution types are available to ensure that tasks get to the right person. If you have experience with business process management, you probably are familiar and use some of the routing and distribution types. The routing options include:

- The **Lane** option assigns the task to the lane that the activity is in.
- **Team** assigns the task to a team designated by a string or JavaScript object.
- **Custom** routing assigns the runtime task according to the JavaScript expression that you provide in the corresponding field.

10.2. Custom routing

Custom routing

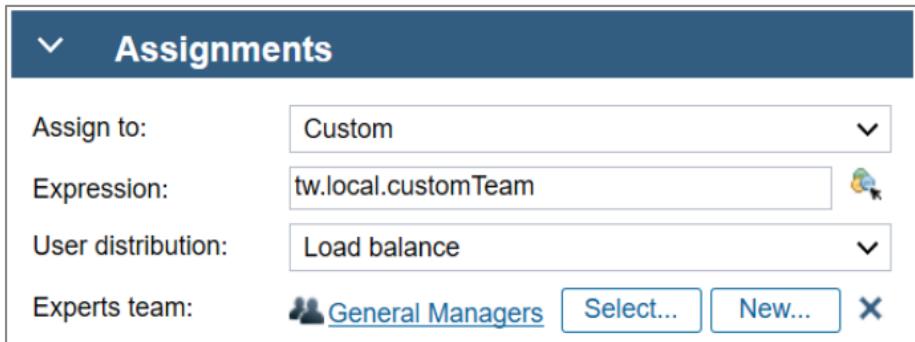
Advanced routing

© Copyright IBM Corporation 2020

Figure 10-8. Custom routing

Custom routing (1 of 3)

- Assigns the runtime task according to the specified JavaScript expression
- Tasks that are assigned by using this option do not appear in the Team Performance dashboard
- Simpler to use than Team filtering
- Useful to quickly create a custom routing solution
- When the logic and algorithms are too complex for other types of routing



The screenshot shows a configuration interface titled "Assignments". It includes the following fields:

- Assign to: Custom
- Expression: tw.local.customTeam
- User distribution: Load balance
- Experts team: General Managers (with buttons for Select..., New..., and X)

Advanced routing

© Copyright IBM Corporation 2020

Figure 10-9. Custom routing (1 of 3)

Custom routing (2 of 3)

Structure of the JavaScript expression depends on the level of assignment

- **USER:<user_name>**,
- **ROLE:<group_name>**
- **TEAM:<team_name>**

where `user_name` is the name of a user (such as `author1`), `group_name` is the name of a security group (such as `tw_authors`), and `team_name` is the name of a team (such as `Managers`).

Note: If the expression evaluates to '`USER:`', then the **User distribution** field has no effect.

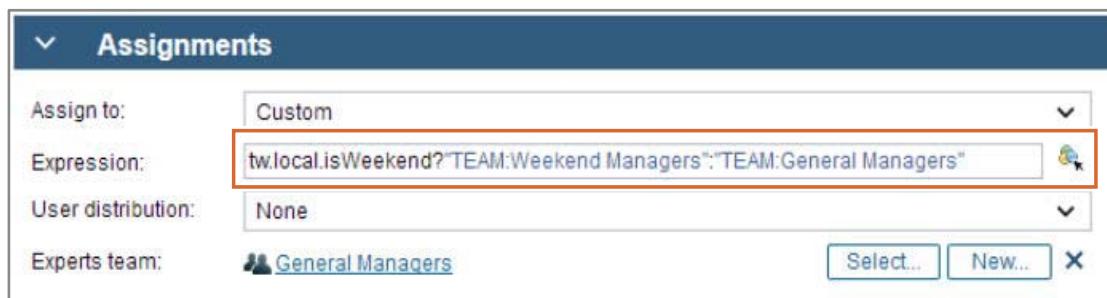
The screenshot shows a configuration interface titled 'Assignments'. It includes fields for 'Assign to' (set to 'Custom'), 'Expression' (containing 'tw.local.customUser'), 'User distribution' (set to 'None'), and 'Experts team' (listing 'General Managers' with options to 'Select...' or 'New...').

Assignments	
Assign to:	Custom
Expression:	tw.local.customUser
User distribution:	None
Experts team:	General Managers Select... New... X

Custom routing (3 of 3)

- Complex JavaScript expressions can be typed in or pasted into the **Expression** field and can be customized as required.
- More valid expressions can be chained together to produce a complex JavaScript expression, for example:

```
tw.local.isWeekend?"TEAM:Weekend Managers":"TEAM: General Managers"
```



Advanced routing

© Copyright IBM Corporation 2020

Figure 10-11. Custom routing (3 of 3)

The JavaScript expression:

`tw.local.isWeekend?"TEAM:Weekend Managers":"TEAM: General Managers"`

evaluates to: If it is the weekend (isWeekend Boolean = true), assign the task the Team: Weekend Managers, otherwise if it is not the weekend (isWeekend Boolean = false), assign the task to the Team: General Managers.

You have an opportunity to develop a solution that uses this complex expression at the end of this unit.

10.3. Team filters

Team filters

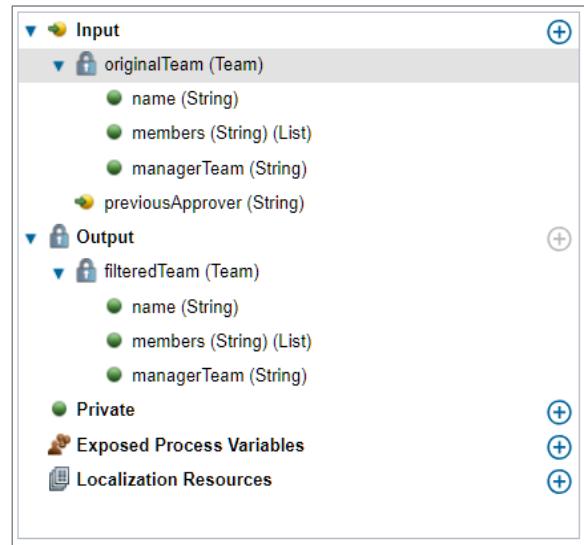
Advanced routing

© Copyright IBM Corporation 2020

Figure 10-12. Team filters

Creating team filters

- Create a team filter with the template
- You can add as many inputs to the team filter as necessary
 - The originalTeam (Team) is automatically added as a default
- The output contains a list of strings that stores the usernames, teams, or security groups that are part of the filtered team
- When filtering, if an originalTeam member contains a team or a security group and you want to remove a username, you must:
 - Create a list of users that the team or security group contains
 - Remove the user when found
 - Return the filtered list



[Advanced routing](#)

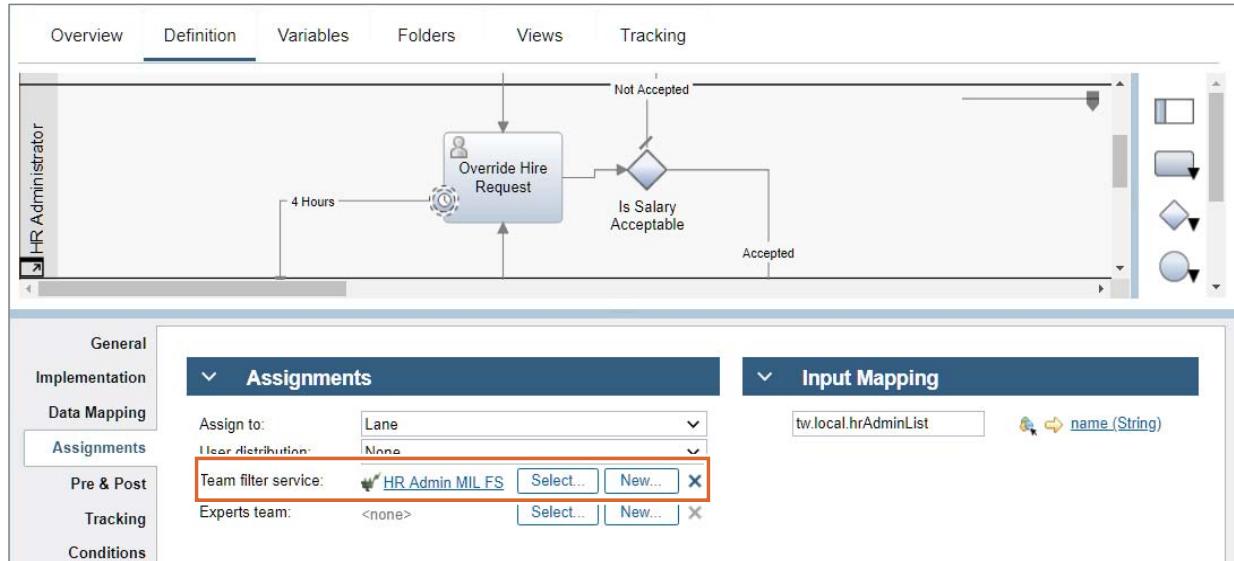
© Copyright IBM Corporation 2020

Figure 10-13. Creating team filters

- The quickest way to create a team filter is to use the team filter template. This template defines the required inputs and outputs of the service. From there, you can add any number of inputs necessary to filter the filteredTeam object.
- The Team object contains a name (String) that is the name of the team. It also contains a managerTeam (String) object that is the name of the team that manages the filteredTeam. Finally, the members (String)(List) object contains a list of usernames, teams, or security groups that are part of the filtered team. When users log in to the portal, IBM Business Automation Workflow checks this list to see whether the user is assigned to this filtered team, and if so, shows the task inside the user's portal inbox.
- When filtering, it is possible that the originalTeam contains a security group or a team. Therefore, if you aim to remove a user from this team, you must first retrieve the members of the team manually before removing the user.
- Because the filtered team service is run when a token is created, you have no way to use the debugger in the Process Designer to step through the filter service at the Process level. Debug your filter service with the service debugger, and if further debugging is necessary at the Process level, use the system log and log.debug (message) to assist you.

Applying a team filter service

The filtering can be based on any criteria and can use input parameters from relevant process variables to determine which users to filter out.



Advanced routing

© Copyright IBM Corporation 2020

Figure 10-14. Applying a team filter service

- Apply the team filter to any activity on a process. The inputs to the team filter can take a list of users and can output the filtered team. If the process runtime variables do not affect the final team list, the input is optional – you can generate the list inside the team filter. When you select an integration service to implement the team filter service for your activity, the input mapping section appears, and you can map your process variables to the integration service inputs. The output of your team filter service is the team that the activity is assigned to.
- You can use a team filter service to dynamically prevent certain users from being assigned to an activity. The filtering can be based on any criteria and can use input parameters from relevant process variables to determine which users to filter out. You can implement the team filter service as either a service flow or a heritage integration service. To define the team filter service, you first define the input variables that the service flow or integration service receives with the input team object. Then you implement the service flow or integration service so that it eliminates ineligible users and returns a team object that contains the remaining users who can be assigned.

10.4. Experts team

Experts team

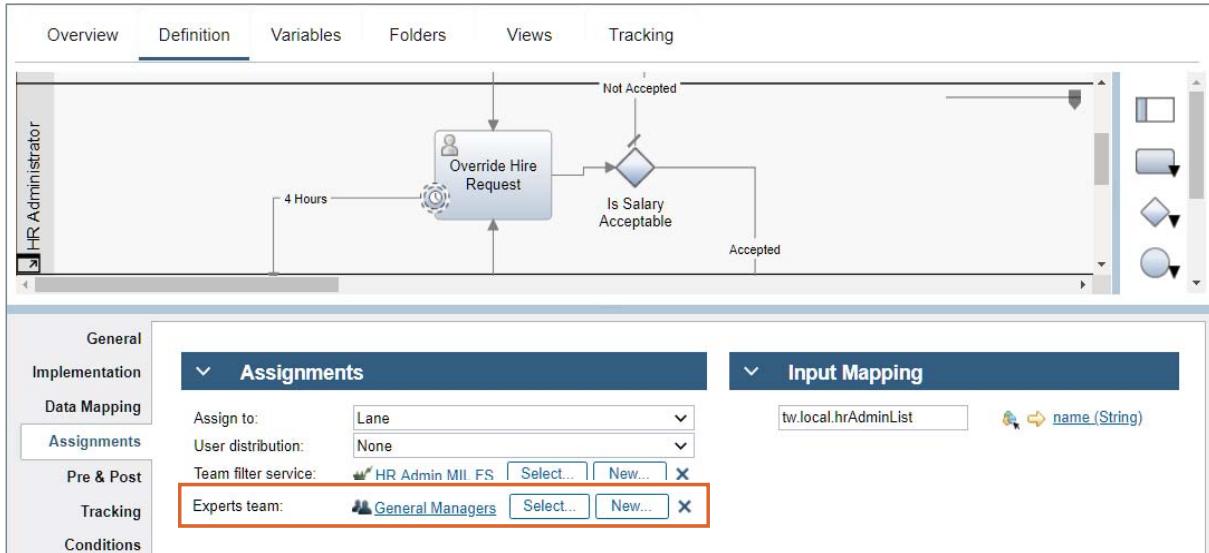
Advanced routing

© Copyright IBM Corporation 2020

Figure 10-15. Experts team

Designating an experts team

Business users that work with your process applications can collaborate or request assistance from expert users who are associated with a task or activity. The list of experts appears in the Experts panel in the Process Portal environment.



Advanced routing

© Copyright IBM Corporation 2020

Figure 10-16. Designating an experts team

Business users that work with your process applications can collaborate or request assistance from expert users who are associated with a task or activity. The list of experts appears in the Experts panel in the Process Portal environment. An activity must be associated with a human service before it can be assigned experts.

Each activity in Process Portal can list two types of experts:

- Users who completed this activity in the past, based on historical analysis. This list is limited to a small group of users who completed the activity most frequently.
- The second type of expert consists of users that belong to a team that is explicitly specified in IBM Process Designer as an expert group for this activity.

Procedure

To explicitly specify an expert group for an activity, open the business process diagram in Process Designer and select the activity. In the **Properties > Assignments > Assignments** section, specify the relevant team in the **Experts team** field. If you didn't already create a team that defines the experts for this task, you can create a team to use.

10.5. Four eyes

Four eyes

Advanced routing

© Copyright IBM Corporation 2020

Figure 10-17. Four eyes

Four eyes

- Team Filter services are used to filter certain team members from receiving a task.
- These are appropriate for many business policies that involve quality controls.
- Four eyes is an example of a business policy relevant to implementing Team Filter services.
- The four eyes principle requires that two people approve some activity.
- To implement the Four eyes policy, you need to know who the previous approver for the activity was, then filter that person out using Team Filter services.

Advanced routing

© Copyright IBM Corporation 2020

Figure 10-18. Four eyes

Four eyes is an example of a business policy relevant to implementing Team Filter services. The four eyes principle requires that two people approve some activity. In the exercise lab at the end of this unit, you implement Team Filter services to ensure that two people perform an approval activity.

Unit summary

- Explain user distribution in a process application
- Determine the best source for team data
- Model decision authority for a process
- Determine when to implement various types of routing
- Explain how to create a routing design by using a team filter service
- Explain when to use team filter services to support business policy

Advanced routing

© Copyright IBM Corporation 2020

Figure 10-19. Unit summary

Review questions

1. True or False: It is good to integrate your company organizational chart into your process model.
2. When should you use custom routing?
 - A. When you assign a task to a group that has many users
 - B. When assigning a task to the team assigned to the lane
 - C. When you want to assign a task to a list of `String()` variables that contains a list of user IDs
 - D. When the logic and algorithms are too complex for other types of routing
3. To implement the “four eyes” policy, you use what type of routing?
 - A. Custom
 - B. List of users
 - C. Team
 - D. Routing policy



Advanced routing

© Copyright IBM Corporation 2020

Figure 10-20. Review questions

Write your answers here:

- 1.
- 2.
- 3.
- 4.

Review answers (1 of 2)

1. True or False: It is good to integrate your company organizational chart into your process model.
The answer is True.

2. When do you use custom routing?
 - A. When you assign a task to a group that has many users
 - B. When assigning a task to the team assigned to the lane
 - C. When you want to assign a task to a list of `String()` variables that contains a list of user IDs
 - D. When the logic and algorithms are too complex for other types of routing
The answer is D. Custom routing allows developers to create a string to assign tasks to users or security groups.



Advanced routing

© Copyright IBM Corporation 2020

Figure 10-21. Review answers (1 of 2)

Review answers (2 of 2)

4. To implement the “four eyes” policy, you use what type of routing?
 - A. Custom
 - B. List of users
 - C. Team
 - D. Routing policy

The answer is D. Use a routing policy to implement the “four eyes” policy.



Exercise: Implementing a custom routing solution

Advanced routing

© Copyright IBM Corporation 2019

Figure 10-23. Exercise: Implementing a custom routing solution

Exercise introduction

- Implement a custom routing solution



Advanced routing

© Copyright IBM Corporation 2019

Figure 10-24. Exercise introduction

Exercise: Implementing the “four eyes” policy by using a team filter

Advanced routing

© Copyright IBM Corporation 2019

Figure 10-25. Exercise: Implementing the “four eyes” policy by using a team filter

Exercise introduction

- Implement the “four eyes” policy by using a team filter



Advanced routing

© Copyright IBM Corporation 2019

Figure 10-26. Exercise introduction

Unit 11. Managing complex tasks and process interactions

Estimated time

01:30

Overview

Process application interactions depend on comprehensive solutions to function correctly. Without the correct approach to implementing complex tasks and interactions, the business process becomes inefficient. This unit covers the methods that developers use to build effective complex tasks and interactions.

How you will check your progress

- Review
- Exercise

Unit objectives

- Manage parallel activity execution
- Implement a parallel task approval within a single process instance
- Manage messaging between processes
- Determine how to access data that is shared across multiple process activities
- Cancel a process at any time
- Determine when to use a multi-instance loop
- Implement multi-instance loops in IBM Business Automation Workflow
- Implement complex end conditions in a multi-instance loop

Topics

- Parallel tasks and messaging
- Multi-instance loops

Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

Figure 11-2. Topics

11.1. Parallel tasks and messaging

Parallel tasks and messaging

Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

Figure 11-3. Parallel tasks and messaging

Parallel tasks

- An implementation requirement that has more than one process activity that is accomplished at any time
- Four main points about parallel tasks:
 - Data flow
 - Sharing data among tasks
 - Variable number of tokens
 - Canceling parallel tasks

[Managing complex tasks and process interactions](#)

© Copyright IBM Corporation 2020

Figure 11-4. Parallel tasks

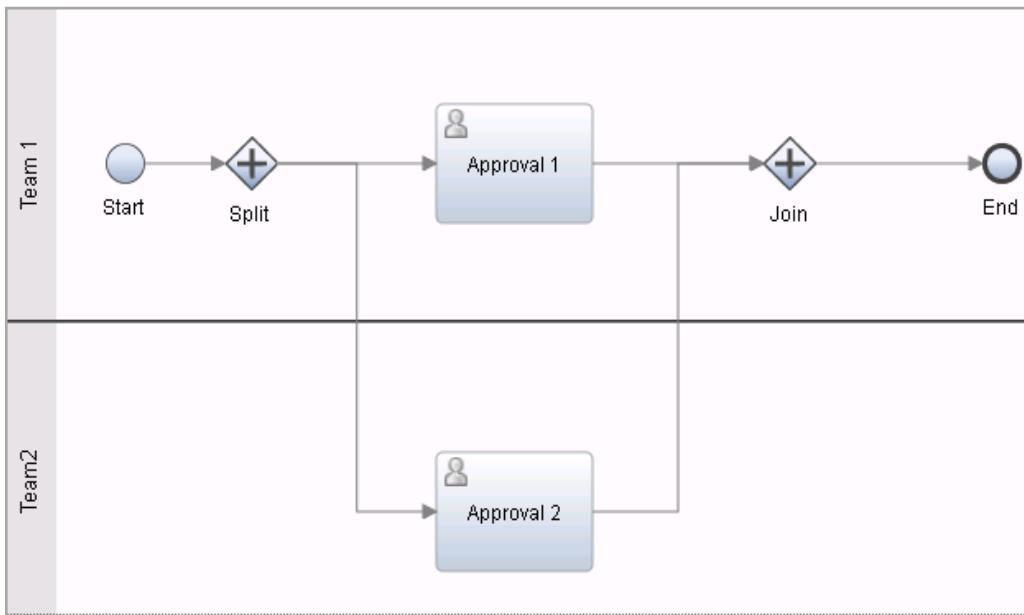
A common implementation requirement is to have more than one process activity that is accomplished at any time. Some interdependencies between these tasks might exist, so it is important to model your process to allow for both multi-participant task accomplishment and data sharing between each.

When attempting to model and implement parallel tasks, consider the following topics:

- Data flow
- Sharing data among tasks
- Variable number of tokens
- Canceling parallel tasks

Data flow

- With parallel tasks, output data flows out of both tasks
 - If you map the output variable for both tasks to the same complex business object, the task that completes last overwrites the data from all prior tasks



Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

Figure 11-5. Data flow

With parallel tasks, output data flows out of both tasks. If you map the output variable for both tasks to the same complex business object, the task that completes last overwrites the data from all prior tasks. To avoid this problem, do not map to the entire complex object as an input; map to the business objects inside the large complex business object. Consider what data might be overwritten when mapping the outputs. Some post-processing might be required to consolidate the input from the parallel activities to repopulate the complex object that is output from both activities. Using a list object that holds the numerous complex object outputs is one approach you can use to avoid losing data.

Sharing data among tasks

- Tasks do not automatically share data, nor do they listen for messages
- In many cases, it is necessary for parallel tasks to “see” what happened in other tasks, such as updated information or recommendations vital to the accomplishment of the task

Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

Figure 11-6. Sharing data among tasks

In many cases parallel tasks must “see” what happened in other tasks, such as updated information or recommendations vital to the accomplishment of the task. Remember, the data that is mapped as an input variable is set when the process token arrives at the task. Tasks do not automatically share data, nor do they listen for messages.

How variables are passed (1 of 2)

- When working with variables in IBM Business Automation Workflow, you must understand how variables are passed
- Several factors determine whether variables pass by value or pass by reference, as described in the following table:

From	To	Pass by
Process activity	Nested process	Value, if simple business object (variable type)
Process activity	Nested process	Reference, if complex business object (variable type)
Process activity	Service	Value
Service	Nested service	Value, if simple business object (variable type)
Service	Nested service	Reference, if complex business object (variable type)

Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

Figure 11-7. How variables are passed (1 of 2)

Also, when passing variables:

- If you pass a variable by value, the process or service that receives the value can manipulate it. The change does not affect the original value unless the receiving process or service returns the variable as an output.
- If you pass a variable by reference, the changes that are made by the process or service that receives the reference affect the original value. These changes affect the variable even if the receiving process or service does *not* return the variable as an output.

Because of the way IBM BAW handles variables, you should follow these guidelines:

- If the variable is a simple type, declare the variable as an input and an output in nested processes, services, and nested services.
- If the variable is a complex type, you must declare the variable as an input. Although the output declaration is not required (because you pass complex types by reference), it is a good idea to also declare the variable as an output. Creating the output variable ensures that other developers are aware that the nested process, service, or nested service returns a complex variable.
- Always use an identical name and data type for a set of input and output variables for data that is passed in, processed, and then passed back.

How variables are passed (2 of 2)

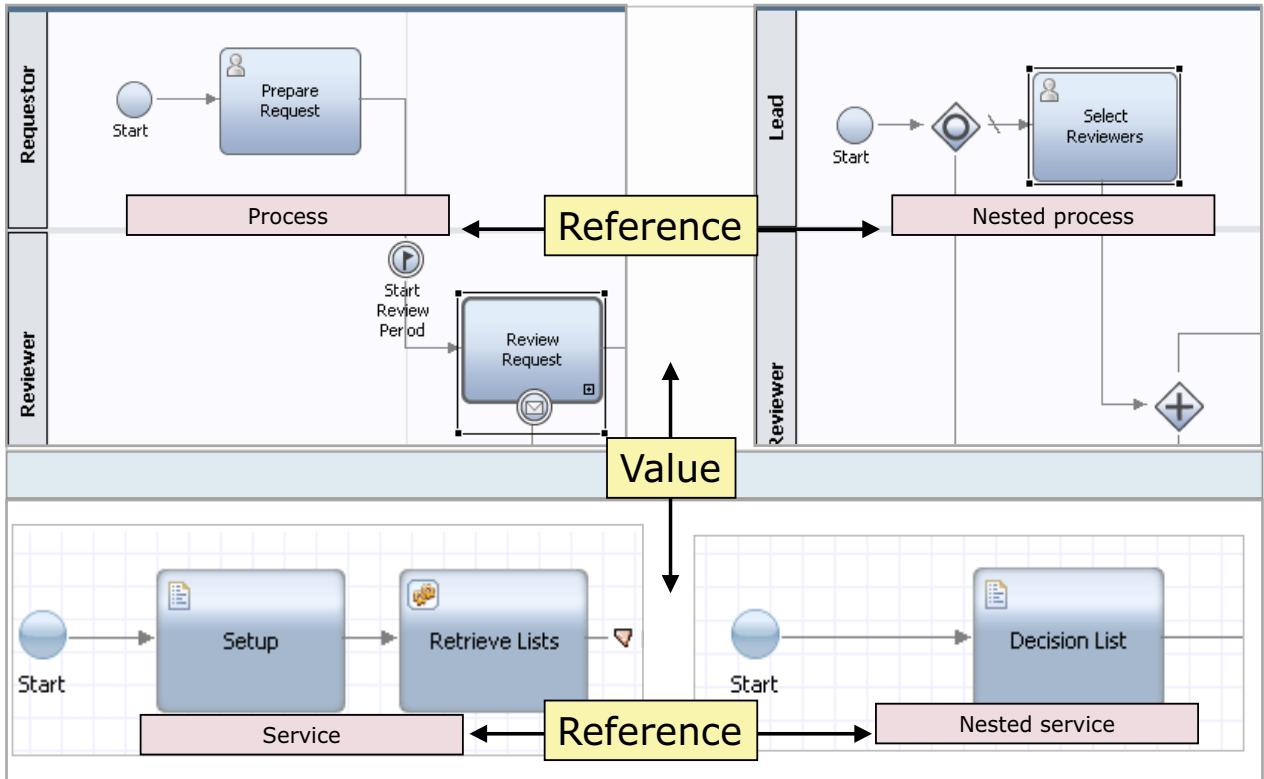
- Changes in the referenced complex object values are reflected at a nested level
- Pass by value creates a copy of the object, so changes are not reflected unless remapped back to the object
- Always map complex variable inputs as outputs
 - Even though you can pass the variable by reference, mapping variables as outputs increases maintainability and clarity

Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

Figure 11-8. How variables are passed (2 of 2)

Passing complex objects



Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

Figure 11-9. Passing complex objects

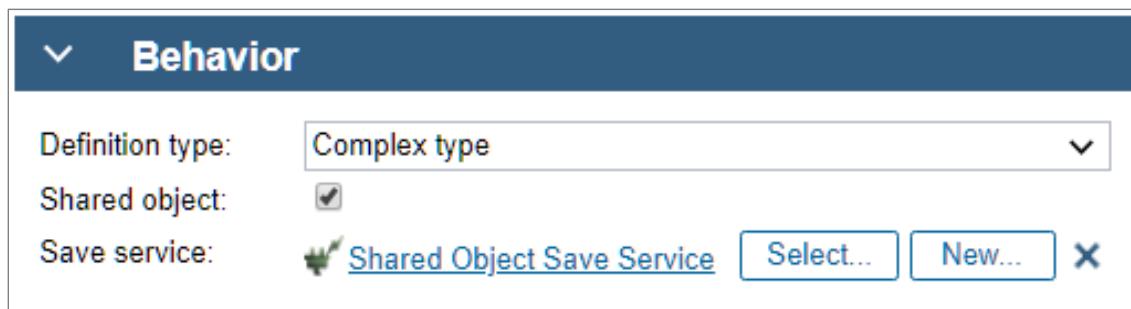
For example, a complex object is created at the highest level process. When the variable is mapped to a nested process, you pass the object by reference, so any changes made to the object in the nested process are reflected in the parent process.

When an activity creates a task, the service creates a copy of the complex object, and the values are passed to the service. Changes to this object are not reflected in the process unless the object is remapped to the parent object, or the object is designated as a shared object.

When the service passes the complex object to a nested service, you pass the object by reference, so any changes that are made at the nested service level are reflected at the parent service level.

Shared objects

- Using the shared object checkbox on the business object **General > Behavior** properties section designates the variable as shared
 - The values of the complex object are persisted automatically to a data store
 - At each process, service, or message event boundary, the local variables with the same object key are refreshed from the data store
 - The save service is a data validation service that is started after updates to the shared business object are merged and before they are saved



Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

Figure 11-10. Shared objects

Shared business objects apply only to a complex structure type. The data within a shared business object is shared between business processes and tasks.

Shared business objects allow concurrent modification. For example, two users can view and modify the same shared business object instance in a human service. When users trigger a boundary event, the data of the shared business object instance is saved. Only the fields that each user modifies are saved. Therefore, if two users modify different fields, both changes are saved. If both users modify the same field, the last user's update is saved. In addition to having multiple users, you can have a situation with automated steps, for example, a server script that modifies shared business objects.

You can send data from one process to a second process by using a message event or by using the unique key of the shared business object to load the data into the second process. To load the data, get the unique identifier key and then use the key to load the instance.

For example, in the following code, `sharedBusinessObjectKey` would be obtained by running:

```
tw.local.myVariable.metadata("key");
tw.local.myVariable =
new tw.object.mySharedBusinessObject(sharedBusinessObjectKey);
```

A shared business object uses database resources. The data within a shared object is persisted to the database and synchronized across all scopes when any of the following events occur:

- The shared object is created.
- The state of a process instance is persisted and automatic synchronization is enabled for the process.
 - If you use linked processes, at run time, the setting of the top-level process is taken.
- The state of a task or service instance is persisted and automatic synchronization is enabled for the service definition.
 - If you use nested services, the setting of the top-level server definition is taken.
- If automatic synchronization is not enabled for the process, or the service implementation does not support automatic synchronization, you must invoke the JavaScript save() method on the shared business object to persist the object data. You must use the load method to load the latest data from the data store into the variable.

Each shared object is logically connected to the business process instance that created it. When the business process instance is deleted (for example, when you delete a business process instance that was running in the Inspector), the shared object data in the database is also deleted. If you clear the **Shared Object** checkbox later, the business object is not accessible to other instances at run time. If the shared business object is created within a human service that can be started and not bound to a business process instance, the shared business object is connected to the activity task instance. In such cases, the shared business object is deleted if the task instance is deleted. If a shared business object is deleted from the database, the behavior of the tasks or processes that reference the shared business object is undefined.

If you want to create shared business objects that have a long lifetime, design a business process that acts as a factory (that is, it is based on a factory method pattern). The result is that your shared business objects remain in the database until the factory business process is deleted.

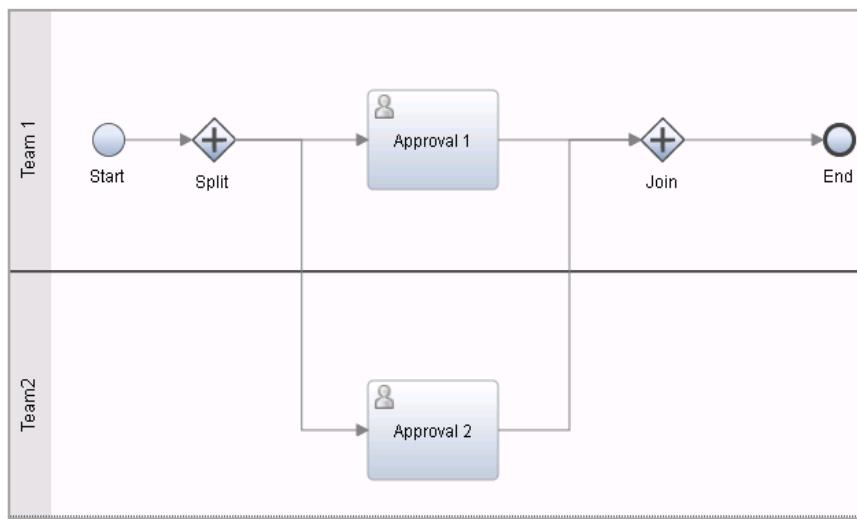
An output shared business object is for an external service, such as a web service, and an Advanced Integration service returns a new and typically updated copy of the original input shared business object. Therefore, if your application is working with an external service or an Advanced Integration service, your application should not reference the input shared business object, and you can expect an updated value. It should reference the new output shared business object.

The system incurs a performance cost for designating variables as shared, so you should not designate objects as shared unless those variables must be accessed from another step inside or outside the process.

The save service is a data validation service that is invoked after updates to the shared business object are merged and before they are saved. The service must have three input parameters of this shared business object type. One parameter of type `BPMBOPropertyChange` (`List`), and one output parameter of type `BPMBOValidationErrors` (`List`) that indicates whether the merged data is valid.

Variable number of tokens

- As tokens are created in the process, all tokens must run to the end to allow the process to complete
- With parallel tasks, it is inevitable to have a variable number of tokens that flow through your process
- A conditional join might be necessary to consume all the tokens



Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

Figure 11-11. Variable number of tokens

As tokens are created in the process, all tokens must run to the end to allow the process to complete. With parallel tasks, it is inevitable to have a variable number of tokens that flow through your process. A conditional join might be necessary to consume all the tokens.

Canceling parallel tasks

- The dynamics of a parallel task often allow for certain tasks to bring a process or the parallel tasks to a stop
 - The process requirements and implemented rules dictate how and when the process would be canceled or closed
- To cancel a parallel task, consider how the cancellation affects all the process activities or tokens when:
 - Canceling and enclosing the process
 - Canceling only the parallel tasks in question with a unique ID (for example, Task ID)
 - Sending a cancel message

Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

Figure 11-12. Canceling parallel tasks

Messaging patterns

- Message events are used to represent a point in your process where an incoming message is received
- Processes listen for these messages, which can originate from multiple sources:
 - An internal web service message
 - A message posted to the JMS listener
 - Calling an undercover agent (UCA) in a service
- These messages trigger events such as start or cancel a process
 - The message can trigger an event at the process or nested process level

Figure 11-13. Messaging patterns

Message listeners are an important component of your process design. Messages can trigger responses that are based on the message payload, and they are a key component to managing the process and data flow. You should realize that the message itself can originate from multiple sources at any time in your process. When creating an Intermediate Message Event (IME), you can set two options:

- The **Consume** option is used to determine whether the IME consumes an incoming message. If consumed, the message is not available to any other message events in the process instance. If it is not consumed, other IMEs in the process can also receive the same message.

Message events

- The message start event functions as a start event based on a received message in your process
- In contrast to the message start event, message intermediate events can be placed anywhere in the flow of your process
- Message intermediate events are commonly used to:
 - Cancel a task
 - Restart a task
 - Update a process state
 - Cancel a process
 - Stop or resume the flow of a process

Parallel tasks and messaging

- All message intermediate events need a correlation ID
 - If you have parallel tasks, the process instance ID is not enough
- A common scenario is a parallel task process where a rejection within one task requires the termination of all the other incomplete tasks
- For a parallel task, you must implement a good pattern to accomplish the cancellation
- Canceling a process at any time requires cleanup steps that might include a notification to process participants that the process is canceled

Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

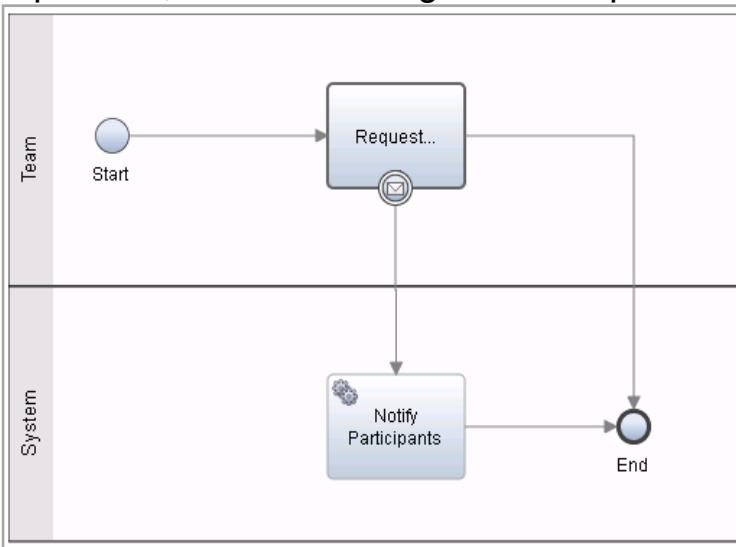
Figure 11-15. Parallel tasks and messaging

Often you must cancel a task or even an entire process instance. Message intermediate events are useful for this purpose.

When you create parallel process tasks, it is common for those tasks to influence each other. A common scenario is a parallel task process where a rejection within one task requires the termination of all the other incomplete tasks. Canceling tasks would not be necessary in a multi-instance loop because you might use an end condition. However, for a parallel task, you must implement a good pattern to accomplish the cancellation. Remember, canceling a process at any time might require cleanup steps that include a notification to process participants that the process is canceled.

Nested process cancellation pattern

- To cancel a process, create a higher-level process and place the nested process on the canvas with a single attached message intermediate event
- This approach replaces any terminate events that might be in the process, and allows elegant cleanup of tasks after the cancellation



- Through user education, business users must understand that this high-level process is the cancellation pattern, and the implementation details are found in the nested process

[Managing complex tasks and process interactions](#)

© Copyright IBM Corporation 2020

Figure 11-16. Nested process cancellation pattern

To use this cancellation pattern, wrap your entire process in a nested process and attach a message intermediate event to listen for a cancellation at the highest level activity. This approach replaces any terminate events that might be in the process, and allows elegant cleanup of tasks after the cancellation.

Using this design pattern, most of the process is not apparent to participants who view the process because the top-level process does not show all the details; it shows only the administrative cancellation implementation details. Through user education, business users must understand that the top-level process enables the cancellation pattern, and the implementation details are found in the nested process.

Use this pattern to start any new top-level process. Even if a cancellation is not part of the requirement, include this simple pattern for all processes because of its simplicity. This pattern allows developers to use a top-level cancellation if they encounter a situation where they might need this feature as they complete their process implementation.

11.2. Multi-instance loops

Multi-instance loops

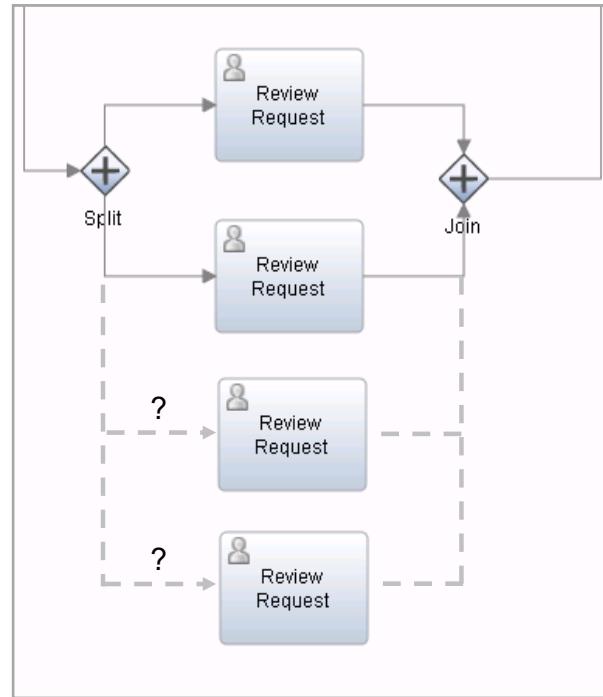
Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

Figure 11-17. Multi-instance loops

Multi-instance loops (MILs)

- How do you model a loop when a variable number of tasks are needed at run time based on business data?
- For what other scenarios would you use a multi-instance loop?



Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

Figure 11-18. Multi-instance loops (MILs)

You saw a pattern to handle parallel tasks in a Process, but how do you model a loop when a variable number of tasks are needed at run time based on business data? This requirement can be fulfilled with a multi-instance loop. The difference between simple looping and multi-instance looping is simple: looping tasks are created and run serially, while multi-instance loops are created and run in parallel.

An example of a multi-instance loop implementation is a loan approval task where multiple signature authorities complete tasks in parallel or in sequence before the process activity “approve the loan” can be considered complete. Multi-instance loops create one token for each instance where simple loops create only one task for all instances.

The trick to getting multi-instance loops to work is to understand how to assign those task instances to multiple users and how to get the data returned to the process. You need a list to hold the users who might complete the activity to get the task to the right person. Most likely, you need special routing to assign tasks to the right person or group. You also need a list to hold the results that come back from the users, similar to creating a simple split to create parallel tasks.

The system variable `tw.system.step.counter` is the index into those lists. The scope of this system variable is limited to the Process level and does not map to the service level. If you plan to use the step variable before, during, or after your loop, you must manually map the step variable to a local variable in your service.

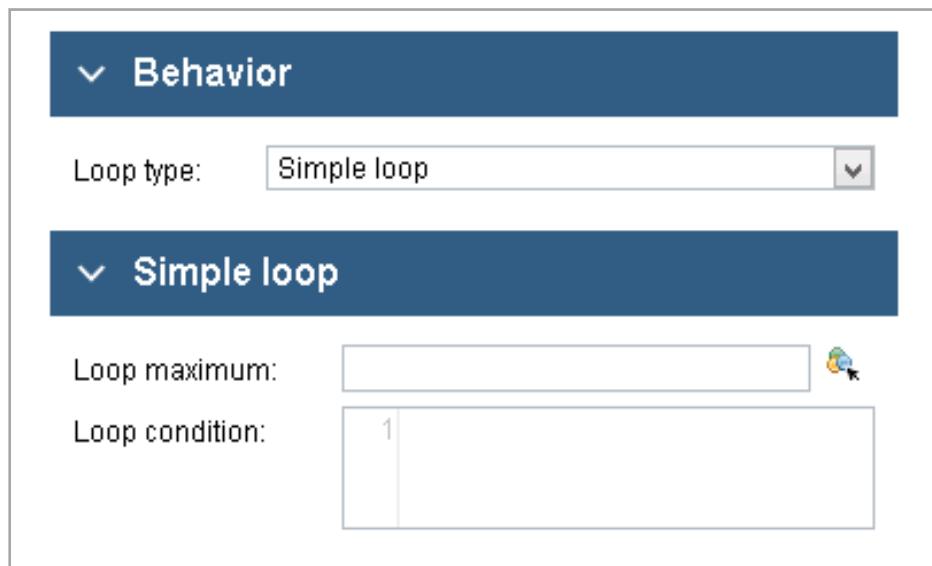
MIL tips:

- Use a list to hold the users who might be assigned the activity to get the task to the right person.
- Most likely, you need special routing to assign tasks to the right person or group.
- You also need a list to hold the results that come back from the users, similar to creating a simple split to create parallel tasks.
- The system variable `tw.system.step.counter` functions as your index into those lists. If you plan to use the step variable before, during, or after your loop, you must manually map the variable to a local variable in your service.

MIL stands for multi-instance loop throughout the rest of this unit.

Simple loop configuration

- Simple loops: Tasks are created and performed serially



Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

Figure 11-19. Simple loop configuration

- In a simple loop, the loop completes when all the tasks that are created complete the loop.
- The loop can implement many different types of artifacts to include subprocesses, linked processes, and simple activities.

Multi-instance loop configuration options

- Multi-instance loops: Greater control over your instances
- Ordering: Run sequentially or in parallel

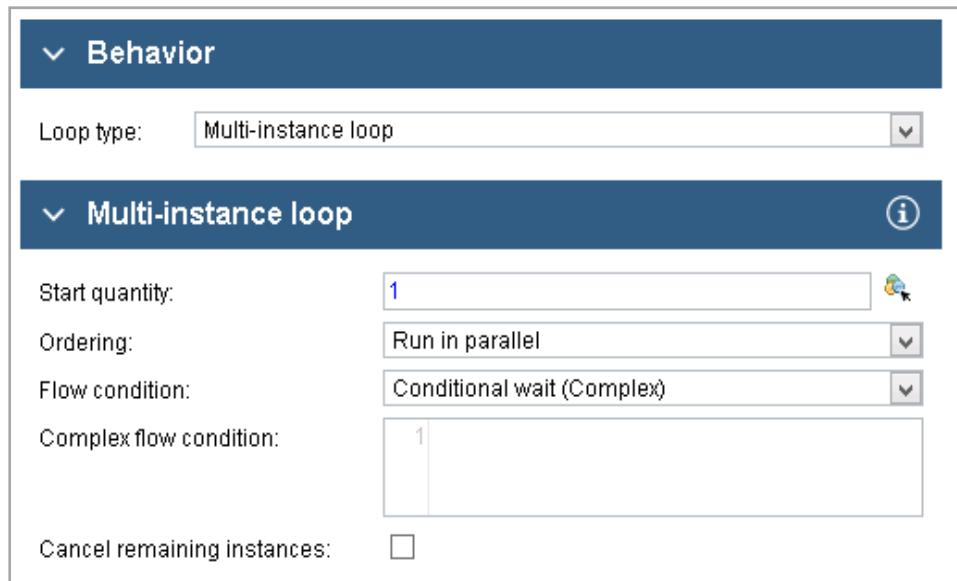


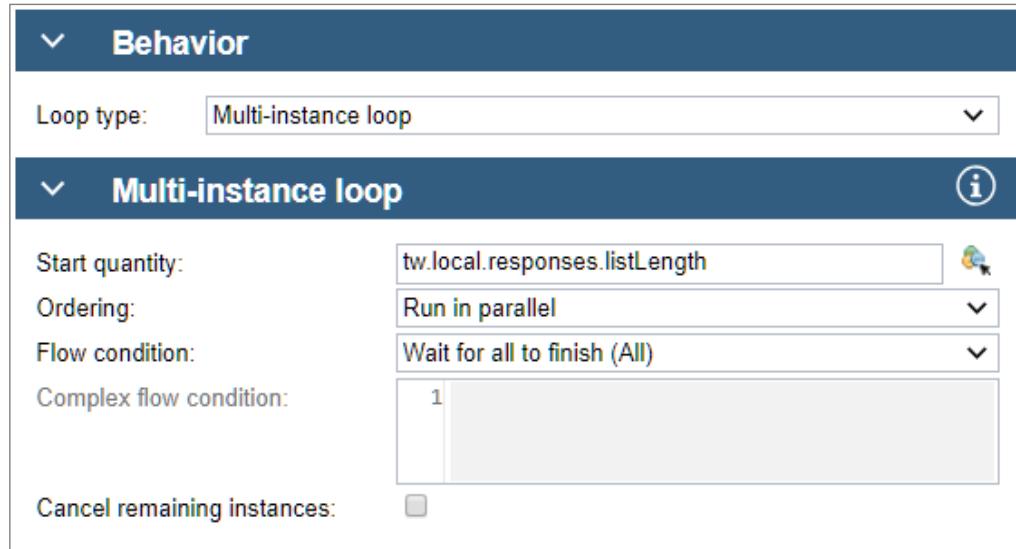
Figure 11-20. Multi-instance loop configuration options

If the loop requires a condition to determine whether the loop should continue, use a multi-instance loop type. A multi-instance loop gives you more control over the instances you create with the loop. The different types of multi-instance loops:

- Sequential: Each task is created after the last one completes
- Parallel: All instances are created when the loop is created

Setting up a multi-instance loop (1 of 2)

- The pattern for setting up a multi-instance loop:
 - Choose the loop type to be a multi-instance loop
 - This example sets the number of tasks created (start quantity) to the number of responses that are stored in a list (array)
 - Choose to run in parallel or serial



Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

Figure 11-21. Setting up a multi-instance loop (1 of 2)

Set up the multi-instance loop on the step tab of the activity. You can choose to run tasks in parallel or run serially, but the true advantage to the multi-instance loop is realized when you run tasks in parallel. On the flow condition, you can either wait for all to finish, or set up a condition so that every time one activity finishes, it checks whether an exit criteria resolves to true. You can use the checkbox to cancel any of the remaining instances when the flow condition resolves to true, and the multi-instance loop can release the token down the path.

Setting up a multi-instance loop (2 of 2)

- Assign a team filter for the lane participants



- Sharing data with parallel tasks

- If the order of the output list must remain intact, use this pattern



- Otherwise, use an initialized list variable without elements, and add to the end of the list



Managing complex tasks and process interactions

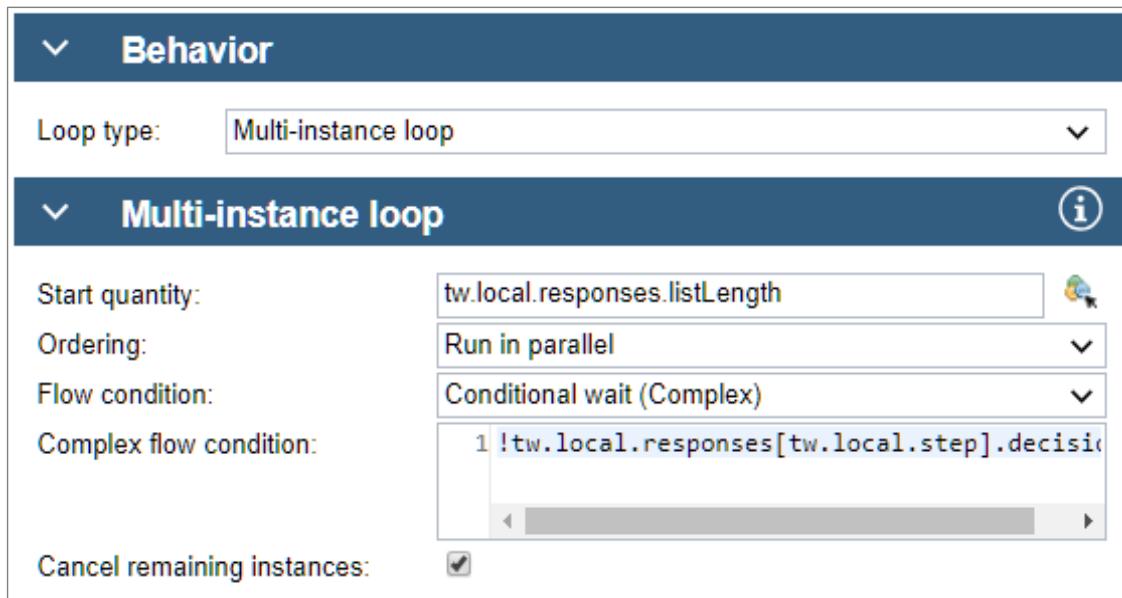
© Copyright IBM Corporation 2020

Figure 11-22. Setting up a multi-instance loop (2 of 2)

- Many times, you must route your tasks to individual users. For example, you might assign the tasks to a team, and it creates four tasks for the group. When members of the group view their inbox, they see all four tasks in their inbox.
- Use a team filter service to route the tasks that are created from the multi-instance loop. For each task in the multi-instance loop, your team filter must output only the user or users to assign the task. The team filter allows input variables to create the resulting team.
- Set up your variables so they store the output correctly, and they do not overwrite one another. Two common ways exist to map the output data. The first is to send into the activity the `tw.system.step.counter` variable. When the step is sent out as an output, use the step as the array indexes for the variables you are trying to map to. If the order of the output variable array is important, use this approach.
- The other approach is to use the list length and save the output to the end of the array. However, with this approach, you cannot keep the order of the array intact. This approach provides only the order of activities that were completed as part of the loop.

Multi-instance loop end conditions

- Use a multi-instance loop end condition when you want to specify what the multi-instance loop should do when an instance completes
 - For example, if you want the loop to terminate after one decision returns false, use a JavaScript condition to express this rule



Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

Figure 11-23. Multi-instance loop end conditions

The flow condition of the multi-instance loop is powerful. It is the only mechanism that makes it easy for you to examine the process state as soon as a parallel activity completes. You can then determine whether to continue the remainder of those activities. If you cannot use a multi-instance loop, you must set up message listeners on all of the parallel activities to cancel the remaining instances. You would also need a conditional join.

The flow condition provides a simple way to create end conditions that are associated with parallel tasks. When the flow condition resolves to true, the loop terminates and the token moves out of the multi-instance loop and down the flow. If you require the cancellation of all remaining tokens, select the **Cancel Remaining Instances** checkbox.

Multi-instance loops and performance

- Never place a system activity as the first activity of a MIL
 - System lane activities lock an event manager engine thread until the activity is complete, so creating multiple instances of the loop can cause deadlock in the system until these activities complete
 - If possible, do the system lane activities that are required before the MIL
- Multi-instance loops create N number of tokens since each nested process that is created gets a new token
- Be wary of exit criteria: The MIL can create more tokens than what the process model can complete and thus cause a loop that never completes

Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

Figure 11-24. Multi-instance loops and performance

Consider a few things when you implement a multi-instance loop. If you are enclosing a subprocess because they are serially processed, as mentioned before, the first activity should not be a system activity. Use a user task as the first activity of a nested process, and serialization is not a problem. Also, a multi-instance loop creates as many tokens as you have instances of the activity. It is possible to create so many tokens that the process would never complete.

Unit summary

- Manage parallel activity execution
- Implement a parallel task approval within a single process instance
- Manage messaging between processes
- Determine how to access data that is shared across multiple process activities
- Cancel a process at any time
- Determine when to use a multi-instance loop
- Implement multi-instance loops in IBM Business Automation Workflow
- Implement complex end conditions in a multi-instance loop

[Managing complex tasks and process interactions](#)

© Copyright IBM Corporation 2020

Figure 11-25. Unit summary

Review questions

1. What type of variable gets passed by reference from a process to a subprocess?
 - A. Simple
 - B. Complex
 - C. List of String()
 - D. Dates

2. True or False: If two parallel activities map their output to the same complex business object, you must use an external system of record to ensure that the variable is not overwritten.

3. True or False: Complex objects pass by value when passed from the process level to the service level.



Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

Figure 11-26. Review questions

Write your answers here:

- 1.
- 2.
- 3.

Review answers

1. B. You pass complex variables by reference from process to subprocess and service to subservice.
2. **False.** Select the **Shared Object** checkbox on the business object to avoid overwriting output variables and use the `save()` method to persist the object.
3. True



Managing complex tasks and process interactions

© Copyright IBM Corporation 2020

Figure 11-27. Review answers

Exercise: Building a cancellation pattern

Managing complex tasks and process interactions

© Copyright IBM Corporation 2019

Figure 11-28. Exercise: Building a cancellation pattern

Exercise introduction

- Implement a cancellation pattern in a process application
- Implement an undercover agent (UCA) to cancel the hiring request



Managing complex tasks and process interactions

© Copyright IBM Corporation 2019

Figure 11-29. Exercise introduction

Unit 12. Integrating with external systems

Estimated time

01:00

Overview

Integrations with external systems are accomplished through integration service flows in IBM Business Automation Workflow. This unit covers the integration service flows that are used to connect to other systems for increased effectiveness of the business process application.

How you will check your progress

- Review
- Exercise

Unit objectives

- Explain how to build services in IBM Business Automation Workflow to integrate with external systems
- Describe the outbound web service integrations and the most common issues that generate connection complexities
- Use the System Data toolkit services when other solutions for external system connection are needed
- Create an inbound web service integration
- Create an event-based undercover agent

Integrating with external systems

© Copyright IBM Corporation 2020

Figure 12-1. Unit objectives

Key concepts in this unit

- **HTTP:** Hypertext Transfer Protocol: a protocol used to transfer data over the World Wide Web.
- **JAR:** Java ARChive: a compressed package file format used to aggregate many Java class files and resources into one file for distribution.
- **JavaBeans:** classes that encapsulate many objects into a single object (bean).
- **SOAP:** Simple Object Access Protocol: Messaging protocol specification for exchanging information over web services.
- **URI:** Uniform Resource Identifier: identifies a resource on a computer network
- **WSDL:** Web Service Definition Language: XML format for describing the functionality offered by a web service.

Topics

- Building outbound integrations
- System toolkit services
- Building inbound integrations

Integrating with external systems

© Copyright IBM Corporation 2020

Figure 12-3. Topics

12.1. Building outbound integrations

Building outbound integrations

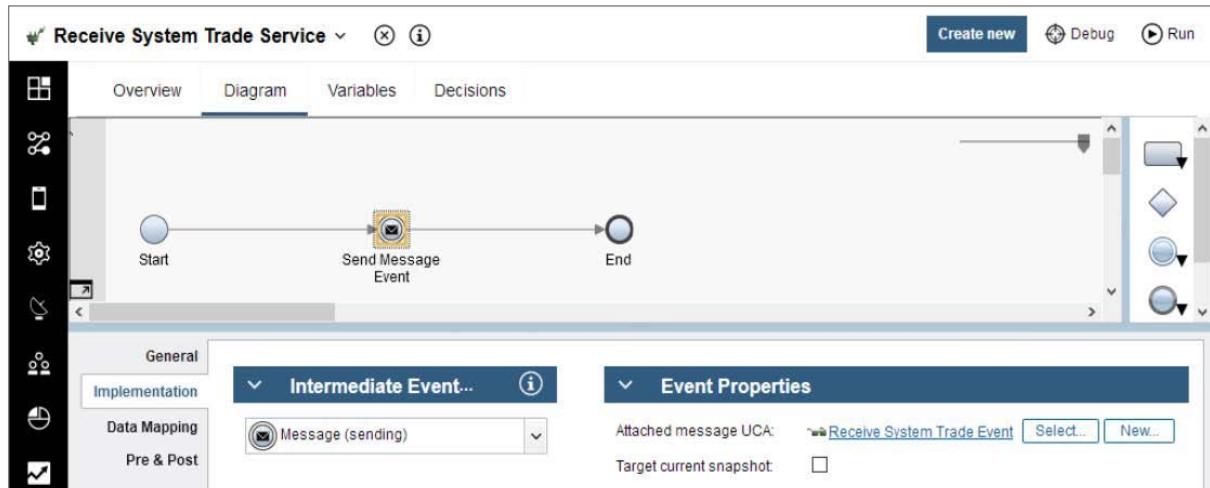
Integrating with external systems

© Copyright IBM Corporation 2020

Figure 12-4. Building outbound integrations

Building an outbound integration

- Most outbound integrations (connection with external systems) can be modeled with the artifacts that are found in the IBM Process Designer service library palette
 - Both the Java and web service integration components can be found here



Integrating with external systems

© Copyright IBM Corporation 2020

Figure 12-5. Building an outbound integration

- Most outbound integrations (connections with external systems) can be modeled with the artifacts that are found in the Process Designer library palette. Both the Java and web service integration components can be found there.
- The web service integration component uses a WSDL discovery SOAP-based web service invocation method to connect to your external web services. This function generates the variable type and auto-maps inputs and outputs.
- However, some WSDLs do not always work well for integrating with other systems, or are not compatible with the web service connection. When you encounter this problem, use a system toolkit service flow with a prebuilt integration or a Java integration.

Integration types

- To accomplish the appropriate integration with an external web service, it is important to have a good understanding of the process implementation requirements
 - Try the built-in services first (web service integration). For the supported WSDLs, it is remarkably quick and easy
 - If a particular WSDL is not supported or more control is needed over a particular request, try the **Call WebService via SOAP** service
 - If all else fails, create a Java connection of your own that uses either SOAP or HTTP protocols

Integrating with external systems

© Copyright IBM Corporation 2020

Figure 12-6. Integration types

- When you are ready to connect an external system, start by creating a service flow. On the service flow palette, you find a web service connector and a Java connector to accomplish your integration. During the upcoming exercise, you use the WSDL discovery method.
- The alternative is to use the Java connection options by using custom JAR files that are created with the custom Java connector class. Understand that all integration connections are Java connections, so the Java option allows for custom connection types when needed.

Java integration

- All integration connections are Java connections, so the Java option allows for custom connection types when needed
- Any static API available in Java can be accessed with a Java integration that is based on your Java connector class
 - Services that are based on a Java component require a Java class found in custom-built Java connectors to work

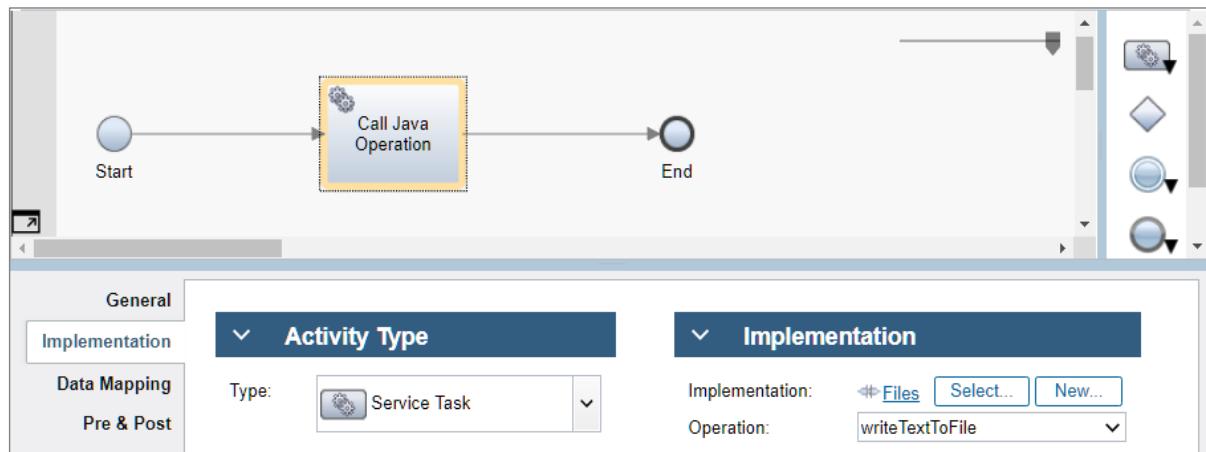


Figure 12-7. Java integration

- Any static API available in Java can be accessed with a Java integration that is based on your Java connector class, as it provides great flexibility. Services that are based on a Java component require a Java class found in custom-built Java connectors to work.
- You can use the Java connector to attach a custom JAR file directly into a process application or toolkit. JAR files have versions just like other assets in the process application or toolkit and are installed alongside the model when installing a process application to a server. If the Java classes are later updated, the version can be updated, and the classes are refreshed in a controlled manner to ensure that developers control the governance over the upgrade path. Developers can test the new JAR file and avoid regression errors in the existing process application.
- If you want the result of the serialized Java method to return to the service as an XML element, select the **Translate JavaBeans** checkbox. The content of the element is based on the properties of the object class. When you select the **Translate JavaBeans** checkbox, the variable type that you select in the service for the value that is returned from the Java method must be `XMLELEMENT` or `ANY`.

Web service integration

- An outbound integration is accomplished through an integration connection that accesses a SOAP web service
- For protected WSDL authentication, you can select the **Protected WSDL** checkbox in the implementation properties for the web service integration component, and then enter the username and password in the provided fields

Integrating with external systems

© Copyright IBM Corporation 2020

Figure 12-8. Web service integration

An outbound integration is accomplished through an integration connection that accesses a SOAP web service. Use a service flow when integrating to an external system. The web service integration provides a Web Services Description Language (WSDL) Uniform Resource Identifier (URI) discovery feature that automatically lists all the advertised WSDL operations. The advantage of the WSDL connection is the discovery of available operations. This feature allows developers to create variable types for each operation automatically. The disadvantage is that not all WSDL definitions can be read properly.

SOAP messages are exchanged in a request/response format. When the system sends a request to a web service, the web service returns the requested values. These values are specified in a SOAP message, which is a block of XML code that contains several elements.

All SOAP messages must contain a SOAP envelope element, which identifies the XML code as a SOAP message. Some WSDLs require that SOAP headers also be passed with each request. A SOAP header is an element in a SOAP message that is contained in the SOAP envelope message and provides detailed information about the SOAP message. IBM Business Automation Workflow supports passing a SOAP header with each request for a WSDL SOAP operation that requires it. The two types of SOAP headers are: headers that are directly specified as part of the SOAP binding, and headers that are not.

IBM Business Automation Workflow supports the following runtime authentication mechanisms that do not require client certificates:

- Protected WSDL
- HTTP basic authentication (described in RFC 2627)
- Username Token authentication

For protected WSDL authentication, you can select the **Protected WSDL** checkbox in the implementation properties for the web service integration component, and then provide the username and password in the provided fields.

Connecting to a web service

Consider:

- If the WSDL Discovery does not work, test for WS-I compliance first
- If the WSDL Discovery does not pull back the operations that are needed to match the business requirement, test the endpoint with a third-party tool like SOAP UI
- If you require Protected WSDL security, ensure that the web service has the right connection settings
- If endpoints change depending on the different environments (development and production), use an environment variable to store the WSDL URI and all the other settings that change between environments

Integrating with external systems

© Copyright IBM Corporation 2020

Figure 12-9. Connecting to a web service

The slide shows common questions to ask when WSDL discovery does not work. Later in the unit, you learn how to mitigate these issues.

Check WSDL for WS-I compliance with the soapUI WS-I compliance test:

<http://www.ibm.com/support/docview.wss?uid=swg21439805>

12.2. System toolkit services

System toolkit services

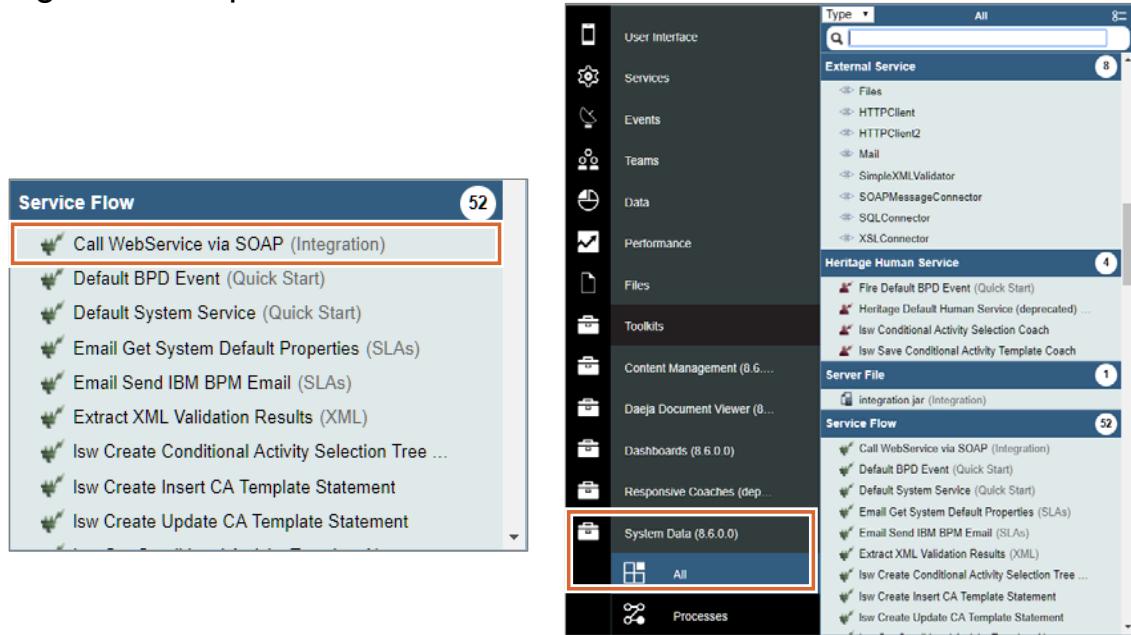
Integrating with external systems

© Copyright IBM Corporation 2020

Figure 12-10. System toolkit services

System toolkit services

- The System Data toolkit includes a service flow, **Call WebService via SOAP**, which provides a different function than the web service integration component



Integrating with external systems

© Copyright IBM Corporation 2020

Figure 12-11. System toolkit services

- A number of prebuilt services are provided in the system toolkit. Search through the service options for unique SOAP or HTTP protocol services. Special connections of this type are initiated through the system toolkit implementation library.
- The SOAP service type generation or automatic mapping from XML to business objects does not occur when you use this approach. You must create the variables manually from the published WSDL.
- The System Data toolkit also includes a service for dealing with HTTP called Read from HTTP. The HTTP service is also a Java integration component that uses Java classes and HTTP methods to invoke the connection and return the data in a structured form such as REST.

Guidelines (1 of 2)

- Advantages to the custom SOAP service
 - If the target endpoint is different for testing and production, you can define the endpoint of the web service call at run time, which can make it easier to promote to different environments
 - Provides complete control in both directions
 - After it is created, the same connection can be reused for many integrations
- Disadvantage of the SOAP service
 - Requires knowledge of SOAP envelopes and translation between business objects and XML

Integrating with external systems

© Copyright IBM Corporation 2020

Figure 12-12. Guidelines (1 of 2)

It is important to understand the differences between creating a SOAP service and the HTTP service. While both are used to integrate with external systems, both have their advantages and disadvantages.

Guidelines (2 of 2)

- Advantages of the HTTP service
 - Maximizes the use of pre-existing, well-defined HTTP features such as caching and security enforcement
 - Minimizes adding application features that must be included in other HTTP protocol web services, such as SOAP
 - Returns XML or some other “well-known” entity type in response to simple HTTP method queries
 - IBM Business Automation Workflow can use XML easily through the XMLElement variable type
- Disadvantage of the HTTP service
 - If mapping between XML and other variable types is necessary, it must be done manually

12.3. Building inbound integrations

Building inbound integrations

Integrating with external systems

© Copyright IBM Corporation 2020

Figure 12-14. Building inbound integrations

Building inbound integrations

- Creating an inbound web service is similar to creating any other artifact in the library, but when a message is received, the system must react
- Many times these inbound messages are intended to affect the flow of a process
 - Associate the inbound web service with a UCA and a receive message event on a process

Integrating with external systems

© Copyright IBM Corporation 2020

Figure 12-15. Building inbound integrations

IBM Business Automation Workflow can publish web services in the same way that it connects to web services. Using a SOAP connection, external applications can call the IBM Business Automation Workflow web service to initiate a particular process, service, or set of services.

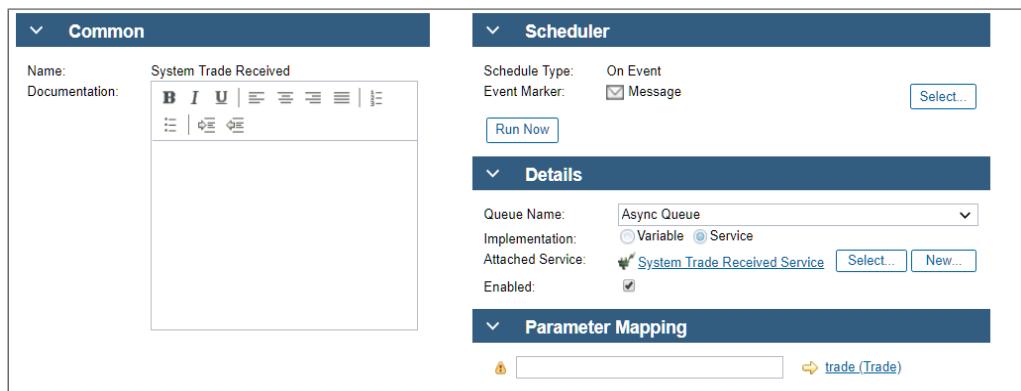
Creating an inbound web service is similar to creating any other artifact in the library, but when a message is received, the system must react. Many times the inbound messages are intended to affect the flow of a process. You can affect the flow by associating the inbound web service with a UCA and a receive message event on a Process.

Until now, UCAs were created but were only triggered internally through test harnesses or internal services. UCAs exposed to external systems by an inbound web service accomplish the same task, but can now be triggered from external systems.

Inbound web services are not always required to use a UCA (stand-alone services can be associated with the web service if needed). But to affect a process, use a UCA with a message start event or message intermediate event on a Process.

Event-based undercover agents (UCA)

- A UCA is a listener that waits to receive a message from an event
 - When a UCA is triggered, it starts a service in response to the event, and that service defines the inputs and outputs for the UCA
 - By creating a caller service with the UCA inside of it, the service can be run to trigger the UCA; the caller service must be used as the attached service for an inbound web service
 - When the inbound web service is started with the caller service-defined inputs, the UCA is triggered, and the message is sent to the listeners defined in the processes



Integrating with external systems

© Copyright IBM Corporation 2020

Figure 12-16. Event-based undercover agents (UCA)

- Undercover agents (UCAs) are a feature of IBM Business Automation Workflow that can be complicated to understand. To simplify, a UCA is a listener that waits to receive a message from an event. A message might trigger these events, or they might be triggered on a specific schedule (time based). When a UCA is triggered, it starts a service in response to the event, and that service defines the inputs and outputs for the UCA. By creating a caller service with the UCA inside it, the service can be run to trigger the UCA. The caller service must be used as the attached service for an inbound web service. When the inbound web service is started with the caller service-defined inputs, the UCA is triggered, and the message is sent to the listeners defined in the process.
- Use the Variable implementation to define the input variable and pass it on to the listener.
- If you need to modify the incoming variable before the data is sent to the listener, use the Service implementation. This action might require a database lookup, stripping of variables that are not needed for the listener, or something else that requires changing the incoming data.

Unit summary

- Explain how to build services in IBM Business Automation Workflow to integrate with external systems
- Describe the outbound web service integrations and the most common issues that generate connection complexities
- Use the System Data toolkit services when other solutions for external system connection are needed
- Create an inbound web service integration
- Create an event-based undercover agent

Integrating with external systems

© Copyright IBM Corporation 2020

Figure 12-17. Unit summary

Review questions

1. True or False: IBM Business Automation Workflow supports protected WSDL authentication.
2. If the WSDL discovery does not work, what is the first thing you must check?
 - A. WS-I compliance
 - B. Network bandwidth capacity
 - C. Updates to Java on the server
 - D. The server hydraulic fluid level
3. A UCA enabling service defines what?
 - A. The input variables of the UCA
 - B. The output variables of the UCA
 - C. When the UCA is triggered, the service that is executed
 - D. All of the above



Integrating with external systems

© Copyright IBM Corporation 2020

Figure 12-18. Review questions

Review answers

1. True or False: IBM Business Automation Workflow supports protected WSDL authentication.
The answer is True. 
2. If the WSDL discovery does not work, what is the first thing you must check?
 - A. WS-I compliance
 - B. Network bandwidth capacity
 - C. Updates to Java on the server
 - D. The server hydraulic fluid level
 The answer is A. Always check WS-I compliance on all web service endpoints.
3. A UCA enabling service defines what?
 - A. The input variables of the UCA
 - B. The output variables of the UCA
 - C. When the UCA is triggered, the service that is executed
 - D. All of the above
 The answer is D. The enabling service is the service that is run when the UCA is triggered. The service defines the inputs and outputs of the UCA.

Exercise: Building web service connections

Integrating with external systems

© Copyright IBM Corporation 2019

Figure 12-20. Exercise: Building web service connections

Exercise introduction

- Create an event-based undercover agent
- Build an inbound web service connection
- Build an outbound web service to message the inbound web service



Integrating with external systems

© Copyright IBM Corporation 2019

Figure 12-21. Exercise introduction

Unit 13. Handling content events in a process

Estimated time

01:00

Overview

In this exercise, you learn how to use the CMIS capabilities in IBM Business Automation Workflow.

How you will check your progress

- Review
- Exercise

Unit objectives

- Use the CMIS capabilities of IBM Business Automation Workfloww
- Explain how to handle content events in a process
- Describe how to add a document to the BPM document store
- Describe how to add a document to a Case solution
- Understand how the Target Object Store (TOS) can be used to share documents between a BPM solution and a Case solution
- Understand how to build a simple Case solution that integrates a process from a BPM solution

Handling content events in a process

© Copyright IBM Corporation 2020

Figure 13-1. Unit objectives

Topics

- CMIS system integration
- Document controls
- Content events
- Case integration

Handling content events in a process

© Copyright IBM Corporation 2020

Figure 13-2. Topics

Key concepts in this unit

- **CMIS:** Content Management Interoperability - Services enables applications and clients that use the OASIS CMIS standard to access content that is stored on Content Platform Engine
- **ECM:** Enterprise Content Management - ECM tools allow the management of an organization's information.
- **TOS:** Target Object Store - Where Case solution definitions are deployed. The data for the cases that are created is also stored in this object store.

Handling content events in a process

© Copyright IBM Corporation 2020

Figure 13-3. Key concepts in this unit

13.1. CMIS system integration

CMIS system integration

Handling content events in a process

© Copyright IBM Corporation 2020

Figure 13-4. CMIS system integration

Goals of integration with enterprise content management systems (ECM)



- Incorporate documents and folders into process applications
- Consume events that occurred in enterprise content management (ECM) systems
- Support access to common ECM systems by using the Content Management Interoperability Services (CMIS) standard

[Handling content events in a process](#)

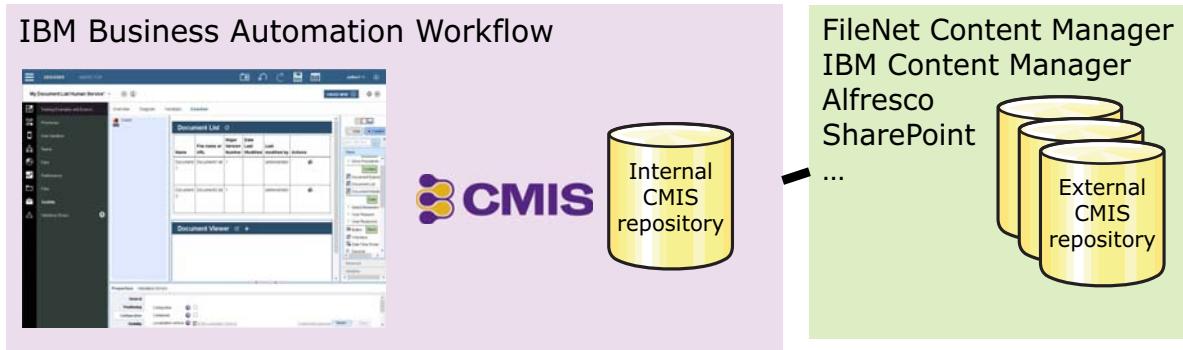
© Copyright IBM Corporation 2020

Figure 13-5. Goals of integration with enterprise content management systems (ECM)

- IBM Business Automation Workflow enables seamless integration with Enterprise Content Management (ECM) systems by providing an embedded CMIS-compliant (Content Management Interoperability Services) internal document store that is known as the BPM document store. This document store addresses the business requirement to include the management of related physical documents in the scope of a business process.
- The Organization for the Advancement of Structured Information Standards (OASIS) administers the CMIS open standard. The standard defines an abstraction layer for controlling diverse document management systems and repositories by using web protocols. CMIS uses Web services and Web 2.0 interfaces to enable rich information to be shared across Internet protocols in vendor-neutral formats, among document systems, publishers and repositories, within one enterprise and between companies. For more information, see: <https://www.oasis-open.org>.

Internal document store

- IBM Business Automation Workflow contains an internal CMIS repository
- Provides consistent document management for external and internal document stores
- Identical user experience, coaches, and views can connect to internal and external ECM the same way
 - If organizations do not currently have an enterprise ECM solution, you can start small and build



Handling content events in a process

© Copyright IBM Corporation 2020

Figure 13-6. Internal document store

The BPM document store is a CMIS-enabled embedded document repository that is used to store documents in IBM Business Automation Workflow. The BPM document store supports most Content Management Interoperability Services (CMIS) operations and a number of inbound events and you can use either coaches or heritage coaches to work with documents in the BPM document store.

IBM Training

Defining ECM server properties

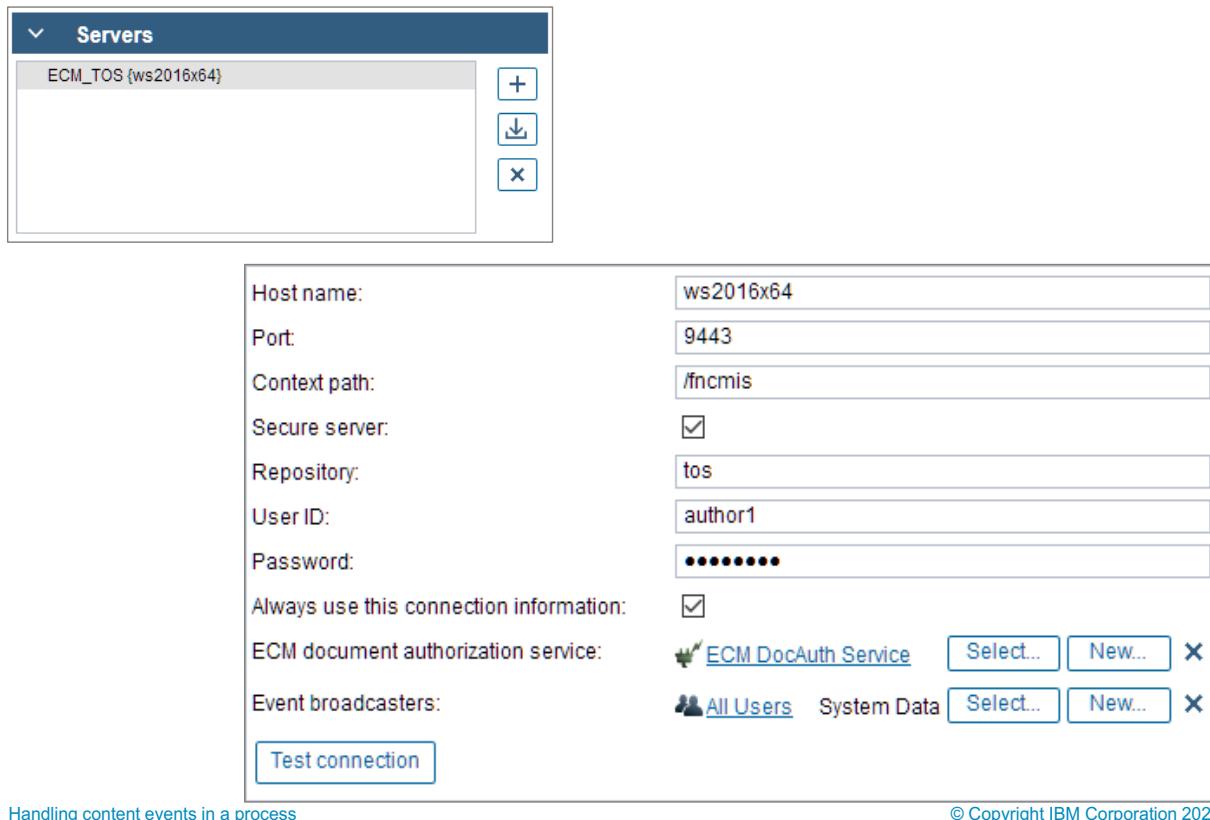


Figure 13-7. Defining ECM server properties

As an alternative to using the internal BPM document store, you can create an ECM server configuration. You can point to any CMIS-compliant content management platform.

To define the server properties, on the **Process Application Settings > Servers** tab, add an ECM server to the servers list and provide the connection details.

You can define the Target Object Store as an ECM server in your BPM solution to support the sharing of content.

The screen capture displays the configuration for the Target Object Store.

ECM Server properties:

- **Host Name:** The host name of the Enterprise Content Management server. Specify an IP address or a host name and domain. For example: myHost.labwide.ibm.com
- **Port:** The port number of the Enterprise Content Management server.
- **Context Path:** The path to the Content Management Interoperability Services (CMIS) web services application on the server. A connection must be established through Content Management Interoperability Services (CMIS) by using the web services protocol rather than the Atom protocol. When you define the ECM server properties for FileNet Content Manager, the default CMIS web service context path is "/fnccmis".

- **Secure Server:** Specify whether you want your service to be secure, that is, to use the Hypertext Transfer Protocol Secure (HTTPS) protocol by selecting this checkbox. If you select the HTTPS protocol, you must configure HTTPS security.
- **Repository:** The name of your repository. If you are using FileNet Content Manager as the ECM server, the object store name is the CMIS repository name.
- **User ID:** The user ID to connect to the Enterprise Content Management server.
- **Password:** The password of the user ID connecting to the Enterprise Content Management server.
- **Always Use This Connection Information:** If selected, which is the default, only this user ID and password are used for authentication. For example, a human service, which your service is associated with when a Document List or Document Viewer is configured, also has a user context. An administrator uses the Manage Users function to specify human service users. Selecting this checkbox means this user ID and password override any other user information.
- **ECM Document Authorization Service:** A service that you create and select if you want to check the permissions of a user. It is used by the Responsive Document Explorer, Responsive Document List, and Responsive Document Viewer coach views from the Content Management (SYSCM) toolkit when they perform operations that cannot be customized by using an Ajax service. It is also used by the older heritage coach views of these controls. These operations are the creation, update, and download of a document. The service is not used when you directly invoke the Content Integration operations in the human service or service flow editors.
- **Event broadcasters:** Specify a team whose content events are able to trigger a document start event or precondition evaluation. The team must include the technical user for this server. Business Automation Workflow processes content events only from these users.

13.2. Document controls

Document controls

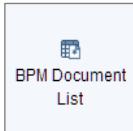
Handling content events in a process

© Copyright IBM Corporation 2020

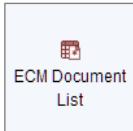
Figure 13-8. Document controls

Document List view (1 of 2)

- The Document List view displays (in tabular form) files in a content store. It provides under one view the ability to upload, delete, and change the properties of documents. It also allows users to view revisions associated with a file.
- The Document List view comes in two implementation types.



The BPM Document List view displays documents from the BPM content store.

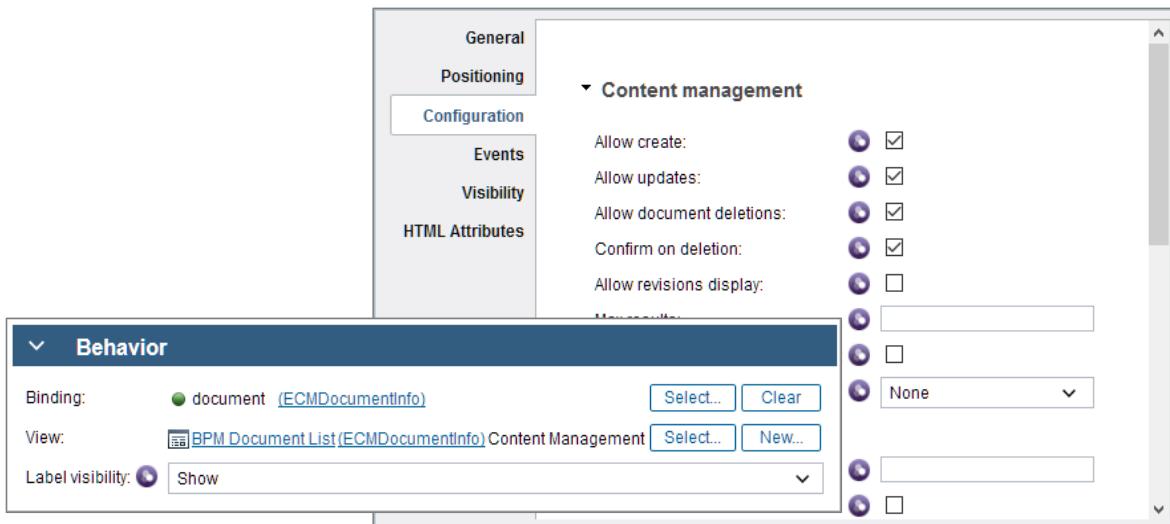


The ECM Document List view displays documents from a configured ECM content store.

- They are contained in the Content Management Toolkit.

Document List view (2 of 2)

- The Document List view is configurable and is bound to the `ECMDocumentInfo` business object. A CMIS query in the document repository generates the Document List.
- You can configure the view to allow the most popular operations such as creating and updating documents.



Handling content events in a process

© Copyright IBM Corporation 2020

Figure 13-10. Document List view (2 of 2)



Running a coach from the Process Portal

You can use coaches to build different types of user interfaces, such as for task completion (in a process), dashboard, startable service, or general purpose (URL-addressable).

The screenshot shows the IBM Process Portal interface. On the left, there is a navigation sidebar with the following items:

- author1 (Profile picture)
- Edit Profile
- Log Out
- Dashboards (selected)
- Create Saved Search
- Work
- Processes
- Process Performance
- Team Performance
- Show less...
- Dashboard** (highlighted with a red box)
- Launch
- Advanced HR Open New Position
- Discover BPM UI
- Startable service** (highlighted with a red box)
- Hiring Request Process

The main content area is titled "Hiring Request From Document" and displays an "ECM Document List". The table has columns for Name, Last Modified, and Version. The data is as follows:

Name	Last Modified	Version	Actions
AAA Employment application.pdf	5/2/2019	1.0	
CmAcmeCaseActivitySweep	4/27/2019	1.0	
CmAcmeCaseHealthAnalysis	4/27/2019	1.0	
CmAcmeCaseOperations	4/27/2019	1.0	
CmAcmeCaseOperationsComponentDefinition	4/27/2019	3.0	
CmAcmeEventHandler	4/27/2019	1.0	
CmAcmeRuleDeploymentAndOperations	4/27/2019	1.0	
CmAcmeRuleOperationsComponentDefinition	4/27/2019	3.0	
Deployment Log	4/30/2019	5.0	
Detail Deployment Log	4/30/2019	4.0	

At the bottom of the page, there are copyright notices: "Handling content events in a process" and "© Copyright IBM Corporation 2020".

Figure 13-11. Running a coach from the Process Portal

You can use coaches to build different types of user interfaces, such as for task completion, dashboard, startable service, or general purpose (URL-addressable). The Dashboard and Startable service can be run from Process Portal directly.

Task completion

When a coach is a task completion user interface, it is part of the human service flow. When the flow enters the coach, the user sees the user interface that is defined for that coach.

Dashboard

A dashboard is a stand-alone user interface that users can run at any time. Users can access dashboards through Process Portal.

Startable service

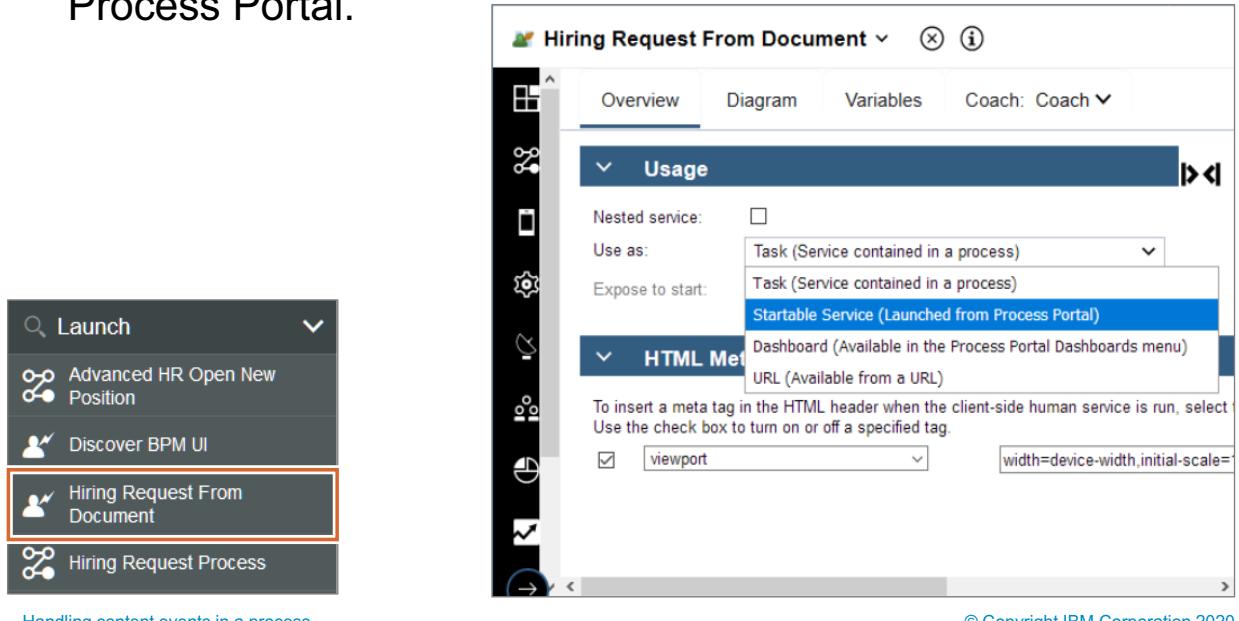
A startable service can be started in Process Portal at any time. The **Discover BPM UI** available on Process Portal is an example of a startable service.

General Purpose

For a general-purpose UI, a URL service creates a stand-alone UI that can be called directly through a URL.

Running a coach as a startable service

- To configure a coach to run as a startable service, select **Startable Service** as the usage under the Overview tab.
- The coach is then immediately available to be launched from Process Portal.



Handling content events in a process

© Copyright IBM Corporation 2020

Figure 13-12. Running a coach as a startable service

13.3. Content events

Content events

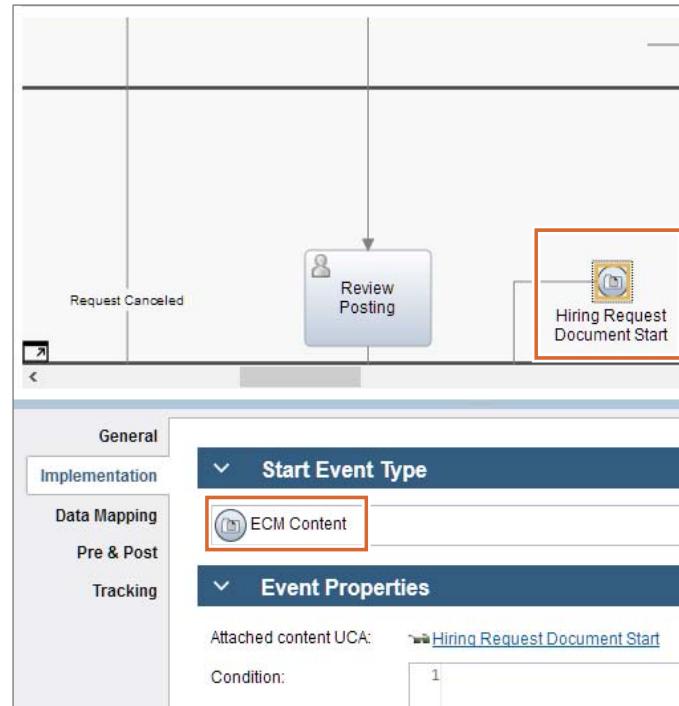
Handling content events in a process

© Copyright IBM Corporation 2020

Figure 13-13. Content events

Modeling content events on a process

- Use start and intermediate events on the process canvas to model content events.
- By configuring the event as an ECM Content event type, the process can react to any content event associated with the attached undercover agent.



Handling content events in a process

© Copyright IBM Corporation 2020

Figure 13-14. Modeling content events on a process



Content Event Subscription for BPM Documents

- You create event subscriptions to signal IBM Business Automation Workflow when ECM events occur.
- You must also create an attached service that is run when the document event is triggered.

Common

Name: Hiring Request Document Start
Type: ECM
Modified: author1 (May 1, 2019, 3:57:19 PM)

Documentation:

Details

ECM Server: BPM document store [Learn More](#)
[Use the Process Application Settings editor to add a server](#)

Event Class: Document
Object Type: IBM Business Automation Workflow Document Attachme
Include Subtypes:
Event Type: Created
Attached Service: **Hiring Request Document Start** [Select...](#) [New...](#)

Exposing

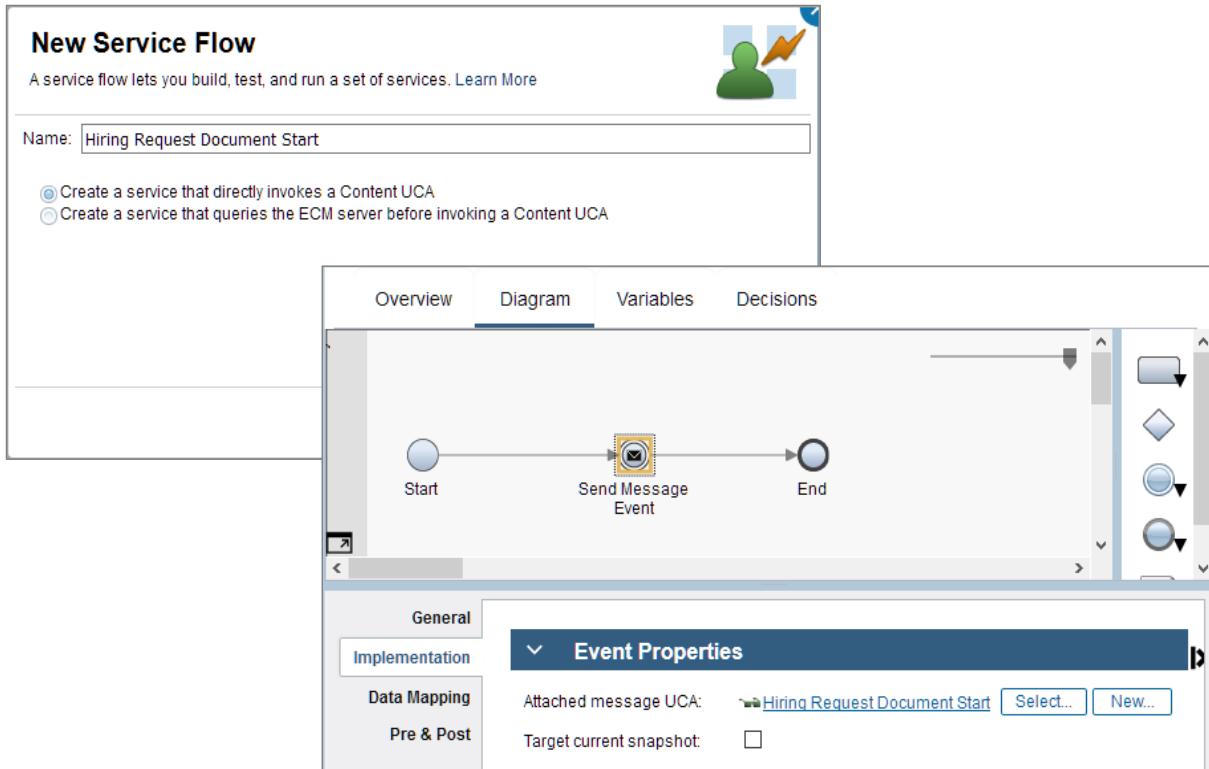
Event Broadcasters: **All Users** [Select...](#) [New...](#) [X](#)

Handling content events in a process

© Copyright IBM Corporation 2020

Figure 13-15. Content Event Subscription for BPM Documents

Creating the attached service



Handling content events in a process

© Copyright IBM Corporation 2020

Figure 13-16. Creating the attached service

When you create the attached service, IBM Business Automation Workflow can create a service with an Undercover Agent automatically, or the second option provides logic to query the ECM server before invoking the Undercover Agent. You use the Undercover Agent when implementing the event you model on the process.

Case integration

Handling content events in a process

© Copyright IBM Corporation 2020

Figure 13-17. Case integration

Case integration (1 of 2)

- IBM Business Automation Workflow supports integration between Case solutions and BPM solutions.
- This is achieved by adding a process from a BPM solution as an activity to the Case.

The screenshot shows the 'Add Activity' dialog on the left and the 'Activities' sidebar on the right.

Add Activity Dialog:

- General:**
 - Name:** Hiring Request Process
 - Unique Identifier:** HRCS_HiringRequestProcess
 - Description:** This activity starts the Hiring Request Process of the HR Recruitment Process BPM solution.
 - This activity starts:** Automatically (radio button selected)
 - This activity is:** Hidden (checkbox)

Activities Sidebar:

- Precondition: Case Start
- Set: <None>
- Activities:
 - View by: Priority
 - Manage Sets
 - Add Activity
 - Activity with New Process
 - To-do Task
 - Add Container
 - Activity with Existing Process** (highlighted with a red border)
 - Activity with New FileNet PB Process
 - Activity with Existing FileNet P8 Process

Handling content events in a process

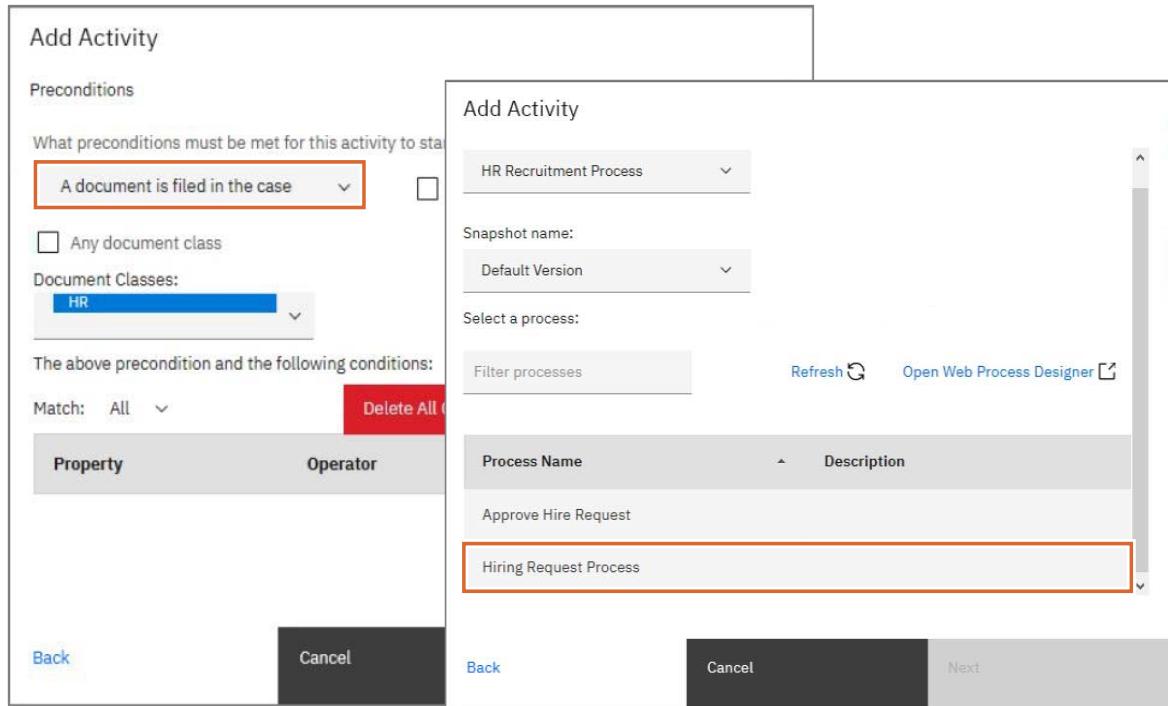
© Copyright IBM Corporation 2020

Figure 13-18. Case integration (1 of 2)

This slide and the next provide the details for how a BPM solution is integrated into a Case solution. The integration is built within the Case solution by adding a structured process from a BPM solution as an activity to the Case solution.

IBM Training

Case integration (2 of 2)



Handling content events in a process

© Copyright IBM Corporation 2020

Figure 13-19. Case integration (2 of 2)

One way to configure the integration is to set a precondition of a document being filed in the case. The document being filed starts the process that you select in the Add Activity screen.

There is no need to build anything in the process that uses this integration. The process is started from the manual start event.

You have a chance to build this integration in the next exercise.

To test the solution, you run the Case solution from the enhanced Workflow Center.



Running a Case solution (1 of 2)

- You can run Case solutions directly from the Workflow Center.

IBM Business Automation Workflow Case Client

Cases Work

Add Case

Search: Added On 9/17/2020 Search Advanced Search

No items to display

HR Case Solution | General Manager

HR Case Solution

Click Add Case to add a new Case to the solution.

Click the Play Solution icon to run the application.

© Copyright IBM Corporation 2020

Handling content events in a process

Figure 13-20. Running a Case solution (1 of 2)

You can run Case solutions directly from the enhanced Workflow Center. To add a document to a Case solution, you must first add a new Case.

Running a Case solution (2 of 2)

- Once a case is added, you can add a document to the Case
- This triggers the precondition and starts the structured process

The screenshot shows the IBM Case Solution interface. On the left, a 'General' panel shows a file named 'AAA Employment application.pdf' being selected for upload. On the right, the 'Add Document' panel is open, providing instructions for document properties. The main workspace displays a case titled 'Case HRCS_HR_000000100001'. The document 'AAA Employment application.pdf' is listed under the 'Documents' tab. A JavaScript code snippet at the bottom demonstrates how to add a comment to the parent case.

```
1 tw.system.currentProcessInstance.addCommentToParentCase("Your project has been approved! Good luck.", true)
```

Handling content events in a process

© Copyright IBM Corporation 2020

Figure 13-21. Running a Case solution (2 of 2)

The document in the screen capture is loaded as a member of the HR document class.

Because there is integration between the Case and BPM solution, the BPM solution can access details regarding the Case by using JavaScript methods.

Unit summary

- Use the CMIS capabilities of IBM Business Automation Workflow
- Explain how to handle content events in a process
- Describe how to add a document to the BPM document store
- Describe how to add a document to a Case solution
- Understand how the Target Object Store (TOS) can be used to share documents between a BPM solution and a Case solution
- Understand how to build a simple Case solution that integrates a process from a BPM solution

Handling content events in a process

© Copyright IBM Corporation 2020

Figure 13-22. Unit summary

Review questions

1. True or False: You must integrate an external CMIS system with IBM Business Automation Workflow to model and implement a content event on a process.
2. When you create a content event subscription and create an attached service, IBM Business Automation Workflow offers to automatically create:
 - A. A service flow with an undercover agent that is connected in the flow
 - B. A service flow with a web service that is connected in the flow
 - C. A client-side human service with a coach that is connected in the flow
 - D. A subprocess with a content start event
3. True or False: When integrating a process from a BPM solution in a Case solution, it is added as an activity of the Case.



Handling content events in a process

© Copyright IBM Corporation 2020

Figure 13-23. Review questions

Write your answers here:

- 1.
- 2.
- 3.

Review answers (1 of 2)

1. True or False: You must integrate an external CMIS system with IBM Business Automation Workflow to model and implement a content event on a process.
The answer is False. If your organization does not have an enterprise CMIS, IBM Business Automation Workflow includes an internal document store.
2. When you create a content event subscription and create an attached service, IBM Business Automation Workflow offers to automatically create:
 - A. A service flow with an undercover agent that is connected in the flow
 - B. A service flow with a web service that is connected in the flow
 - C. A client-side human service with a coach that is connected in the flow
 - D. A subprocess with a content start event

The answer is A. The wizard creates a general system service with the required undercover agent to trigger an event on a process.



Review answers (2 of 2)

3. True or False: When integrating a structured process in a Case solution, it is added as an activity of the Case.
The answer is True. The structured process runs when the activity is reached in the Case solution.



Handling content events in a process

© Copyright IBM Corporation 2020

Figure 13-25. Review answers (2 of 2)

Exercise: Handling content events in a process

Handling content events in a process

© Copyright IBM Corporation 2019

Figure 13-26. Exercise: Handling content events in a process

Exercise introduction

- Use the CMIS capabilities of IBM Business Automation Workflow
- Implement a content event in a process
- Use the BPM document store to add a document to a process
- Use the Target Object Store to share documents between a BPM solution and a Case solutions
- Build a simple HR Case solution and create an activity to start the Hiring Request Process in the HR BPM solution
- Demonstrate integration between the Case solution and the BPM solution



Handling content events in a process

© Copyright IBM Corporation 2019

Figure 13-27. Exercise introduction

Unit 14. Course summary, badge, and other resources

Estimated time

00:30

Overview

This unit summarizes the course and provides information for future study.

Unit objectives

- Describe the course objectives and what you learned
- Identify and describe product certifications that are related to this course
- Identify resources that can help you learn more

© Copyright IBM Corporation 2020

Figure 14-1. Unit objectives

Course objectives (1 of 7)

- Understand the key capabilities of Business Automation Workflow
- Describe how to use IBM Business Automation Workflow to accomplish process modeling goals
- Describe the purpose of the Process Portal, Process Designer, and Workflow Center repository
- Describe the high-level architecture for Business Automation Workflow
- List and describe the core notation elements that are used in the IBM Process Designer
- Describe the purpose and function of Blueworks Live
- Explain the benefits of using IBM Business Automation Workflow on Cloud

© Copyright IBM Corporation 2020

Figure 14-2. Course objectives (1 of 7)

Course objectives (2 of 7)

- Define Business Process Management (BPM)
- Understand the process spectrum
- List and describe the phases in the IBM Playback methodology
- Describe Playback 0 and the achievements that are reached during this stage
- Examine a defined workflow from detailed process requirements and identify the interrelated process activities
- Describe how to model a structured process
- Decompose activities into a nested process

© Copyright IBM Corporation 2020

Figure 14-3. Course objectives (2 of 7)

Course objectives (3 of 7)

- Describe process sequence flow and the runtime use of process tokens
- Explain how to evaluate and model conditions for a gateway
- List and describe intermediate event types that are used in the IBM Process Designer
- Model a business process escalation path with an attached timer intermediate event
- Describe the differences between process flow data and business flow data
- Add variables and business objects to a process
- Describe teams and process lanes

© Copyright IBM Corporation 2020

Figure 14-4. Course objectives (3 of 7)

Course objectives (4 of 7)

- Explain user distribution in a process application
- Implement routing for tasks
- Create a routing design by using a team filter service
- Assign an expert group to an activity
- Expose a process application to a team
- Use coaches to define and implement guided user interactions
- Implement a service for an activity in a process

© Copyright IBM Corporation 2020

Figure 14-5. Course objectives (4 of 7)

Course objectives (5 of 7)

- Enhance coaches by applying a theme and adding tabs
- Create a reusable view
- Create a snapshot
- Share your assets by using a toolkit, and exporting your process application
- Organize assets with favorites, tagging, and smart folders
- Conduct a Playback session
- Explain how to create a decision service

© Copyright IBM Corporation 2020

Figure 14-6. Course objectives (5 of 7)

Course objectives (6 of 7)

- Describe how to create and configure an undercover agent (UCA)
- Describe how to start a process with a message start event
- Define the basic function of services
- Configure and define services for outbound integration
- Create an inbound web service
- Describe the differences between an environment variable and an exposed process variable
- Catch an error in a process and service

© Copyright IBM Corporation 2020

Figure 14-7. Course objectives (6 of 7)

Course objectives (7 of 7)

- Explain when to use team filter services to support business policy
- Use parallel tasks and messaging in a BPMN model
- Use Multi-instance loops to efficiently route work
- Explain how to integrate with external systems in IBM Business Automation Workflow
- Explain how to handle content events in a process
- Understand how the Target Object Store (TOS) can be used to share documents between a BPM solution and a Case solution
- Understand how to build a simple Case solution that integrates a process from a BPM solution

© Copyright IBM Corporation 2020

Figure 14-8. Course objectives (7 of 7)

IBM credentials: Badges and certifications

- **Get certified**
 - Search IBM certification offerings across a broad range of technology areas.
 - <https://www.ibm.com/certify/certs>
- **Take an exam**
 - Search exams available for the IBM Professional Certification program.
 - <https://www.ibm.com/certify/exams>
- **Search badges**
 - Find IBM badges for skill development activities and other achievements.
 - <https://www.ibm.com/training/search?query=%20&trainingType=Badge>
- **News**
 - Catch up on the latest IBM credential news.
 - <https://www.ibm.com/blogs/ibm-training/category/badging-certification>

© Copyright IBM Corporation 2020

Figure 14-9. IBM credentials: Badges and certifications

Learn more about this product

- **IBM Middleware User Community**

- Learn about API Connect, App Connect, MQ, DataPower, Aspera, Event Streams, and Cloud Pak for Integration
- <https://community.ibm.com/community/user/middleware/communities/cloud-integration-home>

- **IBM Automation Community**

- Learn about Blockchain, Blueworks Live, BPM, Workflow, Case, Content Management, Decision Management, Robotic Process Automation, Platform, and Cloud Pak for Automation
- <https://community.ibm.com/community/user/automation/home>

© Copyright IBM Corporation 2020

Figure 14-10. Learn more about this product

Additional resources (1 of 3)

- **IBM Cloud Education course information**
 - View and download course materials and course corrections.
 - <http://ibm.biz/CourseInfo>
- **IBM Developer**
 - IBM's official developer program offers access to software trials and downloads, how-to information, and expert practitioners.
 - <https://developer.ibm.com>
- **IBM Training**
 - Search the IBM Training website for courses and education information.
 - <https://www.ibm.com/training>

© Copyright IBM Corporation 2020

Figure 14-11. Additional resources (1 of 3)

Additional resources (2 of 3)

- **Learning Journeys**

- Learning Journeys describe a recommended collection of learning content to acquire skills for a specific technology or role.
- <https://www.ibm.com/training/journeys/#tab-ibm-cloud>

- **IBM Redbooks**

- IBM Redbooks are developed and published by the IBM International Technical Support Organization (ITSO). Redbooks typically provide positioning and value guidance, installation and implementation experiences, typical solution scenarios, and step-by-step "how-to" guidelines.
- <http://www.redbooks.ibm.com>

- **IBM Knowledge Center**

- IBM Knowledge Center is the primary home for IBM product documentation.
- <http://www.ibm.com/support/knowledgecenter>

© Copyright IBM Corporation 2020

Figure 14-12. Additional resources (2 of 3)

Additional resources (3 of 3)

- **IBM Marketplace**

- Learn about IBM offerings for Cloud, Cognitive, Data and Analytics, Mobile, Security, IT Infrastructure, and Enterprise and Business Solutions.
- <https://www.ibm.com/products>

- **IBM Training blog, Twitter, and Facebook**

- Official IBM Training accounts provide information about IBM course offerings, industry information, conference events, and other education-related topics.
- <https://www.ibm.com/blogs/ibm-training>
- <https://twitter.com/IBMTTraining>
- <https://www.facebook.com/ibmtraining>

© Copyright IBM Corporation 2020

Figure 14-13. Additional resources (3 of 3)

Unit summary

- Describe the course objectives and what you learned
- Identify and describe product certifications that are related to this course
- Identify resources that can help you learn more

© Copyright IBM Corporation 2020

Figure 14-14. Unit summary

Course completion

You have completed this course:

Developing workflow solutions using IBM Business Automation Workflow V20.0.0.1

Do you have any questions?



© Copyright IBM Corporation 2020

Figure 14-15. Course completion



IBM Training



© Copyright International Business Machines Corporation 2020.