Course Guide

IBM Case Foundation 5.2.1: External Communication

Course code F243 ERC 1.0

IBM Training

June 2017 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.

Contents

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:

DB2®	FileNet®	Power®
Tivoli®	WebSphere®	

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

Java™ and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Social® is a trademark or registered trademark of TWC Product and Technology, LLC, an IBM Company.

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

Course description

IBM Case Foundation 5.2.1: External Communication

Duration: 4 hours

Purpose

This course is intended to teach the skills that are needed to integrate; an external database stored procedure, a Java component, and an external web service.

Audience

This course is intended for Workflow Authors, responsible for planning, designing, and implementing a workflow application with IBM Case Foundation.

Prerequisites

- Terminology:
  - Content Platform Engine
  - Process Services
  - Workflow
  - Workflow definition
- Concepts:
  - Workflow
- Skills:
  - Use Process Designer
  - Create workflow maps
  - Use Process Tracker

Objectives

- Use a DbExecute step in a workflow to integrate an external database.
- Identify Component Integrator concepts and components.
- Integrate a CE_Operations component in a workflow.
- Integrate a Java component in a workflow.
- Invoke a Web Service in a workflow.

Contents

NOTE: In this course a unit is the same as a lesson. For example, Unit 1/Lesson 1.1, means Unit 1 or Lesson 1.1.

Agenda

Note

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

(00:15) Course introduction

(00:50) Unit 1/Lesson 1.1 External database integration
- Activity: Use information from an external database in a workflow

(00:20) Unit 2/Lesson 1.2 External database integration
- Review questions

(01:05) Unit 3/Lesson 1.3 Using components
- Activity: Integrate components in a workflow

(00:55) Unit 4/Lesson 1.4 Using components
- Review questions
- Activity: Invoke a web service

Unit 1. External database integration

Estimated time

00:50

Overview

In this unit/lesson shows you how to use a DbExecute step in a workflow to call an external database stored procedure.

How you will check your progress

- Successfully complete the lesson exercises.

References

IBM Knowledge Center

`http://www.ibm.com/support/knowledgecenter/SSNW2F_5.2.1/welcome`

IBM Training

Why is this lesson important to you?

- You are designing a workflow application. The workflow needs to run a stored procedure in an external database to integrate information, contained in the database, with the workflow application. You must add a DbExecute step to your workflow and test the workflow to verify the expected outcome.

Figure 1-1. Why is this lesson important to you?

IBM Training

Lesson objectives

- Use a DbExecute step in a workflow to integrate with an external database

Figure 1-2. Lesson objectives

IBM Training

DbExecute system function

- Runs a stored procedure in a database
  - Specify the external database, the name of the stored procedure, and parameters.
  - Database and stored procedure must exist and the connection be configured.

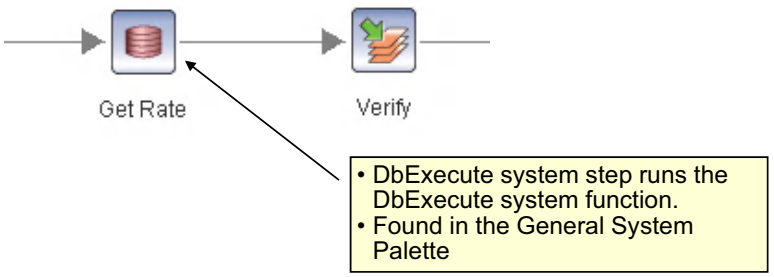


Figure 1-3. DbExecute system function

Help path

FileNet P8 Platform 5.2.1>Integrating workflow into document management>Designing workflows>About steps>System functions>General step activity

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.designerui.doc/bpfdh101.htm`

Use the DbExecute system function to run a stored procedure in a specified, external database, such as DB2, Oracle, or SQL Server. Before you can run a workflow that contains a DbExecute system function, a couple of configuration steps must be completed:

- The database administrator must set up the database and the stored procedures.
- The IBM FileNet P8 system administrator must configure the connection to the external database.

IBM Training

Example: Using DbExecute

- Use case scenario
  - When a new loan is being prepared, the current interest rate is retrieved from the database.
- Solution example
  - A stored procedure - “GetInterestRate” - was defined in a LoanDB database.
  - Data field - ‘interest_rate’ - is set by the stored procedure that runs at the Get Rate step.

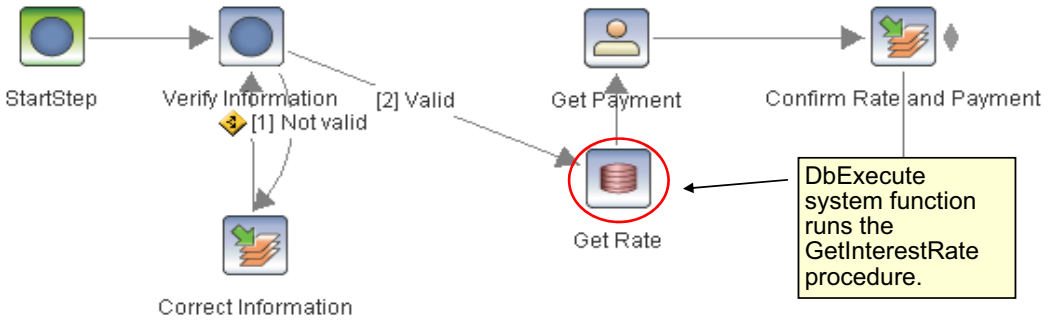


Figure 1-4. Example: Using DbExecute

Loan processing application example (1)

The diagram shows an example of the DbExecute system function in a loan processing application. During loan preparation, the Get Rate step obtains the current interest rate available, based on the loan term, and loan amount requested by the customer. The Get Rate step uses DbExecute to call a stored procedure, named GetInterestRate. This stored procedure assigns a value to the interest_rate data field. The GetInterestRate stored procedure was previously defined in a database named LoanDB.

IBM Training

DbExecute step parameters

- Database connection alias is configured by the system administrator for the object store's workflow system.
- Stored procedure name is determined by the database programmer.
- Parameters are assigned in the workflow definition.

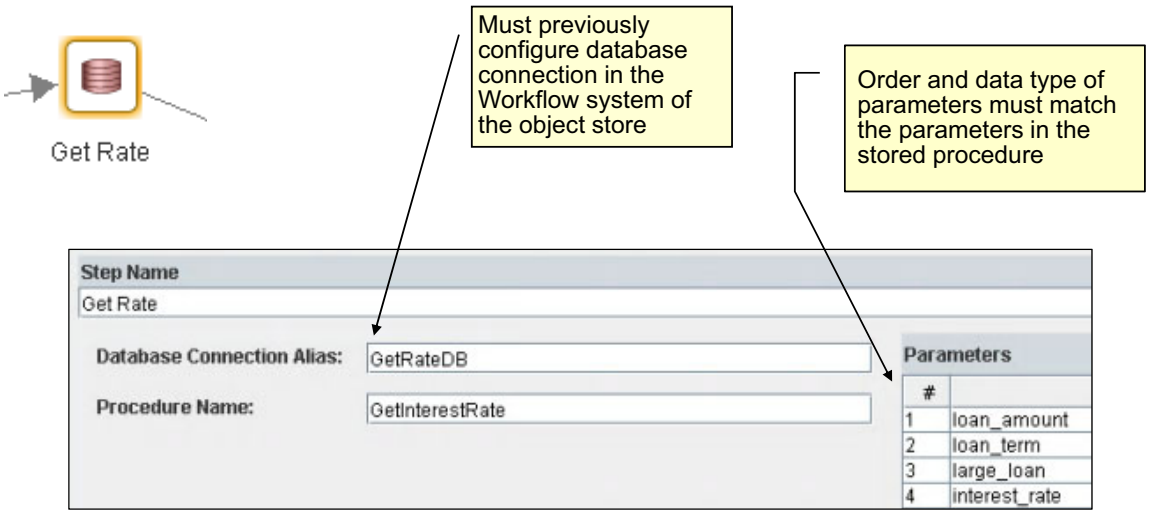


Figure 1-5. DbExecute step parameters

Loan processing application example (2)

The diagram shows how the step parameters are set for the DbExecute step. The Get Rate step uses DbExecute to call the stored procedure GetInterestRate. The GetInterestRate stored procedure is defined in the database, LoanDB. The database connection alias for the LoanDB database is named GetRateDB. The database connection alias must be previously configured on the workflow system of the object store.

The GetInterestRate stored procedure uses four parameters. The first three parameters, loan_amount, loan_term, and large_loan, are input parameters to the stored procedure. The fourth parameter, interest_rate, is an output parameter. The workflow author defines the interest_rate workflow field as a float field. This field holds the value of the interest rate that is returned by the stored procedure.

The parameters that are assigned in the DbExecute step parameters must match the order and data type of the parameters that are used in the stored procedure. The data type matches are described on the next page.

The workflow author typically works with the system administrator and database programmer to gather the necessary information about the stored procedure. For example, the database connection alias, the stored procedure name, and the stored procedure parameters.

IBM Training

Parameter data types

- The data type used in a Process Designer field must match parameter data type in the stored procedure.
  - Depends on database type
  - Parameter cannot be an array.
  - Parameter can be a single element of an array.

DbExecute Parameter Types			
Process Designer	**DB2**	**SQL Server**	**Oracle**
Boolean	number	bit	number
Float	float	float	number
Integer	int	int	number
String	varchar	varchar	varchar
Time	timestamp	datetime	date

Figure 1-6. Parameter data types

Help path

FileNet P8 Platform 5.2.1>Integrating workflow into document management>Designing workflows>About steps>System functions>General step activity

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.designerui.doc/bpfdh101.htm`

The parameter data types in the DbExecute function must match the data types that are specified as parameters in the stored procedure. The matching data type for each supported database is shown in the chart.

Prerequisite administrative task

- Define and validate a database connection for the object store's Workflow System.

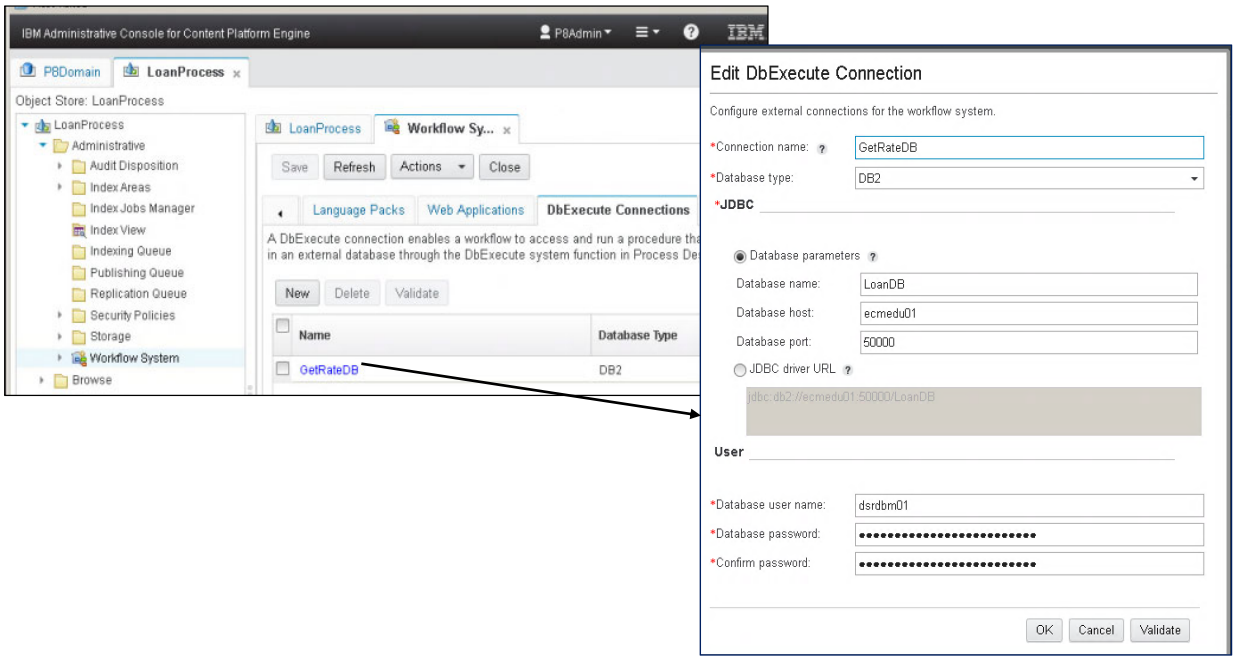


Figure 1-7. Prerequisite administrative task

Help path

FileNet P8 Platform 5.2.1>Integrating workflow into document management>Configuring the workflow system>Managing the workflow system>Setting web application properties for the workflow system

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.configui.doc/bpfc079.htm`

Before you can run a workflow that uses the DbExecute system function, the workflow administrator must configure a DbExecute connection in the object store's workflow system.

- The DbExecute connection is configured in Administration Console for Content Platform Engine, in the workflow system's DbExecute Connections tab.
- The database administrator provides the database connection information.
  - For example, database type, database name, database host and port, and database user name and password.

The stored procedure can be located in the same database that contains the workflow system or an external database.

IBM Training

Considerations when using DbExecute

- Null returned from a stored procedure
  - If String data type, results in an empty string.
  - If Boolean, float, integer, or time, the item fails with a workflow exception.
- If stored procedure does not return within timeout limit, item fails with a workflow exception.
  - Default timeout limit is 1 hour.
  - Define workflow system property, SQLServerStoredProcWait, to change time limit
- Work with your database administrator or programmer to design appropriate business exception handling routines.

Figure 1-8. Considerations when using DbExecute

Help paths

FileNet P8 Platform 5.2.1>Administering>Administering Content Platform Engine>Defining the workflow system>Maintaining the workflow system

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.ce.admin.tasks.doc/bpfad001.htm`

FileNet P8 Platform 5.2.1>Administering>Administering Content Platform Engine>Defining the workflow system>Maintaining the workflow system>Workflow system advanced properties

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.ce.admin.tasks.doc/bpfad029.htm`

Null returned from a stored procedure

- If a String parameter is set to null by a stored procedure, the workflow field receives an empty string.
- If a Time, Integer, Float, or Boolean parameter is set to null by a stored procedure, the work item goes to an exception state. Malfunction is called and a message is logged.

Timeout for returned parameters

A timeout limit exists for the DbExecute system function to complete the communication from the workflow to the database. The default timeout value is 3600 seconds (1 hour). The workflow system administrator can adjust this timeout limit in the workflow system advanced properties, by defining the workflow system property, SQLServerStoredProcWait.

Workflow exception handling

The workflow author works with the workflow system administrator, database administrator, and application programmer to design process exception handling routines that are appropriate for the business case.

IBM Training

Instructor demonstration

- This demonstration shows you how to add and configure a DbExecute system step in a workflow definition.

Figure 1-9. Instructor demonstration

Demonstration notes

This demonstration shows you how to add and configure a DbExecute system step in a workflow definition.

1. Create a Database Connection Alias.
2. Open a new workflow definition.
3. Add a DbExecute step to the map.
4. Assign the step properties.
5. Assign the step parameters. The DbExecute step parameters must agree in number and type with the stored procedure.

IBM Training

Unit summary

- Use a DbExecute step in a workflow to integrate with an external database

Figure 1-10. Unit summary

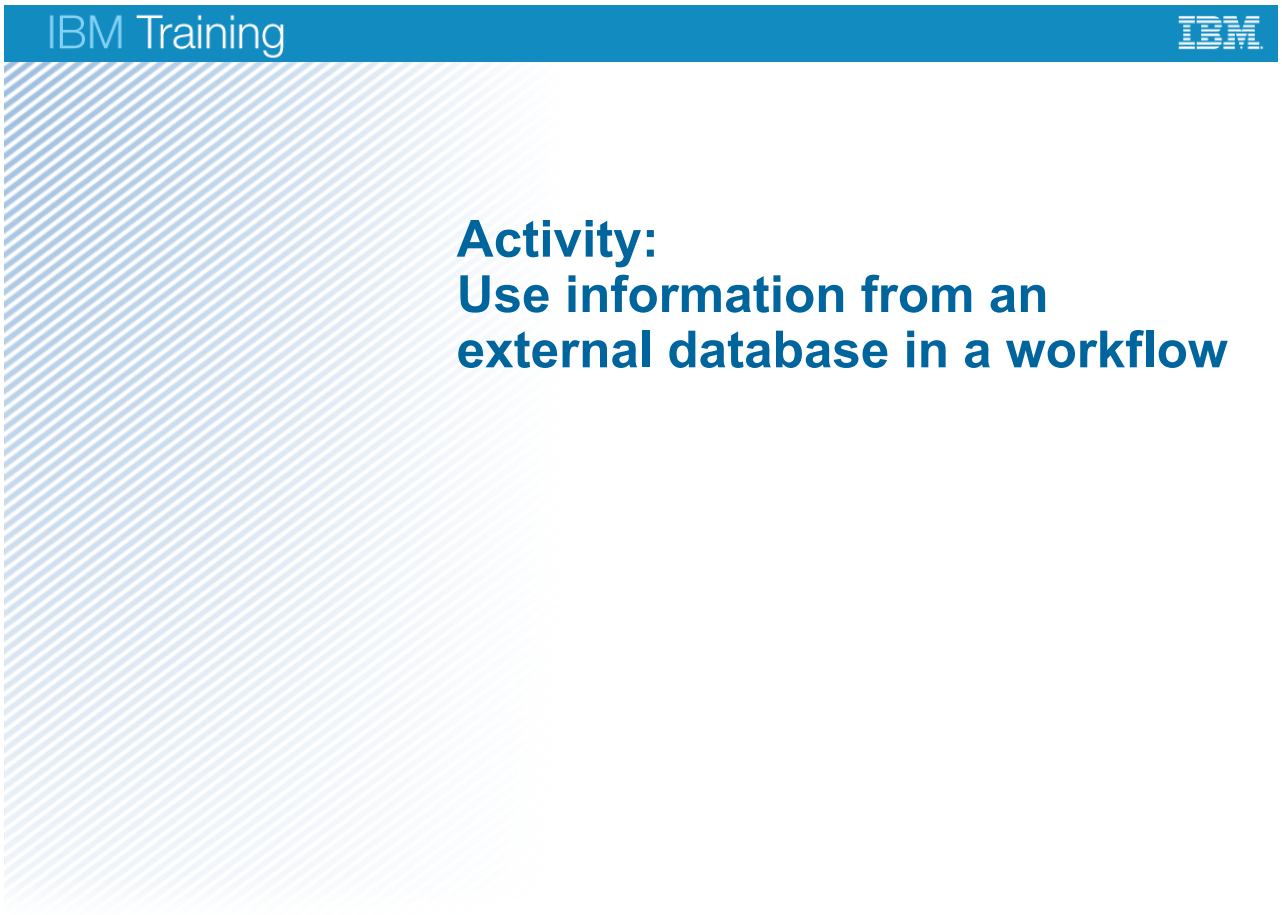


Figure 1-11. Activity: Use information from an external database in a workflow

IBM Training

Activity introduction

- Configure a database connection alias
- Define a simple workflow with a DbExecute system step
- Test the workflow

Figure 1-12. Activity introduction

Use your Course Exercises Guide to perform the activities listed.

Unit 2. Component integrator architecture

Estimated time

00:20

Overview

This unit/lesson describes Component Integration concepts, components, and architecture.

How you will check your progress

- Complete the review questions.

References

IBM Knowledge Center

`http://www.ibm.com/support/knowledgecenter/SSNW2F_5.2.1/welcome`

`http://www.ibm.com/support/docview.wss?uid=swg27043131#FrameworkComparison`

IBM Training

Why is this lesson important to you?

- You are designing a workflow application. You want to integrate CE_Operations, Java or Java Message Service (JMS) components as steps in your workflow. In order to use component steps in a workflow, you must understand the fundamental concepts and architecture of the Component Integrator.

Figure 2-1. Why is this lesson important to you?

IBM Training

Unit objectives

- Identify Component Integrator concepts and components

Figure 2-2. Unit objectives

Purpose of component integration

- Extend business functionality easily without full application development.
- Automate work processing.
- Perform external functions from within a workflow.
- Use existing Java business objects and components.
- Integrate with a Java Message Service (JMS).

Figure 2-3. Purpose of component integration

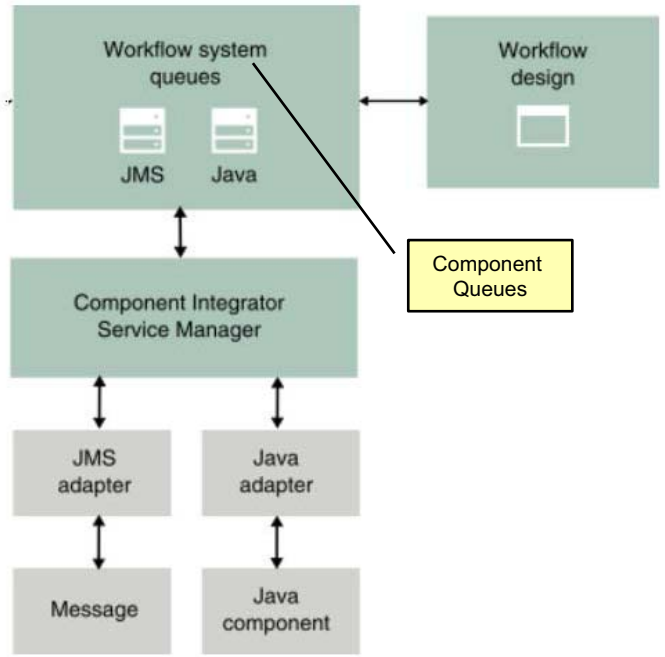
With component integration, you can extend business functions easily without requiring full application development.

You can run external functions from within the workflow application, and use existing Java business objects and components or a Java Message Service.

IBM Training

Component integrator

- The component integrator architecture



Component integrator architecture

Figure 2-4. Component integrator

Help path

FileNet P8 Platform 5.2.1>Developing FileNet P8 applications>Process Development>Process Java Developer's Guide>Developing Process Applications>Developing Work Performers

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.dev.doc/custom/dev_work_performer_overview.htm`

The Component Integrator is also known as the Component Manager Framework.

The diagram shows the Component integrator architecture.

Component Integrator Service Manager connects a work item that is requesting a component to the appropriate service adapter.

Component Integrator Service Manager communicates with the service adapters. At run time, Component Integrator Service Manager polls component queues, labeled as Workflow system queues, in the diagram, for work items that request Java Message Service (JMS) or processing by Java components.

The JMS adapter places messages on the JMS Queue and dispatches the associated work item. The JMS adapter handles posting of Process events to a message queue. The posting is in the form of an XML event, based on the step element for the operation. For JMS information, see `http://docs.oracle.com/javaee/6/tutorial/doc/bncdq.html`.

The Java adapter handles process calls to Java objects. The calls are presented to the Content Platform Engine as operations on queues (work items). Each operation is completed by a method of the Java class. The Java adapter runs the following sequence of actions:

- Loads the Java component class.
- Runs the interface that is associated with the Java component.
- Waits for a response from the Java component.
- Updates the work item field values.
- Dispatches the work item to the next workflow step.

Note

The diagram on this slide shows only the portion of the architecture that applies to the new component queues.

IBM Training

Component Manager Frameworks

- Two Component Manager Frameworks
  - New Component Manager Framework (V2)
  - Old/Legacy Component Manager Framework (V1)
- Both frameworks are fundamentally the same

Facts	New Component Manager framework	Old Component Manager framework
Where the component manager runs	Content Platform Engine (CPE) server	Workplace XT or Application Engine servers
Starting and stopping the component manager	Runs automatically on all CPE nodes. You can disable a specific component queue.	Each instance is started and stopped independently with Process Task Manager.
Location of the JAR files	JAR files are added to an object store as a code module.	JAR files are copied to all appropriate instances of Workplace XT or Application Engine servers.
When changes take effect	When component configuration changes are saved or committed	When Process Task Manager is stopped and restarted

Figure 2-5. Component Manager Frameworks

Help path

Migrating to the IBM FileNet P8 5.2 Component Manager Framework technote, Section: Comparing the two Component Manager Frameworks

`http://www.ibm.com/support/docview.wss?uid=swg27043131#FrameworkComparison`

Starting with the IBM Case Foundation 5.2 release, a new component manager framework was introduced which runs within the Content Platform Engine server.

Two frameworks are supported:

- The new Component Manager framework, also called Component Manager V2.
- The old or legacy Component Manager framework, also called Component Manager V1.

Customers can choose to use the following frameworks

- New component manager framework only.
- Old/legacy component manager framework only.
- A combination of the two frameworks.

The two frameworks are fundamentally the same. Both frameworks use component queues and component manager to connect to external entities. The main difference is where the component manager runs.

The new component manager runs in the Content Platform Engine server. The old component manager requires Workplace XT or Application Engine servers to run.

The table lists some of the major differences. For a complete list, see the help link.

IBM Training

Deciding on a Component Manager Framework

- New Component Manager Framework
  - Packaged and deployed with CPE server, runs as a background task.
    - Implements consistent load balancing and High Availability model.
  - Easier to administer and maintain.
  - More performance information and logs available.
  - Does not require Workplace XT or Application Engine servers
- Old/legacy Component Manager Framework
  - More flexibility but more maintenance required. You can:
    - Run different component adapters on different servers.
    - Define multiple component managers per connection point.
    - Assign different priorities to each component queue.
    - Have more control over the class path.

Figure 2-6. Deciding on a Component Manager Framework

Which Component Manager Framework is right for you?

The new Component Manager Framework:

- Is packaged and deployed with the Content Platform Engine server, and runs as a background task.
- Does not require extra steps to configure it.
- Is easier to administer and maintain.
- Provides more performance information and logs for maintenance and troubleshooting.
- Does not require Workplace XT or Application Engine, which aligns with IBM's strategic plan of using IBM Content Navigator as the application framework.

Note

No enhancements are being made to either Workplace XT or Application Engine.

Component behavior

- A component is an application that performs an operation in a workflow.
  - It is used to process workflow data.
  - It typically has no user interface.
  - It typically performs automatic operations on work items.
- A work item waits in a component queue for processing.
  - The work item makes a request of a component.
  - The component can perform other actions outside the isolated region.
  - The component can provide a response.

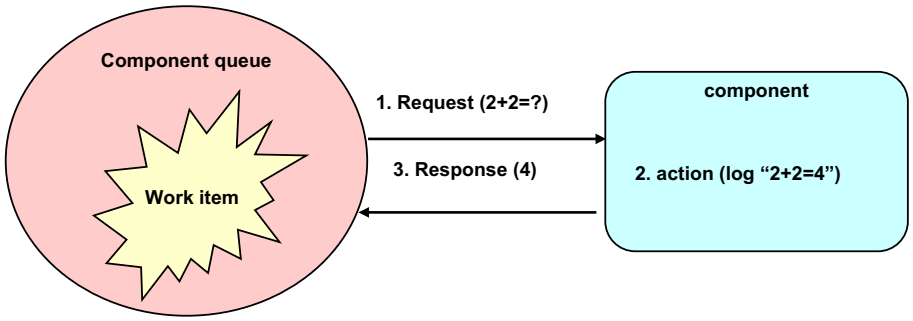


Figure 2-7. Component behavior

Help path

FileNet P8 Platform 5.2.1>Developing FileNet P8 applications>Process Development>Process Java Developer's Guide>Developing Process Applications>Developing Work Performers

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.dev.doc/custom/dev_work_performer_overview.htm`

Components typically run automated functions that do not require human intervention.

The activity initiates the work item, which sends data to the component as a request. The component can return a response from the request, if configured to do so, or it can run some function outside the isolated region. For example, file a document in an object store. Both behaviors are optional.

For example, if a work item requests the sum of two numbers, the work item can send them to a component that performs addition. The component can send the sum back to the work item, or the component can send the sum (or the original parameters) elsewhere. The diagram illustrates this example.

Other operations that are associated with workflow steps can include:

- Logging in and establishing a Process Services session.
- Polling a user, or system queue (to find operations that are related to a particular workflow step).
- Locking the retrieved object, processing the work (such as updating data or saving), and cycling back to queue polling.

Definition: A component queue is a queue that holds work items that an external entity that interacts with the workflow by using the Component Integrator, can complete.

Component queues reside in an isolated region.

IBM Training

Types of components

- Content Extended Operations
  - Provides content operations on IBM FileNet Content Manager repositories from workflows.
  - CE_Operations component
    - Includes many methods to facility working with content in a repository
- Java component
  - Component functionality is provided by a Java archive file.
  - Java classes contain methods and operations to perform the work.
  - Data can move in both directions:
    - From the isolated region to the component
    - From the component to the isolated region
  - Uses a configured Java component adapter.
- JMS component
  - Component functionality sends a message to a specified queue in a JNDI QueueConnectionFactory.
  - Data moves only from the isolated region to the JMS queue.
  - Functionality is contained in the JMS component adapter.

Figure 2-8. Types of components

Help path

FileNet P8 Platform 5.2.1>Integrating workflow into document management>Process applications concepts>Design and run workflows>Using Content Extended Operations in a workflow definition

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.user.doc/bpfwd029.htm`

FileNet P8 Platform 5.2.1>Developing FileNet P8 applications>Process Development>Process Java Developer's Guide>Developing Process Applications>Developing Work Performers>Developing Component Integrator-Based Work Performers

`https://www.ibm.com/support/knowledgecenter/SSNW2F_5.2.1/com.ibm.p8.pe.dev.doc/custom/ci_dev.htm`

Content Extended Operations

Provides content operations on IBM FileNet Content Manager repositories from workflows.

Content Platform Engine includes the CE_Operations component, which includes many methods to facilitate working with content in a repository.

Java

Java objects are archived classes or collections of classes that perform a function. IBM Case Foundation provides a Java component named CE_Operations to provide workflow definitions programmatic access to FileNet P8 domain object store objects.

The Java adapter allows the solution builder to expose public methods from a Java class as operations on a queue.

Java Message Service (JMS)

The Java Message Service (JMS) was designed to develop business applications that asynchronously send and receive business data and events.

IBM Training

Content Extended Operations: CE_Operations

- System-provided Java component queue
- Use to manipulate objects in an object store from within an executing workflow.
- Provides access to Content Platform Engine objects, properties, and actions from a workflow component step.
- Examples of available operations

Operation	Description
checkin	Checks in attached document and set property values
checkout	Checks out attached document from a repository
createDocument	Create a document using the specified class, properties and content
createFolder	Create a folder in the object store
getContent	Retrieves content or the attached document

Figure 2-9. Content Extended Operations: CE_Operations

Help path

FileNet P8 Platform 5.2.1>Integrating workflow into document management>Process applications concepts>Design and run workflows>Using Content Extended Operations in a workflow definition

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.user.doc/bpfwd029.htm`

FileNet P8 Platform 5.2.1>Developing FileNet P8 applications>Process Development>Process Development>Content Integrator Reference

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.dev.java.ceops.doc/filenet/contentops/ContentOperations.html`

Content Extended Operations allow a workflow to access and manipulate objects in an object store. An initialized region includes a system-provided Java component queue named CE_Operations.

The Content Extended Operations have access to Content Engine objects and properties and can perform many functions. For example,

- Get and set document property values of a document or folder.
- File and unfile a document in an object store folder.
- Publish a document.
- Check in and check out a document.
- Create a document.
- Create a folder.
- Delete an object.
- Apply a security template.
- Send an email with or without attachments.

For a complete list of available operations, refer to the Component Integrator Reference help path link.

To be effective in using Content Extended Operations, it is necessary to understand the properties of documents in an object store. Each document in an object store has a set of properties that are defined by the system and by a user, such as the display name (the name of the document), the creator, the date last modified, and other properties. In addition, each document is associated with a particular document class that usually defines more properties specific to that type of document. For example, a Loan document class might have the user-defined property CustomerName that is specified for loan documents when they are filed in the object store.

IBM Training

Java adapter

- The Java adapter:
  - Handles Process calls to Java objects.
    - Process calls are represented as operations on queues (work items). Each operation is run by a method of a Java class.
- The Java adapter performs the following sequence of actions:
  - Runs the interface to the Java component.
  - Automatically waits for a response from the component.
  - Updates the work item.
  - Dispatches the work item to the next workflow step.

Figure 2-10. Java adapter

Help path

FileNet P8 Platform 5.2.1>Developing FileNet P8 applications>Process Development>Process Java Developer's Guide>Developing Process Applications>Developing Work Performers

`https://www.ibm.com/support/knowledgecenter/SSNW2F_5.2.1/com.ibm.p8.pe.dev.doc/custom/dev_work_performer_overview.htm`

The Java adapter handles Process calls to Java objects. The calls are represented as operations on queues (work items), where each operation is run by a method of the Java class.

The Java adapter runs the following sequence of actions:

- Runs the interface to the Java component.
- Automatically waits for a response from the component.
- Updates the work item.
- Dispatches the work item to the next workflow step.

When multiple operations exist in a component queue, which is defined in a single step, the operations run serially. For example, if you have op1 and op2 defined on queue X, in the same step, the component manager will:

- Finish the operation, op1.
- Dispatch the work item to queue X again.
- Query and finish operation, op2.

Java Message Service (JMS) adapter

- The JMS adapter:
  - Places messages on the JMS component queue.
  - Dispatches the associated work item.
  - Handles posting of Process events to a message queue.
    - Posting is in the form of an XML event.
      Based on the step element of the operation.

Figure 2-11. Java Message Service (JMS) adapter

The JMS adapter, places messages on the JMS component queue and dispatches the associated work item.

The JMS adapter handles posting of the Process events to a message queue. The posting is in the form of an XML event, based on the step element for the operation.

IBM Training

Unit summary

- Identify Component Integrator concepts and components

Figure 2-12. Unit summary

IBM Training

Review questions

1. Select the option that is **NOT** a purpose of component integration.
   A. Extend business functionality easily without full application development.
   B. Use existing Java business objects and components.
   C. Perform external functions from within a workflow.
   D. Provide integration for a component step processor.
2. Starting with the IBM Case Foundation 5.2 release, a new Component Manager Framework was introduced. Where does this framework run?
   A. Within the Content Platform Engine server.
   B. Within the Process Task Manager, running on a Workplace XT server.
   C. As a task within the Administration Console for Content Platform Engine.
   D. Within the Process Configuration Console.
3. True or False: IBM Case Foundation 5.2 supports Java components only.

Figure 2-13. Review questions

Write your answers here:

1.

2.

3.

Review questions

4. Which of the following statements is not an advantage to using the new Component Manager Framework?
   A. Easier to administer and maintain.
   B. More control over the class path.
   C. More performance information and logs available.
   D. Does not require Application Engine servers.
5. Which one of the following elements is used in component queues for authentication?
   A. JNDI
   B. Active Directory
   C. JAAS
   D. Component Manager
6. True or False: A component is an operation that waits in a component queue for processing.

Figure 2-14. Review questions(2)

Write your answers here:

1.

2.

3.

IBM Training

Review questions

7. Which of the following component integration elements sends a message to a queue in a JNDI QueueConnectionFactory?
   A. Process Task Manager
   B. Java adapter
   C. CE_Operations
   D. JMS adapter
8. True or False: The two Component Manager Frameworks, supported starting with IBM Case Foundation 5.2, are fundamentally the same.
9. You want to create a folder in the repository from a workflow step. Which component do you need to use?
   A. Java component
   B. JMS component
   C. CE_Operations component
   D. You can’t create a folder from a workflow step

Figure 2-15. Review questions(3)

Write your answers here:

1.

2.

3.

Review answers

1. Select the option that is **NOT** a purpose of component integration.
   A. Extend business functionality easily without full application development.
   B. Use existing Java business objects and components.
   C. Perform external functions from within a workflow.
   D. Provide integration for a component step processor.
2. Starting with the IBM Case Foundation 5.2 release, a new Component Manager Framework was introduced. Where does this framework run?
   A. Within the Content Platform Engine server.
   B. Within the Process Task Manager, running on a Workplace XT server.
   C. As a task within the Administration Console for Content Platform Engine.
   D. Within the Process Configuration Console.
3. True or False: IBM Case Foundation 5.2 supports Java components only.

Figure 2-16. Review answers

IBM Training

Review answers

4. Which of the following statements is not an advantage to using the new Component Manager Framework?
   A. Easier to administer and maintain.
   B. More control over the class path.
   C. More performance information and logs available.
   D. Does not require Application Engine servers.
5. Which one of the following elements is used in component queues for authentication?
   A. JNDI
   B. Active Directory
   C. JAAS
   D. Component Manager
6. True or False: A component is an operation that waits in a component queue for processing.

Figure 2-17. Review answers(2)

IBM Training

Review answers

7. Which of the following component integration elements sends a message to a queue in a JNDI QueueConnectionFactory?
   A. Process Task Manager
   B. Java adapter
   C. CE_Operations
   D. JMS adapter
8. True or False: The two Component Manager Frameworks, supported starting with IBM Case Foundation 5.2, are fundamentally the same.
9. You want to create a folder in the repository from a workflow step. Which component do you need to use?
   A. Java component
   B. JMS component
   C. CE_Operations component
   D. You can't create a folder from a workflow step

Figure 2-18. Review answers(3)

Unit 3. Using components

Estimated time

01:05

Overview

This unit/lesson shows you how to integrate CE_Operations and Java components into a workflow.

How you will check your progress

- Successfully complete the lesson exercises.

References

IBM Knowledge Center

`http://www.ibm.com/support/knowledgecenter/SSNW2F_5.2.1/welcome`

IBM Training

Why is this lesson important to you?

- You are designing a workflow application. The workflow needs to create a document in the object store and run a Java operation. You want to use component steps in the workflow to accomplish this goal.

Figure 3-1. Why is this lesson important to you?

IBM Training

Lesson objectives

- Integrate a CE_Operations component in a workflow
- Integrate a Java component in a workflow

Figure 3-2. Lesson objectives

IBM Training

Prepare the system for component use

- Prior to using a component in a workflow definition, the workflow administrator must complete specific configuration tasks.

Content Extended Operations (CE_Operations)

- Configure JAAS credentials for the CE_Operations component queue

Custom components (Java or JMS)

- Create component queues
- Configure component queues with the appropriate adapter
- Define component queue operations

Figure 3-3. Prepare the system for component use

Help paths

FileNet P8 Platform 5.2.1>Integrating workflow into document management>Designing workflows>About steps>About component steps

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.designerui.doc/bpfdh000.htm`

FileNet P8 Platform 5.2.1>Integrating workflow into document management>Configuring the workflow system>Configuring workflow queues>Manage component queues

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.configui.doc/bpfc056.htm`

Before you can define, transfer, and launch a workflow definition that contains component steps, the workflow administrator needs to complete specific configuration tasks, dependent on the type of component. The workflow administrator uses the Administration Console for Content Platform Engine to complete the configuration tasks.

Content Extended Operations (CE_Operations component)

- Configure the JAAS credentials for the CE_Operations component queue.

Custom components (Java or JMS)

1. Create a custom component queue in the isolated region.

2. Configure the component queue with the appropriate service adapter: Java or JMS.
3. Define component queue operations for the component.

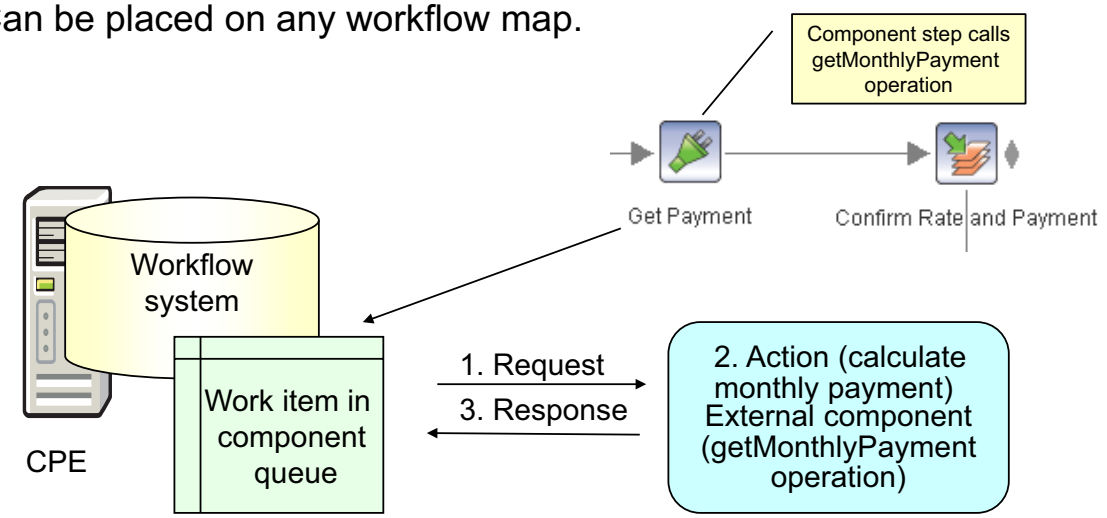
Note

You can also use the Process Configuration Console to create and configure a component queue. However, it is best to use Administration Console for Content Platform Engine because it is a web application and not a Java applet.

IBM Training

Component step

- Process Designer step type
  - Located in BPM palette
- Routes work to operations in custom Java or JMS components.
- Connects to a component queue configured for one or more operations
- Can be placed on any workflow map.



Using components

Figure 3-4. Component step

After the workflow system is prepared for component use, the workflow author places a component step on a workflow map to route work to an operation

A component step makes it possible to route work to operations in custom Java™ or Java Message Service (JMS) components. In the workflow, the component step connects to a component queue configured for one or more operations in the custom component. You can place a component step on any workflow map.

At run time, the work item is placed in the component queue that is configured for the component and the specified operation (in this case, the getMonthlyPayment operation) is called. The external component manages the operation action and sends a response to the Process Services of the Content Platform Engine (CPE). Only Java components can provide a return response.

The diagrams show part of a workflow map for a loan processing workflow that includes a component step named Get Payment. The Get Payment step calls a Java component operation that calculates and returns a monthly loan payment based on input3

IBM Training

What is a queue operation?

- A queue property that specifies input and output parameters for a task (function)
  - Used to define parameters based on the requirements for data type and direction
  - Typically used for tasks processed by automated programs, such as component steps
- One or more operations can be defined for a queue and used in a workflow.
  - An operation is defined using Administration Console for Content Platform Engine > Administrative > Workflow System
  - An operation is assigned to a step using Process Designer.
- An operation definition includes
  - Operation name
  - Operation parameters

Figure 3-5. What is a queue operation?

Help paths

FileNet P8 Platform 5.2.1>Integrating workflow into document management>Process applications concepts>Design and run workflows>About operations

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.user.doc/bpfwd009.htm`

FileNet P8 Platform 5.2.1>Integrating workflow into document management> Configuring the workflow system>Configuring workflow queues>Manage component queues >Defining component queue operations

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.configui.doc/bpfc013.htm`

FileNet P8 Platform 5.2.1>Integrating workflow into document management> Configuring the workflow system>Configuring workflow queues>Manage work queues> Defining queue operations

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.configui.doc/bpfc015.htm`

A queue operation is defined as a queue property that specifies input and output parameters for a function. A queue operation is used to exchange data between work items and the function that accesses the queue, such as a step processor, component, or automated process. The workflow author can use the operation parameters to require input from and provide output to a work item at a specific step.

Queue operations can be created for work queues, component queues, and user queues. In Java component queues, the operation is a method of the component. In JMS component queues, the operation is converted from a defined event on the messaging queue. In the activity for this lesson, you work with component queue operations.

When an operation and its parameters are being defined during Case Foundation solution development, the workflow author, system administrator, and application developer typically work together to determine the operation names and usage. A step processor references the operation in its step properties. The properties for each parameter must correspond to the properties expected by a step processor that is processing the work item in the selected queue.

When the administrator defines a queue operation, an operation definition includes the following items:

- Operation name, which identifies the operation in the workflow definition
- Operation parameters

Benefits of using queue operations

The purpose of an operation is to ensure that a set of data meets defined requirements (data type and data direction, also called data flow).

Typically, operations are defined on queues for work items that are going to be processed by automated programs, such as components. Operations can also be defined on queues from which participants process work items, including user inboxes, although this practice is less common.

Queue operations are used by the workflow author to control how data is presented to users or to automatic processes, when many different sources can be submitting information.

F_Operation system field at run time

At run time, the string system field, F_Operation, contains the name of the operation that is being processed or going to process next for a work item. You can use this system field in Process Administrator to determine the operation name for a work item at a step.

IBM Training

Operation parameters

- Operation parameter name does **not** have to match the workflow data field name.
- Operation parameter data type **must** match the expression data type.
- Data access assigned for a parameter controls the data flow to and from the operation.
  - Read Only: 
    Operation reads from the work item, when step is opened.
  - Write Only: 
    Operation writes to the work item, when step is completed.
  - Read/Write: 
    Both actions are allowed.

Figure 3-6. Operation parameters

Help paths

FileNet P8 Platform 5.2.1>Integrating workflow into document management>Designing workflows>About steps>About Activity steps>Activity step - parameter properties>Specify operation parameters

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.designerui.doc/bpfdh031.htm`

FileNet P8 Platform 5.2.1>Integrating workflow into document management>Designing workflows>About steps>About component steps>Select operations

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.designerui.doc/bpfdh151.htm`

The operation parameter data types that are defined in Administration Console for Content Platform Engine by the workflow administrator must map to workflow data field types defined in Process Designer in the step that uses the queue operation.

Data flow behavior

The page lists the three possible modes of data access that can be assigned for operation parameters and lists the corresponding icon that is displayed for each access mode.

- Read -The operation parameter is read from the work item.
- Write -The workflow field value is written to the operation parameter when the step is completed.
- Read/Write -A value that passes from the work item to the operation where it is processed, and then passes back to the work item.

Limitations on the expression assigned to a parameter

For a Read parameter, the parameter can be a simple or complex expression that consists of literals, data fields, attachments, or workflow groups.

For Write or Read/Write parameters, the parameter must be the name of a data field that is defined in the workflow definition. If the data field is an array, you can enter an index in the array, for example, array_field[index], where index is an integer expression.

For JMS component queue operations, parameters are read-only.

IBM Training

Use a component step in a workflow definition

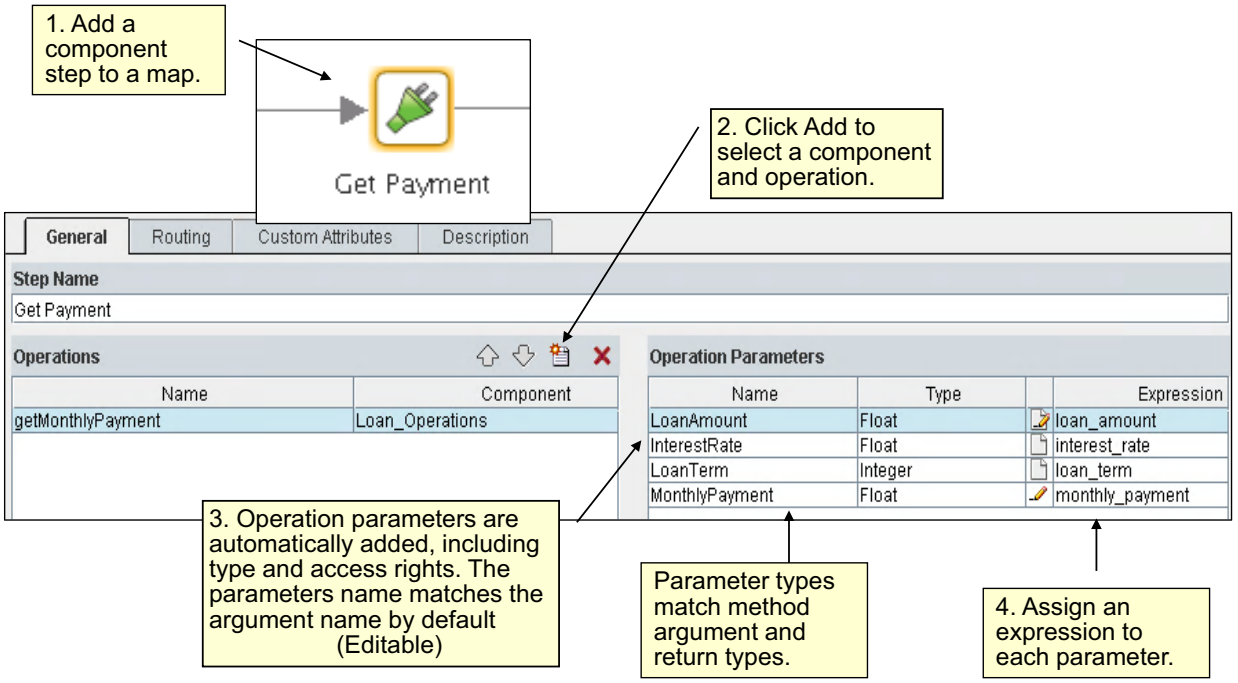


Figure 3-7. Use a component step in a workflow definition

Help paths

FileNet P8 Platform 5.2.1>Integrating workflow into document management>Designing workflows>About steps>About component steps> Component step – general properties

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.designerui.doc/bpfdh058.htm`

1. To use component operations, you add a component step to a workflow definition with Process Designer. A component step represents an activity in the workflow that is assigned to one or more operations in a component queue.
2. In Process Designer, in the General tab of the component step Properties pane, click **Add** to display the Operation Selection window. Select a component, and then select one or more operations for use in this step.
3. In the Operations list, select an operation to display the list of parameters. The operation parameters are automatically added to the Operation Parameters list, including the parameter types and access rights. Parameters cannot be added or modified for an operation.
4. In the Expression field, specify an expression for each parameter. You can select from a list of workflow data fields, create a new data field, or use the Expression Builder. Each parameter that is specified in the operation must be assigned an expression that evaluates to the correct data type in the Expression field.

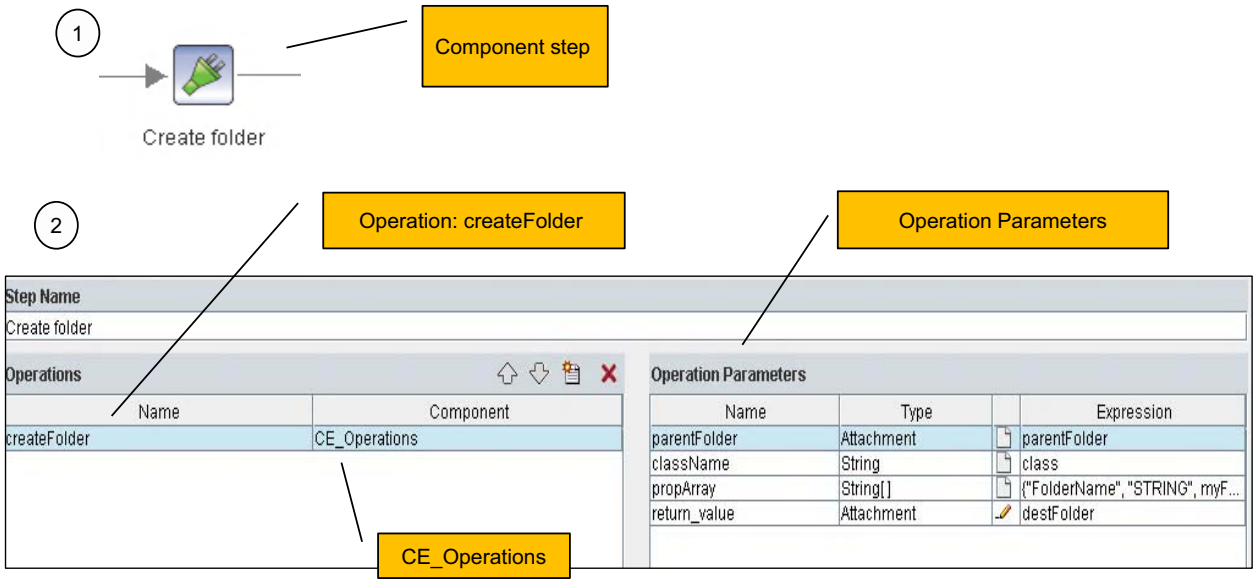
You can also specify the incoming and outgoing routing properties for the component step, similar to other step types in Process Designer.

The diagram shows an example component step along with its defined operation and operation parameters.

IBM Training

CE_Operations component

- Use a component step to call CE_Operations.
  - Specify the operation
  - Specify the expression for each operation parameter



Using components

Figure 3-8. CE_Operations component

To add a CE_Operations operation to a workflow, you use the same component step that you use for a Java component.

The screen captures on this slide show;

1. A component step from a workflow map, with the step name, Create folder.
2. The step properties window, which shows:
   - The CE_Operations component selected.
   - The operation, createFolder.
   - The Operation Parameters for the createFolder operation.

IBM Training

CE_Operations component

- Use a component step to call CE_Operations.
  - Specify the operation
  - Specify the expression for each operation parameter

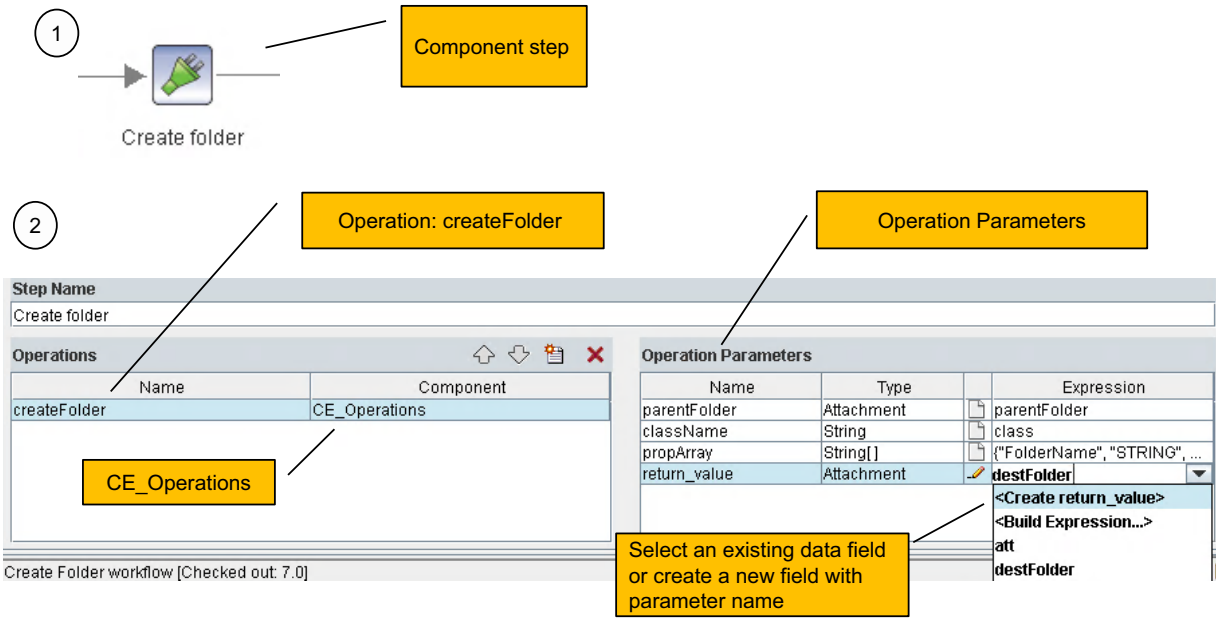


Figure 3-9. CE_Operations component

Help Paths

FileNet P8 Platform 5.2.1>Integrating workflow into document management>Process applications concepts>Design and run workflows>Using Content Extended Operations in a workflow definition

`https://www.ibm.com/support/knowledgecenter/SSNW2F_5.2.1/com.ibm.p8.pe.user.doc/bpfwd029.htm`

FileNet P8 Platform 5.2.1>Integrating workflow into document management>Process applications concepts>Design and run workflows>Using Content Extended Operations in a workflow definition>Creating a folder

`https://www.ibm.com/support/knowledgecenter/SSNW2F_5.2.1/com.ibm.p8.pe.user.doc/bpfwd025.htm`

FileNet P8 Platform 5.2.1>Developing FileNet P8 applications>Process Development

`https://www.ibm.com/support/knowledgecenter/SSNW2F_5.2.1/com.ibm.p8.pe.dev.java.ceops.doc/filenet/contentops/ContentOperations.html`

To add an extended content operation to a workflow, you use the same component step that you use for a Java component.

Review the content operation parameters first by referring to the Content Operations help path provided. Scroll down to the operation. For example, for createFolder, you see;

Method and Type	Method and Description
filenet.vw.api.VWAttachment	**createFolder**(filenet.vw.api.VWAttachment parentFolderAtt,java.lang.String className,java.lang.String[] propArray) Creates a folder, either at the root of the object store or as a child of an attached folder.

From this information, you know that the returned value is an Attachment. To call the method, you need:

- An attachment to specify the parentFolder (parentFolderATT).
- A string data field for the class (java.lang.String className).
- A string array to define the folder properties (java.lang.String[] propArray).

The screen captures on this slide show;

1. A component step from a workflow map, with the step name, Create folder.
2. The step properties window, which shows:
   - The CE_Operations component selected.
   - The operation, createFolder.
   - The Operation Parameters for the createFolder operation.

When you define the expression, you can choose from existing data fields or attachments, or create new data fields and attachments that use the parameter name.

Information

If the value field of the attachment, parentFolder, is left blank, then a value can be assigned by the launch user or by a participant during processing.

IBM Training

CE_Operations - use case scenarios

- File a document based on a response in a previous step.
- File a document based on the value of a specified document property.
- Create a document, folder, or custom object.
- Retrieve the content of an attached document.

Figure 3-10. CE_Operations - use case scenarios

Run content operations on FileNet Content Manager repositories from workflows, by using the Extended Content Operations, CE_Operations.

The slide lists some use case scenarios.

IBM Training

Instructor demonstration

- Use a component step in a workflow definition

Figure 3-11. Instructor demonstration

Demonstration notes

This demonstration shows you how to add and configure a component step in a workflow definition.

1. From the Workflow Author desktop, open Process Designer.
2. Drag a component step onto the map, to the left of the LaunchStep.
3. In the step Properties pane, add a component step Operation. For example, Loan_Operations>getMonthlyPayment.
4. Notice that the Operation parameters are added automatically, including type and access rights.
5. Assign an expression to each of the component step Operation Parameters.

IBM Training

Unit summary

- In this lesson you learned how to integrate a Java component in a workflow.

Figure 3-12. Unit summary

Figure 3-13. Activity: Using components

IBM Training

Activity introduction

- Integrate a CE_Operations component in a workflow
  - Configure the JAAS credentials for CE_Operations
  - Use a CE_Operations component in a workflow
  - Test the workflow
- Integrate a Java component in a workflow
  - Create a code module for a Java object
  - Create a Java component queue
  - Import the component queue operations
  - Use a Java component in a workflow
  - Test the workflow

Figure 3-14. Activity introduction

Use your Course Exercises Guide to perform the activities listed.

Unit 4. Web services

Estimated time

00:55

Overview

This unit/lesson shows you how to call an external web service from a workflow.

How you will check your progress

- Complete the review questions.
- Successfully complete the lesson exercises.

References

IBM Knowledge Center

`http://www.ibm.com/support/knowledgecenter/SSNW2F_5.2.1/welcome`

IBM Training

Why is this lesson important to you?

- You are designing a workflow application. The workflow needs to call an external web service. To accomplish this, you need to specify a partner link and use a Web Service Invoke step in a workflow definition.

Figure 4-1. Why is this lesson important to you?

IBM Training

Unit objectives

- Invoke a web service in a workflow.

Figure 4-2. Unit objectives

IBM Training

XML Web Service

- A programmatic, XML-based component that performs a specific set of related tasks
  - Interface conforms with accepted standards and specifications.
- A set of protocols defines how a service can be published, discovered, and used in a technology-neutral format.
  - WSDL – Web Services Description Language
  - XML – Extensible Markup Language
  - SOAP – Simple Object Access Protocol
  - HTTP – Hypertext Transfer Protocol
  - UDDI – Universal Description, Discovery and Integration

Figure 4-3. XML Web Service

Definition

An XML Web Service is a programmatic, XML-based, component that can run a specific set of related tasks and whose interface is in conformance with accepted web services standards and specifications. Web Services are platform and language independent and are used to develop and deliver distributed applications via the internet or intranet. Web services provide transactions that are secure and functions that can be replicated and reused.

The services contract for a web service is specified by its interface, which is defined in the WSDL. Providers of the web service implement the interface, and the consumers of the service use the interface. Other systems, both external and internal to the organization, can interact with the web service by using SOAP messages, typically over HTTP.

Web Services specifications

Several technology vendors defined a comprehensive set of specifications that provide an infrastructure for enterprise-class web services interoperability.

Standards

A number of standards define how a web service can be published, discovered, and used in a technology-neutral, standard format. Among these standards are the following.

- WSDL is the interface definition for web services. The standard format for describing a web service. Expressed in XML, a WSDL definition describes how to access a web service and what operations it performs. A WSDL file is an XML document that describes a set of SOAP messages and how those messages are exchanged. WSDL is seen (along with SOAP and UDDI) as one of the three foundation standards of web services.
- XML is a widely used standard from the World Wide Web Consortium (W3C) that facilitates interchange of data between computer applications. XML is a programming language that enables designers to create their own customized tags to indicate specific information. It is a flexible way to create standard information formats and share both the format and the data on the World Wide Web. In web services, the messages are written by using XML.
- SOAP is the messaging protocol for XML Web Services. The specification defines the XML format for messages. SOAP functions like an envelope for sending messages over the internet. The SOAP envelope contains the body of the XML message, identifies who can process it, and describes how to process it. One advantage of SOAP is that it is implemented on various hardware and software platforms. Therefore, SOAP can be used to link disparate systems both within and outside of an organization.
- HTTP is the underlying protocol that is used by the World Wide Web. HTTP defines how messages are formatted and transmitted, and what actions web servers and browsers need to take in response to various commands. This protocol is the transport layer for XML Web Services.
- UDDI is a specification for maintaining standardized directories of information about web services, and recording their capabilities, location, and requirements in a universally recognized format. UDDI is the means for discovering and publishing a web service. A UDDI registry is a standardized repository for locating and publishing web services. It allows for the Search and Discovery of web services and is accessed by using SOAP messages. UDDI and WSRR (WebSphere Services Registry and Repository) were deprecated, starting with IBM Case Foundation 5.2.1.

Why use Web Services?

Some major features and benefits of web services are:

- Platform independence enables interoperability among distributed applications that span diverse hardware and software platforms. The cross-platform and cross-language data model (XML) of Web Services facilitates heterogeneous distributed applications.
- Web Services provides easy and widespread access to applications through firewalls by using standard web protocols.
- Web Services are self-describing. That is, they contain a service description for defining how to use the services.

IBM Training

Web services example

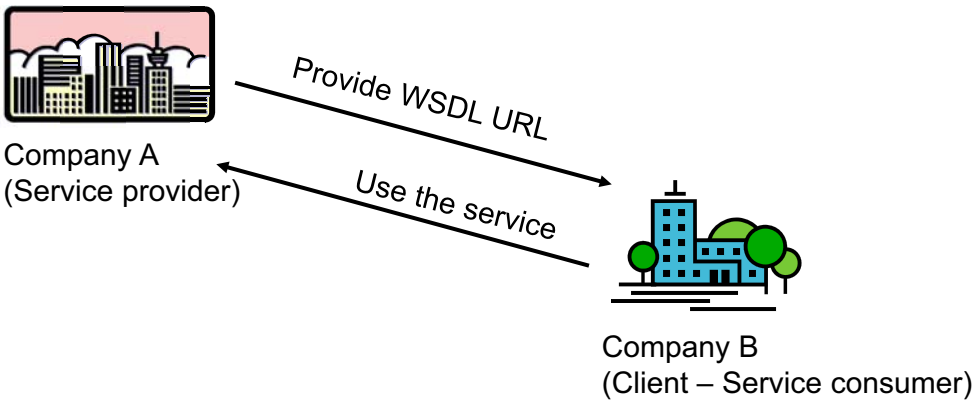


Figure 4-4. Web services example

In a typical use case for web services, Company A has an existing service that it wants offer to other companies. Company A provides the WSDL URL to companies that it wants to share the web service with.

Company B finds that it needs a particular service. Company B obtains the WSDL URL from Company A to use of the web service.

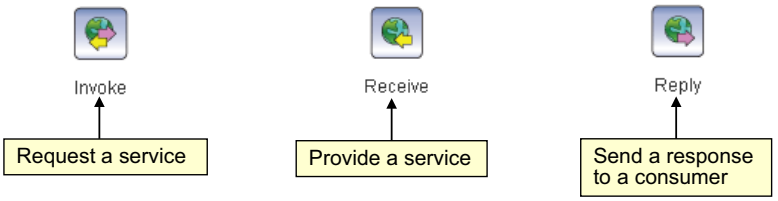
In this example, Company A offers a global weather-reporting web service. Company B runs a travel reservation system that uses the service in its business processes.

The diagrams illustrate this example use case.

IBM Training

Web services in Process Designer

- Web Services Palette contains the following steps:



- At design time
  - You define partner links.
  - You add Web Services steps to maps.
  - You use the partner links to link steps to Web Services.
  - You map workflow parameters to service parameters.
- At run time
  - Web Services are called.

Web services

Figure 4-5. Web services in Process Designer

In Process Designer, the Web Services Palette contains three steps for including web services functions in a workflow; Invoke, Receive, and Reply. You can also access these web services system functions by placing a System step on a workflow map and selecting the appropriate system function from the list of available functions. Before you can use an Invoke, Receive, or Reply function, you must specify the partner link in the Workflow Properties. The partner link is a specification that is used to access the web service. The partner link consists of a name and the web service WSDL URL for Invoke functions, or a name and port definition for Receive functions.

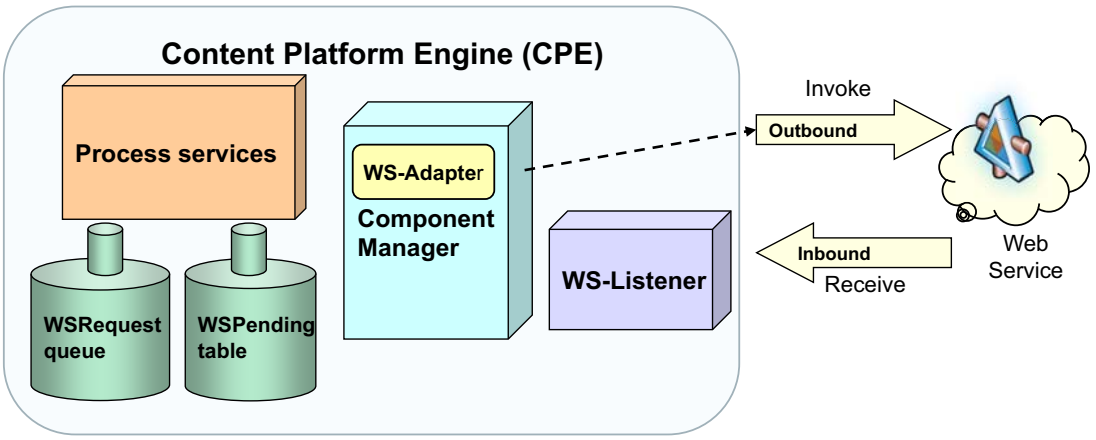
The three web services system functions are:

- **Invoke** requests a web service from a selected partner link. You must specify any data fields and attachments that are required for input to the services, and any data fields or XML fields required for returned values. If attachments are to be returned, you must also specify the default folder in your object store or library where returned attachments are stored.
- **Receive** provides a service in response to a web services request. To make this service available to other FileNet workflows, you need to transfer this workflow to the Process Engine. To make this service available to others, you publish it to a UDDI or WSRR Registry. The placement of a Receive step as the first step in a workflow effectively publishes the workflow as a web service after the workflow is transferred.
  - Note, UDDI and WSSR registries are not widely used and deprecated, starting with IBM Case Foundation 5.2.1.

- **Reply** sends a response to a web service consumer associated with a previously accepted Receive system function. At design time, the workflow designer uses the partner link to link the web services process steps to the web service. The designer maps workflow parameters to the service parameters. At run time, the web services are called.

Web services architecture

- Interaction with XML Web Services is handled by the Content Platform Engine.
- Supports both polling and event-driven modes
- WS-Adapter and WS-Listener provide interface to outbound and inbound Web Service transmissions.



Web services

Figure 4-6. Web services architecture

Starting with the 5.2.x release, interaction with XML Web Services is handled by the Content Platform Engine (CPE). This communication is transparent to both the system administrator and the user. The architecture supports both polling and event-driven modes for the Component Manager. Outbound requests for web services are handled through WS-Adapter, running as part of the Component Manager Framework. Inbound requests for web services are handled by WS-Listener, running in the CPE. The CPE uses process services to interact with the WSRequest queue and a WSPending table, which are system-managed data structures that are used in processing the requests for web services. The workflow author and system administrator do not directly access or manage these region data structures. However, in Process Administrator, work items might appear in the WSRequest queue when they are waiting to be sent to a web service.

IBM Training

Steps to invoke a Web Service

1. Administrator uses Process Configuration Console to enable direct entry of WSDL URL.
2. Workflow author configures the Workflow Properties.
   - Define an Invoke partner link:
     - Select the Web Service (WSDL).
     - Select the Port Type.
3. Workflow author adds an Invoke step to the workflow map.
   - Select the operation.
   - Assign parameters.

Figure 4-7. Steps to invoke a Web Service

Help path

FileNet P8 Platform 5.2.1>Integrating workflow into document management>Process applications concepts>Design and run workflows>About Web Services in workflows

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.user.doc/bpfwd017.htm`

The overall procedure for invoking a web service is:

1. The workflow administrator uses the Process Configuration Console or Process Designer Configuration view to configure the UDDI Registry List or the WSRR Registry List for the isolated region. This registry list includes an entry for the web service.
2. The workflow designer uses Process Designer to configure the workflow properties for the workflow definition. The Invoke Partner Link is defined by selecting the WSDL for the web service and its Port Type. Other workflow data fields are defined, as needed for the web service.
3. The workflow designer adds an Invoke step to the workflow map. In the Invoke step Properties, the partner link, and operation are selected. The workflow designer assigns workflow data fields to the web service parameters, as required by the web service.

Define an Invoke partner link

- In Process Designer, define a Web Services partner link in Workflow Properties > Web Services > Partner Links.
  - Type: Invoke
  - WSDL URL
  - Port Type

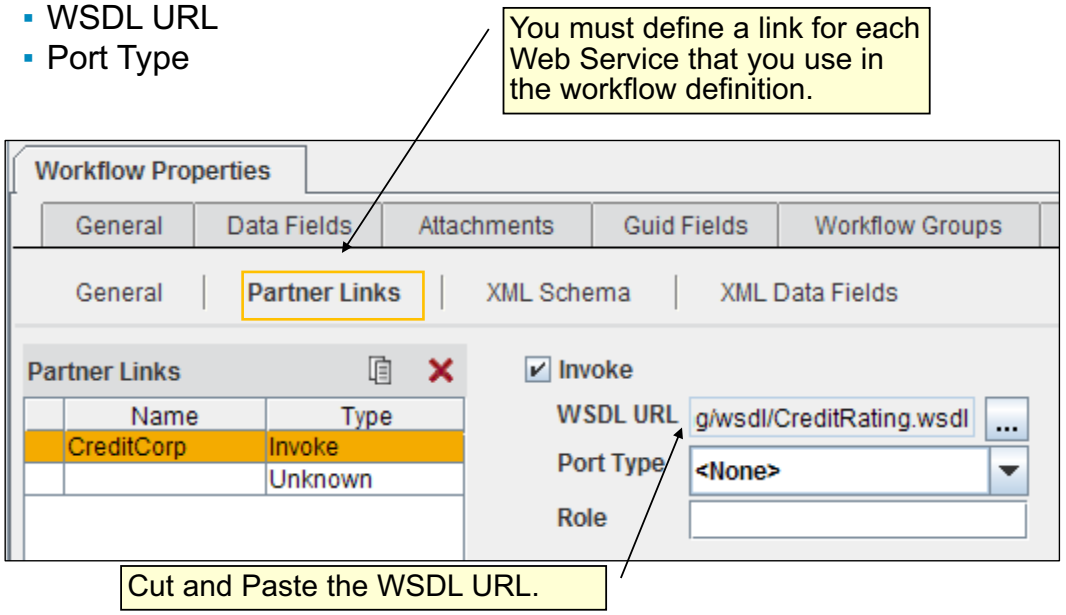


Figure 4-8. Define an Invoke partner link

Help path

FileNet P8 Platform 5.2.1>Integrating workflow into document management>Designing workflows

Define workflow properties>Workflow properties - Web Services>Web Services – Partner Links

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.designerui.doc/bpfdh178.htm`

In Process Designer, you must define the Workflow Properties for the web service that you want to invoke.

If a workflow definition invokes or receives requests for web services, you specify those web services on the Partner Links tab. If XML data fields and XML schemas are required for Invoke, Receive, or Reply tasks, define them on the appropriate tab, XML Data Fields or XML schemas. On the Web Services General tab, you can specify a folder in your object store or library where any incoming attachments are stored.

In the Partner Links tab, cut and paste the WSDL URL provided by the web service provider.

IBM Training

Use Invoke step in a workflow map

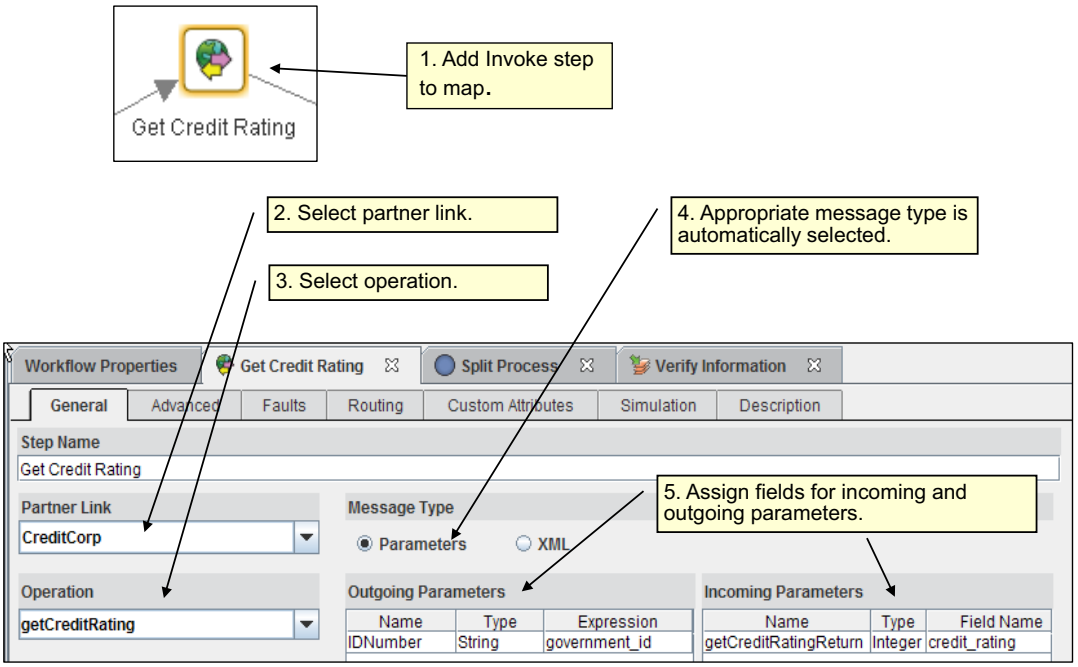


Figure 4-9. Use Invoke step in a workflow map

Help path

FileNet P8 Platform 5.2.1>Integrating workflow into document management>Designing workflows>About steps>System functions>Web services>Invoke system function

`https://www.ibm.com/support/knowledgecenter/en/SSNW2F_5.2.1/com.ibm.p8.pe.designerui.doc/bpfdh106.htm`

Place an Invoke step on a map where you want to invoke a web service. The Invoke step is found in the Web Services Palette. The diagrams show how to specify the General step properties for an Invoke system step.

Invoke message types

Two message types can be uses to invoke a web service:

- Parameters Message Type - With this option you can invoke or create web services without requiring knowledge or understanding of XML or XML schemas. This mode is simple to use and is similar to using a Work queue operation.
- XML Message Type - The XML message type requires XML and XML schema knowledge, and lets you deal with complex XML messages that might not be suitable for the parameters message type. The workflow author must provide the XML and XML schema for the message, and parse the return XML.

Message type is automatically selected

Process Designer selects the message type automatically when you select the Partner Link and operation for the web service. The selection is based on the complexity of the WSDL message definition for the operation. Parameters can be overwritten by the workflow author by using XML. However, if XML is selected, the workflow author cannot override this selection by selecting Parameters.

Example: Invoking a Web Service

- Loan Processing workflow
  - Step on Process Loan Map gets a customer's credit rating using a Web Service.

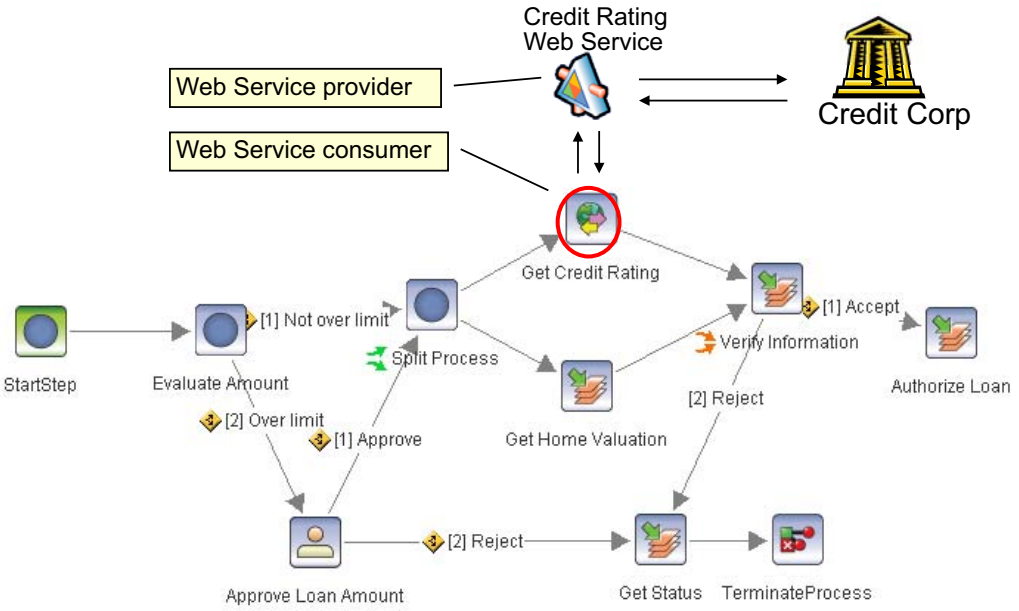


Figure 4-10. Example: Invoking a Web Service

In the Loan Processing workflow, the customer's credit rating is required in the Process Loan Map submap to continue processing and to evaluate the customer's loan application.

In the diagram, the Process Loan Map submap of the Loan Processing workflow contains an Invoke system step to call a web service. At the Get Credit Rating step, an Invoke system function calls a Credit Rating web service to get the credit rating. The customer's ID number is sent to the web service and a credit rating is received from the web service.

The credit rating is evaluated at the next step and then routed to the appropriate steps to authorize the loan based on the credit rating.

In this example, Credit Corp is the web service provider and the Loan Processing workflow is the web service consumer.

IBM Training

Other considerations when using Invoke

- Reliable messaging
  - Option is found in Advanced step properties of Invoke step.
  - Enable this option **only** if a request requires it.
  - If enabled, define fault handling procedures.
- Timeout handling
  - Option to specify timeout expression in Advanced step properties.
  - Submap is executed if timeout occurs.
- Fault handling
  - Options are available on Faults tab in Invoke step properties.
  - Specify String or XML field to hold fault message and submap to execute if fault occurs.
  - Specific and general faults can be specified.

Figure 4-11. Other considerations when using Invoke

Reliable messaging

IBM Case Foundation supports both reliable messaging and non-reliable messaging requests. In most cases, the standard, non-reliable messaging mode is adequate for a web service. This mode requires at most two messages to complete a transaction.

The WS-ReliableMessaging specification describes a protocol that allows messages to be delivered reliably between distributed applications. In Process Designer, the interface for using WS-ReliableMessaging is available on the Advanced tab of the Invoke step Properties. The Receive and Reply system functions require no special user configuration. For a request to an external web service, you must know whether the web service supports WS-ReliableMessaging. If you specify reliable messaging and this functionality is not available in the invoked web service, a runtime error occurs causing the work item to go to the Malfunction system map.

WS-ReliableMessaging uses a minimum of five messages for every request, plus more messages if retransmissions are necessary. Therefore, use WS-ReliableMessaging only for requests that require this type reliable messaging.

When you use reliable messaging, you must specify fault handling in the Invoke system function to handle any problems with the request. Fault handling is specified in the Faults tab in the Invoke step Properties.

Faults

On the Faults tab of the Invoke step properties, you can specify string fields or XML data fields to hold a fault message. Specify submaps to run for faults that are specified by the web service, or for other faults. Specific type faults are named and defined in the web service.

IBM Training

Unit summary

- Invoke a Web Service in a workflow.

Figure 4-12. Unit summary

Review questions

1. Fill in the blanks with the correct letter to complete the following Web Services architecture diagram.
   A. Component Manager
   B. Process services
   C. WS_Adapter
   D. Content Platform Engine

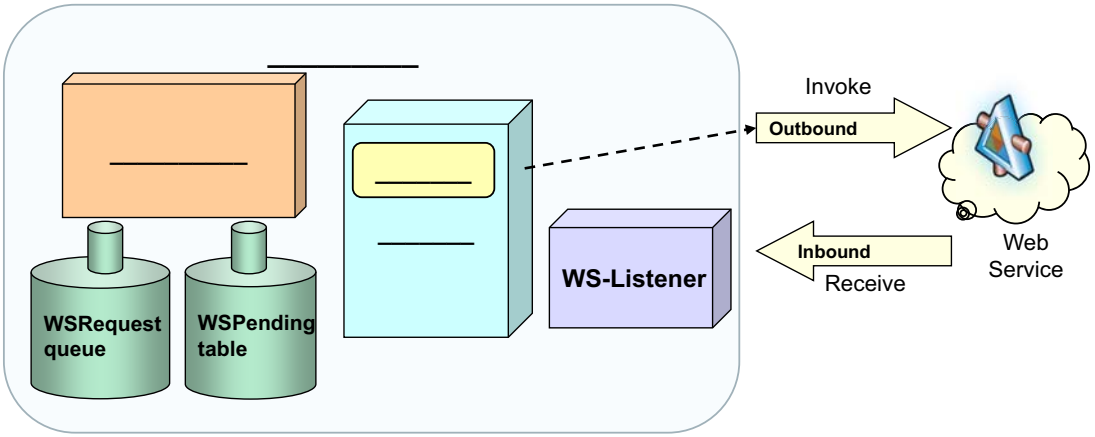


Figure 4-13. Review questions

Review questions

2. Which Web Services Palette system function requests a service?
   A. Invoke
   B. Receive
   C. Reply
   D. Return

3. How does a workflow call a Web Service?
   A. Through a system step that is configured to run through a WS-Adapter
   B. Through a call to a Java adapter in a multi-participant step
   C. By executing a component step in a process map
   D. By executing an Invoke step in a process map that is configured with a partner link pointing to the Web Service

Figure 4-14. Review questions(2)

Review answers

1. Fill in the blanks with the correct letter to complete the following Web Services architecture diagram.
   A. Component Manager
   B. Process services
   C. WS_Adapter
   D. Content Platform Engine

   The answer is:

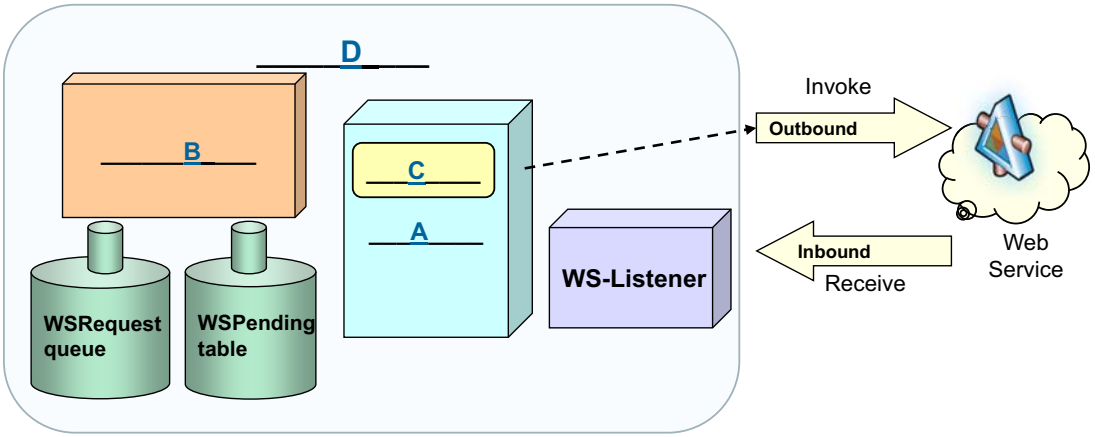


Figure 4-15. Review answers

IBM Training

Review answers

2. Which Web Services Palette system function requests a service?
   A. Invoke
   B. Receive
   C. Reply
   D. Return

   The answer is A.

3. How does a workflow call a Web Service?
   A. Through a system step that is configured to run through a WS-Adapter
   B. Through a call to a Java adapter in a multi-participant step
   C. By executing a component step in a process map
   D. By executing an Invoke step in a process map that is configured with a partner link pointing to the Web Service

   The answer is D.

Figure 4-16. Review answers(2)

Figure 4-17. Activity: Invoke a Web Service

IBM Training

Activity introduction

- Invoke a web service from a workflow
  - Enable option to enter WSDL URL directly for a partner link
  - Invoke a web service in a workflow
  - Test the workflow

Figure 4-18. Activity introduction

Use your Course Exercises Guide to perform the activities listed.

IBM Training