



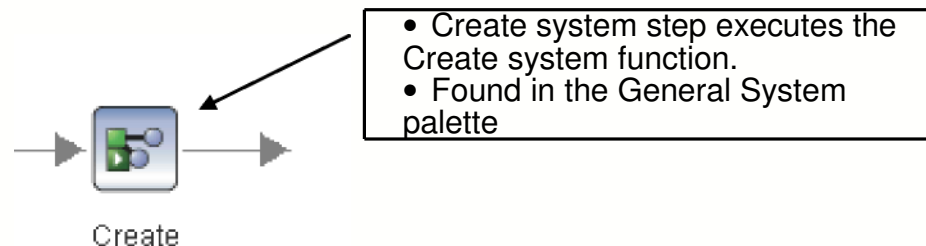
Control of Multiple Workflows

Workflow Designer
IBM FileNet Business Process Manager



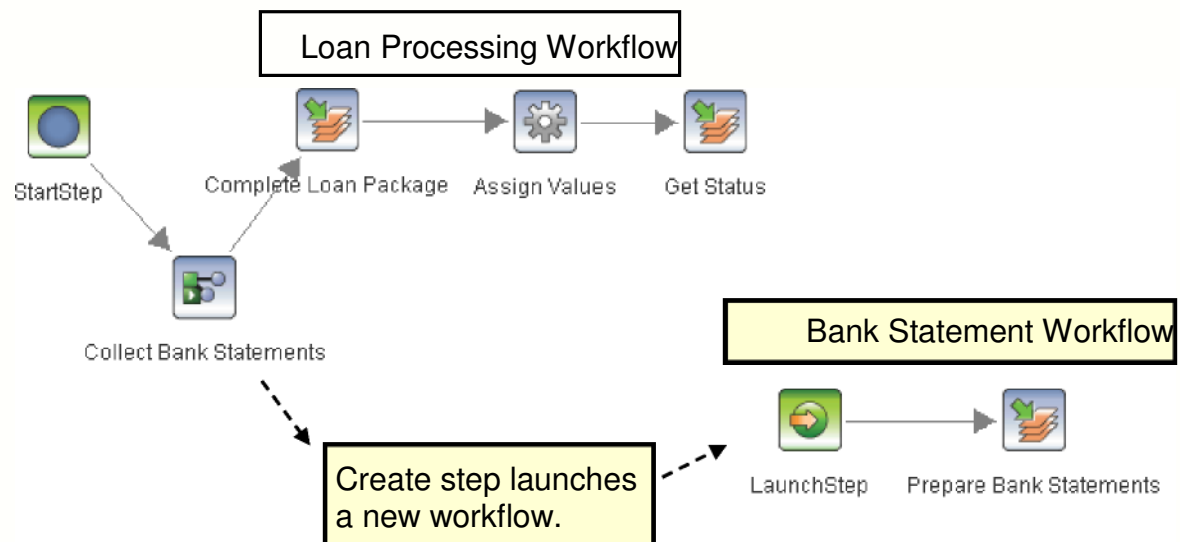
Create system function

- Launches a workflow using a specific workflow definition
 - After being launched, the created workflow executes independently of the creating workflow.
- Allows selection from a list of workflow definitions already transferred
- Assigns values to workflow properties



Example: Using a Create step

- Loan Processing Workflow contains a Create system step.
 - Collect Bank Statements step launches a Bank Statement workflow.
 - Statements are collected and submitted in the new, created workflow.



Assigning Create step properties

- Example: Create system step called Collect Bank Statements



Collect Bank Statements

1. Select a workflow from a list of all previously transferred workflows.

1. Select a workflow field.

Step Name	
Collect Bank Statements	
Workflow Name	
Bank Statement Workflow	
Assignments	
Name	Expression
CustomerID	customer_name

1. Enter a valid expression to assign to the selected field.

Assignment of workflow fields

- Workflow fields from both the existing and the created workflow are displayed in the Assignments table.
- You can assign fields in both created and existing workflows.
- Use the Expression Builder to build the expression assigned to each field or directly type the text expression.

Workflow Name	
Bank Statement Workflow	
Assignments	
Name	Expression
CustomerID	customer_name
BankStatements	
CustomerID	
StatementStatus	
\$bank_statements	
\$condition_row_number	
\$credit_rating	
\$customer_name	
\$down_payment	

Example:
CustomerID is in created workflow.
customer_name is in existing
workflow.

Design considerations for using Create

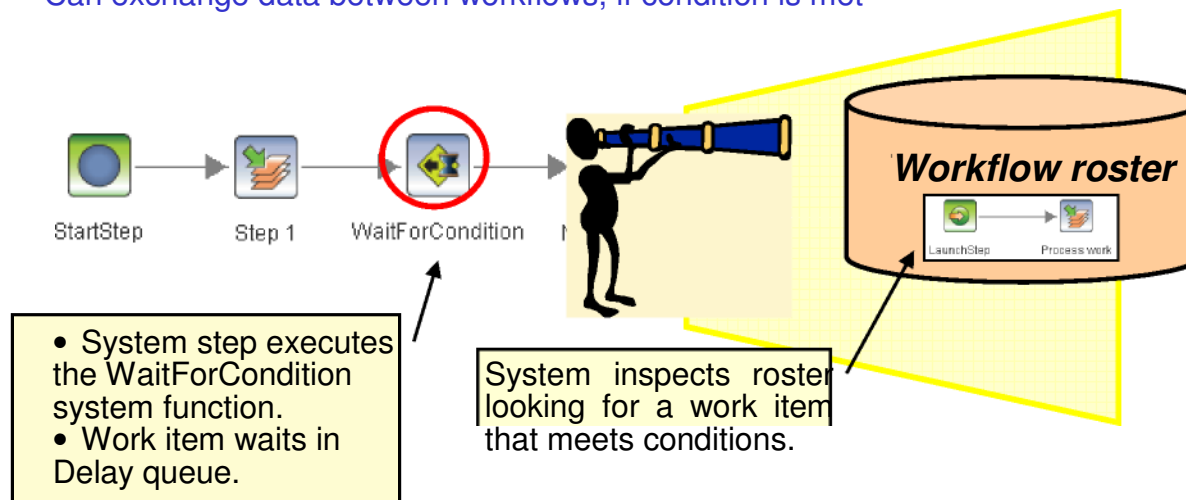


- Launch step of the created workflow cannot contain a response.
 - Results in an error when transferring the **creating** workflow
- Created workflow cannot use F_Originator as a participant or variable.
 - F_Originator contains a null value in workflows launched with Create.
- Created workflow references the creating workflow.
 - Subsequent transferred revisions to the referenced workflow definition must be compatible with Create function used in the creating workflow.
 - Example: You cannot delete a data field in the created workflow definition that is assigned in the Create system step.

WaitForCondition system function

Workflow processing is suspended and work item waits for another work item that meets a specified condition.

- Wait continues until condition met or timeout period expires.
- At run time, only one work item can satisfy the specified condition. – Can define multiple conditions to wait for
- Can exchange data between workflows, if condition is met



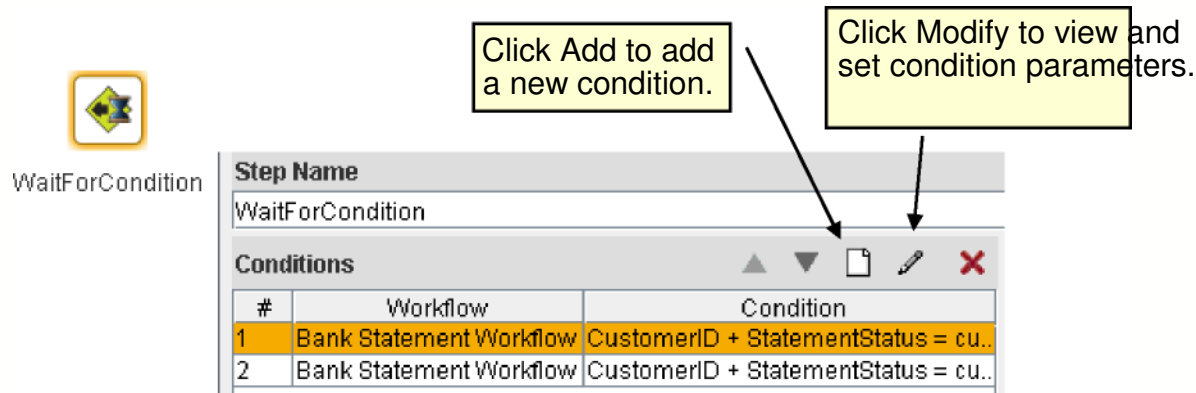
Condition identifier



- Used to define and identify the waited-for item
 - The waiting work item waits for a work item with a specific condition identifier value.
- In the waited-for workflow Advanced Workflow Properties, you define the condition identifier.
 - Can be a complex expression to identify a unique, specific item - For example: `CustomerID + convert (IDNumber, string)`
- At run time, the value of condition identifier is compared to the WaitForCondition parameters.
 - Data field types must be the same.

The Conditions table

- In step properties, specify one or more conditions waited for.
 - Conditions displayed in a table form.
 - Each row in the table references one condition.
 - For each condition, you must configure parameters.
- Conditions are evaluated using logical OR. – Any one of the conditions can satisfy the WaitForCondition.
 - Evaluated in order



WaitForCondition

Step Name
WaitForCondition

Conditions

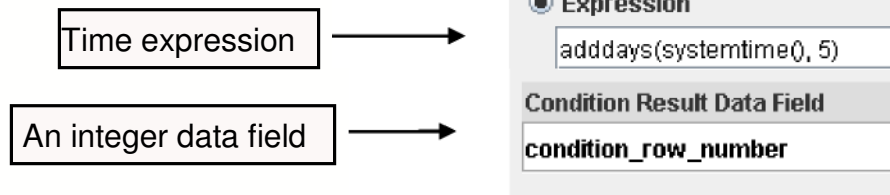
#	Workflow	Condition
1	Bank Statement Workflow	CustomerID + StatementStatus = cu..
2	Bank Statement Workflow	CustomerID + StatementStatus = cu..

Click Add to add a new condition.

Click Modify to view and set condition parameters.

Timeout and condition results

- Specify how long processing is suspended and work item waits in the Delay queue.



- Specify the condition result field.
 - If timeout period expires without meeting condition, 0 value assigned.
 - If condition is met, the row number of condition is assigned and the optional condition map is executed.
- Test the condition result field in a later route or step in the waiting workflow to determine what happened.

Condition parameters

- For **each** condition, you must specify items 1 and 3.
- Assignments and map are optional.

The screenshot shows the 'Condition Parameters' dialog box. It has several sections: 'Condition', 'Expression Data Type', 'Condition Identifier', 'Operator', 'Expression', 'Action', 'Assignments', and 'Map'. Annotations with arrows point to specific fields:

- Annotation 1: '1. Name of waited-for workflow' points to the 'Workflow Name' field, which contains 'Bank Statement Workflow'.
- Annotation 2: '2. Condition identifier is automatically supplied by the system from waited-for workflow.' points to the 'Condition Identifier' field, which contains 'CustomerID + StatementStatus'.
- Annotation 3: '3. Operator and expression to compare to condition identifier' points to the 'Operator' field (containing 'is equal') and the 'Expression' field (containing 'customer_name + "..."').
- An annotation points to the 'Assignments' table, which has two rows: 'bank_statements' with expression 'BankStatements' and 'status' with expression 'StatementStatus'.
- An annotation points to the 'Map' field, which contains '<None>'. The text next to it says: 'If this condition is met, then Fields are assigned. Map is called.'

Name	Expression
bank_statements	BankStatements
status	StatementStatus

Example: Loan processing workflow



- Two workflows exchange data values
 - Bank Statement Workflow – the waited-for workflow
 - Loan Processing – the waiting workflow uses WaitForCondition
- Data field usage **Bank Statement Data flow Loan Processing**

CustomerID Condition Identifier customer_name

StatementStatus Condition Identifier status

BankStatements \neg * bank_statements

- Expressions used to match the workflows
- Condition Identifier of waited-for workflow:

CustomerID + StatementStatus

· WaitForCondition expression in waiting workflow:

customer name + "Have bank statements"

Dependent workflow processing

Using dollar sign (\$) variables in WaitForCondition

- The dollar sign (\$) symbol in Assignments reverses the assumption about the source of the field.

Condition Parameters

Condition

Workflow Name

Bank Statement Workflow

Expression Data Type

String

Condition Identifier Operator Expression

CustomerID + StatementStatus is equal customer_

Action

Assignments

Name	Expression
bank_statements	BankStatements
status	StatementStatus
MAX_AMOUNT	
monthly_payment	
purchase_price	
retry_option	
status	
\$BankStatements	
\$CustomerID	

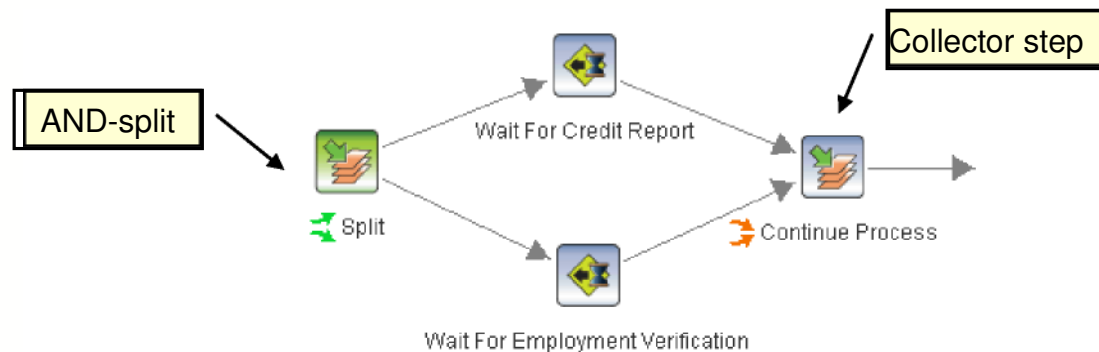
In Expression Builder:
\$Name = Waiting workflow field
Name = waited-for workflow field

Attachment Fields (Waited for workflow)
BankStatements

In Name field:
Name = Waiting workflow field
\$Name = waited-for workflow field

Example: Multiple conditions using AND logic

- Use case scenario
 - _ Credit report and employment verification are both required before the loan processing can be completed.
- Design solution example
 - _ Combine AND-split and collector steps with WaitForCondition steps.
 - _ All conditions must be met to continue processing of waiting workflow.
 - _ Use data field merging to ensure field values are assigned correctly.



WaitForCondition design considerations

- Initial polling for work items to satisfy the WaitForCondition might be resource intensive.
- Consider design strategies to speed search and improve performance.
 - . Use separate roster for work items in the condition.
 - . Delay execution of WaitForCondition to off-peak hours.
- Beware of using data fields of type float for condition identifier. . Exact value cannot be guaranteed.
 - . Use complex expression to approximate value.
 - . For example, if testing for 10.0, use an expression: `data_field < 10.1 and data_field > 9.9`
- Timers and WaitForCondition . An expired timer preempts WaitForCondition processing.

What is a workflow collection?



- Multiple workflow definitions contained in a single file . Used to create a logical group of processes
 - . Stored as an XPDL file
 - . One process is designated as the main workflow.
 - . Each individual process is a workflow and can be managed separately or together as a collection.
- Advantages
 - . Collects a set of related processes for ease of management and purposes of documentation
 - . Expedites transfer, deployment, and maintenance of dependent workflows
 - . Supports import of a multiworkflow process in XPDL 2.0 format from BPMN modeling tools

Workflow definition files

- Two different workflow definition file formats are supported in Process Designer and on the Content Engine:

- .PEP file is used for a single process in a workflow definition file.



- .XPDL file is used for a workflow collection containing two or more processes.



- In addition, from Process Designer you can import and export XPDL 2.0 files to be used with external BPMN modeling tools.

Build a workflow collection



- In Process Designer, use File > Insert command. [Allows you to add a workflow definition to the collection](#) [Can be used only with PEP files](#)
- Specify the main workflow in the collection.
 - [Action > Set as Main Workflow](#)
 - [When you launch the collection, the main workflow is launched.](#)
- Specify the workflow collection properties. [File > Workflow Collection Properties](#)

Workflow collection properties



- Specify a name for the collection.
 - Used as the default file name when saving
- Specify the Application Space.
 - Customized application spaces are used with ECM Widgets or other applications.
 - Default application space is DefaultApplication.
- Specify an optional description.

View and edit a workflow in a collection

- Use View > Workflows option on main menu and select a workflow name from the list
- Only one workflow in a collection can be edited at a time.
Within the workflow, only one map is displayed in the map pane.
- Each workflow in the collection is displayed in a separate workarea tab.

Main workflow is indicated by this symbol:



Switch between workflows by clicking a tab.

Manage a workflow collection



- File menu options
 - Apply to the **entire** workflow collection if the Include options are enabled on the Workflow Properties > Advanced tab (default)
 - Are used to validate, transfer, and save the entire collection
 - Are used to launch the main workflow in the collection
- Action menu options
 - Apply **only** to the currently displayed workflow definition
 - Are used to validate, transfer, launch, and save **only** the current workflow definition
- Add a new workflow definition to the collection. - File > Insert > New Workflow
- Remove a workflow definition from the collection. - Action > Remove Workflow (cannot be used on the main workflow)