



PUBLIC

How-To Send an E-mail Notification During the Governance Process

Applicable Releases:

From MDG 7.0 and from SAP S/4HANA 1511

Version 2.1

September 2024

Document History

Document Version	Description
1.0	First official release of this guide
1.1	Small corrections
1.2	Small corrections
2.0	Adapt current template, elaborate step-by-step description (January 2021)
2.1	Add option based on a new workflow (September 2024)

Table of contents

1.	Introduction.....	4
2.	Email notification using a sub-workflow.....	4
2.1.	Create a sub-workflow.....	4
2.2.	Create a new change request type and configure change request steps.....	15
2.3.	Configure rule-based workflow for the new change request type.....	18
2.4.	Try-out.....	29
3.	Email notification using BAdI System Caller Method.....	32
3.1.	Create a service name in IMG.....	32
3.2.	Create a BAdI implementation for BAdI ‘Rule-Based Workflow – Call System Method’.....	34
3.3.	Create a new change request type and configure change request steps.....	42
3.4.	Configure rule-based workflow for the new change request type.....	42
3.5.	Try-out.....	48
4.	Email notification using a new workflow.....	51
4.1.	Create a new workflow.....	51
4.2.	Modify the workflow container.....	52
4.3.	Modify the activation step	56
4.4.	Implementation of IF_SWF_IFS_WORKITEM_EXIT~EVENT_RAISED.....	63
4.5.	Add an email notification.....	67
4.6.	Define change request step numbers	70
4.7.	Create a new change request type and configure change request steps.....	71
5.	Additional Information	73
5.1.	Further Reading.....	73
5.1.1.	Information on SAP MDG on SAP S/4HANA.....	73
5.1.2.	SAP Roadmap Explorer.....	73
5.1.3.	Related Information.....	73
5.2.	SAP Notes	73

1. Introduction

SAP Master Data Governance (MDG) provides business processes to find, create, change, and mark master data for deletion. It supports the governance of master data in a central hub and the distribution to connected operational and business intelligence systems.

The processes are workflow-driven and can include several approval and revision phases, and the collaboration of all users participating in the master data maintenance.

MDG offers change request (CR)-based processing of master data with integrated workflow, staging, approval, activation, and distribution.

This guide addresses the situation that it is required to notify a person via email (for example the requestor) when the workflow is in a certain step or completed. Two solutions are described. The first one is via configuration only whereas the second one also incorporates coding which provides more flexibility in the way the email notification is formatted as well as the contained information. Both solutions are based on the rule-based workflow.

This guide is not meant describe how to enable notification for work items. Rather, this guide is for non-work item notifications.

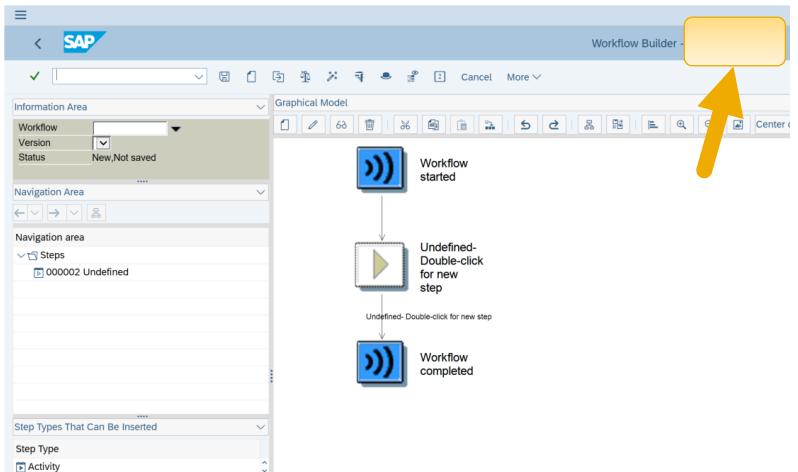
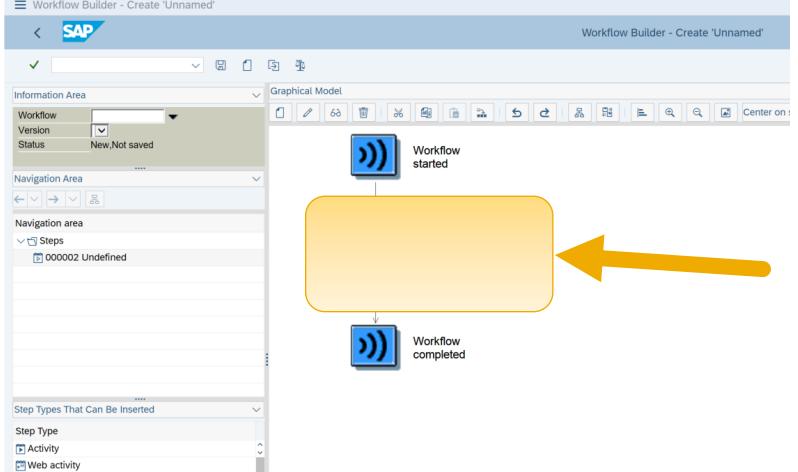
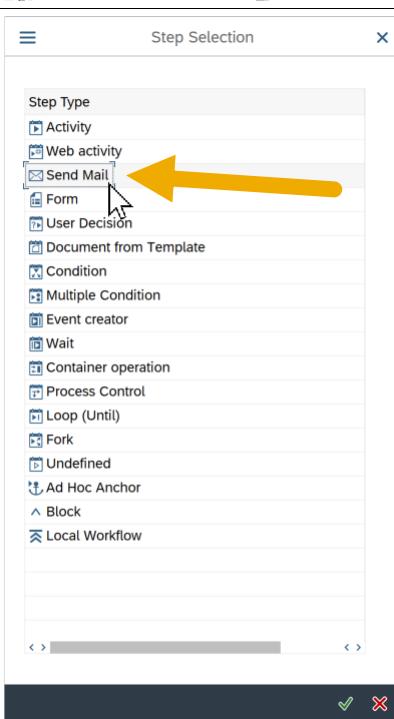
2. Email notification using a sub-workflow

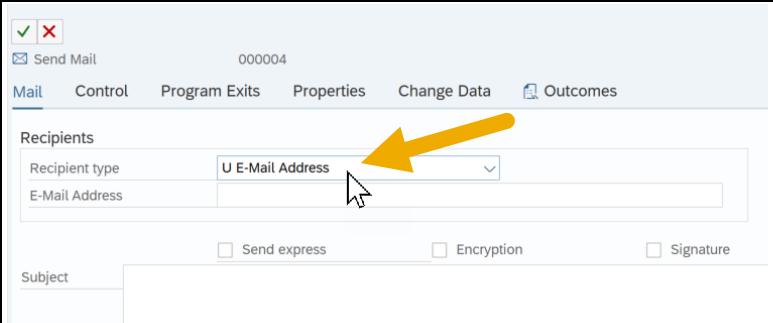
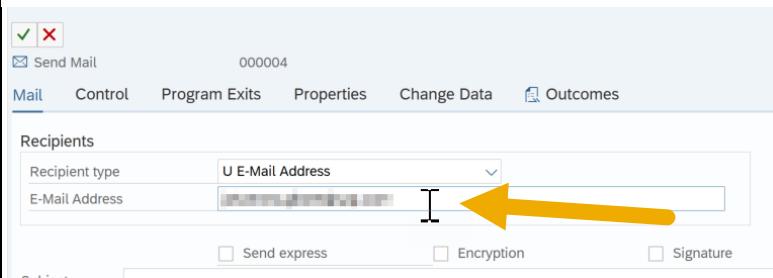
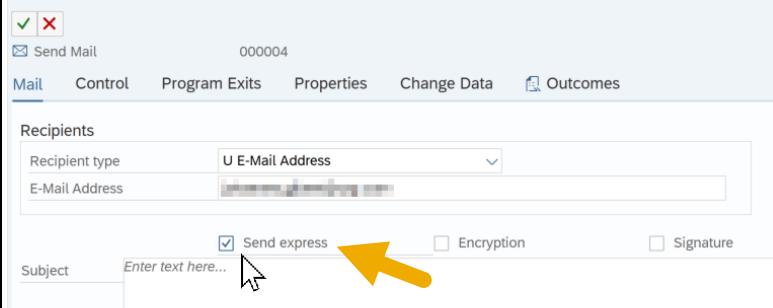
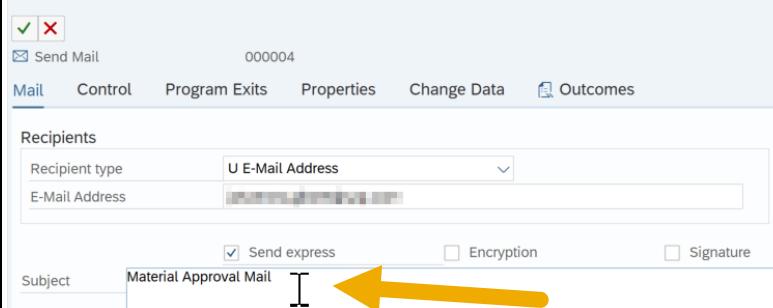
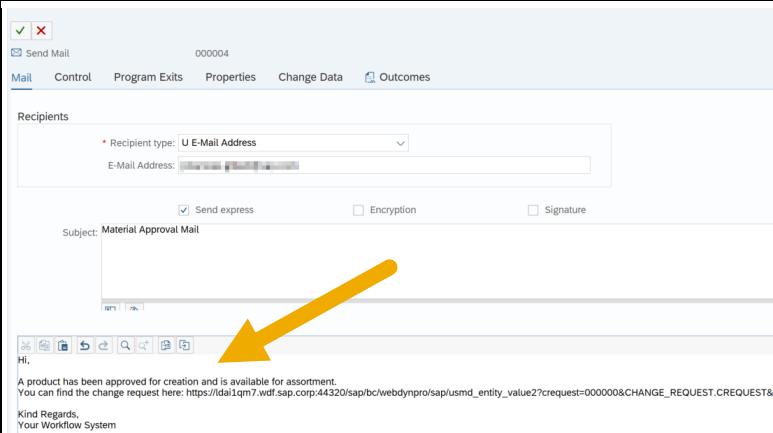
The goal of the approach described in this chapter is to send an email notification for an approved material creation. To achieve this **only configuration** is used (no source code). A new sub-workflow as well as a new change request type and a rule-based workflow are configured.

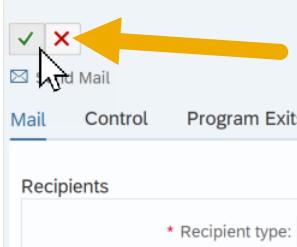
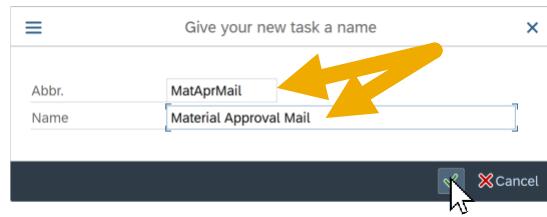
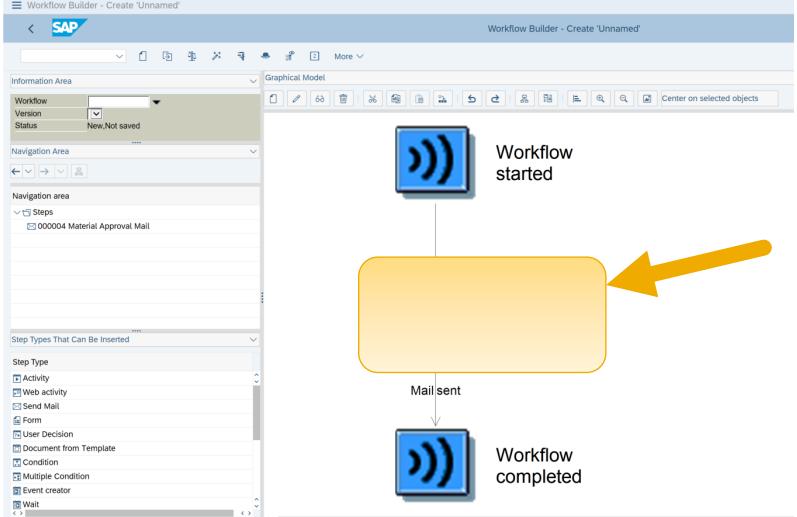
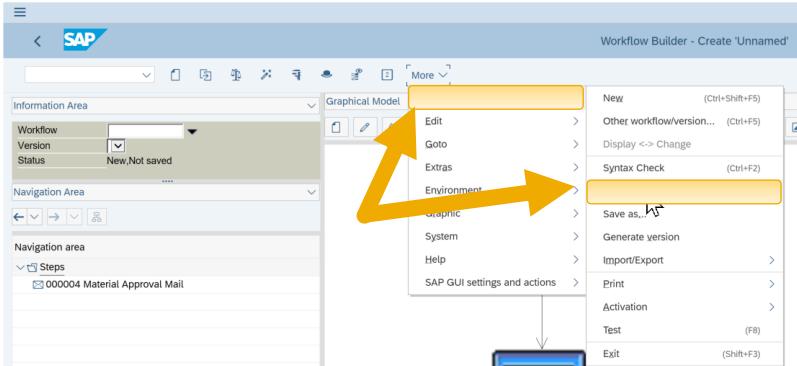
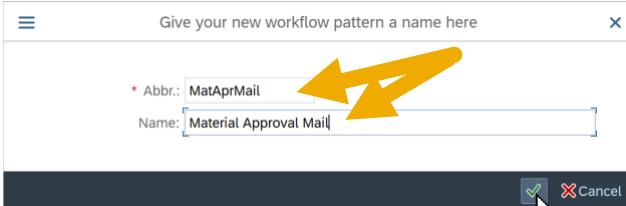
Chapter 2.1 describes the creation of a new sub-workflow which includes a step for sending an email. To bring this sub-workflow into action it shall be incorporated in a rule-based workflow. In this document, for a non-disruptive approach, a new change request type is created and configured (chapter 2.2). The configuration of a rule-based workflow for this new change request type is described in chapter 2.3. To omit starting from scratch the SAP-delivered change request type MAT01 (Create Material) is copied and adjusted. In chapter 2.4 the whole configuration is tested.

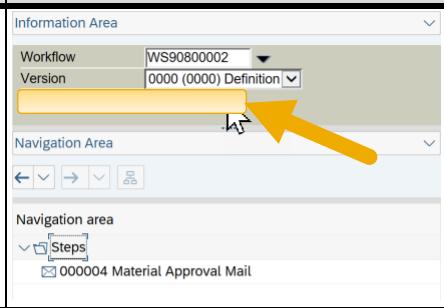
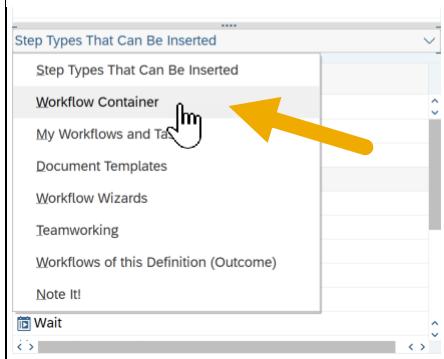
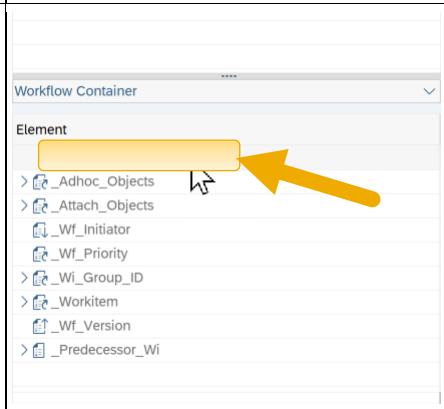
2.1. Create a sub-workflow

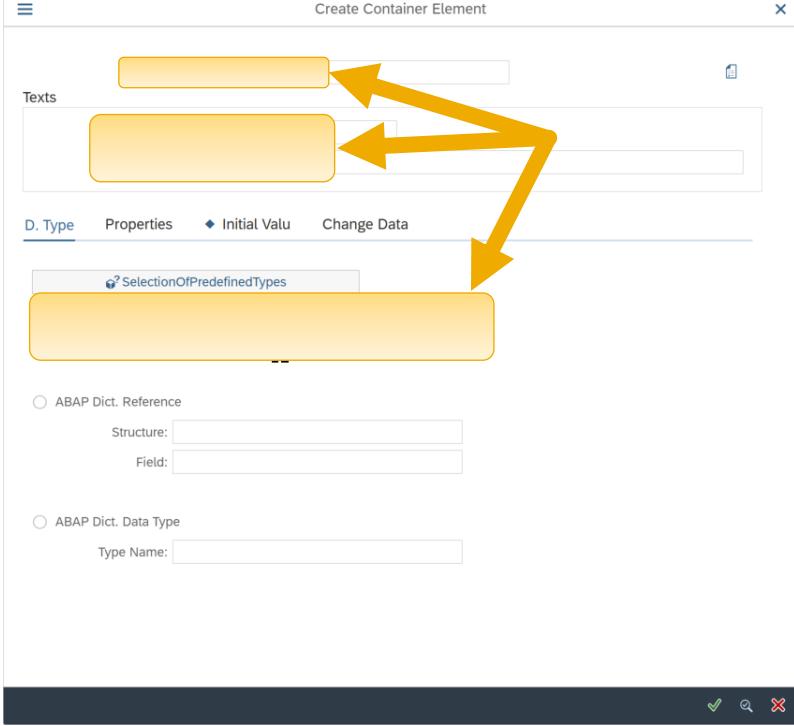
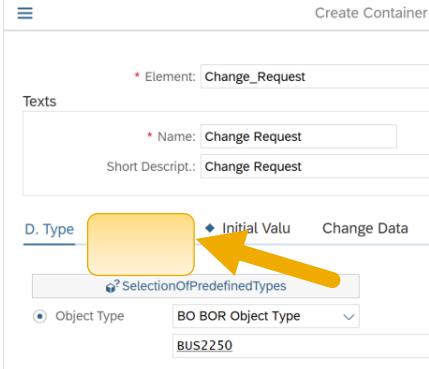
This chapter describes the creation of a new sub-workflow which includes a step for sending an email. The creation starts from scratch and will end with a new (sub-)workflow. All the described steps are configuration only. You might need a customizing request and/or task at hand to save and activate the objects.

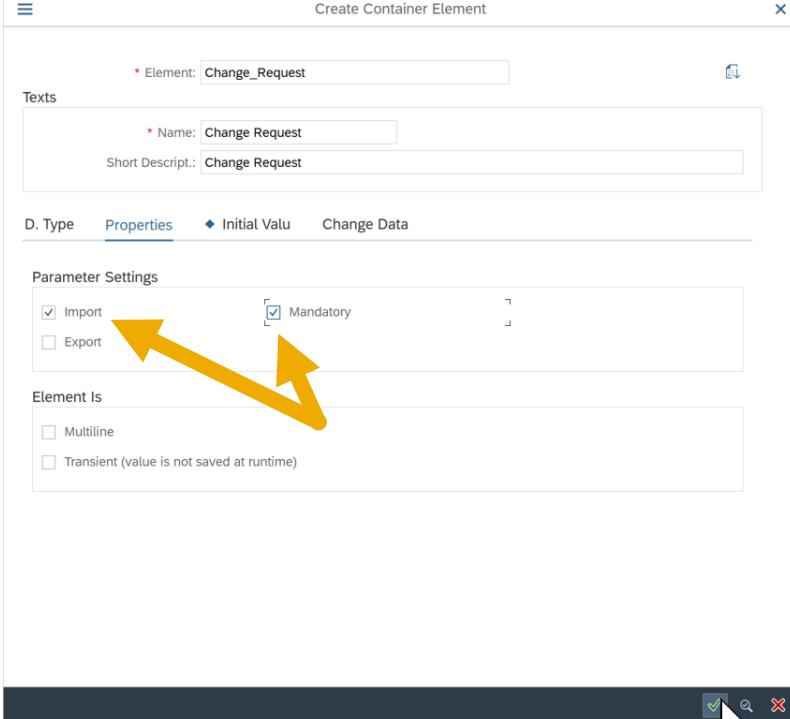
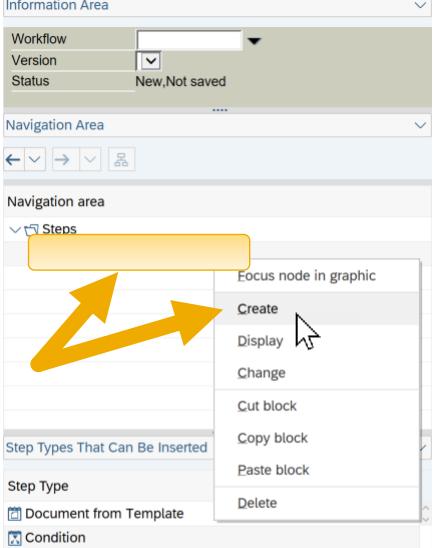
Step	Explanation	Screenshot																			
1	Start the Workflow Builder (transaction SWDD). The Workflow Builder is already in edit mode of a new workflow.	 <pre> graph TD A[Workflow started] --> B[Undefined- Double-click for new step] B --> C[Workflow completed] </pre>																			
2	<p>Double-click on the already existing workflow step named <i>Undefined</i> in the graphical model.</p> <p>Alternatively, navigate to the corresponding step in the navigation area, open the context menu and select <i>Change</i>.</p>	 <pre> graph TD A[Workflow started] --> B[Undefined- Double-click for new step] B --> C[Workflow completed] </pre>																			
3	In pop-up <i>Step Selection</i> chose <i>Send Mail</i> and continue.	 <table border="1"> <thead> <tr> <th>Step Type</th> </tr> </thead> <tbody> <tr><td>Activity</td></tr> <tr><td>Web activity</td></tr> <tr><td>Send Mail</td></tr> <tr><td>Form</td></tr> <tr><td>User Decision</td></tr> <tr><td>Document from Template</td></tr> <tr><td>Condition</td></tr> <tr><td>Multiple Condition</td></tr> <tr><td>Event creator</td></tr> <tr><td>Wait</td></tr> <tr><td>Container operation</td></tr> <tr><td>Process Control</td></tr> <tr><td>Loop (Until)</td></tr> <tr><td>Fork</td></tr> <tr><td>Undefined</td></tr> <tr><td>Ad Hoc Anchor</td></tr> <tr><td>Block</td></tr> <tr><td>Local Workflow</td></tr> </tbody> </table>	Step Type	Activity	Web activity	Send Mail	Form	User Decision	Document from Template	Condition	Multiple Condition	Event creator	Wait	Container operation	Process Control	Loop (Until)	Fork	Undefined	Ad Hoc Anchor	Block	Local Workflow
Step Type																					
Activity																					
Web activity																					
Send Mail																					
Form																					
User Decision																					
Document from Template																					
Condition																					
Multiple Condition																					
Event creator																					
Wait																					
Container operation																					
Process Control																					
Loop (Until)																					
Fork																					
Undefined																					
Ad Hoc Anchor																					
Block																					
Local Workflow																					

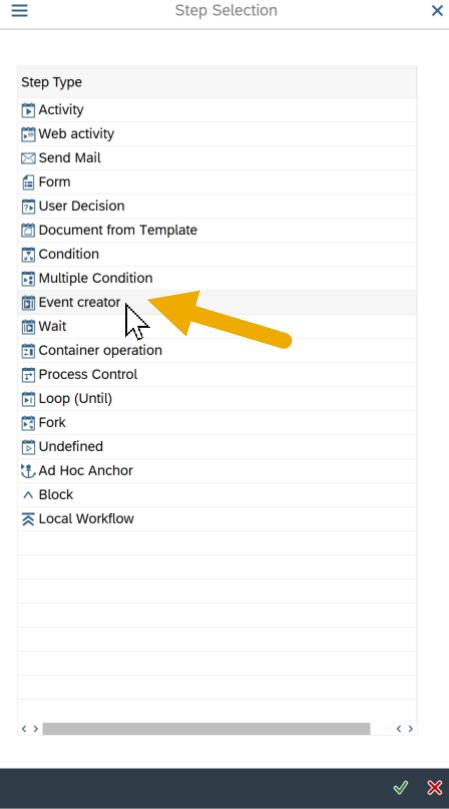
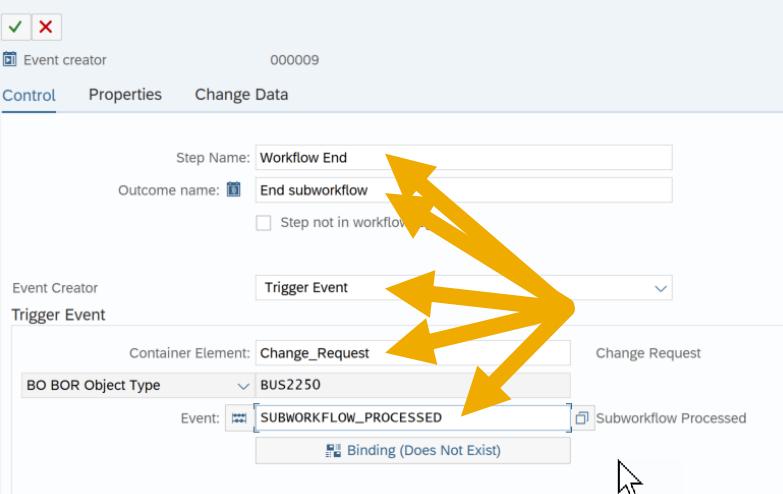
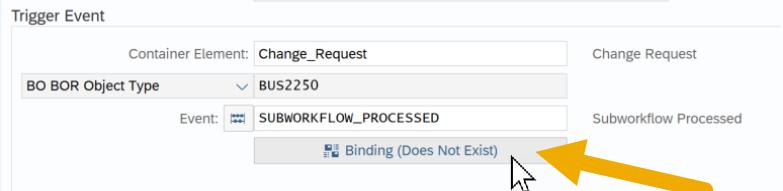
Step	Explanation	Screenshot
4	In section <i>Recipients</i> for <i>Recipient Type</i> chose <i>E-Mail Address (U)</i> .	
5	Enter the email address of the recipient.	
6	Select option <i>Send express</i> .	
7	Enter a suitable <i>Subject</i> like <i>Material Approval Mail</i> for this example.	
8	Enter some text. You may use data from container element CHANGE_REQUEST which will be created soon. Here, it is used to create a link to the change request.	

Step	Explanation	Screenshot
9	Press Transfer and Back.	
10	A pop-up opens where you to give your new task a name. Enter an <i>Abbreviation</i> as well as a <i>Name</i> .	
11	Back on the main screen of the Workflow Builder the workflow should have been adjusted.	
12	Save the workflow via menu <i>Workflow > Save</i> .	
13	Give your new workflow a name. In the corresponding pop-up enter an <i>Abbreviation</i> and name. For example <i>MatAprMail</i> and <i>Material Approval Mail</i> , respectively.	

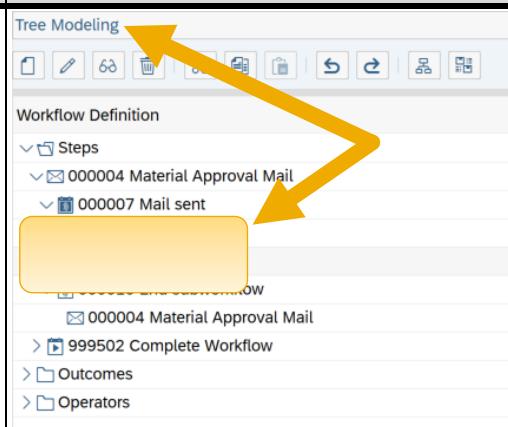
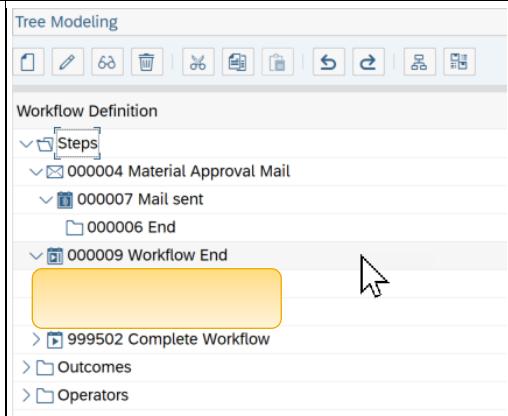
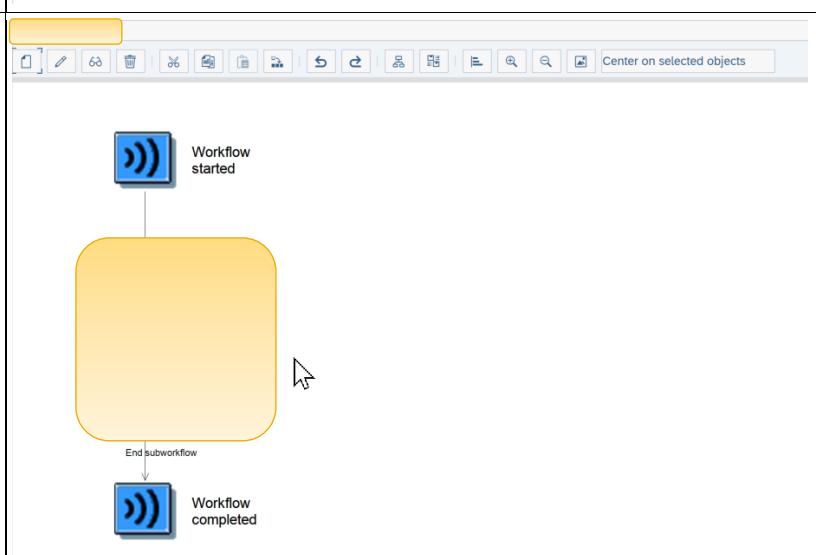
Step	Explanation	Screenshot
14	In the <i>Information Area</i> the assigned number is displayed and the <i>Status</i> is now <i>New, Saved</i> .	
15	Press <i>Step Types That Can Be Inserted</i> from the area header. In the drop down menu chose <i>Workflow Container</i> .	
16	Double click on element <Double-Click to Create>.	

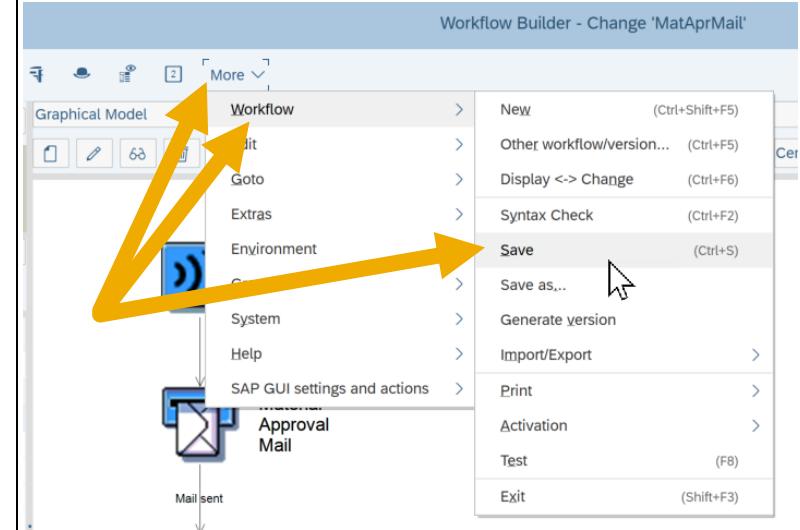
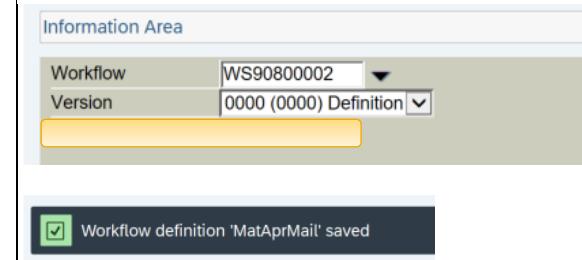
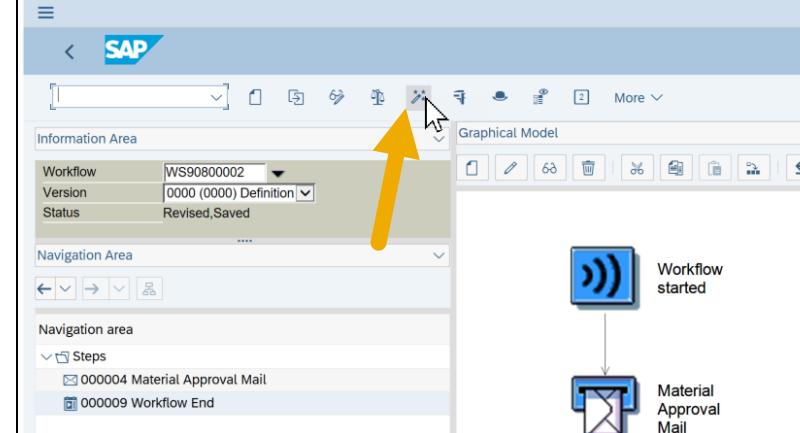
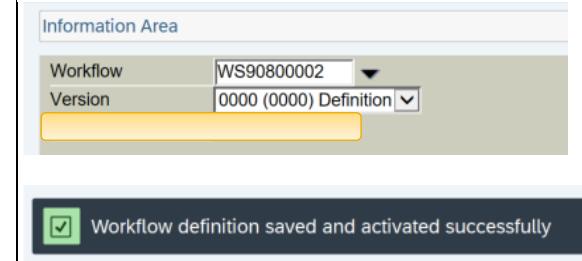
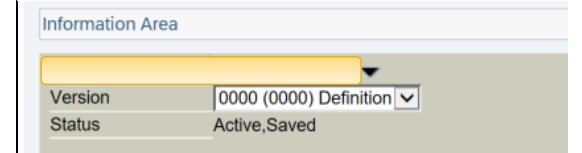
Step	Explanation	Screenshot
17	<p>In pop-up <i>Create Container Element</i> enter the following data:</p> <ul style="list-style-type: none"> • Element: <i>Change_Request</i> • Name: <i>Change Request</i> • Short Description: <i>Change Request</i> <p>On tab <i>D.Type</i> enter the following:</p> <ul style="list-style-type: none"> • Select <i>Object Type</i> • Choose <i>BOR Object Type</i> • Enter: <i>BUS2250</i> 	
18	Navigate to tab <i>Properties</i> .	

Step	Explanation	Screenshot
19	<p>In section <i>Parameter Settings</i> select <i>Import</i> and <i>Mandatory</i> to make the container element a mandatory importing parameter. Press <i>Continue</i>.</p>	
20	<p>In the navigation area open the context menu on the workflow step that you just created and chose <i>Create</i>.</p>	

Step	Explanation	Screenshot
21	In pop-up <i>Step Selection</i> chose <i>Event creator</i> and continue.	
22	<p>Enter the following data:</p> <ul style="list-style-type: none"> • <i>Step Name:</i> Workflow End • <i>Outcome name:</i> End subworkflow <p>For <i>Event Creator</i> chose <i>Trigger Event</i>. In section <i>Trigger Event</i> enter the following:</p> <ul style="list-style-type: none"> • <i>Container Element:</i> Change_Request (you may use the value help) • <i>Event:</i> SUBWORKFLOW_PROCESSED (you may use the value help) 	
23	Press <i>Binding (Does not Exist)</i> .	

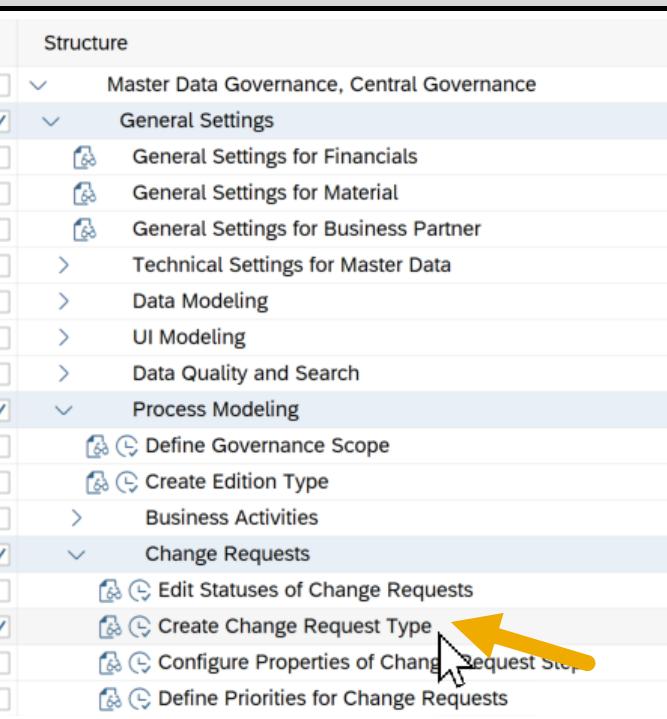
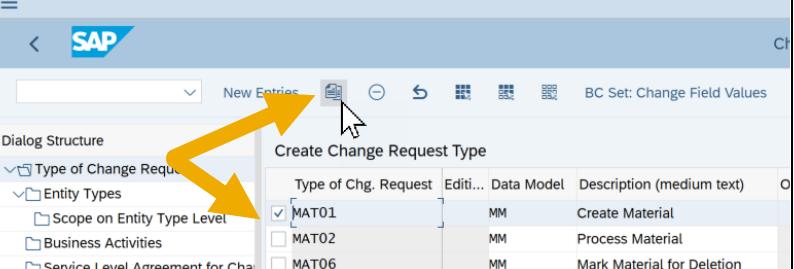
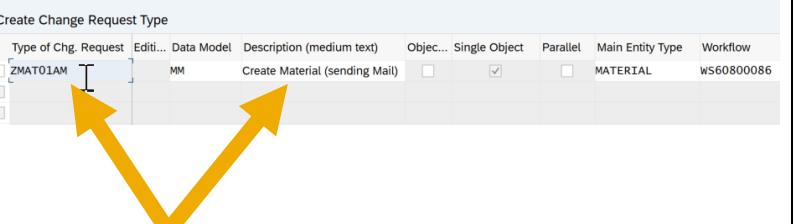
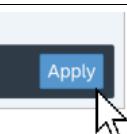
Step	Explanation	Screenshot
24	<p>In pop-up <i>Change Binding For Event</i> the container element <i>Change_Request</i> (left side) is assigned to <i>_EVT_OBJECT</i> (right side). Either enter these values surrounded by &-chars or use drag and drop.</p>	
25	Press <i>Continue</i> .	
26	Back on the <i>Event creator</i> screen, the button text has been changed to <i>Binding (Exists)</i> .	
27	Press <i>Transfer and Back</i> .	
28	In the <i>Graphical Model</i> the new workflow step might be inserted before the <i>Material Approval Mail</i> step.	

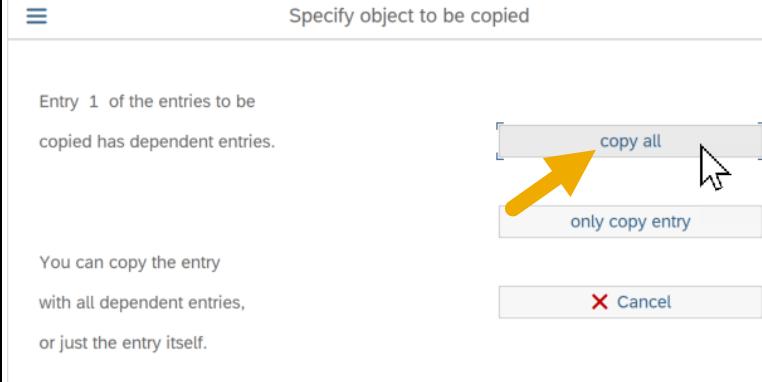
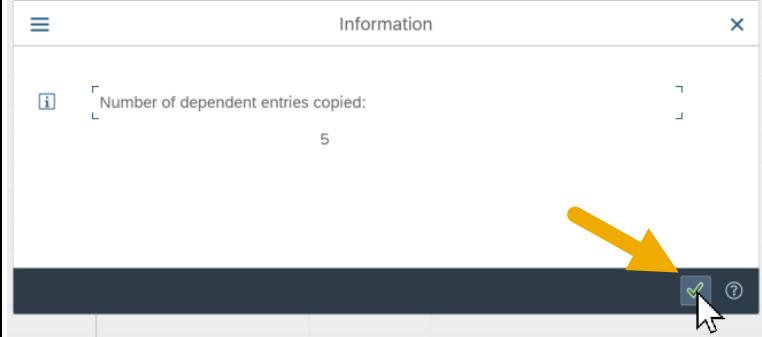
Step	Explanation	Screenshot
29	Switch to <i>Tree Modeling</i> . Drag the <i>Workflow End</i> step and drop it after <i>End of Mail sent</i> .	
30	In the pop-up which opens chose <i>Move</i> .	
31	As result the ordering has been changed.	
32	This is also reflected in the <i>Graphical Model</i> .	

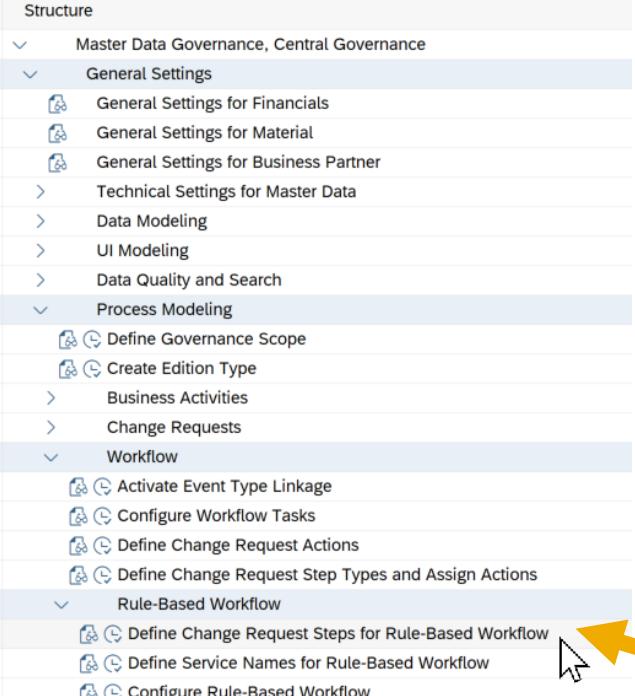
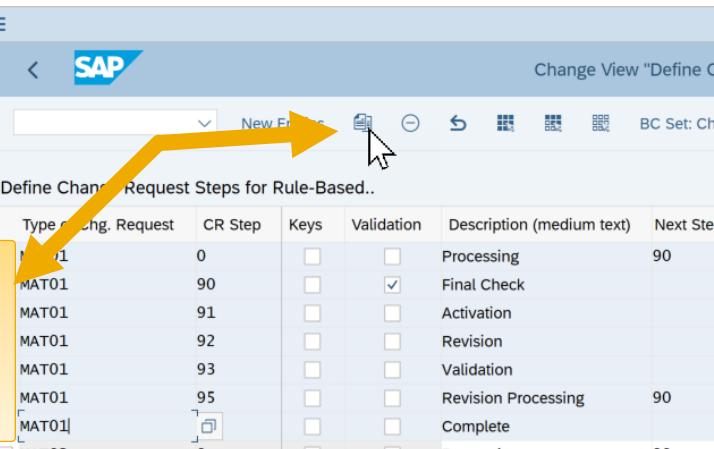
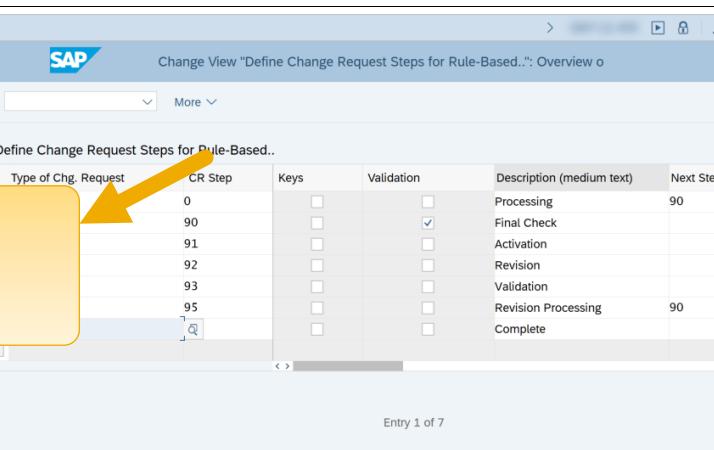
Step	Explanation	Screenshot
33	Save the workflow via menu <i>Workflow > Save</i> .	 <p>The screenshot shows the SAP Workflow Builder interface. The top menu bar has 'Workflow' selected. A yellow arrow points from the left towards the 'Workflow' menu item, and another yellow arrow points from the right towards the 'Save' option in the dropdown menu.</p>
34	A corresponding information message is displayed and the workflow's <i>Status</i> is now <i>Revised, Saved</i> .	 <p>The screenshot shows the SAP Information Area. The 'Workflow' dropdown is set to 'WS90800002' and the 'Version' dropdown is set to '0000 (0000) Definition'. A yellow box highlights the 'Status' field, which shows 'Revised, Saved'. Below the area, a message box displays 'Workflow definition 'MatAprMail' saved' with a checkmark icon.</p>
35	Activate the workflow.	 <p>The screenshot shows the SAP Information Area and the Graphical Model view. The 'Workflow' dropdown is set to 'WS90800002' and the 'Version' dropdown is set to '0000 (0000) Definition'. A yellow arrow points to the 'Activate' button in the toolbar above the graphical model. To the right, a diagram shows a blue circle with double arrows labeled 'Workflow started' connected to a blue envelope icon labeled 'Material Approval Mail'.</p>
36	A corresponding information message is displayed and the workflow's <i>Status</i> is now <i>Active, Saved</i> .	 <p>The screenshot shows the SAP Information Area. The 'Workflow' dropdown is set to 'WS90800002' and the 'Version' dropdown is set to '0000 (0000) Definition'. A yellow box highlights the 'Status' field, which shows 'Active, Saved'. Below the area, a message box displays 'Workflow definition saved and activated successfully' with a checkmark icon.</p>
37	Note down the workflow's number.	 <p>The screenshot shows the SAP Information Area. The 'Version' dropdown is set to '0000 (0000) Definition' and the 'Status' field shows 'Active, Saved'. The 'Workflow' dropdown is visible at the top left.</p>

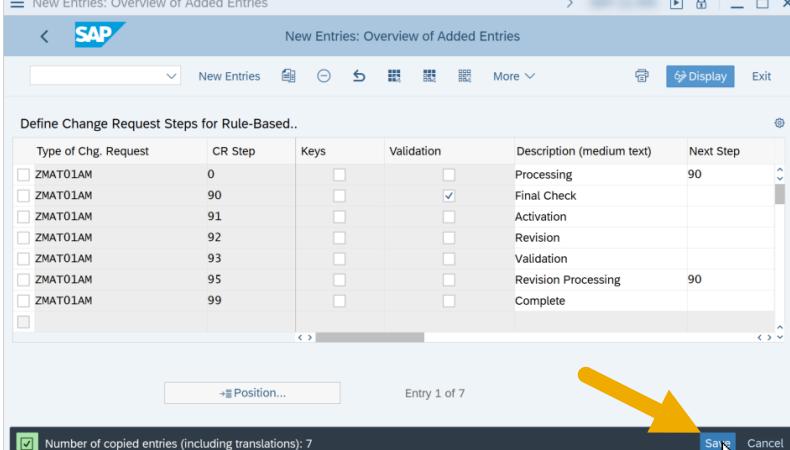
2.2. Create a new change request type and configure change request steps

To bring this sub-workflow into action it must be incorporated in a rule-based workflow. In this document for a non-disruptive approach, a new change request type is created and configured. To omit starting from scratch the SAP-delivered change request type MAT01 (Create Material) is copied and adjusted.

Step	Explanation	Screenshot
1	Start transaction MDGIMG. Navigate to <i>General Settings > Process Modeling > Change Requests > Create Change Request Type</i> .	
2	Select change request type MAT01 and press Copy As.	
3	Enter the following values: <ul style="list-style-type: none"> Type of Change Request: ZMAT01AM Description: Create Material (sending Mail) <p>(Note that the workflow is not the one you just created, it is still the original one.)</p>	
4	Press Apply.	

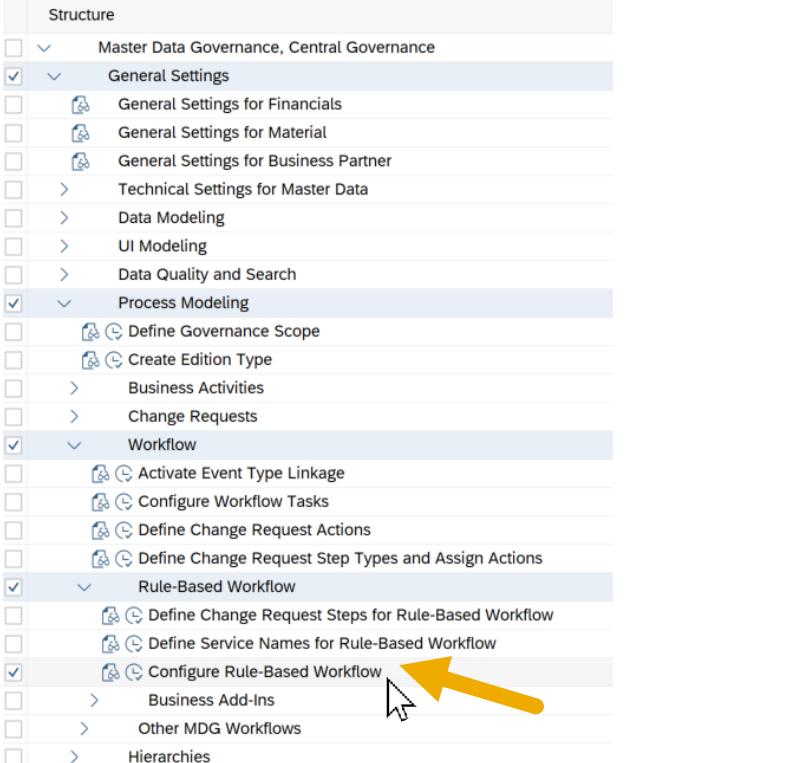
Step	Explanation	Screenshot
5	In pop-up <i>Specify object to be copied</i> press <i>copy all</i> .	 <p>Entry 1 of the entries to be copied has dependent entries.</p> <p>You can copy the entry with all dependent entries, or just the entry itself.</p> <p>copy all</p> <p>only copy entry</p> <p>Cancel</p>
6	Acknowledge the information on <i>Number of dependent entries copied</i> by choosing <i>Continue</i> .	 <p>Number of dependent entries copied: 5</p> <p>Continue</p>
7	Navigate to <i>Type of Change Request</i> .	 <p>Dialog Structure</p> <ul style="list-style-type: none"> Type of Change Request <ul style="list-style-type: none"> Entity Types <ul style="list-style-type: none"> Scope on Entity Type Level Business Activities Service Level Agreement for Cha
8	Press Save.	 <p>Save Cancel</p>

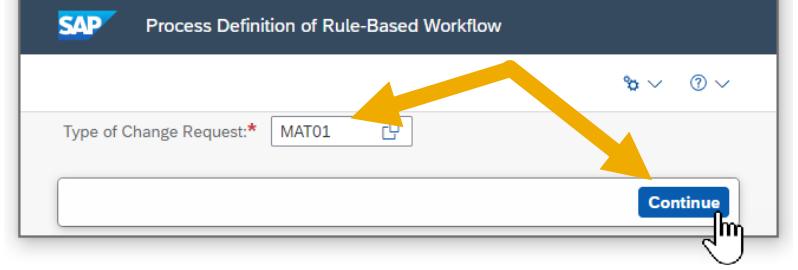
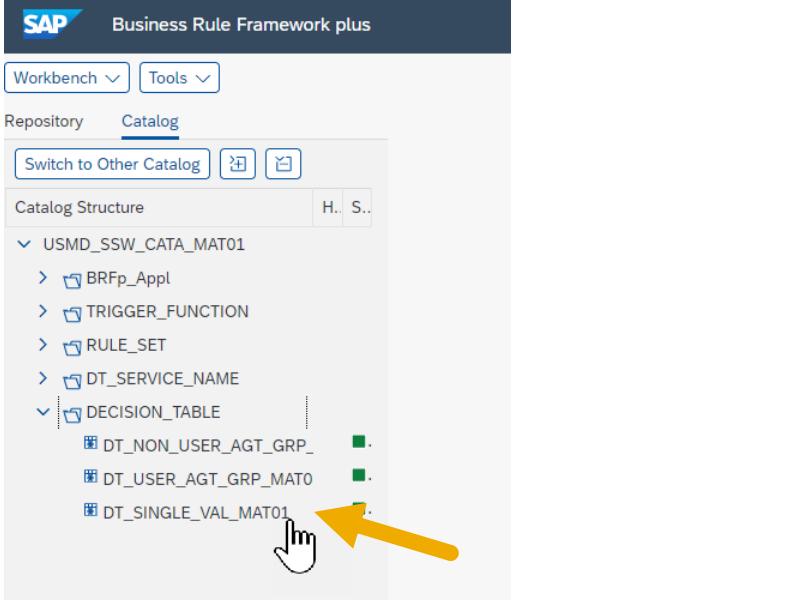
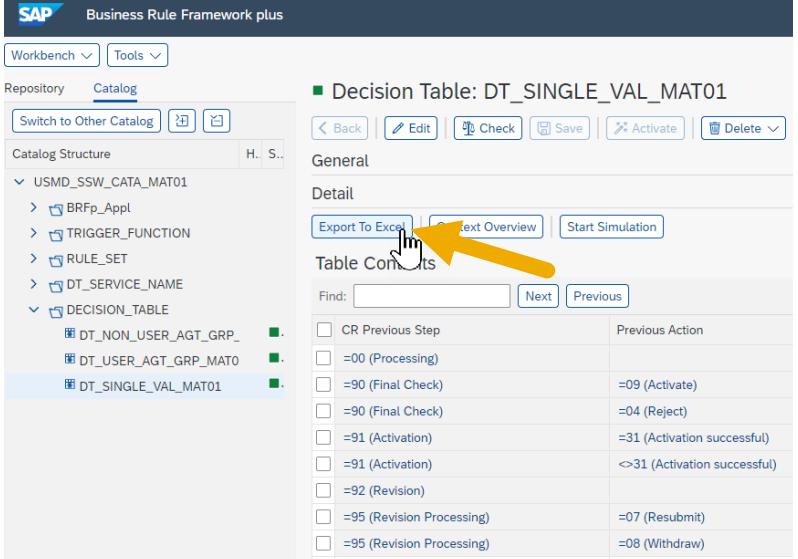
Step	Explanation	Screenshot																																																						
9	<p>Go back to MDGIMG.</p> <p>Navigate to General Settings > Process Modeling > Workflow > Rule-Based Workflow > Define Change Request Steps for Rule-Based Workflow.</p>																																																							
10	<p>Select all entries for change request type MAT01 and press the copy as button.</p>	 <table border="1"> <thead> <tr> <th>Type of Chg. Request</th> <th>CR Step</th> <th>Keys</th> <th>Validation</th> <th>Description (medium text)</th> <th>Next Step</th> </tr> </thead> <tbody> <tr> <td>MAT01</td> <td>0</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>Processing</td> <td>90</td> </tr> <tr> <td>MAT01</td> <td>90</td> <td><input type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td>Final Check</td> <td></td> </tr> <tr> <td>MAT01</td> <td>91</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>Activation</td> <td></td> </tr> <tr> <td>MAT01</td> <td>92</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>Revision</td> <td></td> </tr> <tr> <td>MAT01</td> <td>93</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>Validation</td> <td></td> </tr> <tr> <td>MAT01</td> <td>95</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>Revision Processing</td> <td>90</td> </tr> <tr> <td>MAT01</td> <td></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>Complete</td> <td></td> </tr> <tr> <td>MAT02</td> <td>0</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>Processing</td> <td>90</td> </tr> </tbody> </table>	Type of Chg. Request	CR Step	Keys	Validation	Description (medium text)	Next Step	MAT01	0	<input type="checkbox"/>	<input type="checkbox"/>	Processing	90	MAT01	90	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Final Check		MAT01	91	<input type="checkbox"/>	<input type="checkbox"/>	Activation		MAT01	92	<input type="checkbox"/>	<input type="checkbox"/>	Revision		MAT01	93	<input type="checkbox"/>	<input type="checkbox"/>	Validation		MAT01	95	<input type="checkbox"/>	<input type="checkbox"/>	Revision Processing	90	MAT01		<input type="checkbox"/>	<input type="checkbox"/>	Complete		MAT02	0	<input type="checkbox"/>	<input type="checkbox"/>	Processing	90
Type of Chg. Request	CR Step	Keys	Validation	Description (medium text)	Next Step																																																			
MAT01	0	<input type="checkbox"/>	<input type="checkbox"/>	Processing	90																																																			
MAT01	90	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Final Check																																																				
MAT01	91	<input type="checkbox"/>	<input type="checkbox"/>	Activation																																																				
MAT01	92	<input type="checkbox"/>	<input type="checkbox"/>	Revision																																																				
MAT01	93	<input type="checkbox"/>	<input type="checkbox"/>	Validation																																																				
MAT01	95	<input type="checkbox"/>	<input type="checkbox"/>	Revision Processing	90																																																			
MAT01		<input type="checkbox"/>	<input type="checkbox"/>	Complete																																																				
MAT02	0	<input type="checkbox"/>	<input type="checkbox"/>	Processing	90																																																			
11	<p>Change the name of the change request type to ZMAT01AM. Keep all other values as they are. Press Apply.</p>	 <table border="1"> <thead> <tr> <th>Type of Chg. Request</th> <th>CR Step</th> <th>Keys</th> <th>Validation</th> <th>Description (medium text)</th> <th>Next Step</th> </tr> </thead> <tbody> <tr> <td>0</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>Processing</td> <td>90</td> </tr> <tr> <td>90</td> <td><input type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td><input type="checkbox"/></td> <td>Final Check</td> <td></td> </tr> <tr> <td>91</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>Activation</td> <td></td> </tr> <tr> <td>92</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>Revision</td> <td></td> </tr> <tr> <td>93</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>Validation</td> <td></td> </tr> <tr> <td>95</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>Revision Processing</td> <td>90</td> </tr> <tr> <td></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>Complete</td> <td></td> </tr> </tbody> </table> <p>Entry 1 of 7</p> <p><input checked="" type="checkbox"/> Specify target entries View details</p> <p>Apply</p>	Type of Chg. Request	CR Step	Keys	Validation	Description (medium text)	Next Step	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Processing	90	90	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Final Check		91	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Activation		92	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Revision		93	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Validation		95	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Revision Processing	90		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Complete							
Type of Chg. Request	CR Step	Keys	Validation	Description (medium text)	Next Step																																																			
0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Processing	90																																																			
90	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Final Check																																																				
91	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Activation																																																				
92	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Revision																																																				
93	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Validation																																																				
95	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Revision Processing	90																																																			
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Complete																																																				

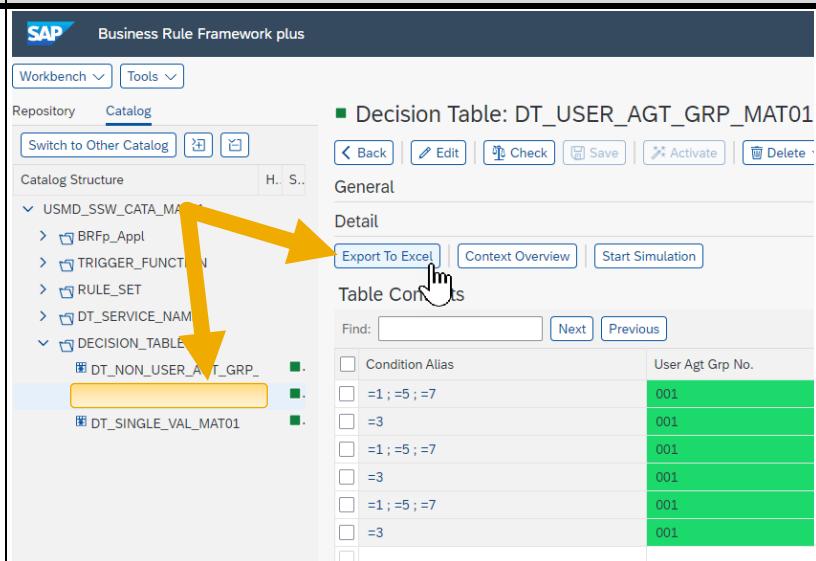
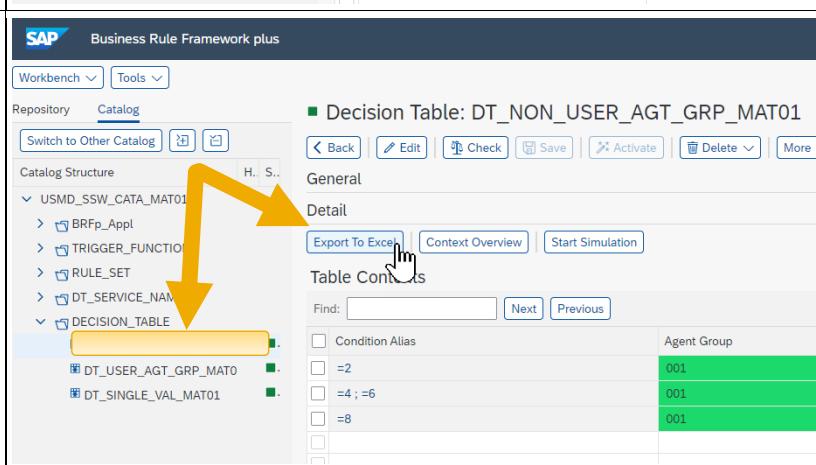
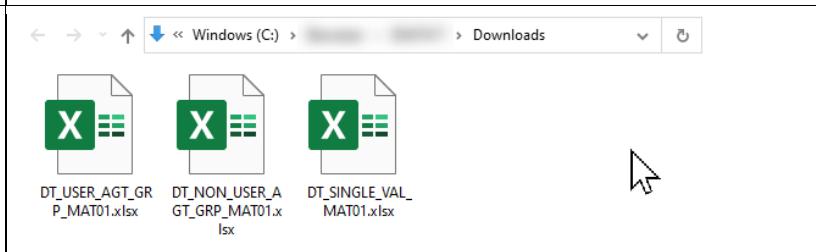
Step	Explanation	Screenshot
12	Press Save.	 <p>The screenshot shows a SAP Fiori application titled 'New Entries: Overview of Added Entries'. The main area displays a table titled 'Define Change Request Steps for Rule-Based..'. The table has columns for 'Type of Chg. Request', 'CR Step', 'Keys', 'Validation', 'Description (medium text)', and 'Next Step'. Several rows are listed, corresponding to ZMAT01AM entries. At the bottom of the screen, there is a footer bar with a checked checkbox labeled 'Number of copied entries (including translations): 7' and a 'Save' button with a yellow arrow pointing to it. Other buttons in the footer include 'Cancel' and icons for 'Display' and 'Exit'.</p>

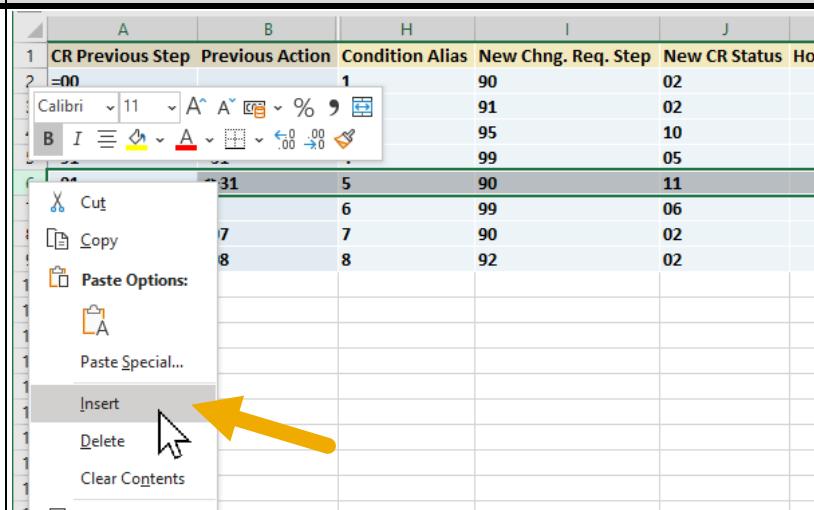
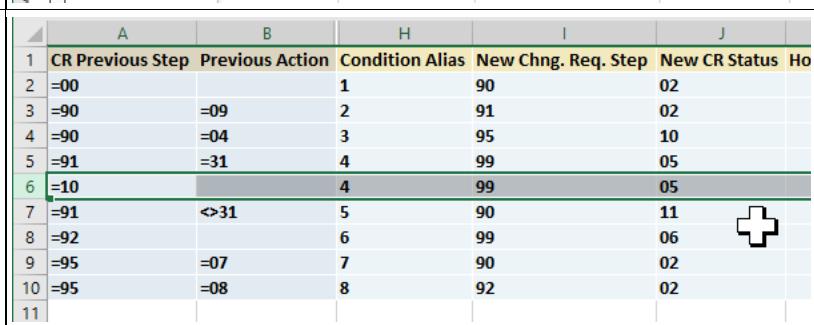
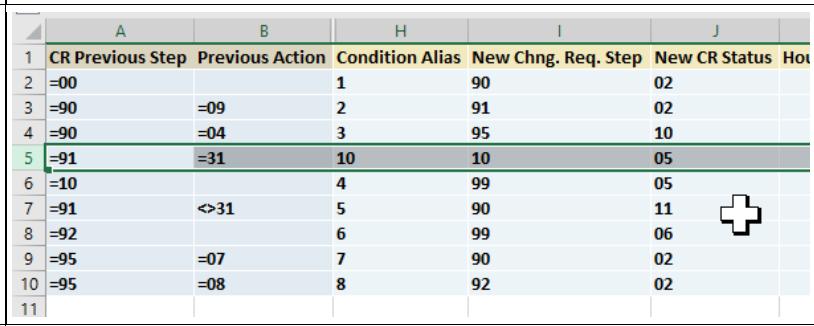
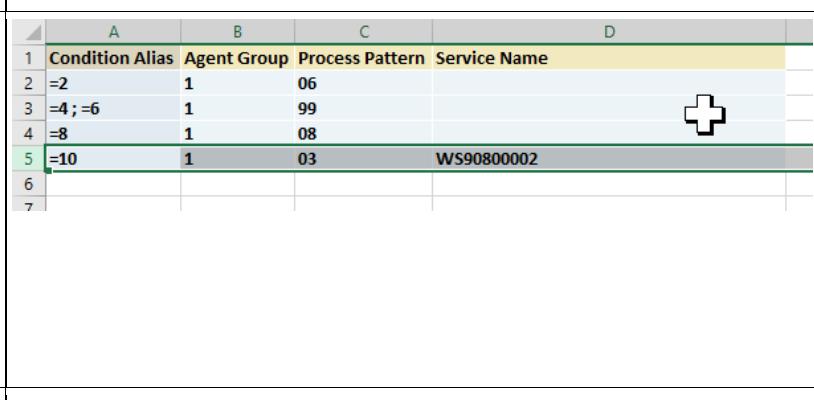
2.3. Configure rule-based workflow for the new change request type

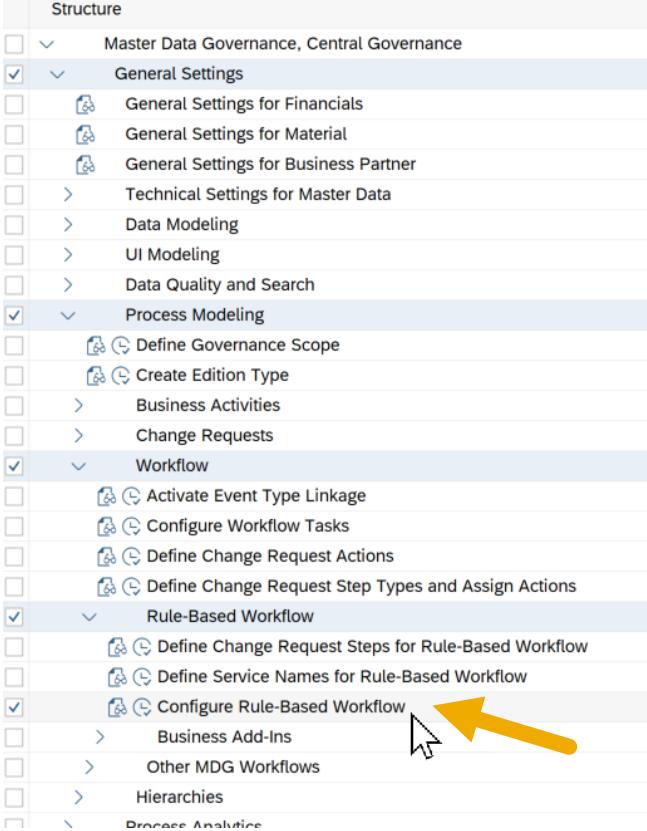
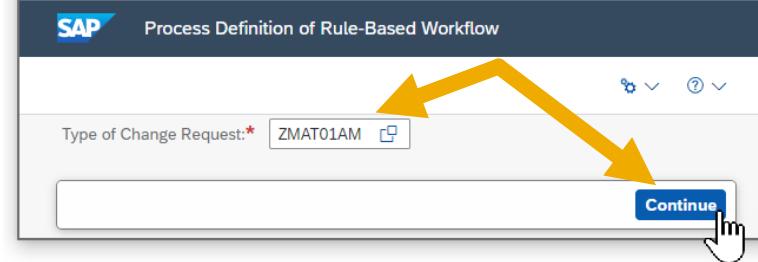
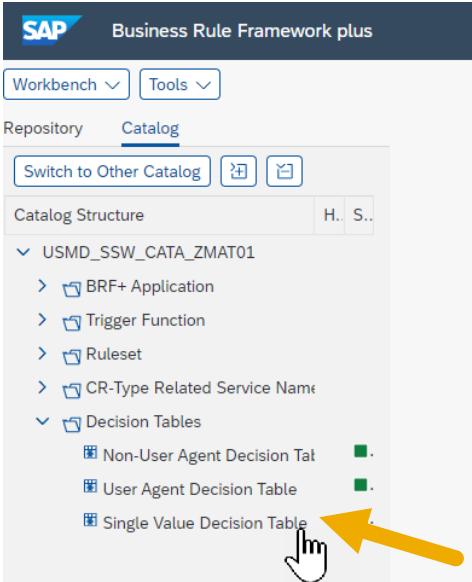
This chapter describes the configuration of a rule-based workflow for the new change request type. To omit starting from scratch the BRF+ decision tables of the SAP-delivered change request type MAT01 (Create Material) are copied and adjusted.

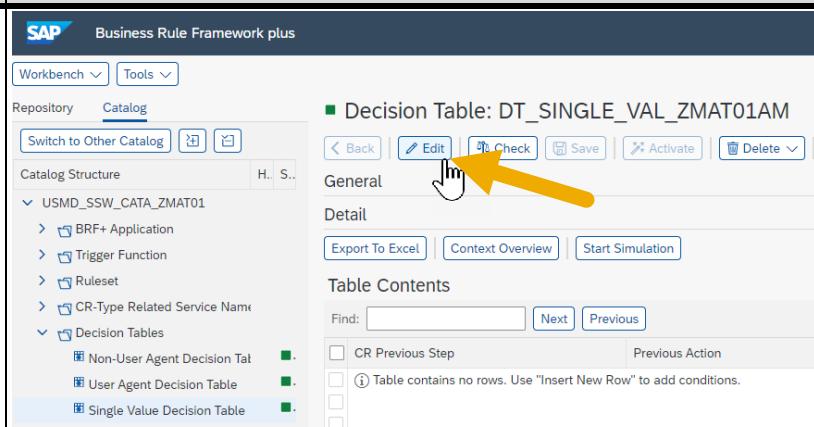
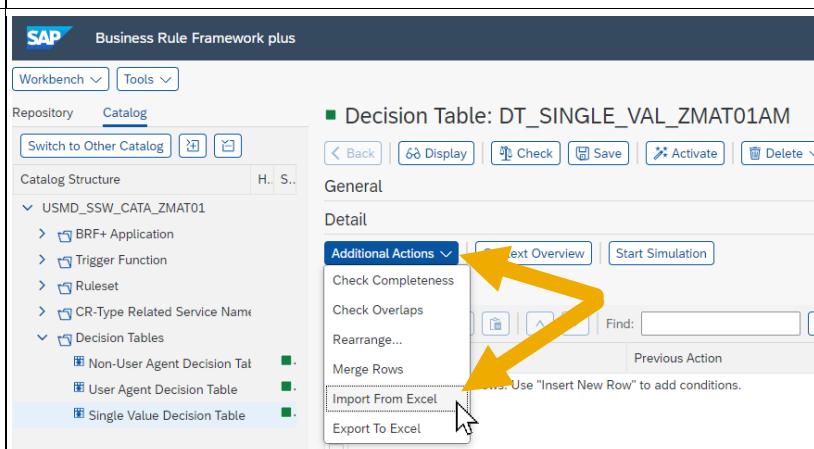
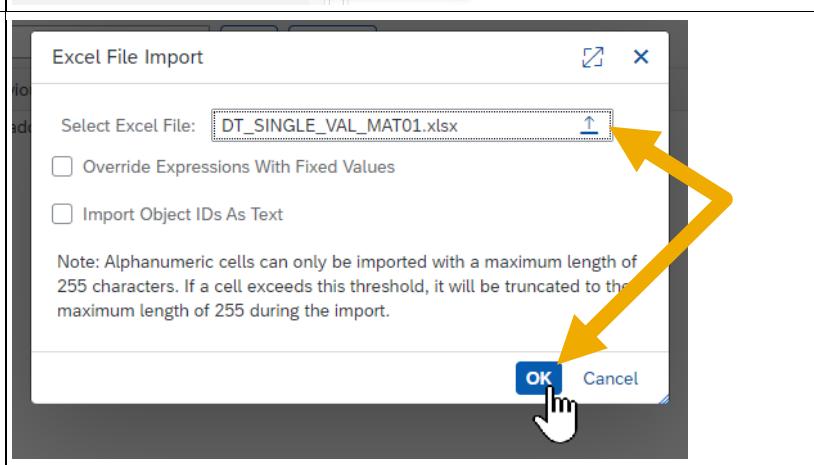
Step	Explanation	Screenshot
1	<p>Start transaction MDGIMG. Navigate to <i>General Settings > Process Modeling > Workflow > Rule-Based Workflow > Configure Rule-Based Workflow</i>.</p>	 <p>The screenshot shows the SAP Fiori navigation tree. The path selected is 'Process Modeling > Rule-Based Workflow > Configure Rule-Based Workflow'. The 'Configure Rule-Based Workflow' option is highlighted with a yellow arrow. The tree also includes other nodes like 'Master Data Governance, Central Governance', 'General Settings', and various modeling and search options.</p>

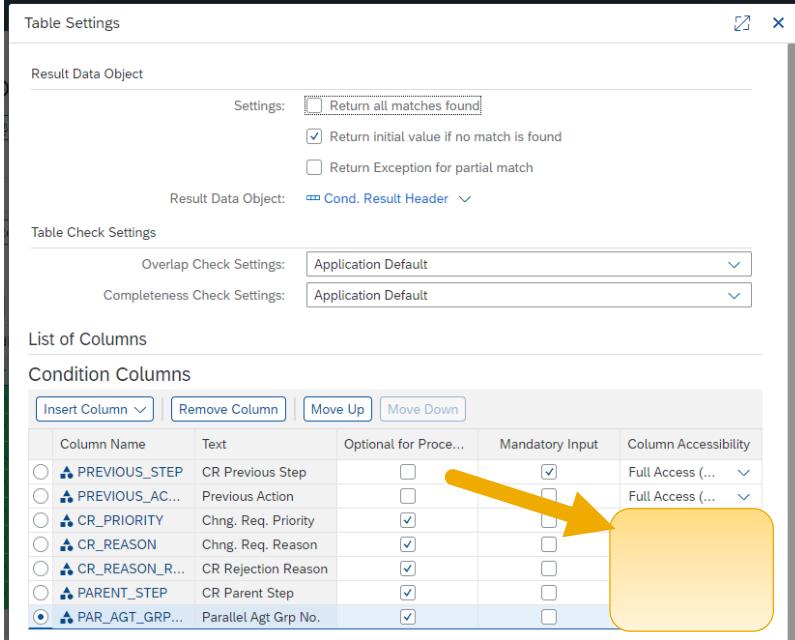
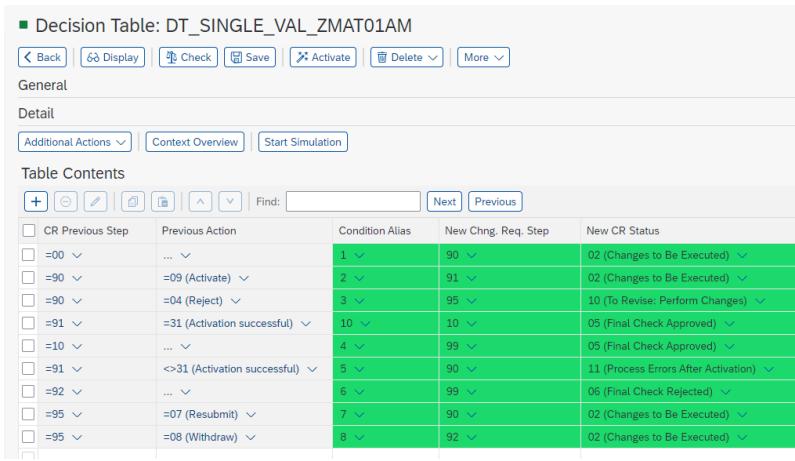
Step	Explanation	Screenshot
2	Enter the created change request type MAT01 and press Continue .	
3	Navigate to the single value decision table (DT_SINGLE_VAL_MAT01).	
4	Press Export To Excel .	

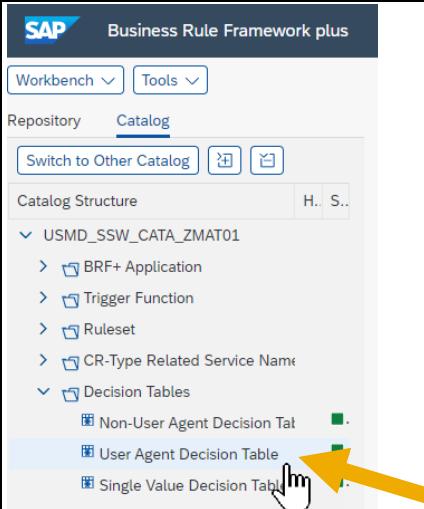
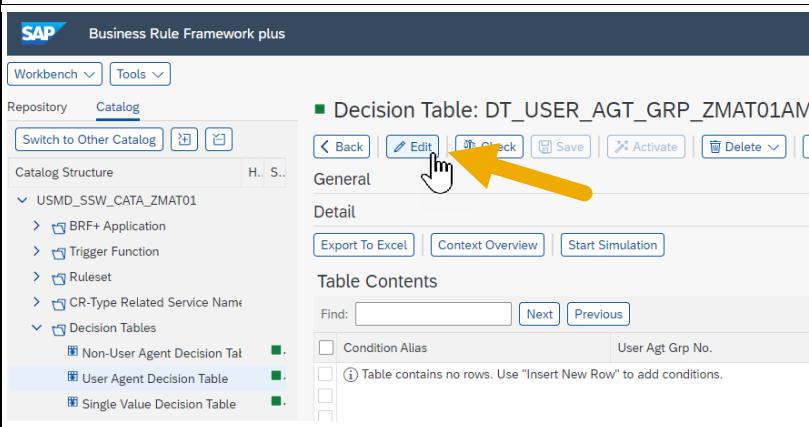
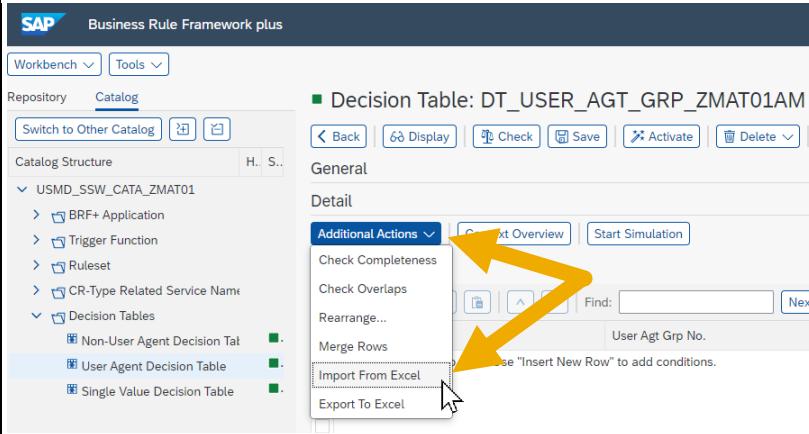
Step	Explanation	Screenshot														
5	Navigate to the user agent decision table (<i>DT_USER_AGT_GRP_MAT01</i>) and press <i>Export To Excel</i> .	 <p>SAP Business Rule Framework plus</p> <p>Decision Table: DT_USER_AGT_GRP_MAT01</p> <p>Detail</p> <p>Table Contents</p> <table border="1"> <thead> <tr> <th>Condition Alias</th> <th>User Agt Grp No.</th> </tr> </thead> <tbody> <tr><td>=1 ; =5 ; =7</td><td>001</td></tr> <tr><td>=3</td><td>001</td></tr> <tr><td>=1 ; =5 ; =7</td><td>001</td></tr> <tr><td>=3</td><td>001</td></tr> <tr><td>=1 ; =5 ; =7</td><td>001</td></tr> <tr><td>=3</td><td>001</td></tr> </tbody> </table>	Condition Alias	User Agt Grp No.	=1 ; =5 ; =7	001	=3	001	=1 ; =5 ; =7	001	=3	001	=1 ; =5 ; =7	001	=3	001
Condition Alias	User Agt Grp No.															
=1 ; =5 ; =7	001															
=3	001															
=1 ; =5 ; =7	001															
=3	001															
=1 ; =5 ; =7	001															
=3	001															
6	Navigate to the non-user agent decision table (<i>DT_NON_USER_AGT_GRP_MAT01</i>) and press <i>Export To Excel</i> .	 <p>SAP Business Rule Framework plus</p> <p>Decision Table: DT_NON_USER_AGT_GRP_MAT01</p> <p>Detail</p> <p>Table Contents</p> <table border="1"> <thead> <tr> <th>Condition Alias</th> <th>Agent Group</th> </tr> </thead> <tbody> <tr><td>=2</td><td>001</td></tr> <tr><td>=4 ; =6</td><td>001</td></tr> <tr><td>=8</td><td>001</td></tr> </tbody> </table>	Condition Alias	Agent Group	=2	001	=4 ; =6	001	=8	001						
Condition Alias	Agent Group															
=2	001															
=4 ; =6	001															
=8	001															
7	There should be three MS Excel files downloaded.	 <p>Downloads</p> <p>DT_USER_AGT_GRP_MAT01.xlsx</p> <p>DT_NON_USER_AGT_GRP_MAT01.xlsx</p> <p>DT_SINGLE_VAL_MAT01.xlsx</p>														

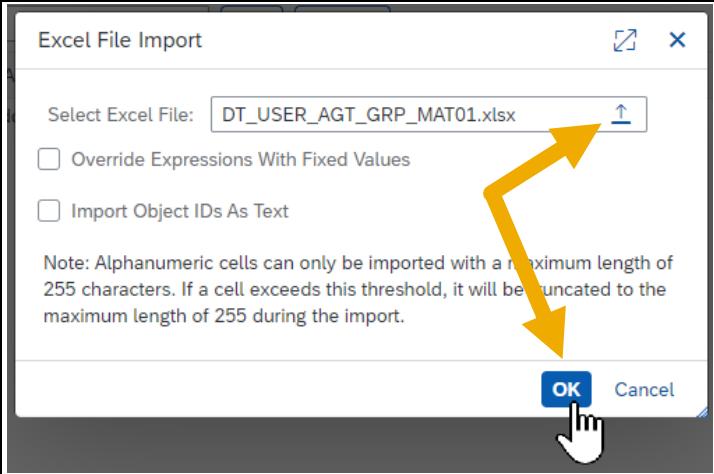
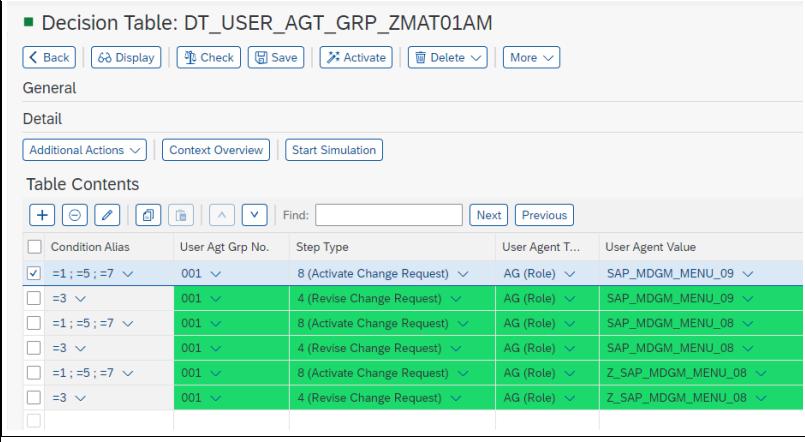
Step	Explanation	Screenshot
8	Open MS Excel document DT_SINGLE_VAL_MAT01.xls and insert an additional row between the original rows 5 and 6.	 <p>A screenshot of a Microsoft Excel spreadsheet titled 'DT_SINGLE_VAL_MAT01.xls'. The spreadsheet has columns A through J. Row 5 contains values: CR Previous Step =00, Previous Action =00, Condition Alias 1, New Chng. Req. Step 90, and New CR Status 02. Row 6 contains values: CR Previous Step =00, Previous Action =00, Condition Alias 1, New Chng. Req. Step 91, and New CR Status 02. Row 7 contains values: CR Previous Step =00, Previous Action =00, Condition Alias 1, New Chng. Req. Step 95, and New CR Status 10. Row 8 contains values: CR Previous Step =00, Previous Action =00, Condition Alias 1, New Chng. Req. Step 99, and New CR Status 05. A context menu is open over the cell in row 5, column H (Condition Alias). The 'Insert' option is highlighted with a yellow arrow. Other options in the menu include Cut, Copy, Paste Options, Paste Special, Delete, and Clear Contents.</p>
9	Insert the following values: <ul style="list-style-type: none">CR Previous Step: =10Previous Action: (empty)Condition Alias: 4New Chng. Req. Step: 99New CR Status: 05	 <p>A screenshot of the same Microsoft Excel spreadsheet. Row 6 now contains values: CR Previous Step =10, Previous Action =00, Condition Alias 4, New Chng. Req. Step 99, and New CR Status 05. A context menu is open over the cell in row 6, column H (Condition Alias). The 'Insert' option is highlighted with a yellow arrow. Other options in the menu include Cut, Copy, Paste Options, Paste Special, Delete, and Clear Contents.</p>
10	Change the values of row no. 5: <ul style="list-style-type: none">Condition Alias: 10New CR Step: 10	 <p>A screenshot of the Microsoft Excel spreadsheet. Row 5 now contains values: CR Previous Step =91, Previous Action =31, Condition Alias 10, New Chng. Req. Step 10, and New CR Status 05. A context menu is open over the cell in row 5, column H (Condition Alias). The 'Insert' option is highlighted with a yellow arrow. Other options in the menu include Cut, Copy, Paste Options, Paste Special, Delete, and Clear Contents.</p>
11	Save the MS Excel file.	
12	Open MS Excel document DT_NON_USER_AGT_GRP_MAT01.xlsx and add the following values to (new) row no. 5: <ul style="list-style-type: none">Condition Alias: =10Agent Group: 10Process Pattern: 03Service Name: WS90800002	 <p>A screenshot of a Microsoft Excel spreadsheet titled 'DT_NON_USER_AGT_GRP_MAT01.xlsx'. The spreadsheet has columns A through D. Row 5 contains values: Condition Alias =10, Agent Group 10, Process Pattern 03, and Service Name WS90800002. A context menu is open over the cell in row 5, column C (Process Pattern). The 'Insert' option is highlighted with a yellow arrow. Other options in the menu include Cut, Copy, Paste Options, Paste Special, Delete, and Clear Contents.</p>
13	Save the MS Excel file.	

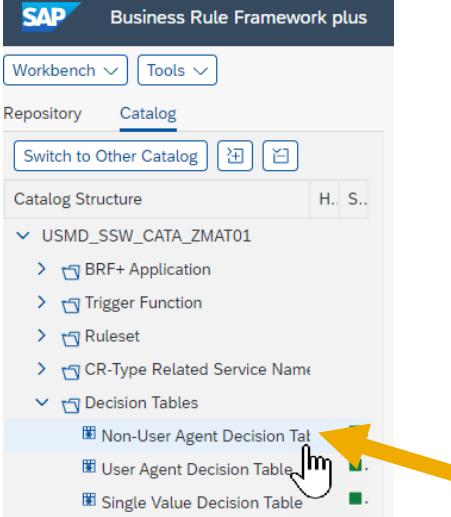
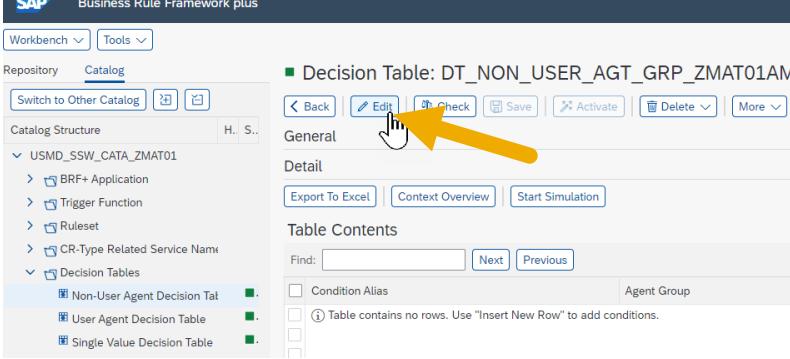
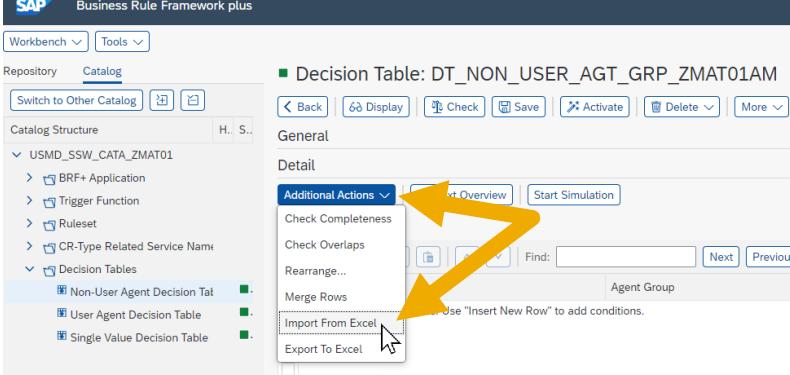
Step	Explanation	Screenshot
14	In the MDGIMG navigate to General Settings > Process Modeling > Workflow > Rule-Based Workflow > Configure Rule-Based Workflow .	
15	Enter the newly created change request type ZMAT01AM and press Continue .	
16	Navigate to the single value decision table (DT_SINGLE_VAL_ZMAT01AM).	

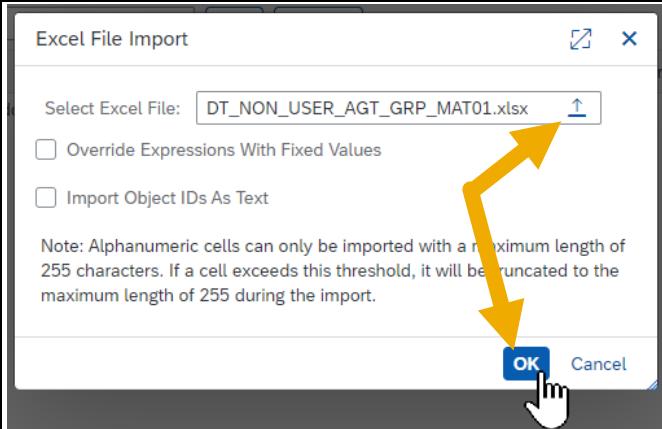
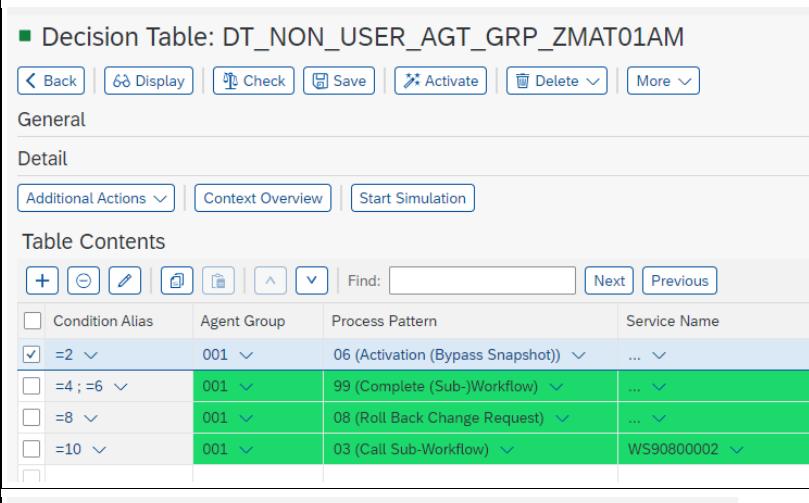
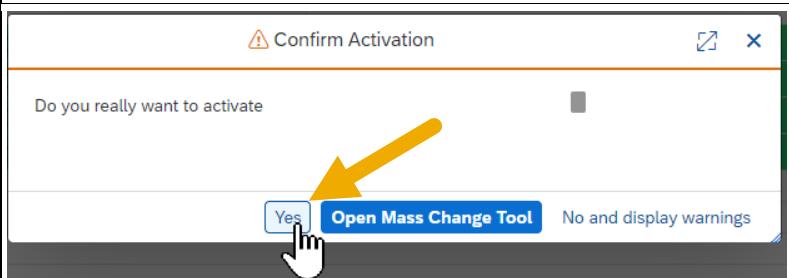
Step	Explanation	Screenshot
17	Press <i>Edit</i> to switch to edit mode.	
18	Press <i>Additional Actions > Import From Excel</i> to import the corresponding MS Excel file and apply its content as settings since the actual settings of the single value decision table are empty.	
19	Choose MS Excel file <i>DT_SINGLE_VAL_MAT01.xlsx</i> and then <i>OK</i> . Having "..._MAT01..." instead of "..._ZMATAM..." in the file name will not be any issue.	
20	You might want to adjust the <i>Table Settings</i> to hide non-relevant columns.	

Step	Explanation	Screenshot
21	Non-relevant columns in this case are the condition columns CR_PRIORITY and the following ones.	 <p>The screenshot shows the 'Table Settings' dialog with the 'Result Data Object' tab selected. Under 'Result Data Object', 'Cond. Result Header' is chosen. In the 'List of Columns' section, there is a table with columns: Column Name, Text, Optional for Proc..., Mandatory Input, and Column Accessibility. The 'CR_PRIORITY' row has 'Mandatory Input' checked. A yellow arrow points to this row, and a yellow box highlights the entire table area.</p>
22	The decision table should contain the adjusted values from the MS Excel file.	 <p>The screenshot shows the 'Decision Table: DT_SINGLE_VAL_ZMAT01AM' interface. At the top, there are buttons for Back, Display, Check, Save, Activate, Delete, and More. Below these are tabs for General, Detail, and Additional Actions. The main area is titled 'Table Contents' and shows a table with columns: CR Previous Step, Previous Action, Condition Alias, New Chng. Req. Step, and New CR Status. The table rows show various activation and rejection scenarios with their corresponding step numbers and status codes.</p>
23	Save the decision table.	 <p>The screenshot shows the 'Decision Table: DT_SINGLE_VAL_ZMAT01AM' interface with the 'General' tab selected. The 'Activate' button is highlighted with a yellow arrow and a mouse cursor icon.</p>
24	Activate the decision table.	 <p>The screenshot shows the 'Decision Table: DT_SINGLE_VAL_ZMAT01AM' interface with the 'General' tab selected. The 'Activate' button is highlighted with a yellow arrow and a mouse cursor icon. A message at the top right says 'Active version available'.</p>

Step	Explanation	Screenshot
25	Navigate to the user agent decision table (<i>DT_USER_AGT_GRP_ZMAT01AM</i>).	
26	Press <i>Edit</i> to switch to edit mode.	
27	Press <i>Additional Actions > Import From Excel</i> to import the corresponding MS Excel file and apply its content as settings since the actual settings of the single value decision table are empty.	

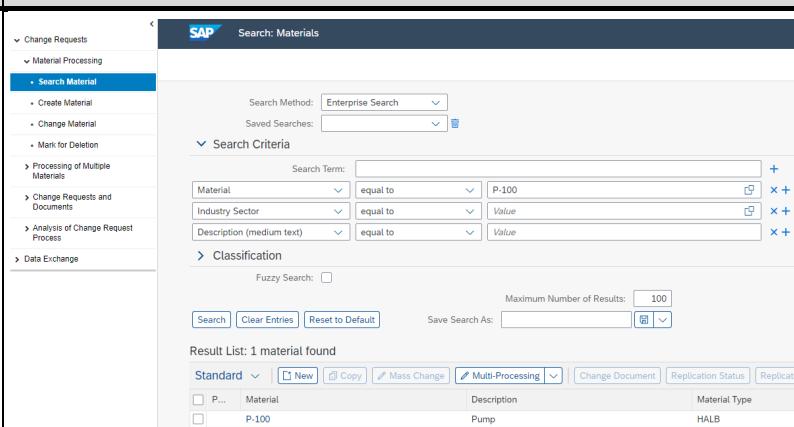
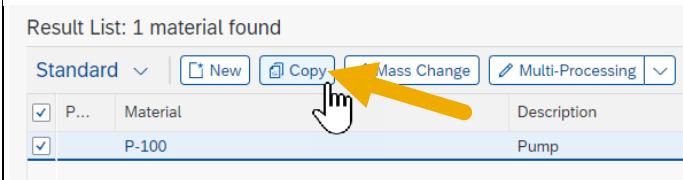
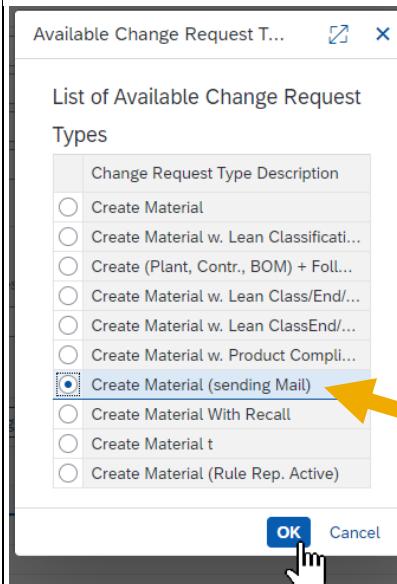
Step	Explanation	Screenshot
28	Select MS Excel file DT_USER_AGT_GRP_MAT01.xlsx and press OK .	
29	The decision table should contain the values from the MS Excel file.	
30	Save the decision table.	
31	Activate the decision table.	

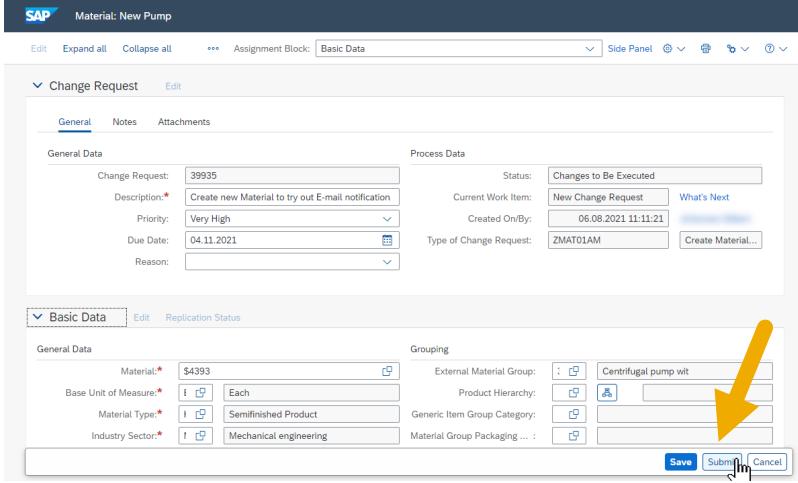
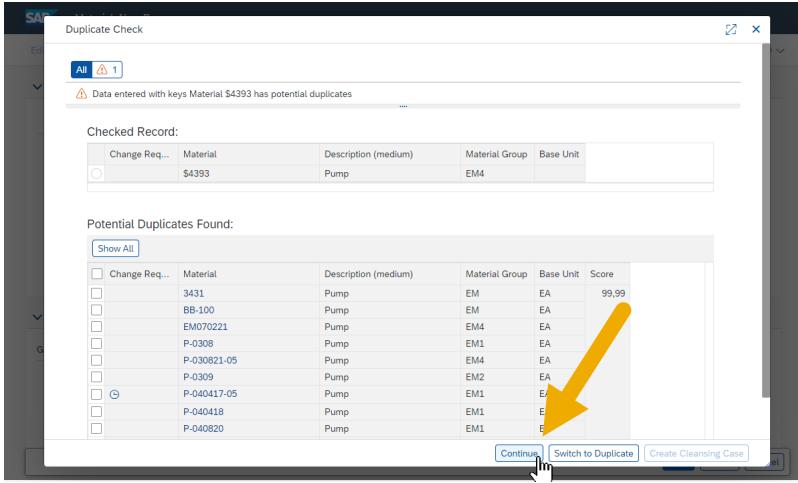
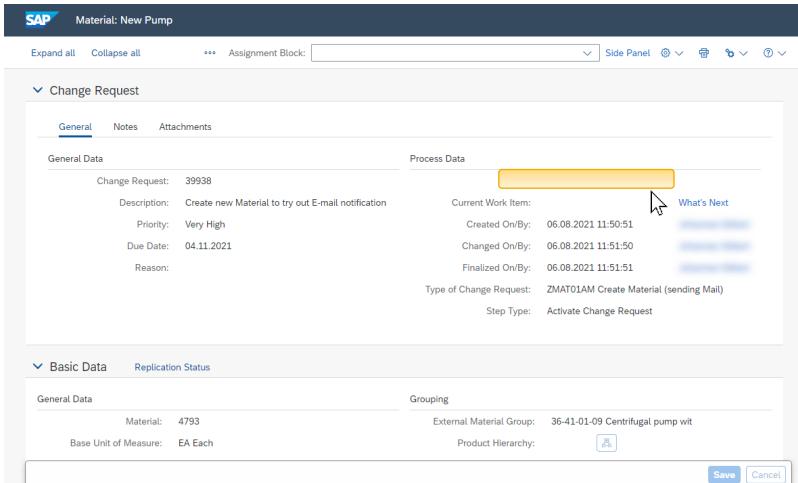
Step	Explanation	Screenshot
32	Navigate to the user agent decision table (<i>DT_NON_USER_AGTE_GRP_ZMAT01AM</i>).	
33	Press <i>Edit</i> to switch to edit mode.	
34	Press <i>Additional Actions > Import From Excel</i> to import the corresponding MS Excel file and apply its content as settings since the actual settings of the single value decision table are empty.	

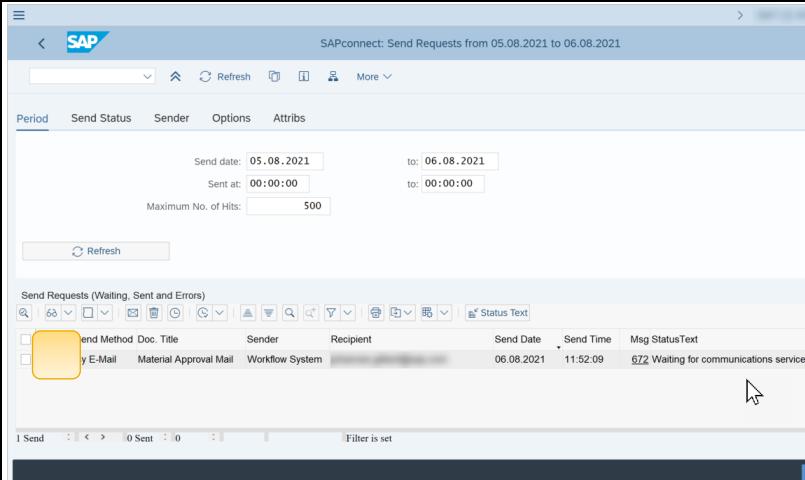
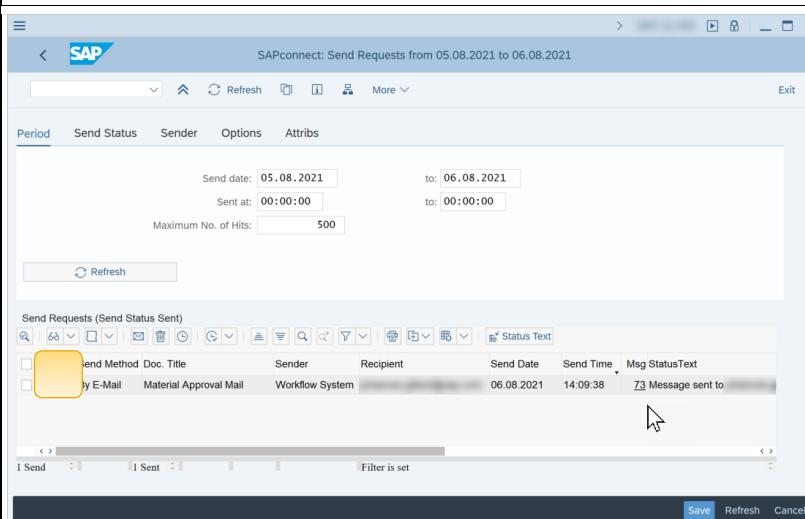
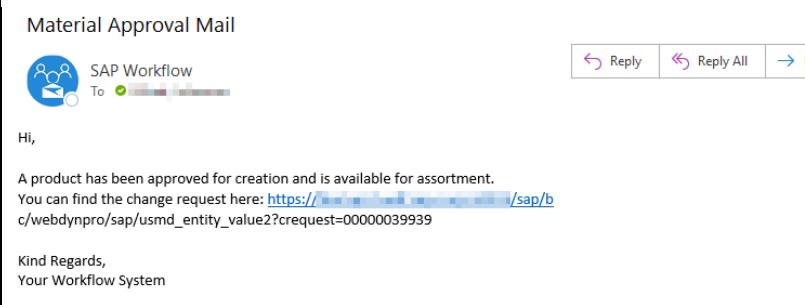
Step	Explanation	Screenshot
35	Select MS Excel file <i>DT_NON_USER_AGT_GRP_MAT01.xlsx</i> and then OK.	
36	The decision table should contain the adjusted values from the MS Excel file.	
37	Save the decision table.	
38	Activate the decision table.	
39	In case a confirmation for the activation is required chose Yes.	

2.4. Try-out

In this chapter the whole new configuration is tested. A new material is created via the newly created and configured change request type ZMAT01AM and an email notification is sent. To reduce the input-effort an existing material is copied.

Step	Explanation	Screenshot
1	Open MDGUI for Material Governance. Navigate to <i>Change Requests > Material Processing > Search Material</i> . Search for some appropriate material (here P-100).	
2	Select the material and press Copy .	
3	In the pop-up <i>Available Change Request Types</i> select the newly create one and press OK .	

Step	Explanation	Screenshot
4	Add an appropriate description and <i>Submit</i> the change request.	
5	In case the duplicate check pops up press <i>Continue</i> to go ahead with the current, new material.	
6	Get your change request processed to <i>Final Check Approved</i> .	

Step	Explanation	Screenshot
7	<p>Start transaction SOST to show <i>Business Communication Services' Send Requests</i>. Depending on the BCS settings the email will need some time to be sent by the system (<i>Status Waiting</i>, ▲). Nonetheless, a corresponding <i>Send Requests</i> should already be available.</p>	 <p>The screenshot shows the SAP Connect interface for sending requests. The search bar at the top contains the query 'SAPconnect: Send Requests from 05.08.2021 to 06.08.2021'. Below the search bar are filter options for Period, Send Status, Sender, Options, and Attrbs. The 'Send Status' dropdown is set to 'Waiting'. The main table displays 'Send Requests (Waiting, Sent and Errors)'. The first row in the table has its status highlighted with a yellow box: 'Send Method: Mail E-Mail', 'Doc. Title: Material Approval Mail', 'Sender: Workflow System', 'Recipient: [REDACTED]', 'Send Date: 06.08.2021', 'Send Time: 11:52:09', and 'Msg StatusText: 672 Waiting for communications service'.</p>
8	<p>The <i>Send Requests</i> is transmitted at some point in time (<i>Status Transmitted</i>, ■).</p>	 <p>This screenshot shows the same SAP Connect interface after the send request has been transmitted. The status for the first row has changed to 'Message sent' (indicated by a green square icon). The rest of the table and interface elements remain the same.</p>
9	<p>Check the corresponding mailbox. An email from the workflow should be received. Note: The step <i>Send Mail</i> does not allow rich text formatting without further actions (like implementing a program exit for the step).</p>	 <p>The screenshot shows an email from 'SAP Workflow' to '[REDACTED]' with the subject 'Material Approval Mail'. The email body starts with 'Hi,' followed by a message: 'A product has been approved for creation and is available for assortment. You can find the change request here: https://[REDACTED]/sap/bc/webdynpro/sap/usmd_entity_value2?crequest=00000039939'. The message concludes with 'Kind Regards, Your Workflow System'.</p>

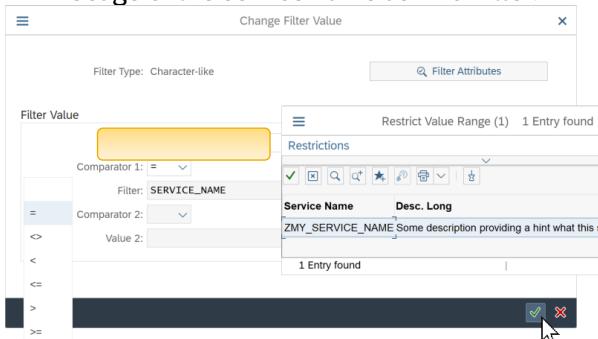
3. Email notification using BAdI System Caller Method

The goal of the approach described in this chapter is to send an email notification for an approved material creation. To achieve this **configuration and coding** is used. A new change request type, a BAdI implementation as well as a rule-based workflow are configured.

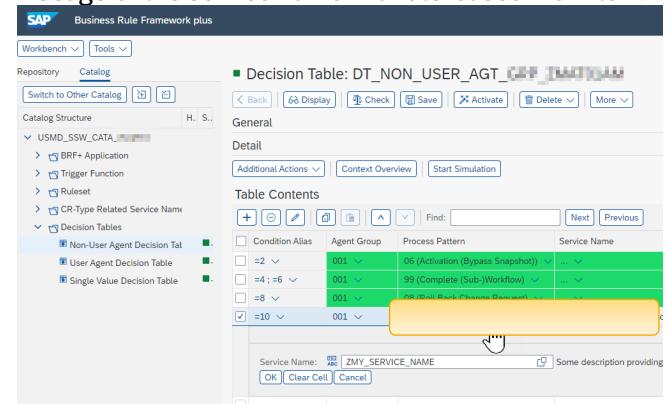
Chapter 3.1 describes the definition of a new service name for rule-based workflow. A new BAdI implementation for BAdI USMD_SSW_SYSTEM_METHOD_CALLER is created to perform the actual sending of the E-mail (chapter 3.2). A new change request type is created and configured (chapter 3.3). The configuration of a rule-based workflow for this new change request type is described in chapter 3.4. To omit starting from scratch the SAP-delivered change request type MAT01 (Create Material) is copied and adjusted. In chapter 3.5 the whole configuration is tested.

3.1. Create a service name in IMG

Usage of the service name as BAdI filter:

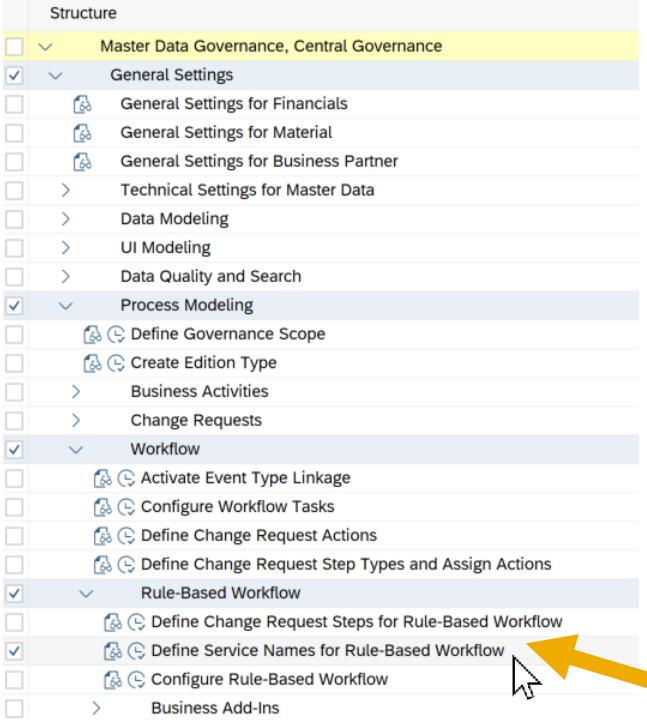
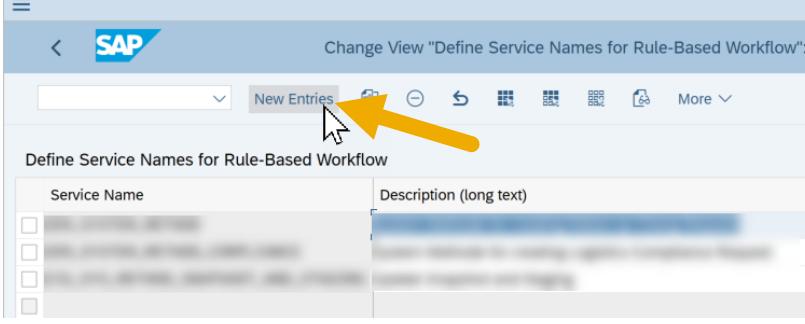
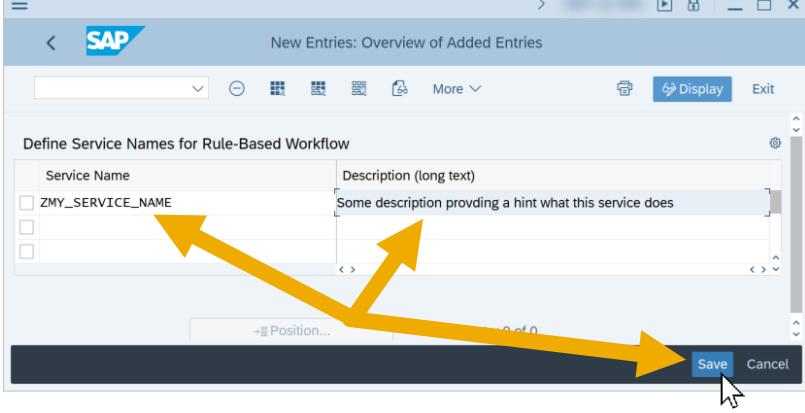


Usage of the service name in a rule-based workflow:



In the following steps, generic names and descriptions are used. For this guide, replace them as follows:

Object	Name / Description	To be replaced by:
Service Name	ZMY_SERVICE_NAME	ZNOTIFY_REQUESTOR
Service Description	Some description providing a hint as to what this service does	Notify Requestor About Approval

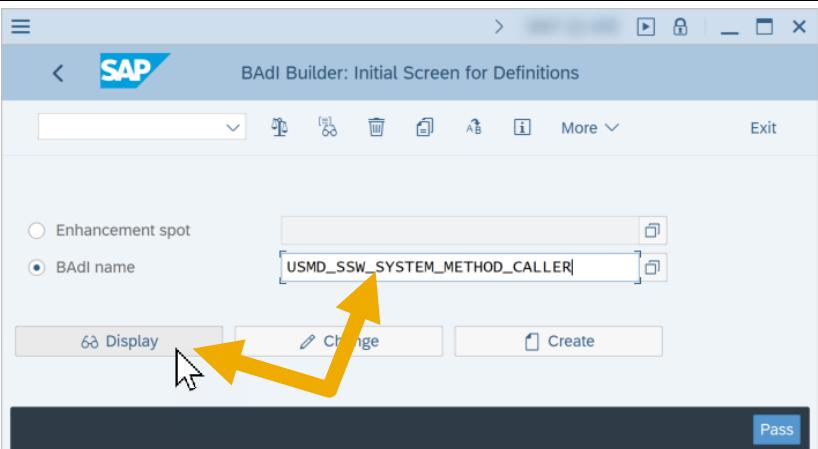
Step	Explanation	Screenshot
1	<p>Start transaction MDGIMG.</p> <p>Navigate to General Settings > Process Modeling > Workflow > Rule-Based Workflow > Configure Rule-Based Workflow > Define Service Names for Rule-Based Workflow.</p>	 <p>The screenshot shows the SAP Fiori navigation tree under the 'Structure' section. The path is: Master Data Governance, Central Governance > General Settings > Process Modeling > Rule-Based Workflow > Define Service Names for Rule-Based Workflow. The 'Define Service Names for Rule-Based Workflow' node is highlighted with a yellow arrow pointing to it.</p>
2	<p>Press New Entries.</p>	 <p>The screenshot shows the SAP Fiori Change View titled 'Change View "Define Service Names for Rule-Based Workflow"'. At the top, there is a 'New Entries' button with a plus sign icon, which is highlighted with a yellow arrow. Below the header, there is a table with columns 'Service Name' and 'Description (long text)'.</p>
3	<p>Create a new entry with the following data:</p> <ul style="list-style-type: none"> A meaningful name for Service Name, here: ZMY_SERVICE_NAME A Description providing a hint as to what this service does <p>Press Save.</p> <p>Note: You might need to chose a transport request to record you changes.</p>	 <p>The screenshot shows the SAP Fiori 'New Entries: Overview of Added Entries' screen. It displays a table with columns 'Service Name' and 'Description (long text)'. A single row is shown with the 'Service Name' field containing 'ZMY_SERVICE_NAME' and the 'Description (long text)' field containing 'Some description providing a hint what this service does'. At the bottom right, there is a 'Save' button, which is highlighted with a yellow arrow.</p>

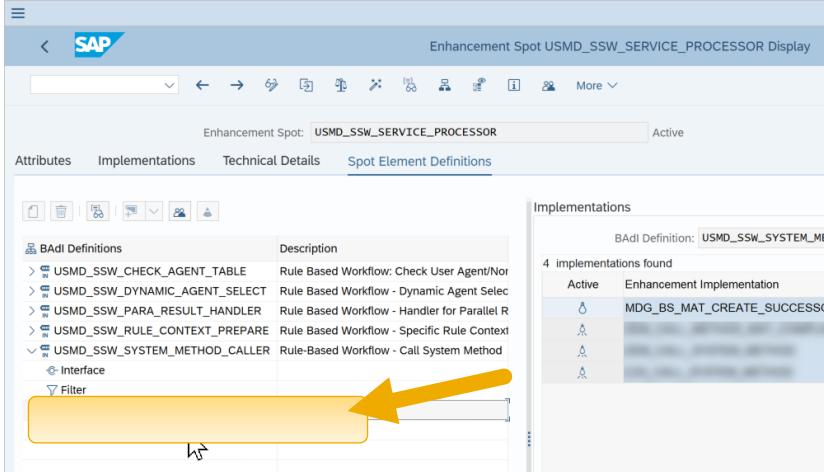
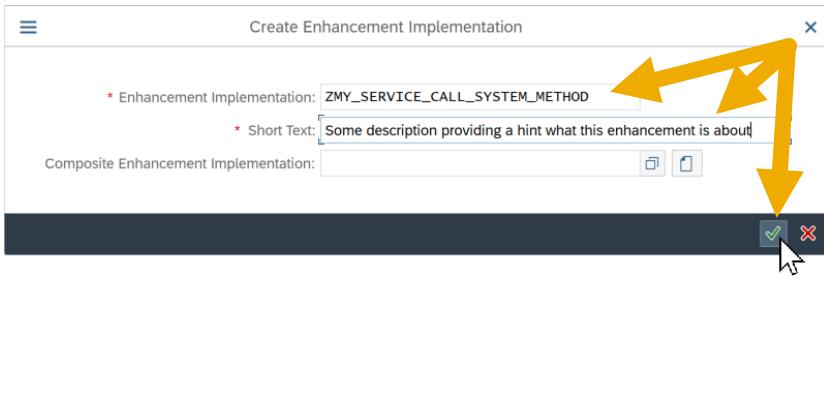
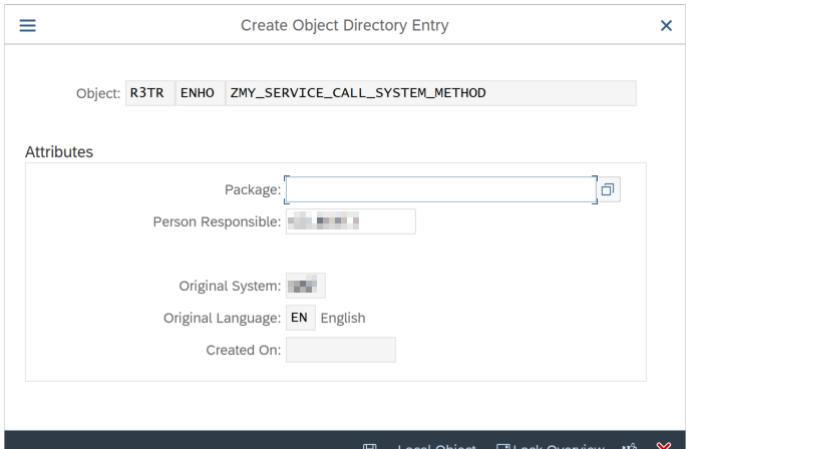
3.2. Create a BAdI implementation for BAdI ‘Rule-Based Workflow – Call System Method’

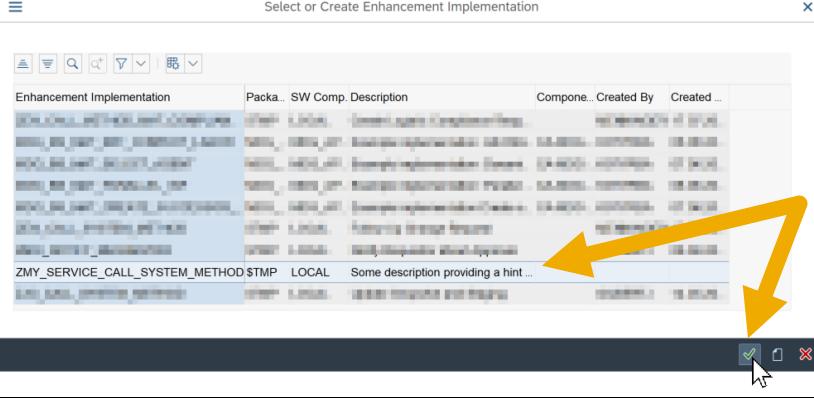
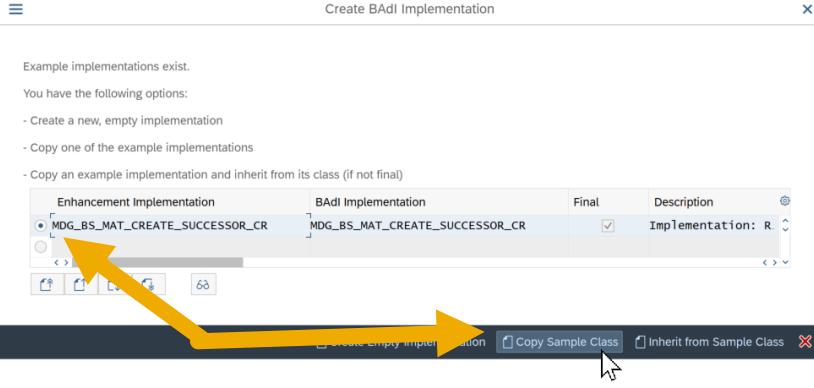
In this chapter a Business Add-In (BAdI) implementation for BAdI USMD_SSW_SYSTEM_METHOD_CALLER (Rule-Based Workflow - Call System Method) is created. It enables calling a system method in a rule-based workflow in SAP Master Data Governance. For more information, see the BAdI documentation. The service name is defined in the previous chapter 3.1 serves as BAdI filter. The actual implementation of the BAdI method depends on the use case.

In the following steps, generic names and descriptions are used. For this guide, replace them as follows:

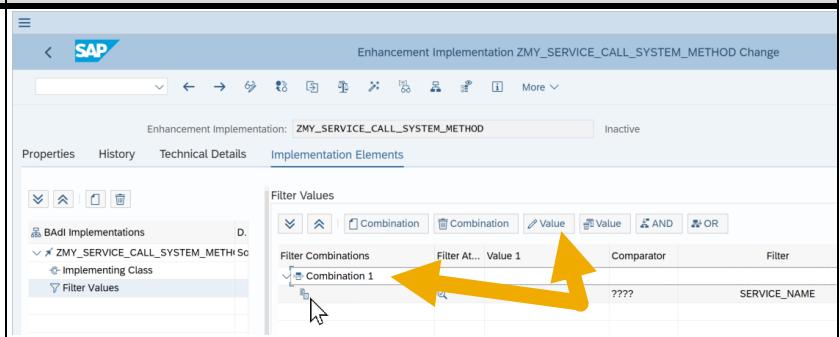
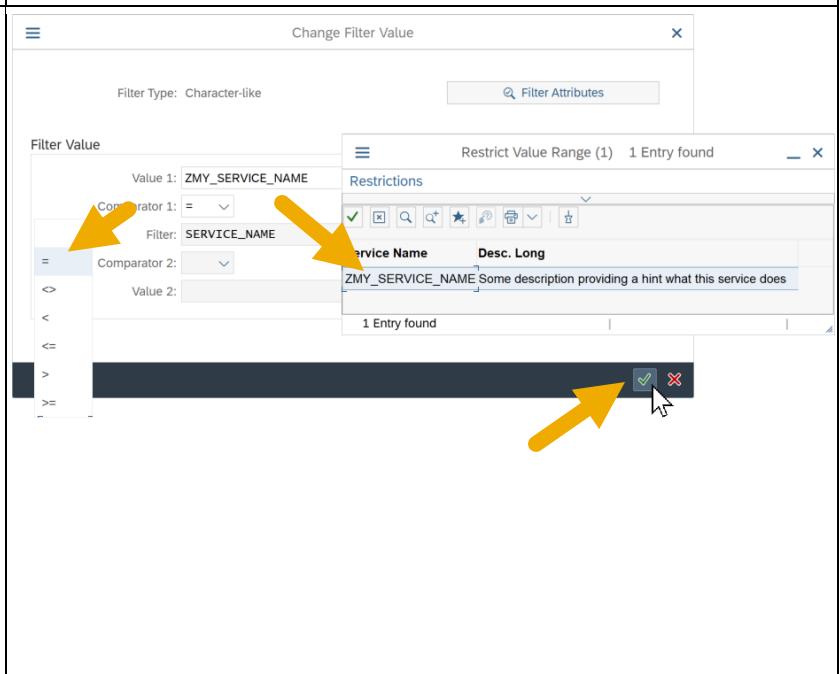
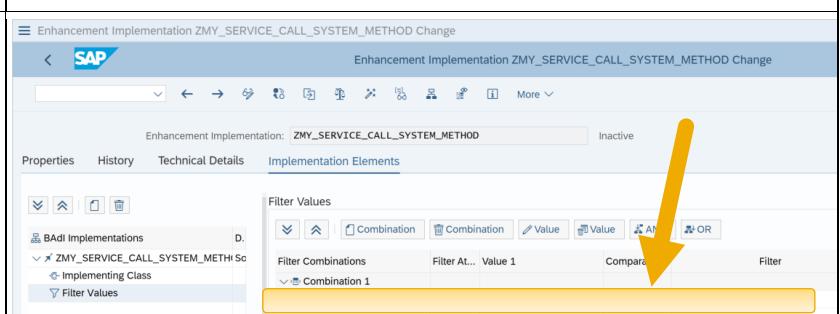
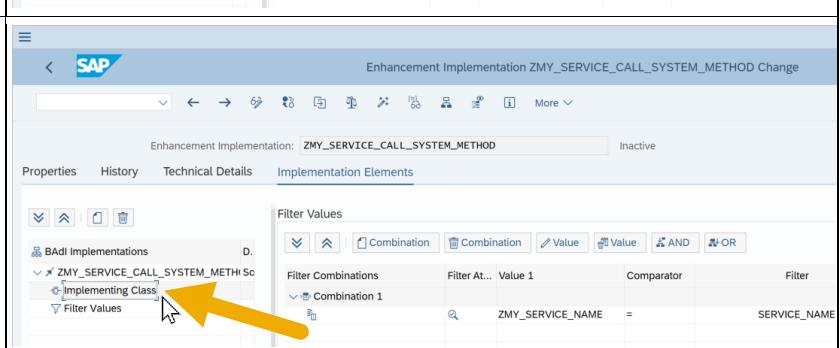
Object	Name / Description	To be replaced by:
Enhancement Implementation	ZMY_SERVICE_CALL_SYSTEM_METHOD	ZBDI_NOTIFY_REQUESTOR
Enhancement Implementation Short Text	Some description providing a hint as to what this enhancement is about	Notify Requestor About Approval
BAdI Implementation	ZMY_SERVICE_CALL_SYSTEM_METHOD	ZBDI_NOTIFY_REQUESTOR
BAdI Implementation Description	Some description providing a hint as to what this BAdI implementation is about.	Notify Requestor About Approval
BAdI Implementing Class	ZCL_MY_SERVICE_CALL_SYS_METHOD	ZCL_BDI_NOTIFY_REQUESTOR
BAdI Filter Value	ZMY_SERVICE_NAME	ZNOTIFY_REQUESTOR

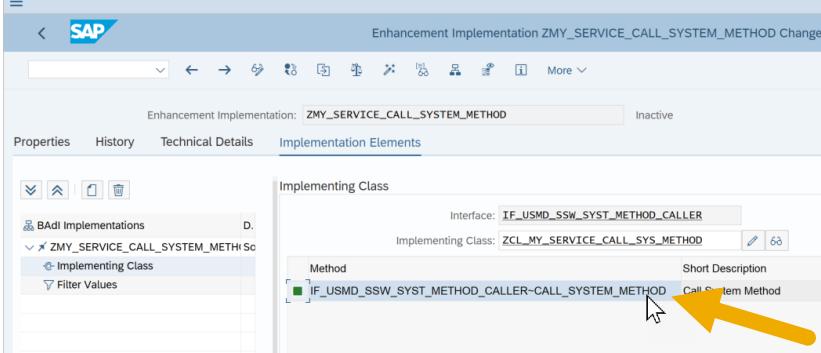
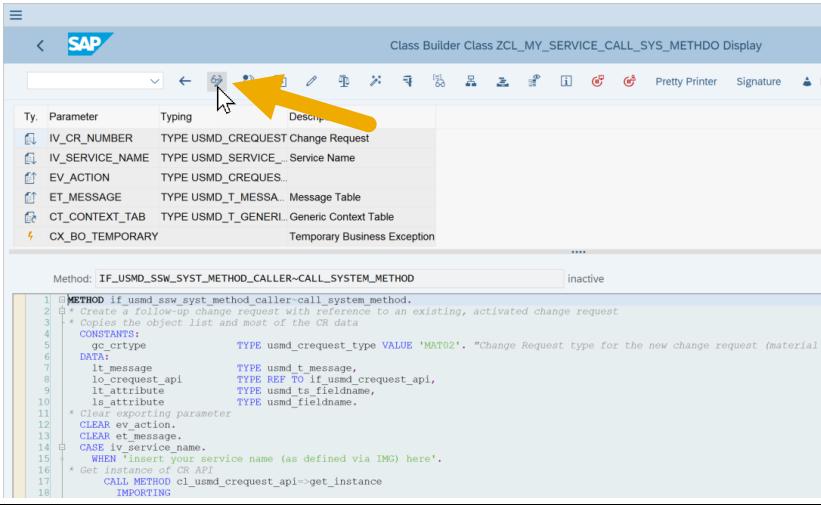
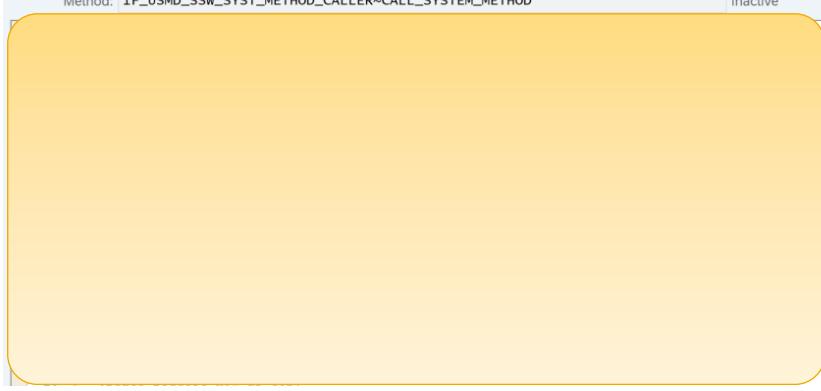
Step	Explanation	Screenshot
1	Start transaction SE18. Select option <i>BAdI Name</i> and enter USMD_SSW_SYSTEM_METHOD_CALLER . Press <i>Display</i> .	

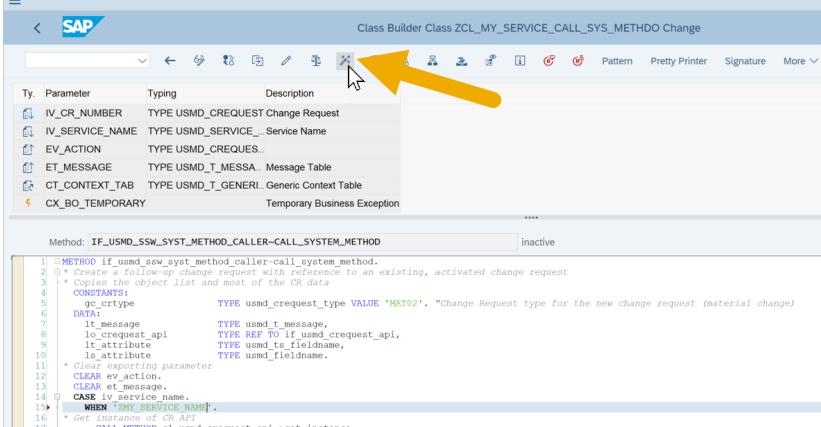
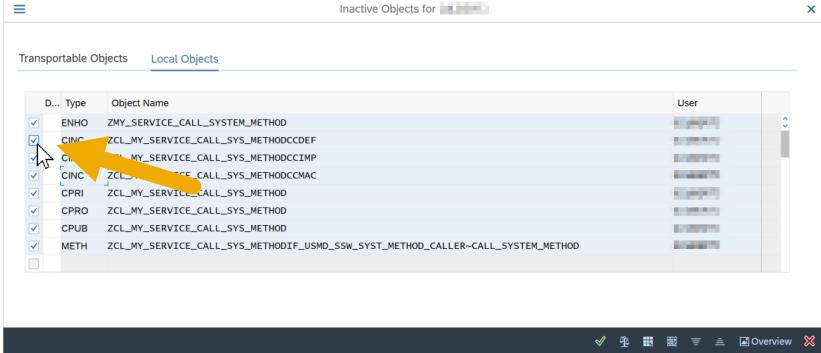
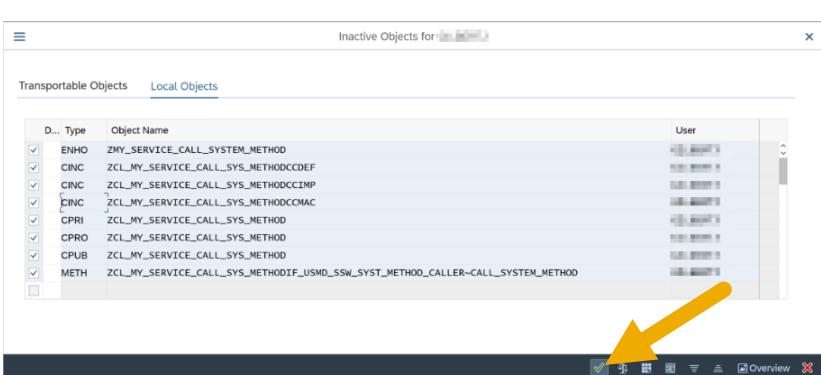
Step	Explanation	Screenshot
2	Open the context menu on the <i>Implementations</i> for BAdI USMD_SSW_SYSTEM_METHOD_CALLER and chose <i>Create BAdI Implementation</i> .	
3	In the pop-up <i>Select or Create Enhancement Implementation</i> , press <i>Create Enhancement Implementation</i> .	
4	In the pop-up <i>Create Enhancement Implementation</i> enter the following data: <ul style="list-style-type: none"> A meaningful name for the <i>Enhancement Implementation</i>, here: ZMY_SERVICE_CALL_SYSTEM_METHOD Some <i>Short Text</i> providing a hint as to what this enhancement is about Press <i>Creation of Enhancement</i> .	
5	Chose a suitable development package or store the enhancement implementation as a local object depending on your needs.	

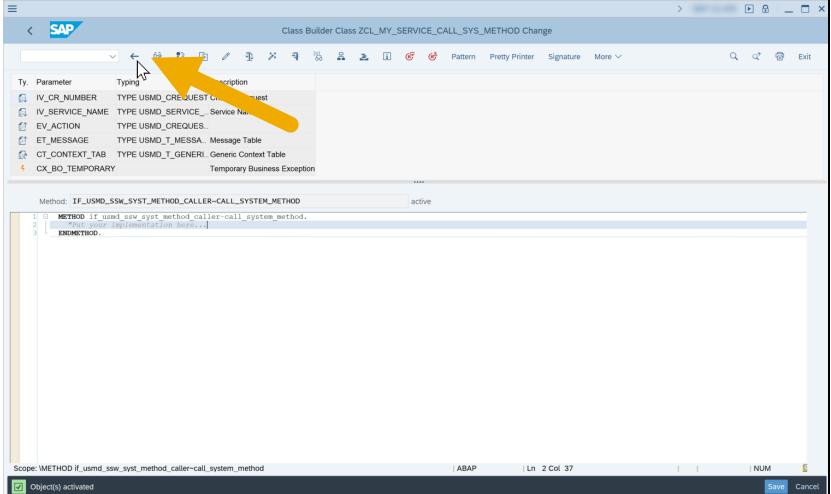
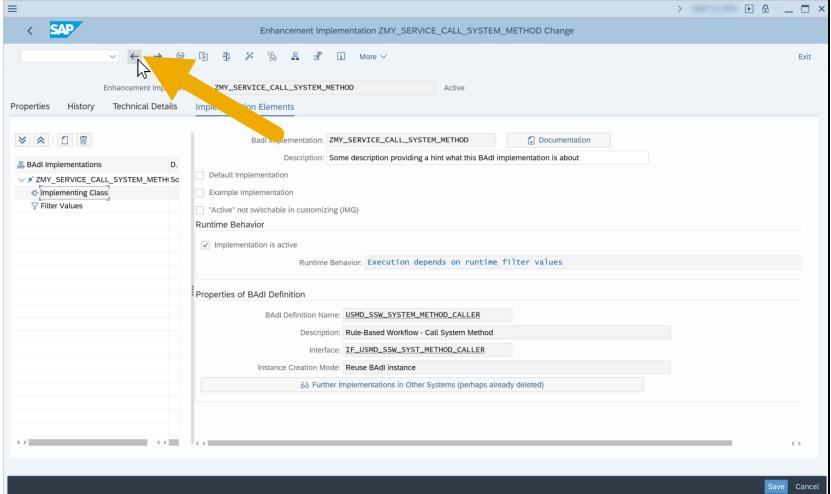
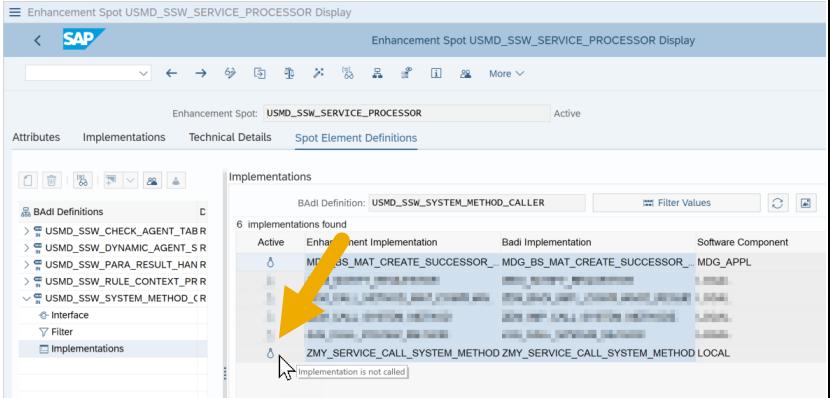
Step	Explanation	Screenshot
6	<p>In the pop-up <i>Select or Create Enhancement Implementation</i> amongst others, the new enhancement implementation is listed. Select it and press <i>Select Specified Enhancement Implementation</i>.</p>	
7	<p>In the pop-up <i>Create BAdI Implementation</i> enter the following data:</p> <ul style="list-style-type: none"> A meaningful name for the <i>BAdI Implementation</i>, here: ZMY_SERVICE_CALL_SYS TEM_METHOD Some <i>Description</i> providing a hint as to what this BAdI implementation is about The name of the <i>Implementing Class</i> which implemented the BAdI interface, here: ZCL_MY_SERVICE_CALL_SYS_METHOD <p>Press <i>Continue</i>.</p>	
8	<p>In the next pop-up chose <i>Enhancement Implementation</i> MDG_BS_MAT_CREATE_SUCCESOR_CR and press <i>Copy Sample Class</i>.</p>	

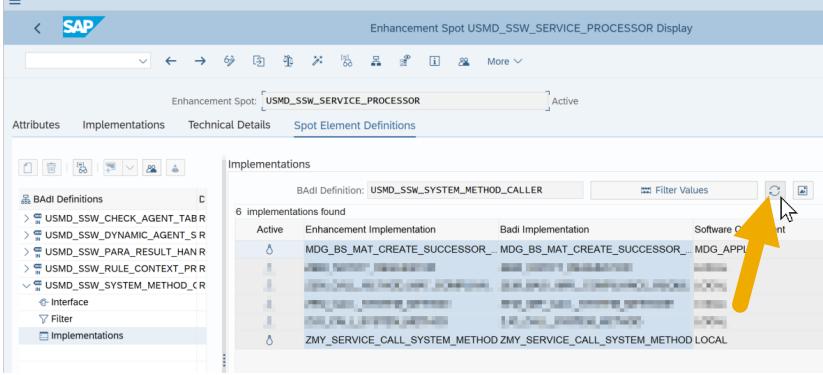
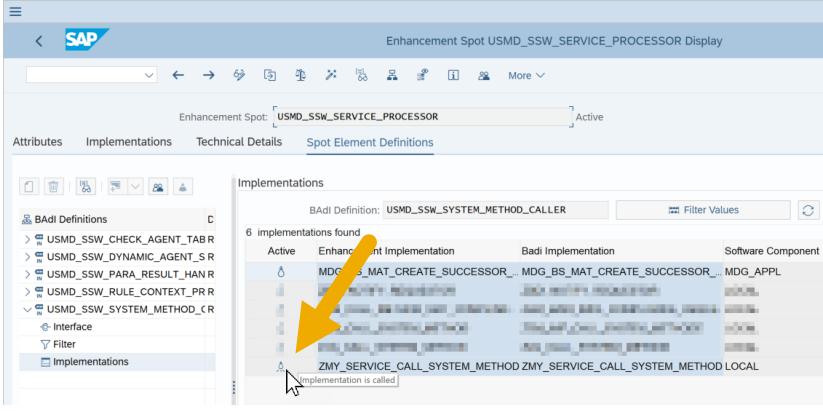
Step	Explanation	Screenshot
9	Chose a suitable development package or store the implementation class as a local object depending on your needs.	
10	Navigate to the <i>Filter Values</i> for the BAdI implementation.	
11	Go to edit mode.	
12	A filter shall be created for filter SERVICE_NAME to be ZMY_SERVICE_NAME, matching the service defined in the Customizing <i>Define Service Names for Rule-Based Workflow</i> (see 3.1). Press <i>Combination</i> .	

Step	Explanation	Screenshot
13	Double-click the combination or press Value.	
14	<p>In the pop-up <i>Change Filter Value</i> enter or choose the following data:</p> <ul style="list-style-type: none"> • Value 1: The service name defined in Customizing <i>Define Service Names for Rule-Based Workflow</i> (see 3.1), here: ZMY_SERVICE_NAME. You can use the value help to choose the service name. • Comparator 1: = Press Continue. <p>Note: One BAdI implementation can handle calls for more than one service name. Use the service name distinguish between use cases.</p>	
15	As result the filter is adjusted.	
16	Navigate to <i>Implementing Class</i> .	

Step	Explanation	Screenshot
17	Double-click BAdI method IF_USMD_SSW_SYS_METHOD_CALLER~CALL_SYSTEM_METHOD.	
18	Go to edit mode.	 <pre data-bbox="609 900 1430 1102"> Method: IF_USMD_SSW_SYS_METHOD_CALLER~CALL_SYSTEM_METHOD inactive 1 METHOD if_usmd_ssw_syst_method_caller-call_system_method. 2 * Create a follow-up change request with reference to an existing, activated change request 3 * Copy the object list and most of the CR data 4 CONSTANTS: 5 gc_crytype TYPE usmd_crequest_change_request. 6 DATA: 7 lt_message TYPE usmd_t_message, 8 lo_crequest_api TYPE REF TO if_usmd_crequest_api, 9 ls_attribute TYPE usmd_ts_attribute, 10 ls_attribute TYPE usmd_fieldname. 11 * Clear exporting parameter 12 CLEAR ev_action. 13 CLEAR et_message. 14 CLEAR ls_attribute. 15 WHEN 'insert your service name (as defined via IMG) here'. 16 * Get instance of CR API 17 CALL METHOD cl_usmd_crequest_api->get_instance 18 IMPORTING </pre>
19	<p>Change the example implementation according to your needs. At least change the CASE-WHEN statement to match the service name defined in the Customizing <i>Define Service Names for Rule-Based Workflow</i> (see 3.1), here: ZMY_SERVICE_NAME.</p>	

Step	Explanation	Screenshot																																				
20	Activate the method. In case other objects are still to be adjusted, select them for activation as well.	 <pre> Method: IF_USMD_SSW_SYS_METHOD_CALLER-CALL_SYSTEM_METHOD inactive 1 METHOD if_usmd_ssw_sys_method_caller-call_system_method. 2 * Create a follow-up change request with reference to an existing, activated change request 3 * Copied the object list and most of the CR data 4 * COUNTRY 5 gc_cryptp TYPE usmd_crequest_type VALUE 'MAT02'. "Change Request type for the new change request (material change) 6 DATA: 7 lt_message TYPE usmd_t_message, 8 lt_crequest_api TYPE REF TO if_usmd_crequest_api, 9 lt_attribute TYPE usmd_ts_fieldname, 10 ls_attribute TYPE usmd_fieldname. 11 * Create a follow-up parameter 12 CLEAR ev_action, 13 CLEAR et_message, 14 CASE iv_service_name. 15 WHEN 'ZCL_MY_SERVICE_CALL_SYS_METHODO'. 16 CALL METHOD cl_usmd_crequest_api->get_instance 17 IMPORTING 18 et_message = lt_message 19 ENDCASE. </pre>																																				
21	The enhancement, the BAdl implementation as well as the BAdl class have not been activated so far. In the pop-up <i>Inactive Objects for ...</i> select all of them.	 <table border="1"> <thead> <tr> <th>D...</th> <th>Type</th> <th>Object Name</th> <th>User</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/></td> <td>ENHO</td> <td>ZMY_SERVICE_CALL_SYSTEM_METHODO</td> <td>[User Icon]</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>CINC</td> <td>ZCL_MY_SERVICE_CALL_SYS_METHODOCCDEF</td> <td>[User Icon]</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>CINC</td> <td>ZCL_MY_SERVICE_CALL_SYS_METHODOCIMP</td> <td>[User Icon]</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>CINC</td> <td>ZCL_MY_SERVICE_CALL_SYS_METHODOCMAC</td> <td>[User Icon]</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>CPRI</td> <td>ZCL_MY_SERVICE_CALL_SYS_METHOD</td> <td>[User Icon]</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>CPRO</td> <td>ZCL_MY_SERVICE_CALL_SYS_METHOD</td> <td>[User Icon]</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>CPUB</td> <td>ZCL_MY_SERVICE_CALL_SYS_METHOD</td> <td>[User Icon]</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>METH</td> <td>ZCL_MY_SERVICE_CALL_SYS_METHODIF_USMD_SSW_SYS_METHOD_CALLER-CALL_SYSTEM_METHOD</td> <td>[User Icon]</td> </tr> </tbody> </table>	D...	Type	Object Name	User	<input checked="" type="checkbox"/>	ENHO	ZMY_SERVICE_CALL_SYSTEM_METHODO	[User Icon]	<input checked="" type="checkbox"/>	CINC	ZCL_MY_SERVICE_CALL_SYS_METHODOCCDEF	[User Icon]	<input checked="" type="checkbox"/>	CINC	ZCL_MY_SERVICE_CALL_SYS_METHODOCIMP	[User Icon]	<input checked="" type="checkbox"/>	CINC	ZCL_MY_SERVICE_CALL_SYS_METHODOCMAC	[User Icon]	<input checked="" type="checkbox"/>	CPRI	ZCL_MY_SERVICE_CALL_SYS_METHOD	[User Icon]	<input checked="" type="checkbox"/>	CPRO	ZCL_MY_SERVICE_CALL_SYS_METHOD	[User Icon]	<input checked="" type="checkbox"/>	CPUB	ZCL_MY_SERVICE_CALL_SYS_METHOD	[User Icon]	<input checked="" type="checkbox"/>	METH	ZCL_MY_SERVICE_CALL_SYS_METHODIF_USMD_SSW_SYS_METHOD_CALLER-CALL_SYSTEM_METHOD	[User Icon]
D...	Type	Object Name	User																																			
<input checked="" type="checkbox"/>	ENHO	ZMY_SERVICE_CALL_SYSTEM_METHODO	[User Icon]																																			
<input checked="" type="checkbox"/>	CINC	ZCL_MY_SERVICE_CALL_SYS_METHODOCCDEF	[User Icon]																																			
<input checked="" type="checkbox"/>	CINC	ZCL_MY_SERVICE_CALL_SYS_METHODOCIMP	[User Icon]																																			
<input checked="" type="checkbox"/>	CINC	ZCL_MY_SERVICE_CALL_SYS_METHODOCMAC	[User Icon]																																			
<input checked="" type="checkbox"/>	CPRI	ZCL_MY_SERVICE_CALL_SYS_METHOD	[User Icon]																																			
<input checked="" type="checkbox"/>	CPRO	ZCL_MY_SERVICE_CALL_SYS_METHOD	[User Icon]																																			
<input checked="" type="checkbox"/>	CPUB	ZCL_MY_SERVICE_CALL_SYS_METHOD	[User Icon]																																			
<input checked="" type="checkbox"/>	METH	ZCL_MY_SERVICE_CALL_SYS_METHODIF_USMD_SSW_SYS_METHOD_CALLER-CALL_SYSTEM_METHOD	[User Icon]																																			
22	To activate the selected object press continue.																																					

Step	Explanation	Screenshot
23	Navigate back to the enhancement implementation.	
24	Navigate back to enhancement spot USMD_SSW_SERVICE_PROCESSOR.	
25	Check the activation status (column Active) for your enhancement implementation. It must be Active or Implementation is called.	

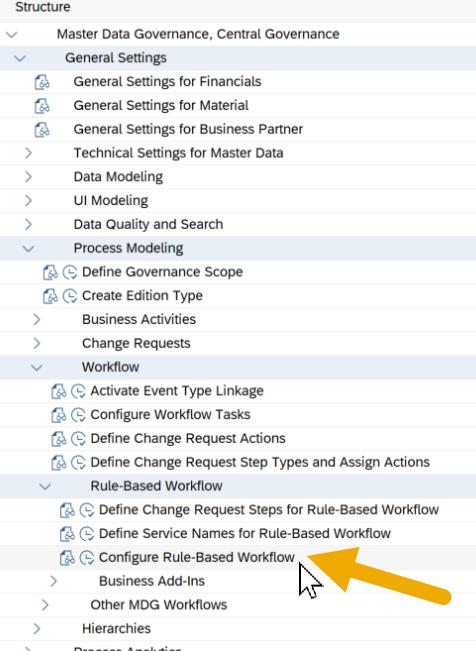
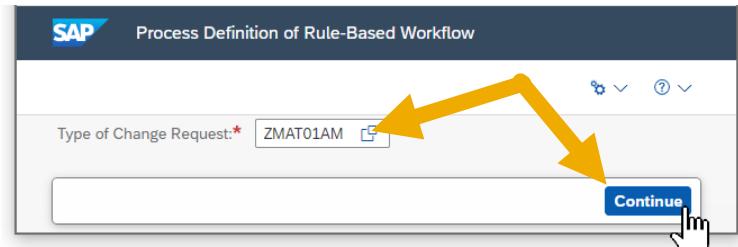
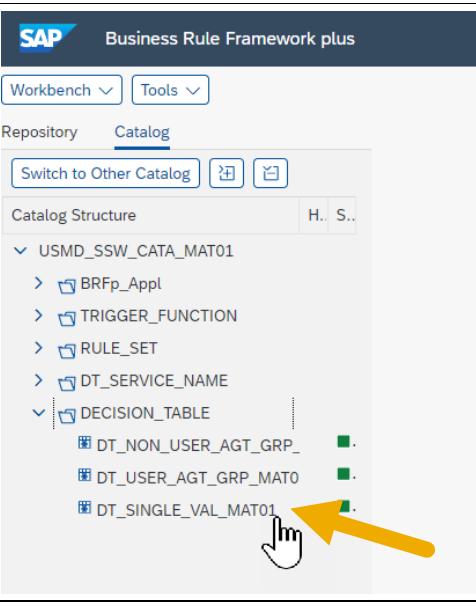
Step	Explanation	Screenshot
26	You may need to press <i>Refresh</i> to get the current status.	 <p>The screenshot shows the SAP Fiori interface for managing enhancement spots. The title bar says "Enhancement Spot USMD_SSW_SERVICE_PROCESSOR Display". The "Implementations" tab is selected. On the left, there's a tree view with nodes like "BAdI Definitions", "Interface", and "Implementations". The main area shows a table with columns: Active, Enhancement Implementation, Badi Implementation, and Software Component. There are 6 implementations found. One implementation is highlighted, showing "MDG_BS_MAT_CREATE_SUCCESSOR_" in the Enhancement Implementation column and "MDG_APPL" in the Software Component column. A yellow arrow points to the "Software Component" column header.</p>
27	After refreshing, the status must be <i>Active</i> or rather <i>Implementation is called</i> .	 <p>This screenshot is similar to the previous one but shows the state after refresh. The "Implementation is called" status is now visible in the "Implementation" column for the highlighted row. A yellow arrow points to this status.</p>

3.3. Create a new change request type and configure change request steps

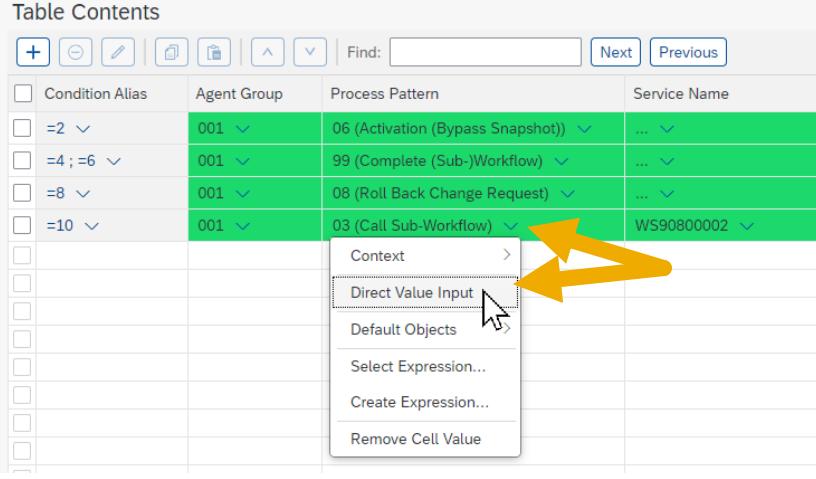
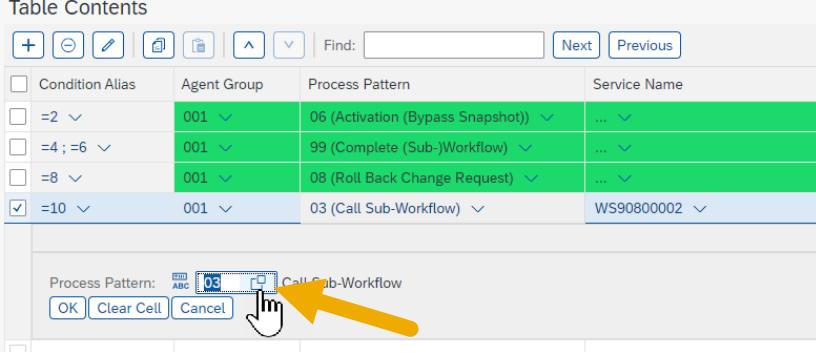
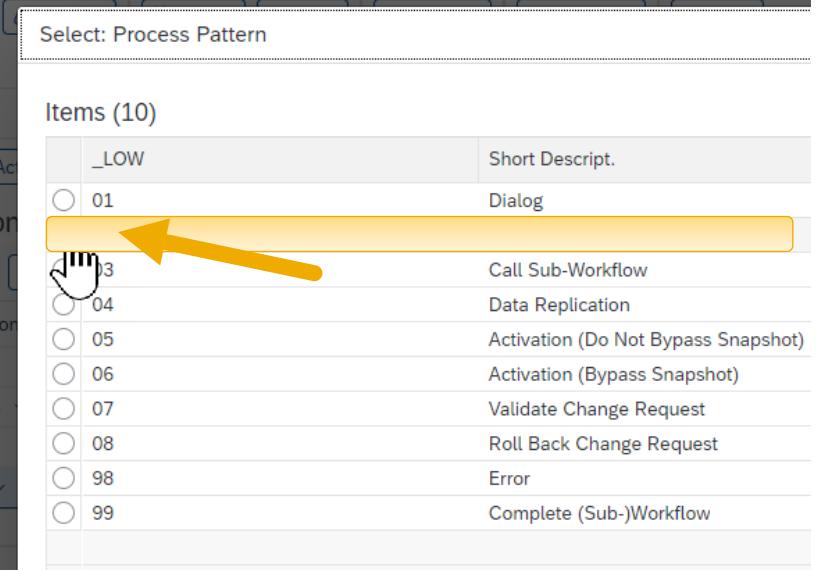
For the steps required see chapter 2.2. The steps described are also valid for email notification via BAdI.

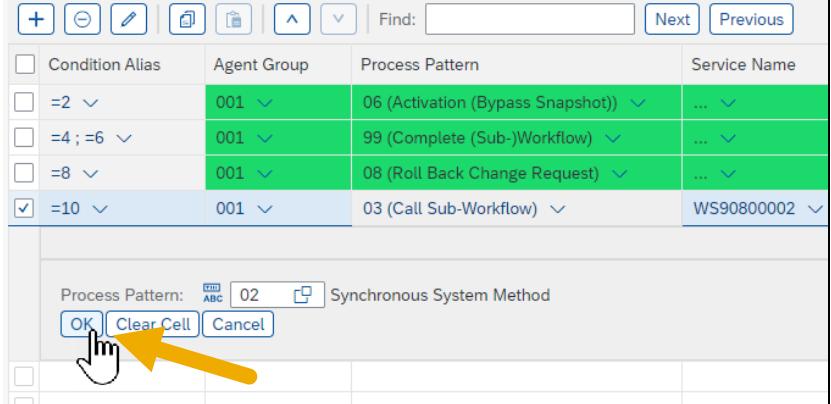
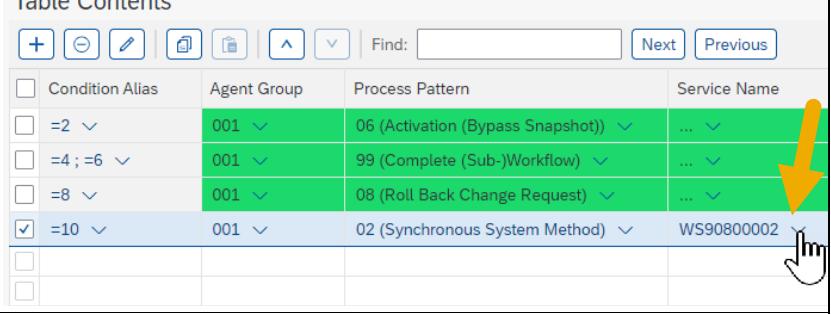
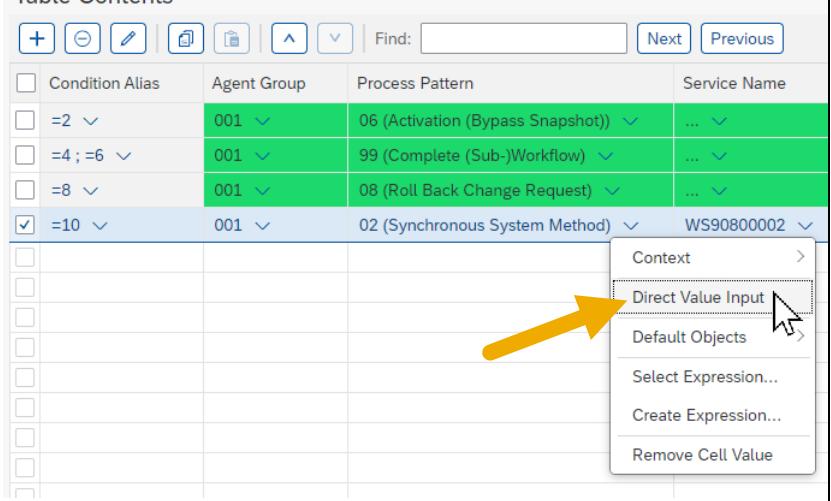
3.4. Configure rule-based workflow for the new change request type

For the steps required see chapter 3.2. The steps described are also valid for email notification via BAdI with some slight adjustments, described in the following.

Step	Explanation	Screenshot
1	In MDGIMG navigate again to General Settings > Process Modeling > Workflow > Rule-Based Workflow > Configure Rule-Based Workflow .	
2	Enter the newly created change request type ZMAT01AM and press Continue .	
3	Navigate to the single value decision table (DT_SINGLE_VAL_MAT01).	

Step	Explanation	Screenshot																																													
4	For CR Previous Step 91 (Activation) and Previous Action 31 (Activation successful) Condition Alias 10 is configured.	<p>Decision Table: DT_SINGLE_VAL_ZMAT01AM</p> <table border="1"> <thead> <tr> <th>CR Previous Step</th> <th>Previous Action</th> <th>Condition Alias</th> <th>New Chg. Req. Step</th> <th>New CR Status</th> </tr> </thead> <tbody> <tr> <td>=00 (Processing)</td> <td>=09 (Activate)</td> <td>1</td> <td>90 (Final Check)</td> <td>02 (Changes to Be Executed)</td> </tr> <tr> <td>=90 (Final Check)</td> <td>=04 (Reject)</td> <td>2</td> <td>91 (Activation)</td> <td>02 (Changes to Be Executed)</td> </tr> <tr> <td>=90 (Final Check)</td> <td>=04 (Reject)</td> <td>3</td> <td>95 (Revision Processing)</td> <td>10 (To Revert: Perform Changes)</td> </tr> <tr style="outline: 2px solid orange;"> <td>=10</td> <td><>31 (Activation successful)</td> <td>4</td> <td>99 (Complete)</td> <td>09 (Final Check Approved)</td> </tr> <tr> <td>=91 (Activation)</td> <td></td> <td>5</td> <td>90 (Final Check)</td> <td>11 (Process Errors After Activation)</td> </tr> <tr> <td>=92 (Revision)</td> <td></td> <td>6</td> <td>99 (Complete)</td> <td>06 (Final Check Rejected)</td> </tr> <tr> <td>=95 (Revision Proc...)</td> <td>=07 (Resubmit)</td> <td>7</td> <td>90 (Final Check)</td> <td>02 (Changes to Be Executed)</td> </tr> <tr> <td>=95 (Revision Proc...)</td> <td>=06 (Withdraw)</td> <td>8</td> <td>92 (Revision)</td> <td>02 (Changes to Be Executed)</td> </tr> </tbody> </table>	CR Previous Step	Previous Action	Condition Alias	New Chg. Req. Step	New CR Status	=00 (Processing)	=09 (Activate)	1	90 (Final Check)	02 (Changes to Be Executed)	=90 (Final Check)	=04 (Reject)	2	91 (Activation)	02 (Changes to Be Executed)	=90 (Final Check)	=04 (Reject)	3	95 (Revision Processing)	10 (To Revert: Perform Changes)	=10	<>31 (Activation successful)	4	99 (Complete)	09 (Final Check Approved)	=91 (Activation)		5	90 (Final Check)	11 (Process Errors After Activation)	=92 (Revision)		6	99 (Complete)	06 (Final Check Rejected)	=95 (Revision Proc...)	=07 (Resubmit)	7	90 (Final Check)	02 (Changes to Be Executed)	=95 (Revision Proc...)	=06 (Withdraw)	8	92 (Revision)	02 (Changes to Be Executed)
CR Previous Step	Previous Action	Condition Alias	New Chg. Req. Step	New CR Status																																											
=00 (Processing)	=09 (Activate)	1	90 (Final Check)	02 (Changes to Be Executed)																																											
=90 (Final Check)	=04 (Reject)	2	91 (Activation)	02 (Changes to Be Executed)																																											
=90 (Final Check)	=04 (Reject)	3	95 (Revision Processing)	10 (To Revert: Perform Changes)																																											
=10	<>31 (Activation successful)	4	99 (Complete)	09 (Final Check Approved)																																											
=91 (Activation)		5	90 (Final Check)	11 (Process Errors After Activation)																																											
=92 (Revision)		6	99 (Complete)	06 (Final Check Rejected)																																											
=95 (Revision Proc...)	=07 (Resubmit)	7	90 (Final Check)	02 (Changes to Be Executed)																																											
=95 (Revision Proc...)	=06 (Withdraw)	8	92 (Revision)	02 (Changes to Be Executed)																																											
5	Navigate to the user agent decision table (DT_NON_USER_AGT_GRP_ZMAT01AM).																																														
6	Press <i>Edit</i> to switch to edit mode.	<table border="1"> <thead> <tr> <th>Condition Alias</th> <th>Agent Group</th> <th>Process Pattern</th> <th>Service Name</th> </tr> </thead> <tbody> <tr> <td>=2</td> <td>001</td> <td>06 (Activation (Bypass Snapshot))</td> <td></td> </tr> <tr> <td>=4 ; =6</td> <td>001</td> <td>99 (Complete (Sub-)Workflow)</td> <td></td> </tr> <tr> <td>=8</td> <td>001</td> <td>08 (Roll Back Change Request)</td> <td></td> </tr> <tr> <td>=10</td> <td>001</td> <td>03 (Call Sub-Workflow)</td> <td>WS90800002</td> </tr> </tbody> </table>	Condition Alias	Agent Group	Process Pattern	Service Name	=2	001	06 (Activation (Bypass Snapshot))		=4 ; =6	001	99 (Complete (Sub-)Workflow)		=8	001	08 (Roll Back Change Request)		=10	001	03 (Call Sub-Workflow)	WS90800002																									
Condition Alias	Agent Group	Process Pattern	Service Name																																												
=2	001	06 (Activation (Bypass Snapshot))																																													
=4 ; =6	001	99 (Complete (Sub-)Workflow)																																													
=8	001	08 (Roll Back Change Request)																																													
=10	001	03 (Call Sub-Workflow)	WS90800002																																												

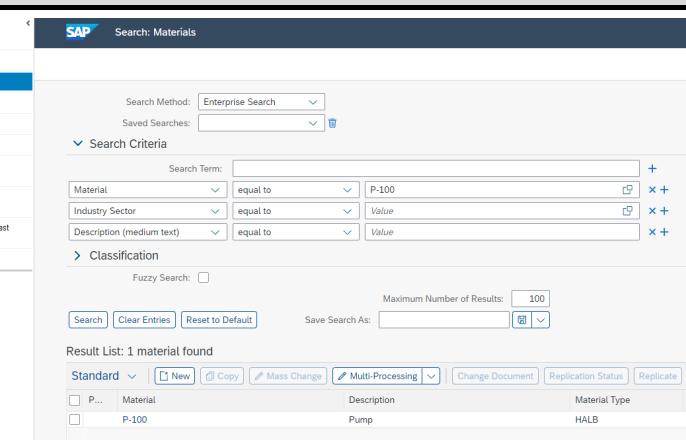
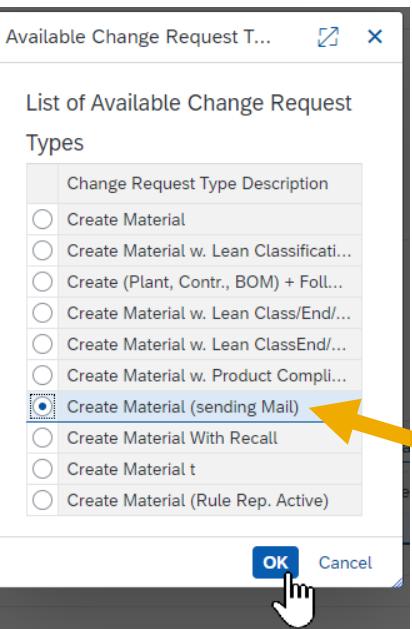
Step	Explanation	Screenshot																						
7	<p>For the row with <i>Condition Alias 10</i> the <i>Process Pattern</i> as well as the <i>Service Name</i> need to be changed. The current values are intended for the approach sending an email notification using a sub-workflow.</p> <p>Open the drop-down menu for <i>Process Pattern</i> for this row and choose <i>Direct Value Input</i>.</p>																							
8	Enter <i>Process Pattern 02</i> (Synchronous System Method) or go via the value help.																							
9	In the value help, choose 02, <i>Synchronous System Method</i> .	 <table border="1"> <thead> <tr> <th colspan="2">Items (10)</th> </tr> </thead> <tbody> <tr> <td>_LOW</td> <td>Short Descript.</td> </tr> <tr> <td>01</td> <td>Dialog</td> </tr> <tr> <td>03</td> <td>Call Sub-Workflow</td> </tr> <tr> <td>04</td> <td>Data Replication</td> </tr> <tr> <td>05</td> <td>Activation (Do Not Bypass Snapshot)</td> </tr> <tr> <td>06</td> <td>Activation (Bypass Snapshot)</td> </tr> <tr> <td>07</td> <td>Validate Change Request</td> </tr> <tr> <td>08</td> <td>Roll Back Change Request</td> </tr> <tr> <td>98</td> <td>Error</td> </tr> <tr> <td>99</td> <td>Complete (Sub-)Workflow</td> </tr> </tbody> </table>	Items (10)		_LOW	Short Descript.	01	Dialog	03	Call Sub-Workflow	04	Data Replication	05	Activation (Do Not Bypass Snapshot)	06	Activation (Bypass Snapshot)	07	Validate Change Request	08	Roll Back Change Request	98	Error	99	Complete (Sub-)Workflow
Items (10)																								
_LOW	Short Descript.																							
01	Dialog																							
03	Call Sub-Workflow																							
04	Data Replication																							
05	Activation (Do Not Bypass Snapshot)																							
06	Activation (Bypass Snapshot)																							
07	Validate Change Request																							
08	Roll Back Change Request																							
98	Error																							
99	Complete (Sub-)Workflow																							

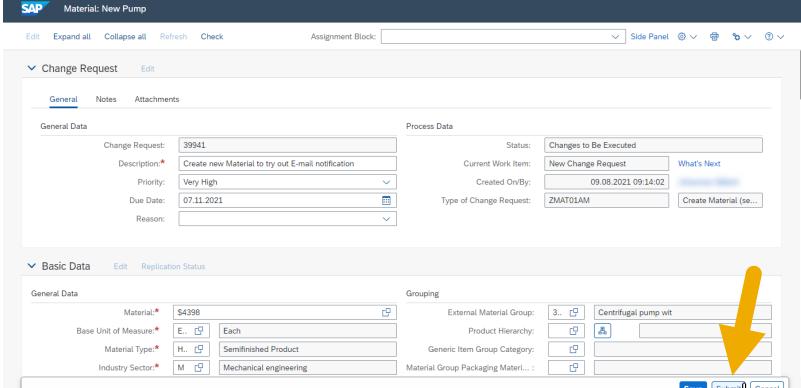
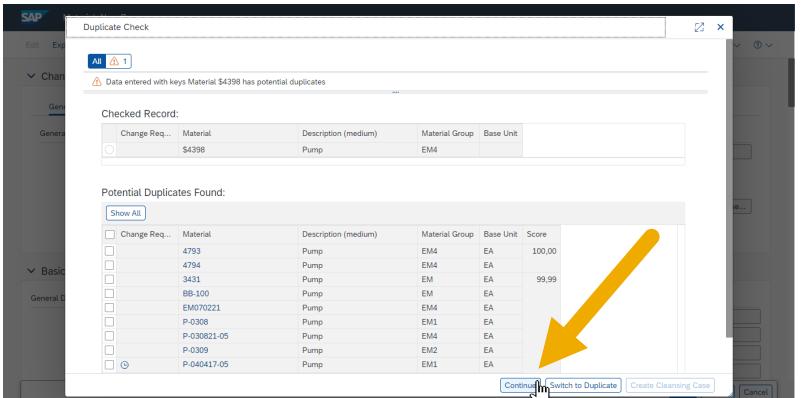
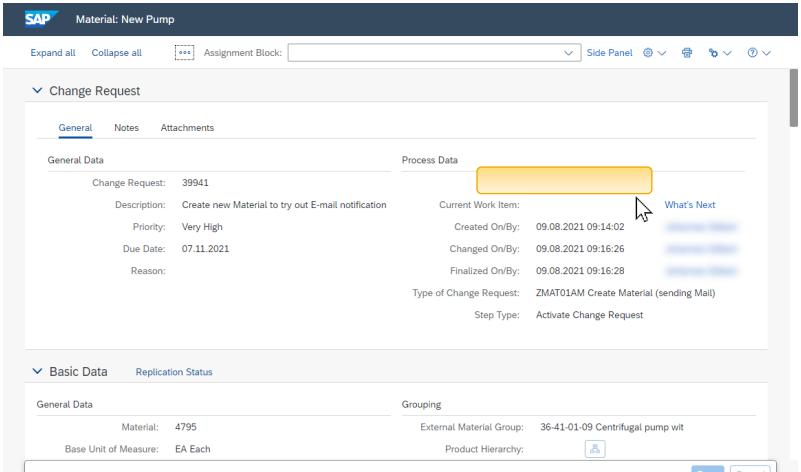
Step	Explanation	Screenshot
10	Press OK.	 <p>Table Contents</p> <p>Condition Alias Agent Group Process Pattern Service Name</p> <ul style="list-style-type: none"> =2 001 06 (Activation (Bypass Snapshot)) ... =4 ; =6 001 99 (Complete (Sub-)Workflow) ... =8 001 08 (Roll Back Change Request) ... =10 001 03 (Call Sub-Workflow) WS90800002 <p>Process Pattern: 02 Synchronous System Method</p> <p>OK Clear Cell Cancel</p>
11	Open the drop-down menu for the Service Name of this row.	 <p>Table Contents</p> <p>Condition Alias Agent Group Process Pattern Service Name</p> <ul style="list-style-type: none"> =2 001 06 (Activation (Bypass Snapshot)) ... =4 ; =6 001 99 (Complete (Sub-)Workflow) ... =8 001 08 (Roll Back Change Request) ... =10 001 02 (Synchronous System Method) WS90800002
12	Chose Direct Value Input.	 <p>Table Contents</p> <p>Condition Alias Agent Group Process Pattern Service Name</p> <ul style="list-style-type: none"> =2 001 06 (Activation (Bypass Snapshot)) ... =4 ; =6 001 99 (Complete (Sub-)Workflow) ... =8 001 08 (Roll Back Change Request) ... =10 001 02 (Synchronous System Method) WS90800002 <p>Context ></p> <p>Direct Value Input</p> <p>Default Objects</p> <p>Select Expression...</p> <p>Create Expression...</p> <p>Remove Cell Value</p>

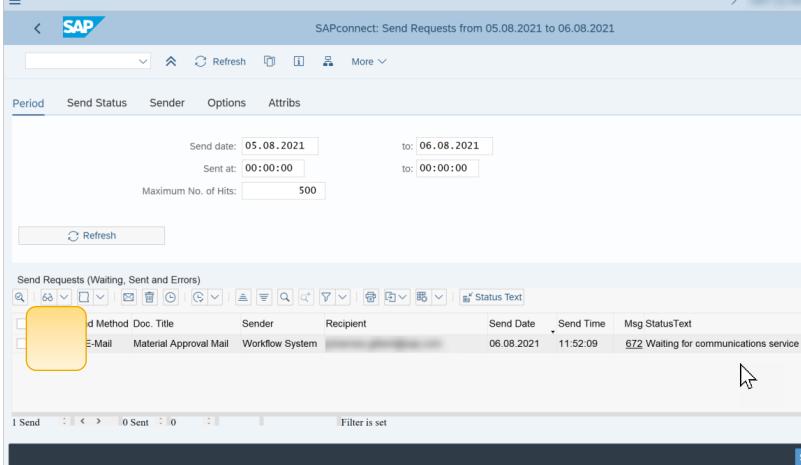
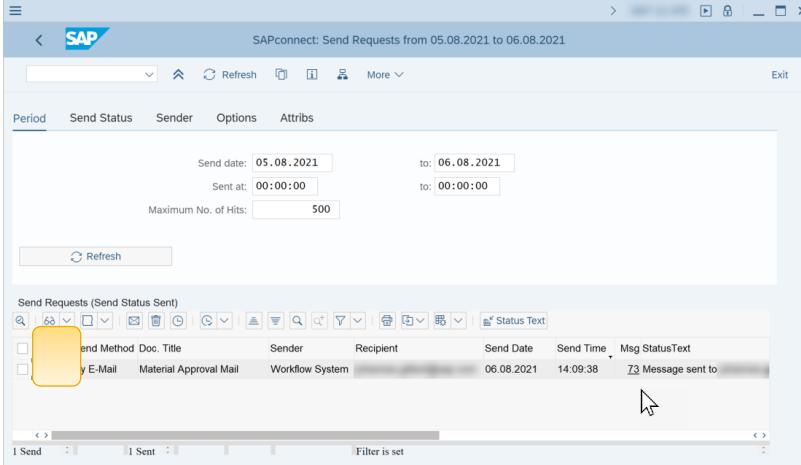
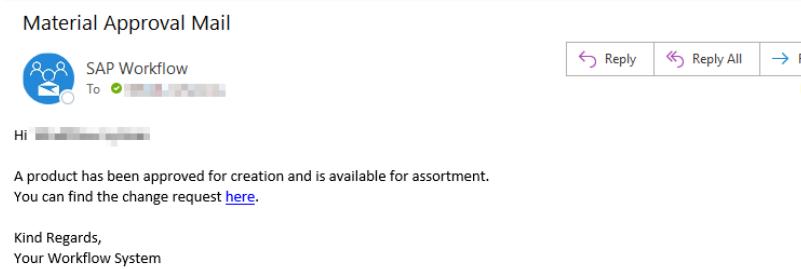
Step	Explanation	Screenshot								
13	Open the value help.	<p>Table Contents</p>								
14	Choose ZNOTIFY_REQUESTOR, the service name created in 3.1.	<p>Select: Service Name</p> <p>Value: <input type="text"/></p> <p>Text: <input type="text"/></p> <p>Items (4)</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Text</th> </tr> </thead> <tbody> <tr> <td>ZEM_SYSTEM_METHOD</td> <td>System Methode for creating follow-up Chang Request</td> </tr> <tr> <td>ZEM_SYSTEM_METHOD_COMPLIANCE</td> <td>System Methode for creating Logistics Compliance Request</td> </tr> <tr> <td>ZG_SYS_METHOD_SNAPSHOT_AND_STAGING_UPDATE</td> <td>Update Snapshot and Staging</td> </tr> </tbody> </table>	Value	Text	ZEM_SYSTEM_METHOD	System Methode for creating follow-up Chang Request	ZEM_SYSTEM_METHOD_COMPLIANCE	System Methode for creating Logistics Compliance Request	ZG_SYS_METHOD_SNAPSHOT_AND_STAGING_UPDATE	Update Snapshot and Staging
Value	Text									
ZEM_SYSTEM_METHOD	System Methode for creating follow-up Chang Request									
ZEM_SYSTEM_METHOD_COMPLIANCE	System Methode for creating Logistics Compliance Request									
ZG_SYS_METHOD_SNAPSHOT_AND_STAGING_UPDATE	Update Snapshot and Staging									
15	Press OK.	<p>Table Contents</p>								
16	Press Activate.	<p>SAP Business Rule Framework plus</p> <p>Decision Table: DT_NON_USER_AGT_GRP_ZMAT01AM</p> <p>Activate</p>								

3.5. Try-out

In this chapter the whole new configuration is tested. A new material is created via the newly created and configured change request type ZMAT01AM and an email notification is sent. To reduce the input-effort an existing material is copied.

Step	Explanation	Screenshot
1	Open the MDGUI for Material Governance. Navigate to Change Requests > Material Processing > Search Material . Search for some appropriate material (here P-100).	
2	Select the material and choose Copy .	
3	In the pop-up Available Change Request Types select the newly create one and choose OK .	

Step	Explanation	Screenshot
4	Add an appropriate description and <i>Submit</i> the change request.	
5	In case the duplicate check pops up choose <i>Continue</i> to go ahead with the current, new material.	
6	Have your change request processed to status <i>Final Check Approved</i> .	

Step	Explanation	Screenshot														
7	<p>Start transaction SOST to show the business communication services' Send Requests. Depending on the BCS settings, the email will need some time to be sent by the system. The Status is waiting, ▲).</p> <p>Nonetheless, a corresponding Send Request is already available.</p>	 <p>SAPconnect: Send Requests from 05.08.2021 to 06.08.2021</p> <p>Period Send Status Sender Options Attrbs</p> <p>Send date: 05.08.2021 to: 06.08.2021 Sent at: 00:00:00 to: 00:00:00 Maximum No. of Hits: 500</p> <p>Send Requests (Waiting, Sent and Errors)</p> <table border="1"> <thead> <tr> <th>Send Method</th> <th>Doc. Title</th> <th>Sender</th> <th>Recipient</th> <th>Send Date</th> <th>Send Time</th> <th>Msg StatusText</th> </tr> </thead> <tbody> <tr> <td>E-Mail</td> <td>Material Approval Mail</td> <td>Workflow System</td> <td>[REDACTED]</td> <td>06.08.2021</td> <td>11:52:09</td> <td>672 Waiting for communications service</td> </tr> </tbody> </table> <p>1 Send : 0 Sent : 0 Filter is set</p>	Send Method	Doc. Title	Sender	Recipient	Send Date	Send Time	Msg StatusText	E-Mail	Material Approval Mail	Workflow System	[REDACTED]	06.08.2021	11:52:09	672 Waiting for communications service
Send Method	Doc. Title	Sender	Recipient	Send Date	Send Time	Msg StatusText										
E-Mail	Material Approval Mail	Workflow System	[REDACTED]	06.08.2021	11:52:09	672 Waiting for communications service										
8	<p>Once the send request is transmitted the Status is transmitted, ■).</p>	 <p>SAPconnect: Send Requests from 05.08.2021 to 06.08.2021</p> <p>Period Send Status Sender Options Attrbs</p> <p>Send date: 05.08.2021 to: 06.08.2021 Sent at: 00:00:00 to: 00:00:00 Maximum No. of Hits: 500</p> <p>Send Requests (Send Status Sent)</p> <table border="1"> <thead> <tr> <th>Send Method</th> <th>Doc. Title</th> <th>Sender</th> <th>Recipient</th> <th>Send Date</th> <th>Send Time</th> <th>Msg StatusText</th> </tr> </thead> <tbody> <tr> <td>E-Mail</td> <td>Material Approval Mail</td> <td>Workflow System</td> <td>[REDACTED]</td> <td>06.08.2021</td> <td>14:09:38</td> <td>Z3 Message sent to</td> </tr> </tbody> </table> <p>1 Send : 1 Sent : Filter is set</p>	Send Method	Doc. Title	Sender	Recipient	Send Date	Send Time	Msg StatusText	E-Mail	Material Approval Mail	Workflow System	[REDACTED]	06.08.2021	14:09:38	Z3 Message sent to
Send Method	Doc. Title	Sender	Recipient	Send Date	Send Time	Msg StatusText										
E-Mail	Material Approval Mail	Workflow System	[REDACTED]	06.08.2021	14:09:38	Z3 Message sent to										
9	<p>Check the corresponding mailbox for the email from the workflow.</p>	 <p>Material Approval Mail</p> <p>SAP Workflow To [REDACTED]</p> <p>Hi [REDACTED]</p> <p>A product has been approved for creation and is available for assortment. You can find the change request here.</p> <p>Kind Regards, Your Workflow System</p>														

4. Email notification using a new workflow

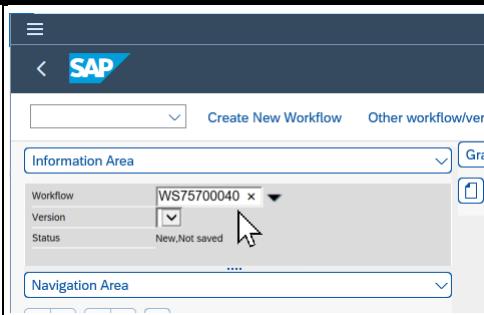
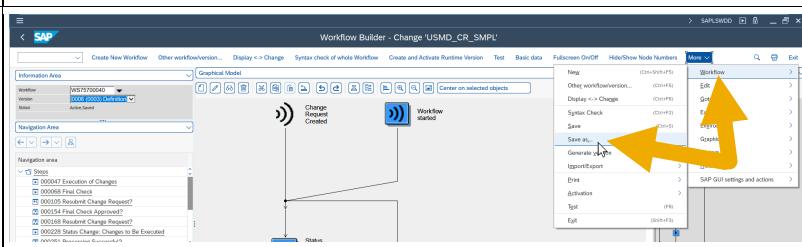
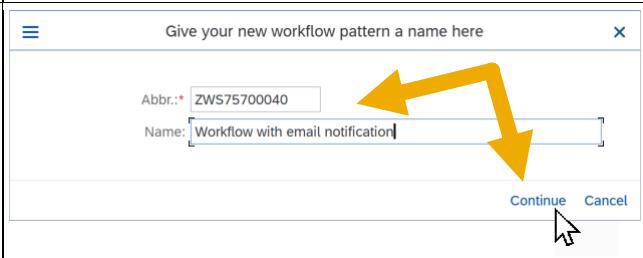
The following is a description of how to send an Email notification after activating a change request. The scenario takes workflow WS75700040 as an example. The workflow WS75700040 only contains three dialog steps.

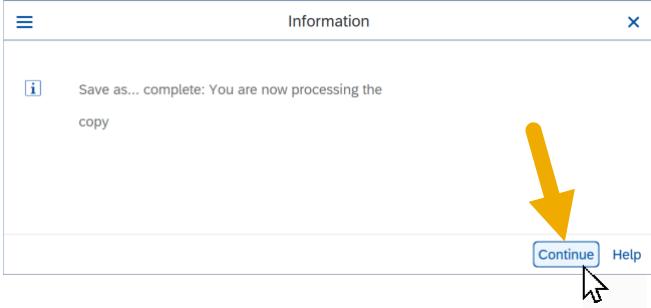
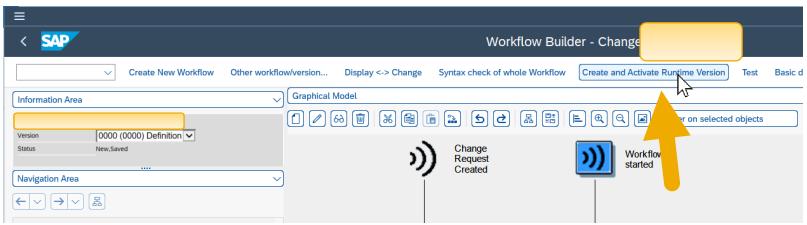
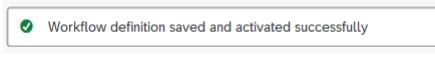
Workflow	Step	Description	Key	Validation	Next Step
WS75700040	0	Submission	-	-	1
WS75700040	1	Processing	X	-	2
WS75700040	2	Final Check	-	X	-
WS75700040	3	Revision	-	-	1

The idea is that add a program-exit to the activation step of workflow. After the activation step add a new step with type “Send Email”.

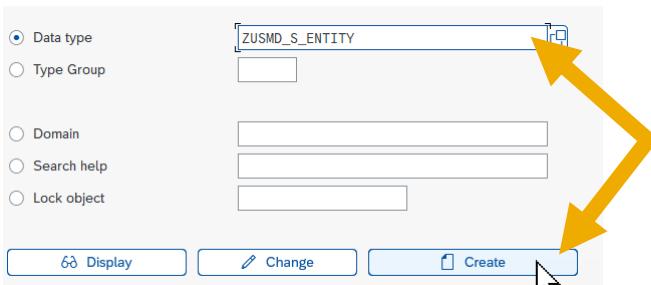
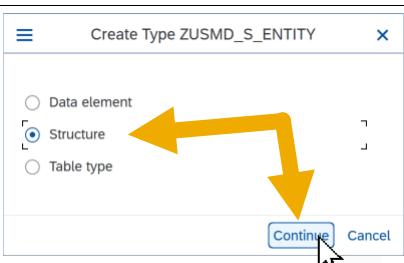
4.1. Create a new workflow

In the chapter workflow WS75700040 is copied, and the copy is activated.

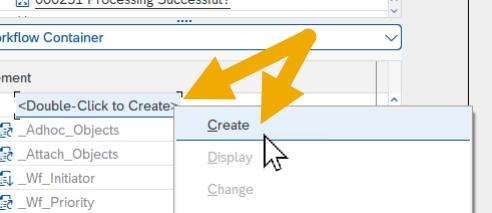
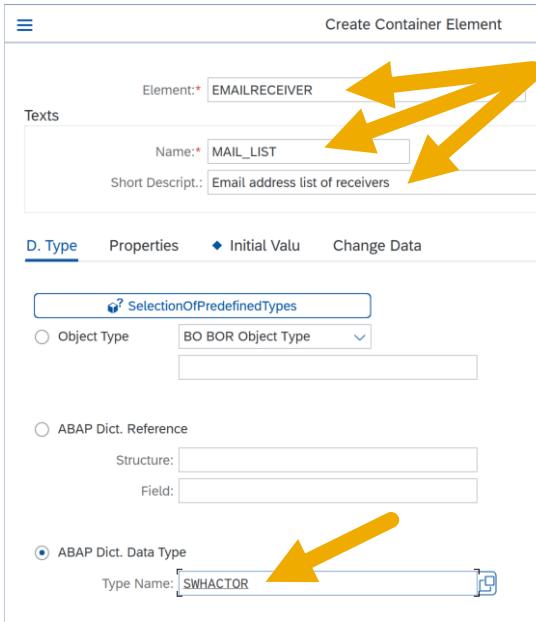
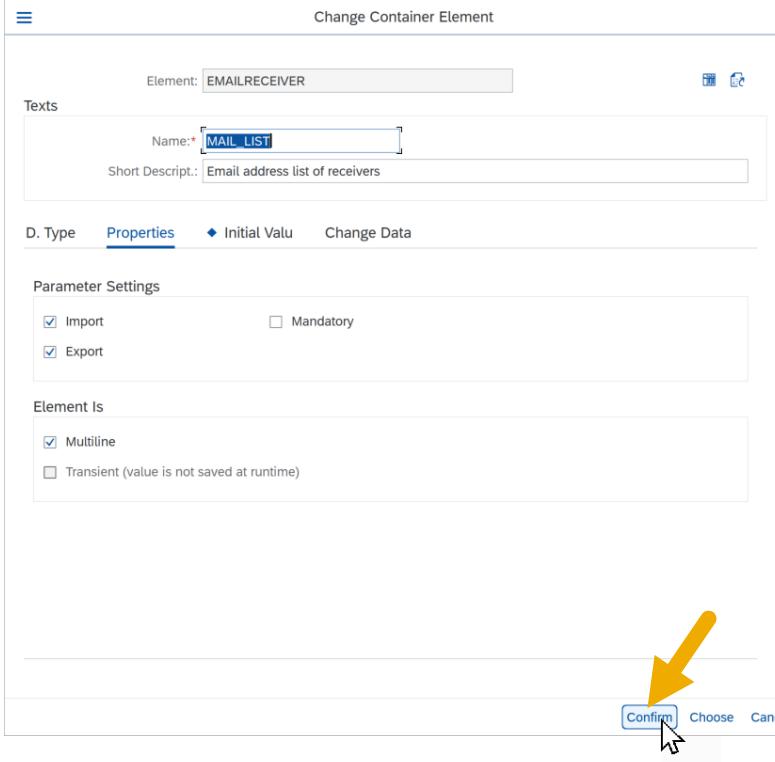
Step	Explanation	Screenshot
1	Start transaction SWBD. Enter Workflow WS75700040 and press <i>Enter</i> . In case there is a popup display confirm/ ignore it.	
2	In the menu select <i>Workflow > Save as</i>	
3	Enter an <i>Abbreviation</i> and a <i>Name</i> and press <i>Continue</i> . In the next popup chose a suitable package.	

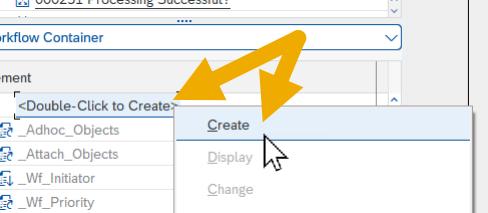
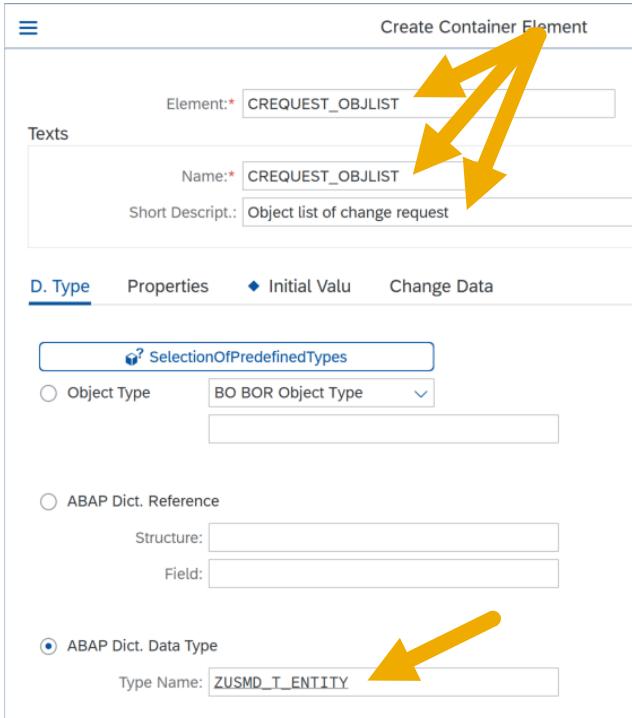
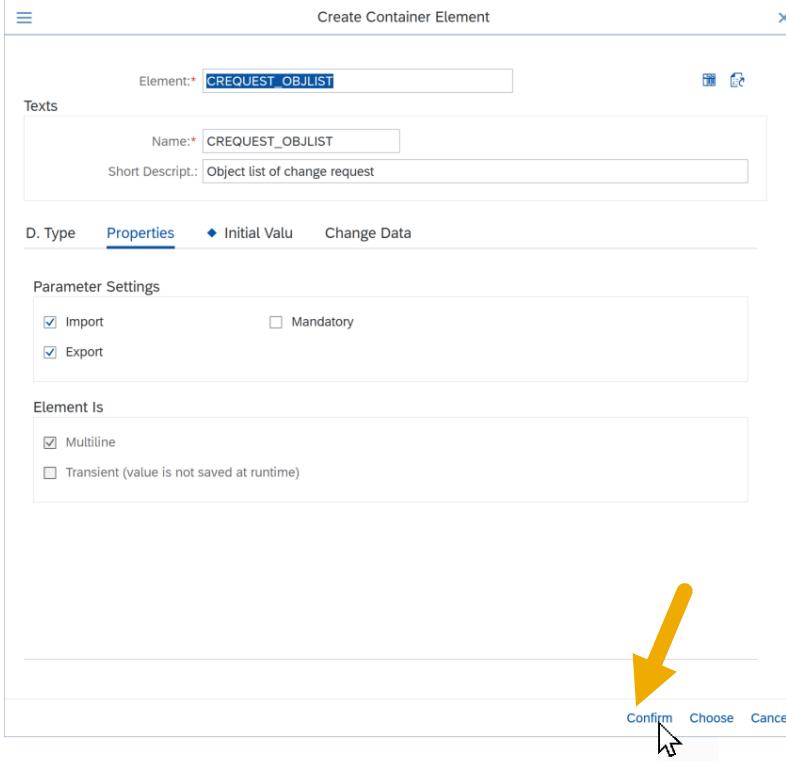
Step	Explanation	Screenshot
4	You are informed the copying is complete. Press <i>Continue</i> to confirm the popup.	
5	A new workflow number is automatically generated. Press <i>Create and Activate Runtime Version</i> .	
6	You are informed that the activation is successful.	

4.2. Modify the workflow container

Step	Explanation	Screenshot
1	Start transaction SE11. To create a new structure chose <i>Data type</i> , enter ZUSMD_S_ENTITY and press <i>Create</i> .	
2	In the popup chose <i>Structure</i> and press <i>Continue</i> .	

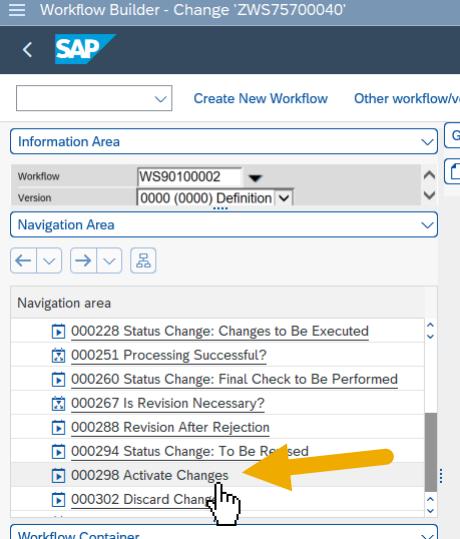
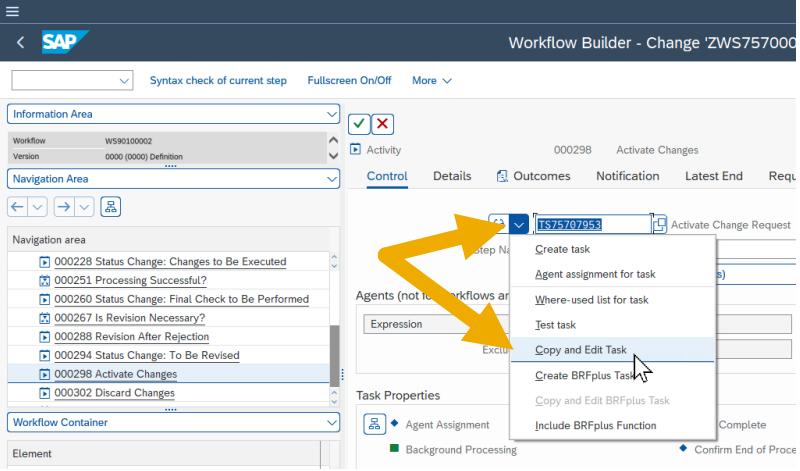
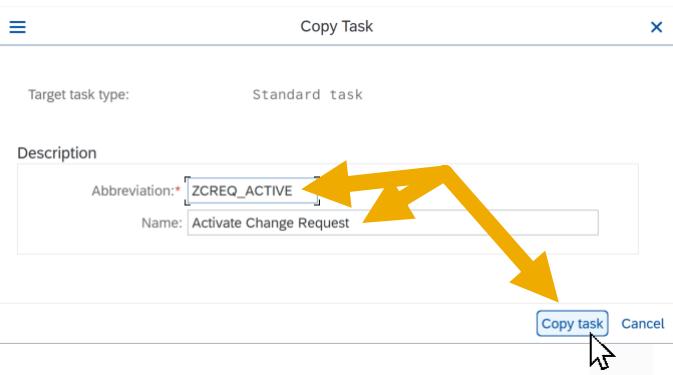
Step	Explanation	Screenshot
3	<p>Enter a <i>Short Description</i> and add the following components:</p> <ul style="list-style-type: none"> Component: ENTITY <ul style="list-style-type: none"> Typing: Types Type: USMD_ENTITY Component: VALUE <ul style="list-style-type: none"> Typing: Types Type: USMD_TXTLG <p>Press Save and Activate.</p>	
4	<p>Go back to the initial screen of SE11. Enter Data type ZUSMD_T_ENTITY and press Create.</p>	
5	<p>In the popup chose <i>Table type</i> and press <i>Continue</i>.</p>	
6	<p>Enter a short text. Chose <i>Line Type</i> and enter ZUSMD_S_ENTITY. Save and activate the table type.</p>	
7	<p>Start transaction SWDD (not SWDB!). Enter the workflow created in 4.1 if it is not automatically entered. In the Navigation Area chose the <i>Workflow Container</i>.</p>	

Step	Explanation	Screenshot
8	From the context menu chose <i>Create</i> to create a new container element.	
9	Enter EMAILRECEIVER as <i>Element</i> , MAIL_LIST as <i>Name</i> and Email address list of receivers as <i>Short Description</i> . Chose <i>ABAP Dictionary Data Type</i> and enter SWHACTOR.	
10	Select tab <i>Properties</i> . Chose <i>Import</i> , <i>Export</i> and <i>Multiline</i> . Press <i>Confirm</i> .	

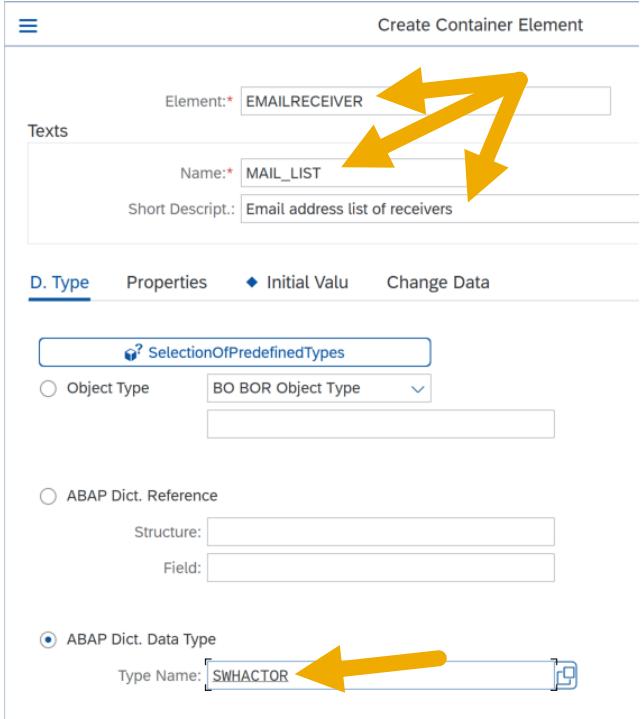
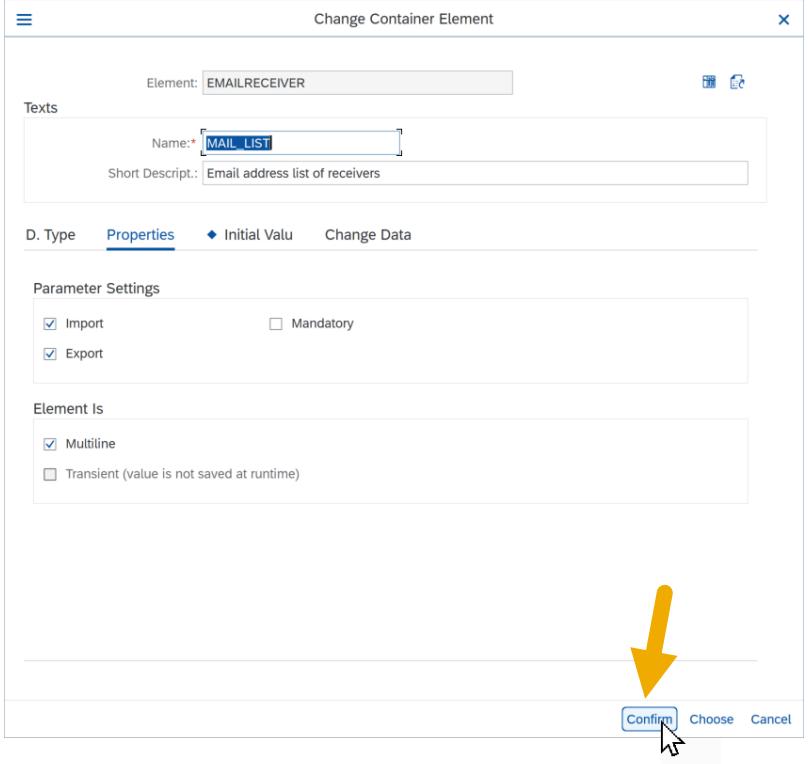
Step	Explanation	Screenshot
11	From the context menu chose <i>Create</i> to create another new container element.	
12	Enter CREQUEST_OBJLIST as <i>Element</i> , CREQUEST_OBJLIST as <i>Name</i> and Object list of change request as <i>Short Description</i> . Choose <i>ABAP Dictionary Data Type</i> and enter ZUSMD_T_ENTITY.	
13	Select tab <i>Properties</i> . Chose <i>Import</i> and <i>Export</i> . Press <i>Confirm</i> .	

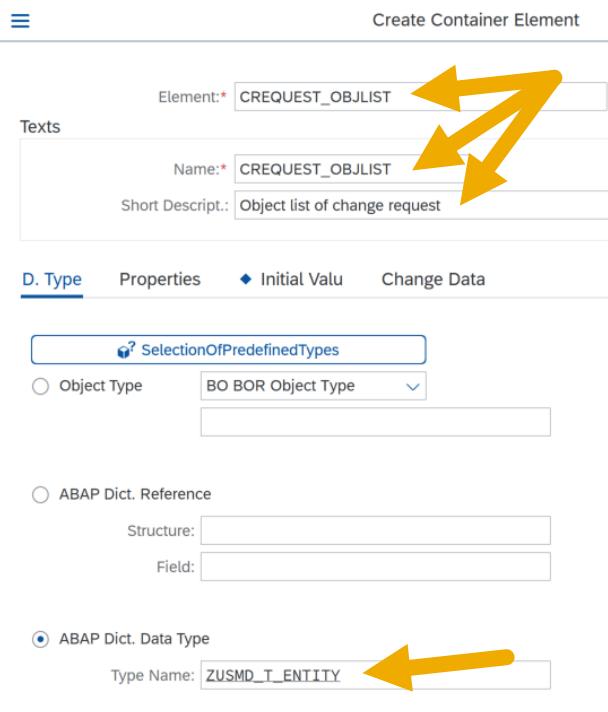
Step	Explanation	Screenshot
14	Press Save.	

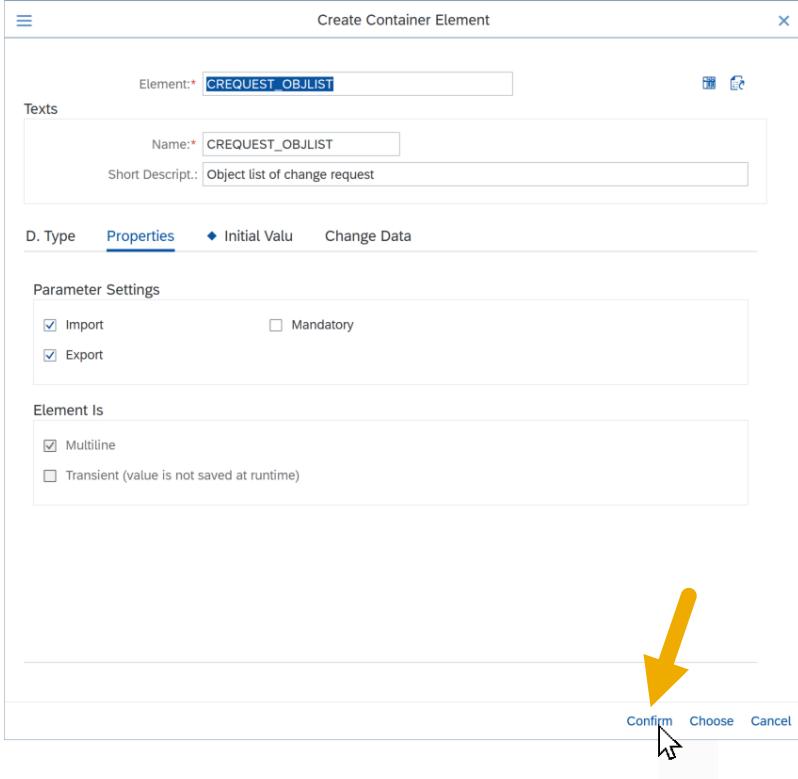
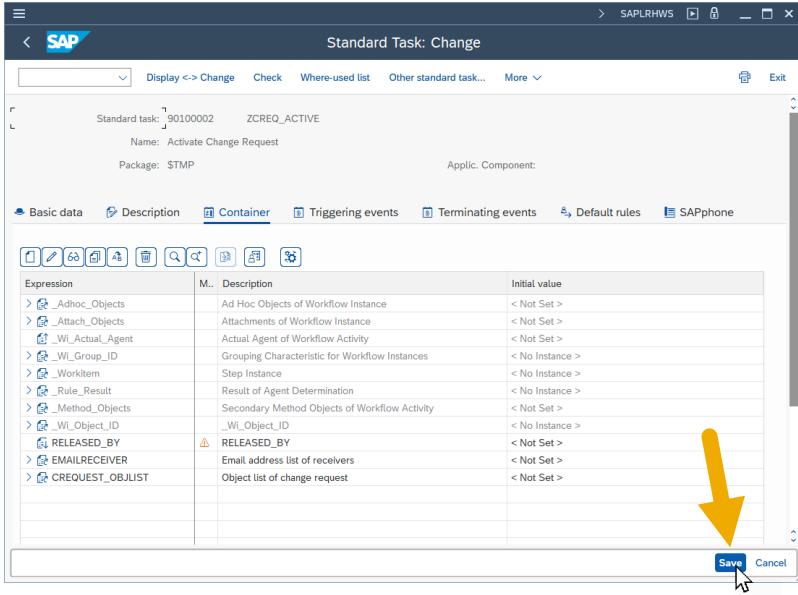
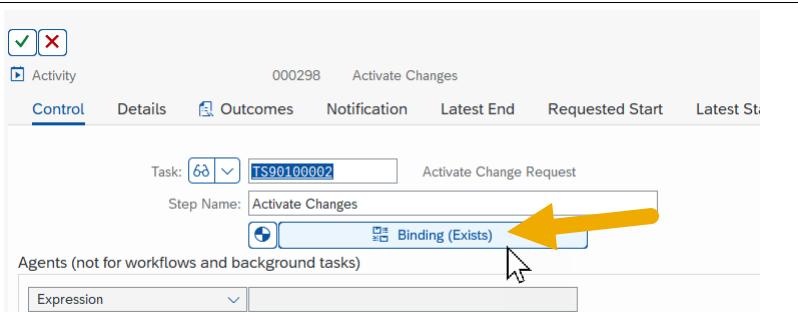
4.3. Modify the activation step

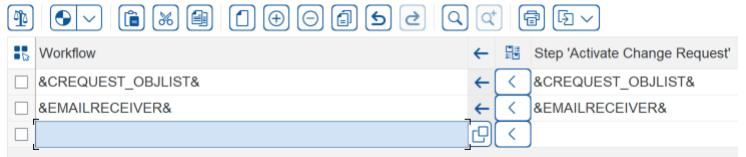
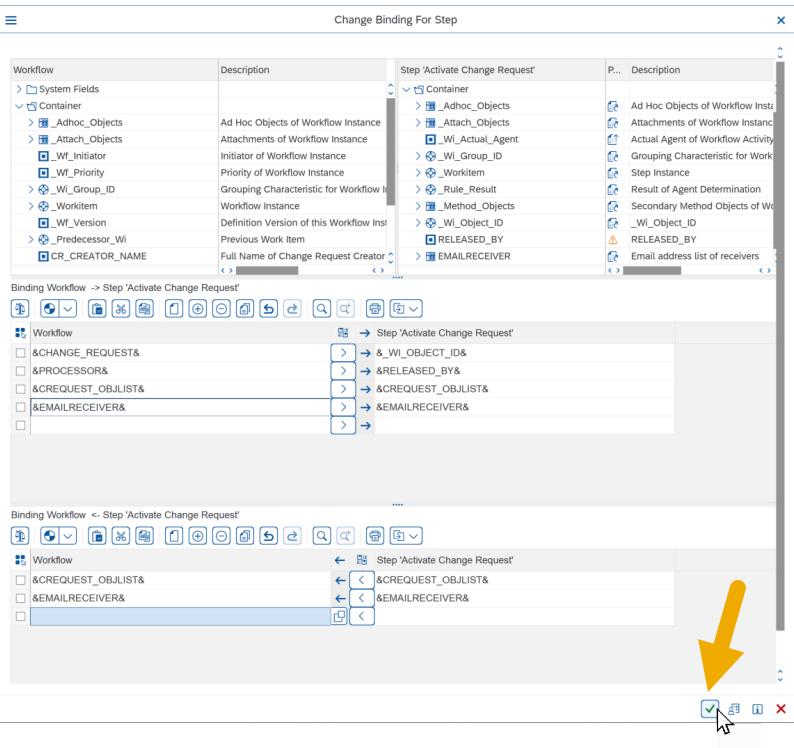
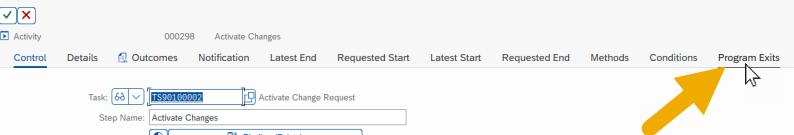
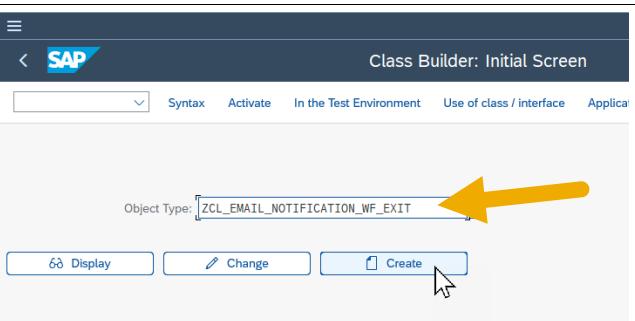
Step	Explanation	Screenshot
1	In transaction SWDD select step 000298 Activate Changes.	
2	In the content are open the task's menu and chose Copy and Edit Task.	
3	In the popup enter Abbreviation ZCREQ_ACTIVE and press Copy task. Chose a suitable package.	

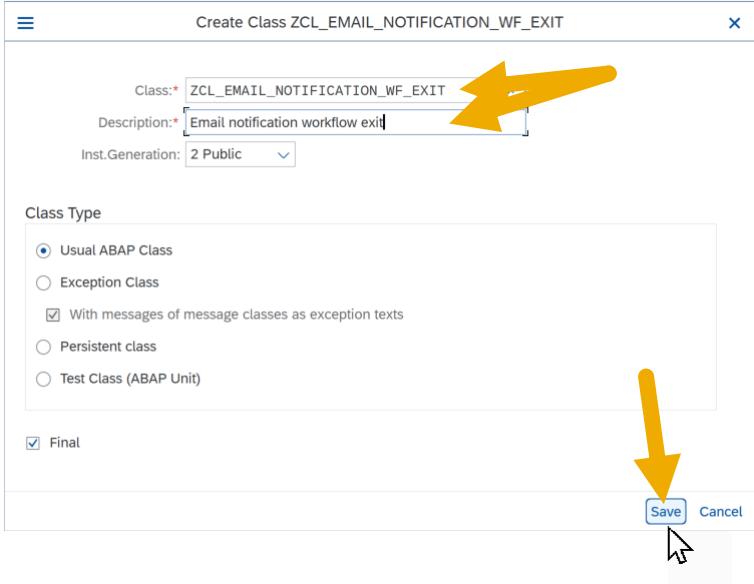
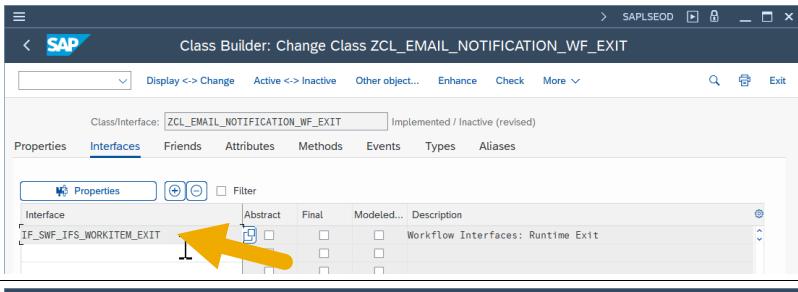
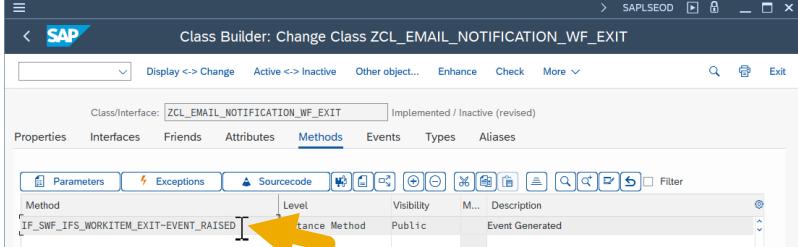
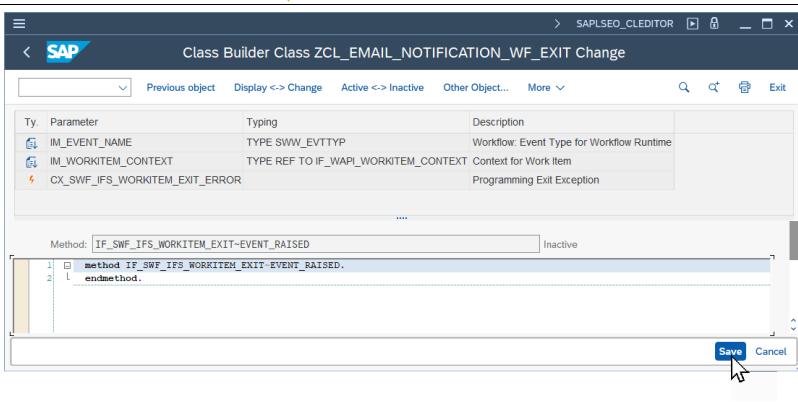
Step	Explanation	Screenshot
4	Now you are editing the copy.	
5	Navigate to tab Container. At popup Transfer missing elements from the object method press Yes.	
6	<p>You are going to create two new container elements EMAILRECEIVER and CREQUEST_OBJLIST (just like previously).</p> <p>Press <i>Create Element</i>.</p>	

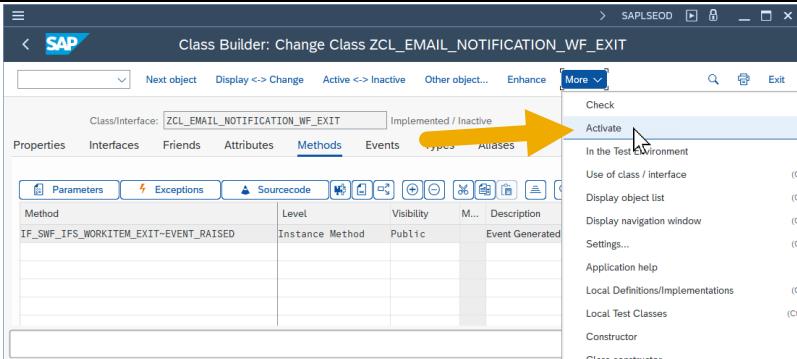
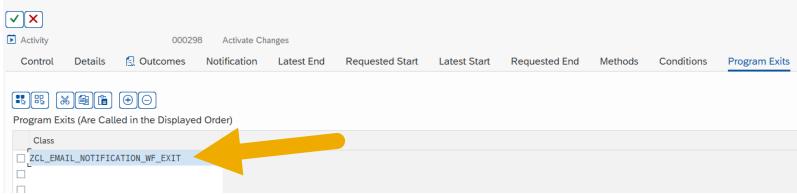
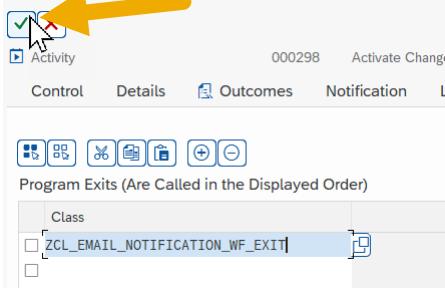
Step	Explanation	Screenshot
7	<p>Enter EMAILRECEIVER as <i>Element</i>, MAIL_LIST as <i>Name</i> and Email address list of receivers as <i>Short Description</i>. Choose ABAP Dictionary Data Type and enter SWHACTOR.</p>	
8	<p>Select tab <i>Properties</i>. Choose <i>Import</i>, <i>Export</i> and <i>Multiline</i>. Press <i>Confirm</i>.</p>	

Step	Explanation	Screenshot
9	Press <i>Create Element</i> .	<p>Standard task: 90100002 ZCREQ_ACTIVE Name: Activate Change Request Package: \$TMP</p> <p>Basic data Description Container Trigger</p> 
10	<p>Enter CREQUEST_OBJLIST as <i>Element</i>, CREQUEST_OBJLIST as <i>Name</i> and Object list of change request as <i>Short Description</i>. Choose <i>ABAP Dictionary Data Type</i> and enter ZUSMD_T_ENTITY.</p>	<p>Create Container Element</p> <p>Element: * CREQUEST_OBJLIST</p> <p>Name: * CREQUEST_OBJLIST Short Descript.: Object list of change request</p> <p>D. Type Properties Initial Value Change Data</p> <p><input type="radio"/> SelectionOfPredefinedTypes</p> <p><input type="radio"/> Object Type BO BOR Object Type</p> <p><input type="radio"/> ABAP Dict. Reference Structure: Field:</p> <p><input checked="" type="radio"/> ABAP Dict. Data Type Type Name: ZUSMD_T_ENTITY</p> 

Step	Explanation	Screenshot
11	Select tab Properties . Choose Import and Export . Press Confirm .	
12	Press Save and navigate back.	
13	Press Binding .	

Step	Explanation	Screenshot
14	In section <i>Binding Workflow</i> -> Step 'Activate Change Request' add bindings for &CREQUEST_OBJLIST& and &EMAILRECEIVER&.	
15	In section <i>Binding Workflow</i> <- Step 'Activate Change Request' add bindings for &CREQUEST_OBJLIST& and &EMAILRECEIVER&.	
16	Press <i>Exit Binding Editor</i> .	
17	Navigate to tab <i>Program Exits</i> .	
18	Open a new SAPgui window. Start transaction SE24. Enter ZCL_EMAIL_NOTIFICATION_WF_EXIT and press <i>Create</i> .	

Step	Explanation	Screenshot
19	Enter <i>Description</i> Email notification workflow exit and press <i>Save</i> . Choose a suitable package.	
20	Navigate to tab <i>Interfaces</i> and enter IF_SWF_IFS_WORKITEM_EXIT . Press <i>Enter</i> and <i>Save</i> .	
21	Navigate to tab <i>Methods</i> . Navigate to interface method EVENT_RAISED .	
22	Keep the method implementation empty and press <i>Save</i> . (The implementation will be done later.)	

Step	Explanation	Screenshot
23	Navigate back and activate the class (select all objects).	
24	Switch to the other SAPGui window. Enter class ZCL_EMAIL_NOTIFICATION_WF_EXIT. Press Save.	
25	Press press Transfer and Back.	

4.4. Implementation of IF_SWF_IFS_WORKITEM_EXIT~EVENT_RAISED

```

CLASS zcl_email_notification_wf_exit DEFINITION
  PUBLIC
  FINAL
  CREATE PUBLIC .

  PUBLIC SECTION.

    INTERFACES if_swf_ifs_workitem_exit .
  PROTECTED SECTION.
  PRIVATE SECTION.
ENDCLASS.

```

CLASS ZCL_EMAIL_NOTIFICATION_WF_EXIT IMPLEMENTATION.

```

METHOD if_swf_ifs_workitem_exit~event_raised.
  DATA: lo_workitem      TYPE REF TO if_swf_ifs_parameter_container,
        lt_objlst       TYPE usmd_ts_crequest_objkey,
        lv_return        TYPE sysubrc,
        ls_swotobjid    TYPE swotobjid,
        lv_crequest      TYPE usmd_crequest,
        lv_edition       TYPE usmd_edition,
        lv_edtn_type    TYPE usmd_edition_type,
        lv_model         TYPE usmd_model,
        lo_model         TYPE REF TO if_usmd_model,
        lt_message       TYPE usmd_t_message,

```

```

lt_message1      TYPE usmd_t_message,
lt_objlst_ele   TYPE zusmd_t_entity,
lr_data          TYPE REF TO data,
ls_objlst        TYPE usmd_s_crequest_objkey,
ls_objlst_ele   TYPE zusmd_s_entity,
lv_obj           TYPE string,
ls_entity_prop  TYPE usmd_s_entity_prop,
ls_fdep          TYPE usmd_s_fdep,
ld_entity         TYPE usmd_entity,
lv_entity_txt   TYPE usmd_value,
lv_field_txt    TYPE usmd_value,
ls_receiver      TYPE swhactor,
lt_receiver_temp TYPE tswhactor,
lt_receiver     TYPE tswhactor,
lr_badi_wf_agent TYPE REF TO usmd_wf_agent,
ld_agent_filter  TYPE swcont-value,
lo_table          TYPE REF TO cl_abap_tabledescr,
lo_struct         TYPE REF TO cl_abap_structdescr,
lo_line           TYPE REF TO cl_abap_datadescr,
ls_comp            TYPE abap_compdescr,
lo_type           TYPE REF TO cl_abap_typedescr.

FIELD-SYMBOLS: <f_obj>      TYPE any,
                <f_value>    TYPE any,
                <f_struct>   TYPE any,
                <f_obj_tab>  TYPE ANY TABLE.

* The email is sent only after the activation of master data
CHECK im_event_name EQ if_swf_ifs_workitem_exit=>c_evtyp_after_execution.

*Get instance of workflow item
lo_workitem = im_workitem_context->get_wi_container( ).

* Get Change request number
TRY.
  CALL METHOD lo_workitem->get
    EXPORTING
      name = '_WI_OBJECT_ID'
    IMPORTING
      value = ls_swotobjid.
  lv_crequest = ls_swotobjid-objkey+0(12).
  CATCH cx_swf_cnt_elem_not_found .
  CATCH cx_swf_cnt_elem_type_conflict .
  CATCH cx_swf_cnt_unit_type_conflict .
  CATCH cx_swf_cnt_container .
ENDTRY.

* Get object list of the change request
* Get data model
  SELECT SINGLE usmd_edition FROM usmd120c INTO lv_edition WHERE usmd_crequest = lv_crequest.
  IF sy-subrc EQ 0.
    SELECT SINGLE usmd_edtn_type FROM usmd020c INTO lv_edtn_type WHERE usmd_edition = lv_edition.
    IF sy-subrc EQ 0.
      SELECT SINGLE usmd_model FROM usmd010c INTO lv_model WHERE usmd_edtn_type = lv_edtn_type.
    ENDIF.
  ENDIF.

  IF lv_model IS NOT INITIAL.
    CALL METHOD cl_usmd_model=>get_instance
      EXPORTING
        i_usmd_model = lv_model
      IMPORTING
        eo_instance  = lo_model
        et_message   = lt_message.
  ENDIF.

  IF lt_message IS INITIAL.
    CALL METHOD lo_model->if_usmd_model_crequest~request_get_objlist
      EXPORTING
        i_crequest      = lv_crequest
      IMPORTING
        ets_crequest_objkey = lt_objlst
        et_message     = lt_message1.
  ENDIF.

```

```

ENDIF.
APPEND LINES OF lt_message1 TO lt_message.

LOOP AT lt_objlst INTO ls_objlst.
  CLEAR ls_objlst_ele.
  ls_objlst_ele-entity = ls_objlst-fieldname.
  ASSIGN ls_objlst-r_data->*> TO <f_obj>.
  lo_type = cl_abap_typedescr->describe_by_data( <f_obj> ).
  CHECK lo_type->kind = cl_abap_typedescr->kind_table. lo_table ?= lo_type.
  lo_line = lo_table->get_table_line_type( ). ASSIGN <f_obj> TO <f_obj_tab>.
  IF lo_line->kind = cl_abap_typedescr->kind_struct.
    lo_struct ?= lo_line.
  IF lo_line->kind = cl_abap_typedescr->kind_struct.
    lo_struct ?= lo_line.
    IF <f_obj_tab> IS ASSIGNED.
      LOOP AT <f_obj_tab> ASSIGNING <f_struct>.
        CLEAR: lv_obj, ld_entity.
      * Get text of entity type
        CALL METHOD cl_usmd_services->fieldname2entity
          EXPORTING
            io_model = lo_model
            i_fieldname = ls_objlst-fieldname
          RECEIVING
            r_entity = ld_entity.

        CALL METHOD cl_usmd_services->entity2text
          EXPORTING
            io_model = lo_model
            i_entity = ld_entity
          RECEIVING
            r_text = lv_entity_txt.
        CONCATENATE lv_entity_txt ';' INTO lv_obj SEPARATED BY space.

      * Get master key of ld_entity
        READ TABLE lo_model->dt_entity_prop INTO ls_entity_prop WITH KEY usmd_entity =
ld_entity.

        LOOP AT lo_model->dt_fdep INTO ls_fdep WHERE fieldname EQ ls_entity_prop-r_fprop-
>fieldname.
          IF sy-subrc EQ 0.
            ASSIGN COMPONENT ls_fdep-masterfield OF STRUCTURE <f_struct> TO <f_value>.
            ASSERT <f_value> IS ASSIGNED.
          ENDIF.
        * Get text of field name
          CALL METHOD cl_usmd_services->fieldname2text
            EXPORTING
              io_model = lo_model
              i_fieldname = ls_fdep-masterfield
            RECEIVING
              r_value = lv_field_txt.
            CONCATENATE lv_obj lv_field_txt ' ' <f_value> ' ' INTO
lv_obj SEPARATED BY space .
          ENDLOOP.
          ls_objlst_ele-value = lv_obj.
          APPEND ls_objlst_ele TO lt_objlst_ele.
        ENDLOOP.
      ENDIF.
    ENDIF.
  ENDLOOP.
TRY.
  CALL METHOD lo_workitem->set
    EXPORTING
      name = 'CREQUEST_OBJLST'
      value = lt_objlst_ele.
CATCH cx_swf_cnt_elem_not_found .
CATCH cx_swf_cnt_elem_type_conflict .
CATCH cx_swf_cnt_unit_type_conflict :

```

```

CATCH cx_swf_cnt_container .
ENDTRY.

*****
*Set parameter EMAILRECEIVER
** Call BAdI to get agents
  IF ld_agent_filter IS INITIAL.
    ld_agent_filter = if_usmd_wf_agent->gc_filter-standard.
  ENDIF.

  GET BADI lr_badi_wf_agent
  FILTERS
    agent_filter = ld_agent_filter.
  IF lr_badi_wf_agent IS BOUND.
    CALL BADI lr_badi_wf_agent->get_agents
    EXPORTING
      id_appstep      = '1'
      id_crequest     = lv_crequest
      if_single_proc  =
      id_index        = 0
    IMPORTING
      et_agent        = lt_receiver_temp
      et_message      = lt_message.
  ENDIF.
  APPEND LINES OF lt_receiver_temp TO lt_receiver.
  IF lr_badi_wf_agent IS BOUND.
    CALL BADI lr_badi_wf_agent->get_agents
    EXPORTING
      id_appstep      = '2'
      id_crequest     = lv_crequest
      if_single_proc  =
      id_index        = 0
    IMPORTING
      et_agent        = lt_receiver_temp
      et_message      = lt_message.
  ENDIF.
  APPEND LINES OF lt_receiver_temp TO lt_receiver.
  IF lr_badi_wf_agent IS BOUND.
    CALL BADI lr_badi_wf_agent->get_agents
    EXPORTING
      id_appstep      = '3'
      id_crequest     = lv_crequest
      if_single_proc  =
      id_index        = 0
    IMPORTING
      et_agent        = lt_receiver_temp
      et_message      = lt_message.
  ENDIF.
  APPEND LINES OF lt_receiver_temp TO lt_receiver.
  INSERT VALUE #( otype = 'US' OBJID = 'GILBERTJ' ) INTO TABLE lt_receiver.

TRY.
  CALL METHOD lo_workitem->set
  EXPORTING
    name  = 'EMAILRECEIVER'
    value = lt_receiver.
  CATCH cx_swf_cnt_elem_not_found .
  CATCH cx_swf_cnt_elem_type_conflict .
  CATCH cx_swf_cnt_unit_type_conflict .
  CATCH cx_swf_cnt_container .

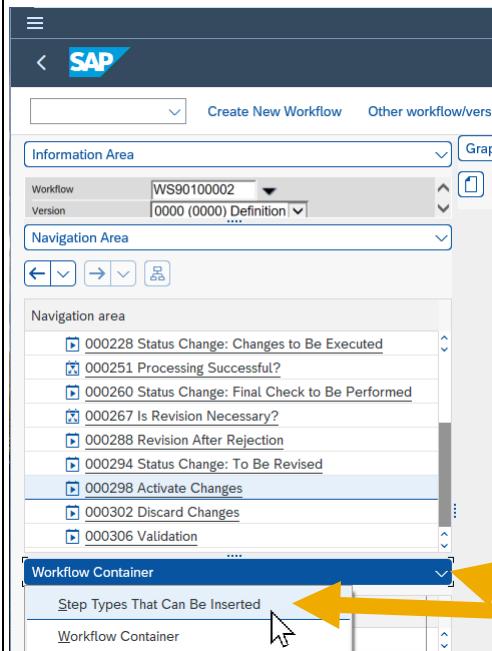
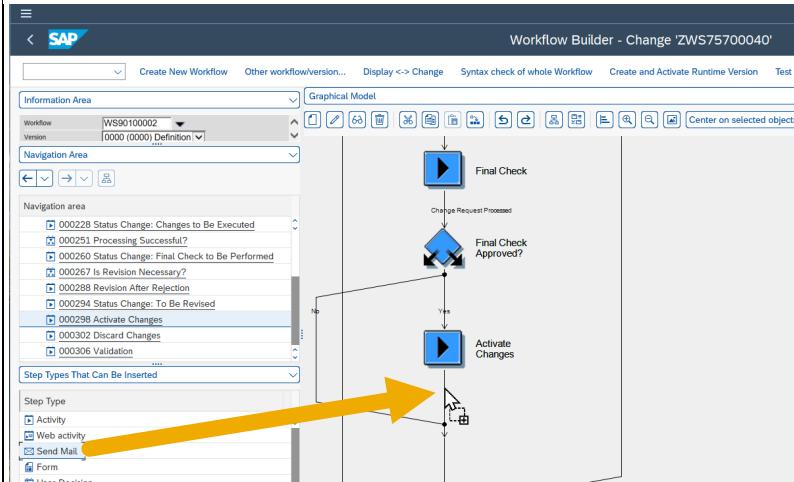
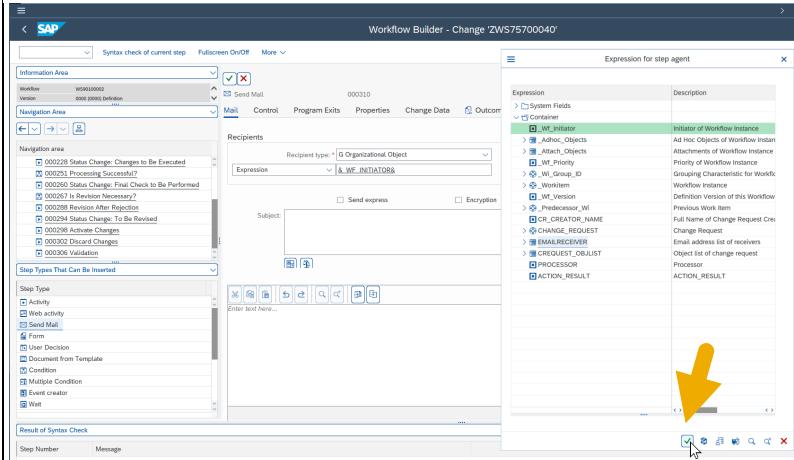
ENDTRY.

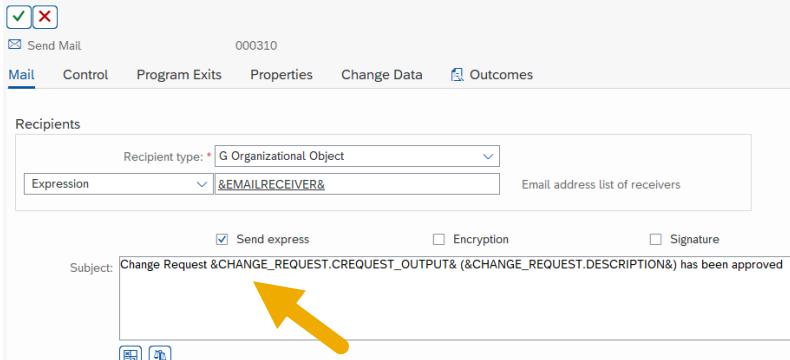
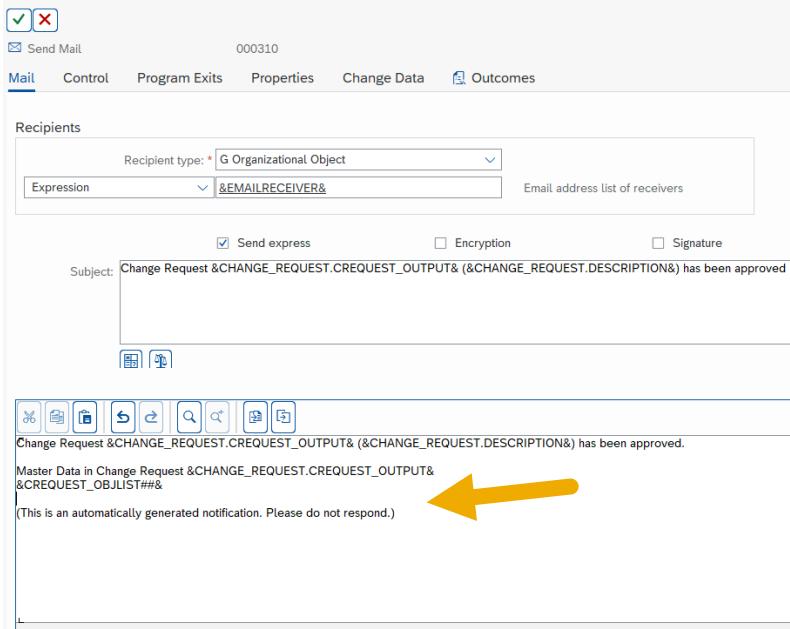
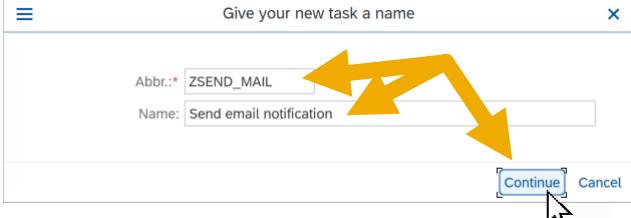
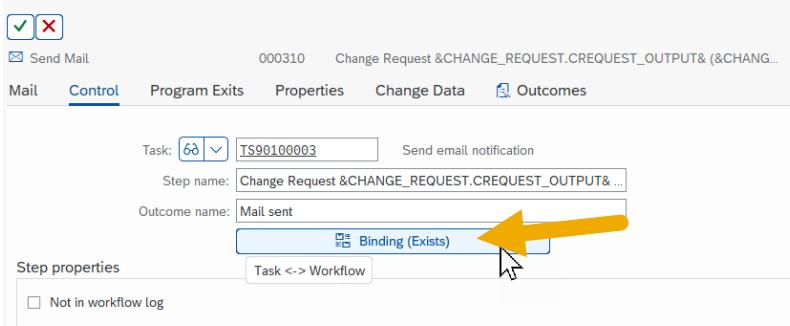
IF lv_return IS INITIAL. ENDIF.

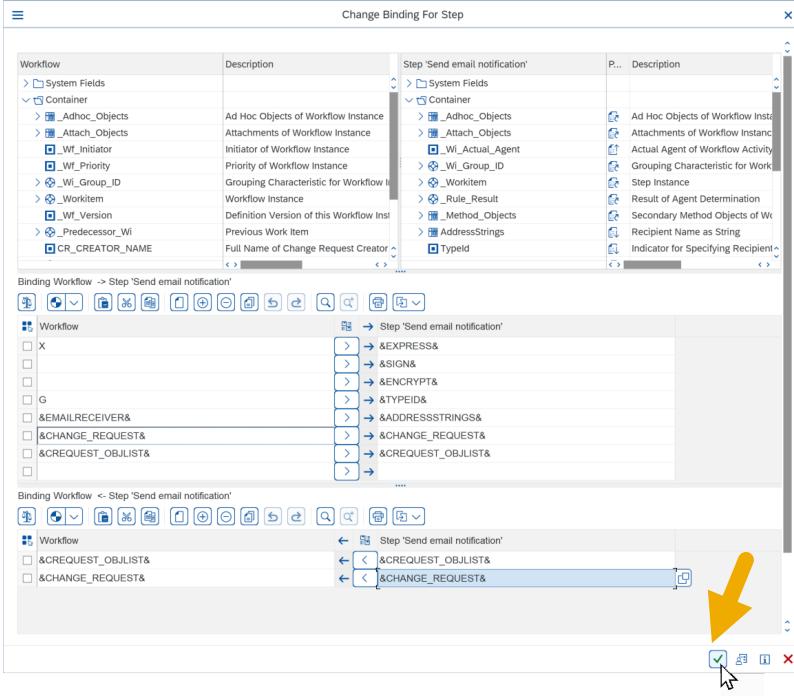
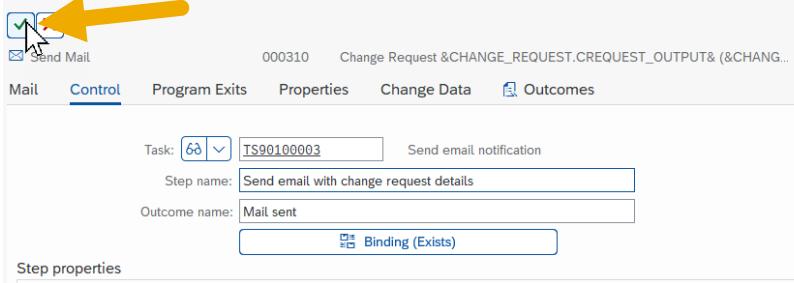
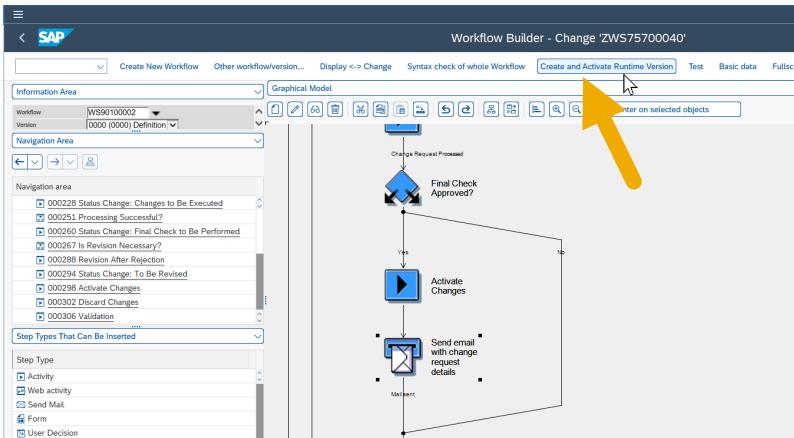
ENDMETHOD.
ENDCLASS.

```

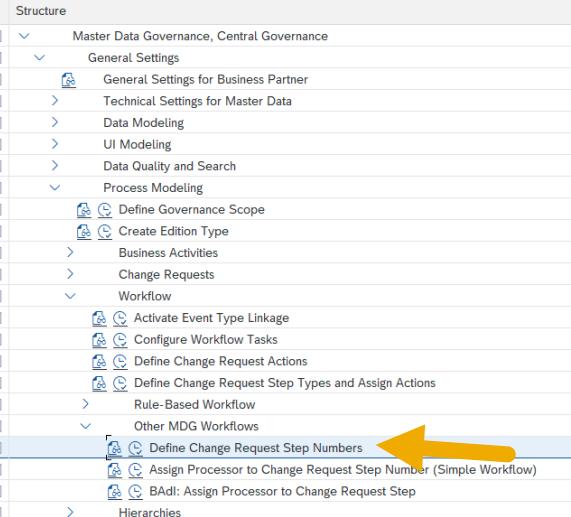
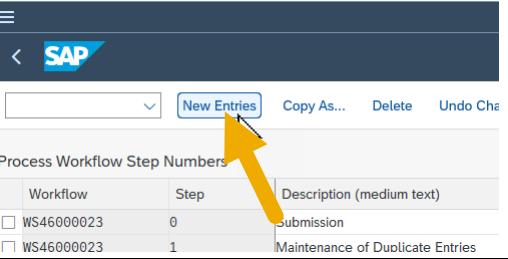
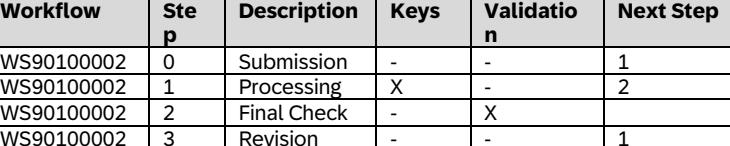
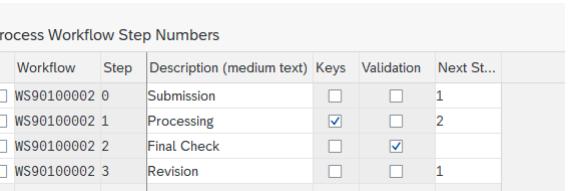
4.5. Add an email notification

Step	Explanation	Screenshot
1	Start transaction SWDD. Enter the workflow created in 4.1 if it is not automatically entered. In the Navigation Area chose the Step Types That Can Be Inserted.	
2	Drag step type Send Mail and drop it after step Activate Changes.	
3	Open the value help for Expression. Select EMAILRECEIVER and press Continue.	

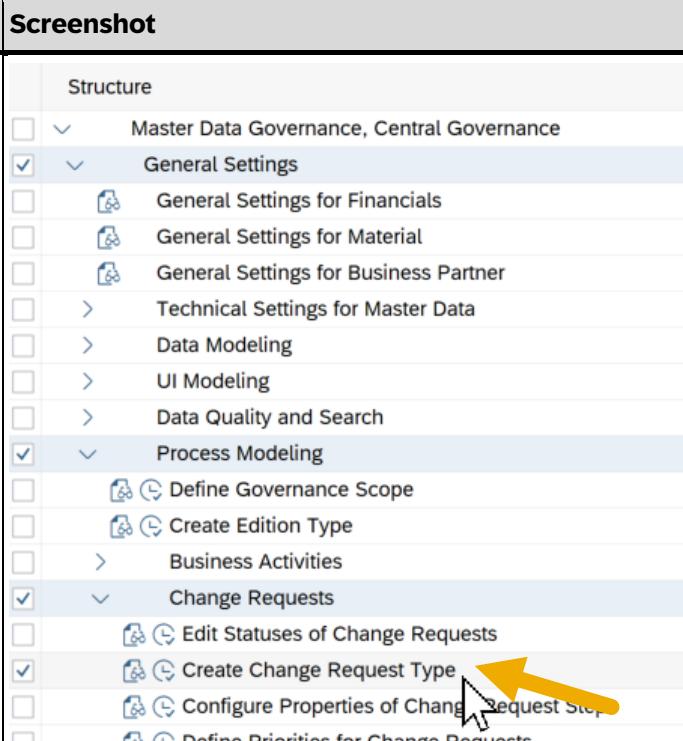
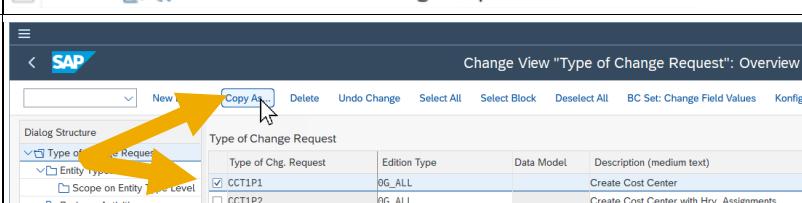
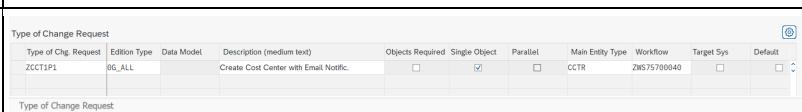
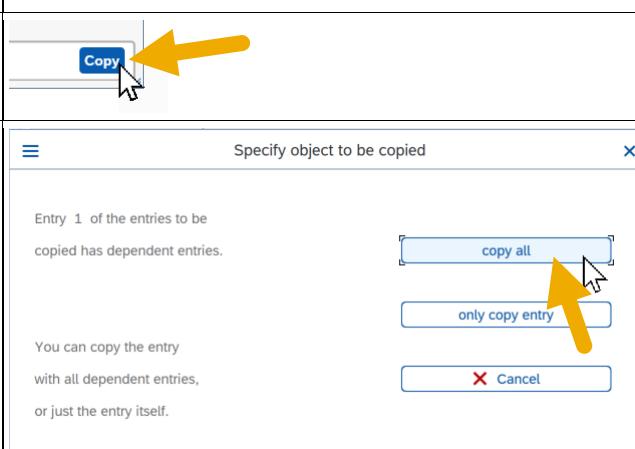
Step	Explanation	Screenshot
4	<p>Select <i>Send express</i>. For <i>Subject</i> enter the following text:</p> <p><i>Change Request &CHANGE_REQUEST.CREQUEST_OUTPUT& (&CHANGE_REQUEST.DESCRIPTION&) has been approved.</i></p>	
5	<p>Enter the following text into the text area at the bottom:</p> <p><i>Change Request &CHANGE_REQUEST.CREQUEST_OUTPUT& (&CHANGE_REQUEST.DESCRIPTION&) has been approved.</i></p> <p><i>Master Data in Change Request &CHANGE_REQUEST.CREQUEST_OUTPUT& &CREQUEST_OBJLIST##&</i></p> <p><i>(This is an automatically generated notification. Please do not respond.)</i></p>	
6	<p>Navigate to tab <i>Control</i>. In the popup enter <i>Abbreviation ZSEND_MAIL</i> and <i>Name</i> <i>Send email notification</i>. Press <i>Continue</i>. Choose a suitable package. Navigate back.</p>	
7	<p>Press <i>Binding</i>.</p>	

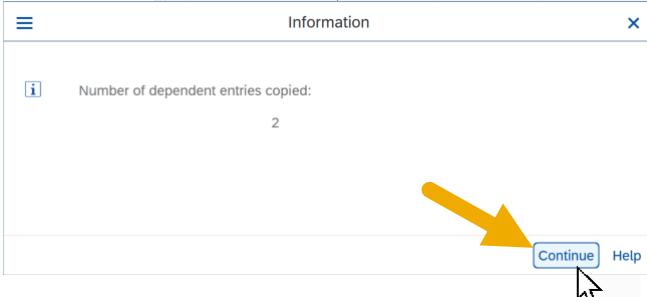
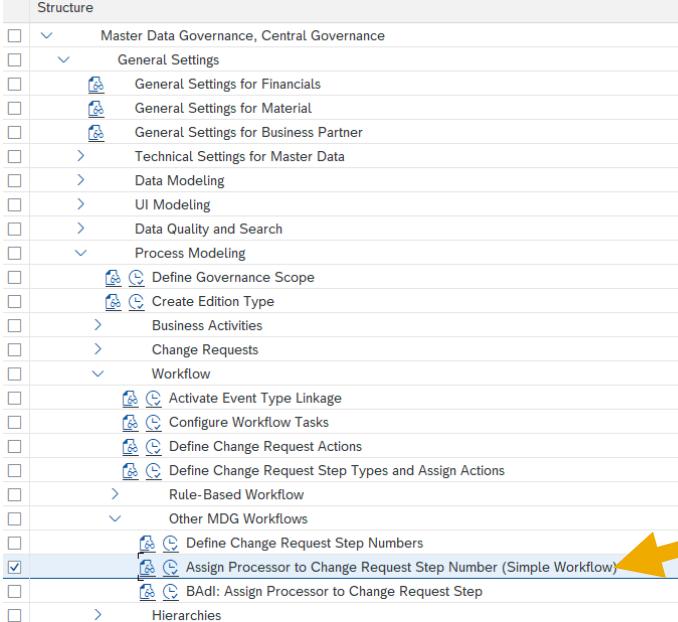
Step	Explanation	Screenshot
8	<p>Make sure the binding is as depicted on the right.</p> <p>The availability of the parameters SIGN and ENCRYPT depend on the setup of the workflow and might not be available.</p> <p>Press <i>Exit Binding Editor</i>.</p>	
9	<p>Change the step name to Send email with change request details. Press Transfer and back.</p>	
10	<p>Press Create and Activate Runtime Version.</p>	

4.6. Define change request step numbers

Step	Explanation	Screenshot																																																												
1	Start transaction MDGIMG. Navigate to <i>General Settings</i> > <i>Process Modeling</i> > <i>Workflow</i> > <i>Other MDG Workflows</i> > <i>Define Workflow Step Numbers</i> . Start this Customizing activity.																																																													
2	Press <i>New Entries</i> .																																																													
	Enter the entries from the right.	 <table border="1"> <thead> <tr> <th>Workflow</th> <th>Step</th> <th>Description</th> <th>Keys</th> <th>Validation</th> <th>Next Step</th> </tr> </thead> <tbody> <tr> <td>WS90100002</td> <td>0</td> <td>Submission</td> <td>-</td> <td>-</td> <td>1</td> </tr> <tr> <td>WS90100002</td> <td>1</td> <td>Processing</td> <td>X</td> <td>-</td> <td>2</td> </tr> <tr> <td>WS90100002</td> <td>2</td> <td>Final Check</td> <td>-</td> <td>X</td> <td></td> </tr> <tr> <td>WS90100002</td> <td>3</td> <td>Revision</td> <td>-</td> <td>-</td> <td>1</td> </tr> </tbody> </table>  <table border="1"> <thead> <tr> <th>Workflow</th> <th>Step</th> <th>Description (medium text)</th> <th>Keys</th> <th>Validation</th> <th>Next Step</th> </tr> </thead> <tbody> <tr> <td>WS90100002</td> <td>0</td> <td>Submission</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>1</td> </tr> <tr> <td>WS90100002</td> <td>1</td> <td>Processing</td> <td><input checked="" type="checkbox"/></td> <td><input type="checkbox"/></td> <td>2</td> </tr> <tr> <td>WS90100002</td> <td>2</td> <td>Final Check</td> <td><input type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td></td> </tr> <tr> <td>WS90100002</td> <td>3</td> <td>Revision</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td>1</td> </tr> </tbody> </table>	Workflow	Step	Description	Keys	Validation	Next Step	WS90100002	0	Submission	-	-	1	WS90100002	1	Processing	X	-	2	WS90100002	2	Final Check	-	X		WS90100002	3	Revision	-	-	1	Workflow	Step	Description (medium text)	Keys	Validation	Next Step	WS90100002	0	Submission	<input type="checkbox"/>	<input type="checkbox"/>	1	WS90100002	1	Processing	<input checked="" type="checkbox"/>	<input type="checkbox"/>	2	WS90100002	2	Final Check	<input type="checkbox"/>	<input checked="" type="checkbox"/>		WS90100002	3	Revision	<input type="checkbox"/>	<input type="checkbox"/>	1
Workflow	Step	Description	Keys	Validation	Next Step																																																									
WS90100002	0	Submission	-	-	1																																																									
WS90100002	1	Processing	X	-	2																																																									
WS90100002	2	Final Check	-	X																																																										
WS90100002	3	Revision	-	-	1																																																									
Workflow	Step	Description (medium text)	Keys	Validation	Next Step																																																									
WS90100002	0	Submission	<input type="checkbox"/>	<input type="checkbox"/>	1																																																									
WS90100002	1	Processing	<input checked="" type="checkbox"/>	<input type="checkbox"/>	2																																																									
WS90100002	2	Final Check	<input type="checkbox"/>	<input checked="" type="checkbox"/>																																																										
WS90100002	3	Revision	<input type="checkbox"/>	<input type="checkbox"/>	1																																																									
	Press <i>Save</i> .																																																													
	You might get a warning message <i>Select a key from the allowed namespace</i> . Ignore it by pressing <i>Enter</i> .																																																													

4.7. Create a new change request type and configure change request steps

Step	Explanation	Screenshot
1	Start transaction MDGIMG. Navigate to <i>General Settings</i> > <i>Process Modeling</i> > <i>Change Requests</i> > <i>Create Change Request Type</i> .	 A screenshot of the SAP navigation tree. The path selected is: Master Data Governance, Central Governance > General Settings > Process Modeling > Change Requests > Create Change Request Type. A yellow arrow points to the 'Create Change Request Type' option.
2	Select change request type CCT1P1 (Create Cost Center) and press Copy As.	 A screenshot of the SAP Change View 'Type of Change Request': Overview screen. It shows a table with two entries: CCT1P1 (selected) and CCT1P2. A yellow arrow points to the 'Copy As' button in the toolbar above the table.
3	Enter the following values: <ul style="list-style-type: none">Type of Change Request: ZCCT1P1Description: Create Cost Center with Email Notific.Workflow: WS90100002	 A screenshot of the Type of Change Request configuration screen. It shows two tabs: 'Type of Chg. Request' and 'Type of Change Request'. The first tab has values: Type of Chg. Request ZCCT1P1, Edition Type OG_ALL, Data Model Create Cost Center with Email Notific. The second tab has values: Objects Required Single Object Parallel Main Entity Type CCTR Workflow WS90100002. A yellow arrow points to the 'Copy' button.
4	Press Copy.	 A screenshot of the 'Specify object to be copied' pop-up dialog. It contains two buttons: 'copy all' (highlighted with a yellow arrow) and 'only copy entry'.
5	In pop-up <i>Specify object to be copied</i> to be copied press copy all.	

Step	Explanation	Screenshot																																																													
6	Acknowledge the information on <i>Number of dependent entries copied</i> by choosing <i>Continue</i> .																																																														
7	Press Save.																																																														
	<p>Navigate back to the IMG. Navigate to General Settings > Process Modeling > Workflow > Other MDG Workflows > Assign processor to Workflow Step Number (Simple Workflow). Start this</p>																																																														
	Press <i>New Entries</i> . Enter the data given on the right. Press <i>Apply</i> .	<table border="1" data-bbox="609 1331 1319 1477"> <thead> <tr> <th>Type of Chg. Request</th> <th>Step</th> <th>Description</th> <th>Ob</th> <th>Agent ID</th> </tr> </thead> <tbody> <tr> <td>ZCCT1P1</td> <td>0</td> <td>Submission</td> <td>O</td> <td>50017103</td> </tr> <tr> <td>ZCCT1P1</td> <td>1</td> <td>Processing</td> <td>O</td> <td>50017103</td> </tr> <tr> <td>ZCCT1P1</td> <td>2</td> <td>Final Check</td> <td>O</td> <td>50017103</td> </tr> <tr> <td>ZCCT1P1</td> <td>3</td> <td>Revision</td> <td>O</td> <td>50017103</td> </tr> </tbody> </table> <table border="1" data-bbox="609 1522 1367 1724"> <thead> <tr> <th colspan="6">Assignment of Processors to Workflow Step Number</th> </tr> <tr> <th>Type of Chg. Request</th> <th>Step</th> <th>Description (medium)</th> <th>Ob</th> <th>Agent ID</th> <th>Full Name</th> </tr> </thead> <tbody> <tr> <td>ZCCT1P1</td> <td>0</td> <td>Submission</td> <td>O</td> <td>50017103</td> <td>MDG Organizational Unit for Financials</td> </tr> <tr> <td>ZCCT1P1</td> <td>1</td> <td>Processing</td> <td>O</td> <td>50017103</td> <td>MDG Organizational Unit for Financials</td> </tr> <tr> <td>ZCCT1P1</td> <td>2</td> <td>Final Check</td> <td>O</td> <td>50017103</td> <td>MDG Organizational Unit for Financials</td> </tr> <tr> <td>ZCCT1P1</td> <td>3</td> <td>Revision</td> <td>O</td> <td>50017103</td> <td>MDG Organizational Unit for Financials</td> </tr> </tbody> </table>	Type of Chg. Request	Step	Description	Ob	Agent ID	ZCCT1P1	0	Submission	O	50017103	ZCCT1P1	1	Processing	O	50017103	ZCCT1P1	2	Final Check	O	50017103	ZCCT1P1	3	Revision	O	50017103	Assignment of Processors to Workflow Step Number						Type of Chg. Request	Step	Description (medium)	Ob	Agent ID	Full Name	ZCCT1P1	0	Submission	O	50017103	MDG Organizational Unit for Financials	ZCCT1P1	1	Processing	O	50017103	MDG Organizational Unit for Financials	ZCCT1P1	2	Final Check	O	50017103	MDG Organizational Unit for Financials	ZCCT1P1	3	Revision	O	50017103	MDG Organizational Unit for Financials
Type of Chg. Request	Step	Description	Ob	Agent ID																																																											
ZCCT1P1	0	Submission	O	50017103																																																											
ZCCT1P1	1	Processing	O	50017103																																																											
ZCCT1P1	2	Final Check	O	50017103																																																											
ZCCT1P1	3	Revision	O	50017103																																																											
Assignment of Processors to Workflow Step Number																																																															
Type of Chg. Request	Step	Description (medium)	Ob	Agent ID	Full Name																																																										
ZCCT1P1	0	Submission	O	50017103	MDG Organizational Unit for Financials																																																										
ZCCT1P1	1	Processing	O	50017103	MDG Organizational Unit for Financials																																																										
ZCCT1P1	2	Final Check	O	50017103	MDG Organizational Unit for Financials																																																										
ZCCT1P1	3	Revision	O	50017103	MDG Organizational Unit for Financials																																																										
	Press Save.																																																														

5. Additional Information

5.1. Further Reading

5.1.1. Information on SAP MDG on SAP S/4HANA

- Exchange knowledge: [SAP Community](#) | [Q&A](#) | [Blog](#)
- Try SAP Master Data Governance on S/4HANA for free: [Trial Version](#)
- Try SAP Master Data Governance on S/4HANA on the SAP Cloud Appliance Library: [S/4HANA 2022 FPS1](#)
- Learn more: [Latest Release](#) | [Help Portal](#) | [How-to Information](#) | [Key Presentations](#)

5.1.2. SAP Roadmap Explorer

- Please see the [roadmap for SAP Master Data Governance](#)

5.1.3. Related Information

- Learn more: [Floorplan Manager for Web Dynpro ABAP](#) | [How to Adapt FPM](#) | [FPM Blog](#) | [How-to Information](#) | [Service Mapping Tool](#) | [SAP S/4HANA Cookbook CVI](#)

5.2. SAP Notes

In addition to the detailed explanations written in this document, please see the following SAP Notes for further important information.

Note	Description
1619534	How to Create, Enhance and Adapt FPM Applications