



PUBLIC

## How-To: Check or Derive an Attribute Value in MDG Central Governance using BRFplus

Applicable Releases:

From EHP5 for SAP ERP 6.0 and from S/4HANA 1511

Version 1.1

January 2024



## Document History

Document Version	Description
1.0	First official release of this guide (October 2012)
1.1	Adapt template (January 2024)



<b>1 BUSINESS SCENARIO.....</b>	<b>4</b>
<b>2 BRFPLUS-BASED VALIDATION RULE.....</b>	<b>4</b>
2.1 Identify the attribute's and entity type's name .....	4
2.2 Create BRFplus objects .....	5
2.3 Try out.....	11
<b>3 BRFPLUS-BASED DERIVATION RULE .....</b>	<b>12</b>
3.1 Try out.....	17
<b>4 HINTS FOR TROUBLESHOOTING .....</b>	<b>17</b>
<b>5 ADDITIONAL INFORMATION .....</b>	<b>21</b>
5.1 Further Reading .....	21
5.1.1 Information on SAP MDG on SAP S/4HANA .....	21
5.1.2 SAP Roadmap Explorer .....	21
5.1.3 Related Information .....	21

# 1 Business Scenario

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.

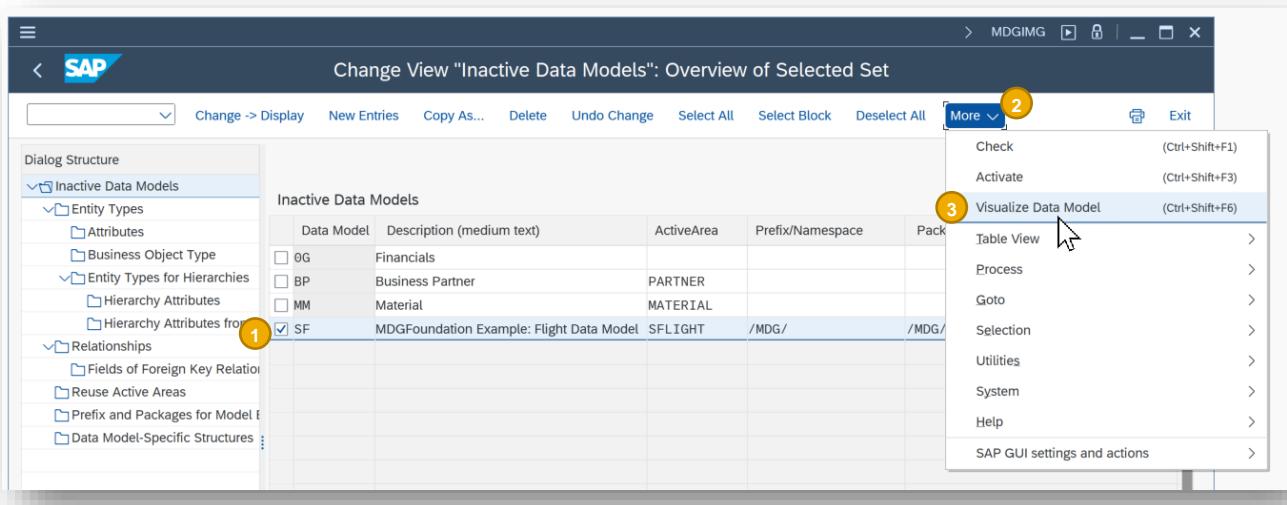
MDG offers change request (CR)-based processing of master data with integrated workflow, staging, approval, activation, and distribution.

This guide describes how to use BRFplus to validate master data attributes as well as derive them. As an example, the value of the airline local currency should be checked: It must not be **DEM** (Deutsche Mark) or **GRD** (Greek Drachma). To demonstrate a BRFplus-based derivation the airline URL is derived from the airline's name.

## 2 BRFplus-based validation rule

### 2.1 Identify the attribute's and entity type's name

Let's first examine the airline's local currency in the MDG data model. Open transaction **MDGIMG**. Navigate to *Master Data Governance > Classic Mode in SAP MDG > Central Governance > General Settings > Data Modeling* and execute Customizing Activity *Edit Data Model*. Select data model **SF** and press Visualize Data Model.

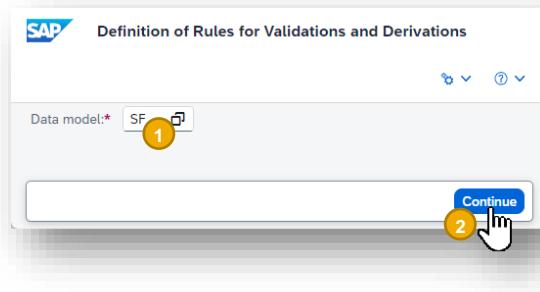


By examining the structure, you see that the attribute *Airline local currency* has technical name **CURRCODE** and belongs to entity type **Airline** (**CARR**).

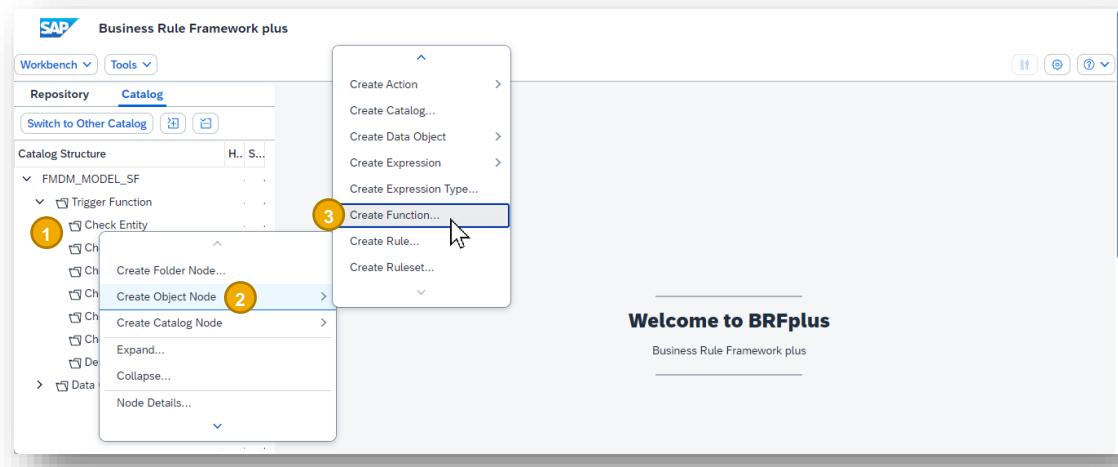
Data Model	Name	Field Type	Sto...	Data Element	Referenced Entity Type
SF	Flight Connection				
PFLI	Ownership-Link Between...				
CARR_LINK	Names for Hierarchies o...				
CARR_HIER					
CARR	Airline	Entity Type Itself	<input checked="" type="checkbox"/>	MDG_S_CARR_ID	
CARR	Airline	Attribute	<input type="checkbox"/>	MDG_S_CARRNAME	
CARRNAME	Name of Airline	Attribute	<input type="checkbox"/>	MDG_S_CURRCODE	
CURRCODE	Airline local currency	Attribute	<input type="checkbox"/>	MDG_S_CARRURL	
URL	Airline URL	Attribute	<input type="checkbox"/>		

## 2.2 Create BRFplus objects

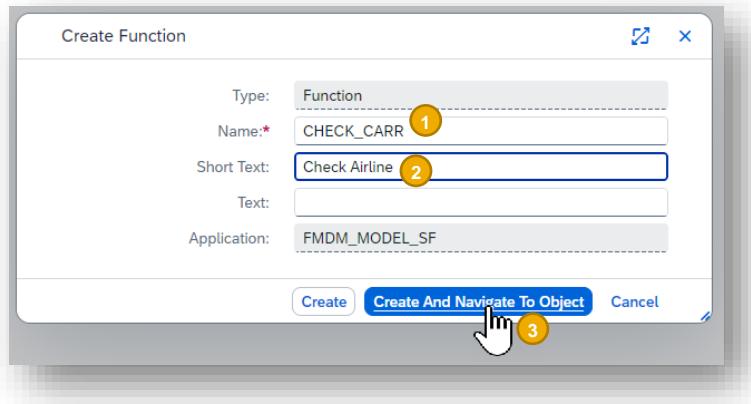
Go back to the IMG again or start transaction **MDGIMG**. Navigate to *Master Data Governance > Classic Mode in SAP MDG > Central Governance > General Settings > Data Quality and Search > Validations and Enrichments* and execute Customizing Activity *Define Validation and Derivation Rules*. You can also simply start transaction **USMD\_RULE**. Enter data model **SF** and press *Continue*.



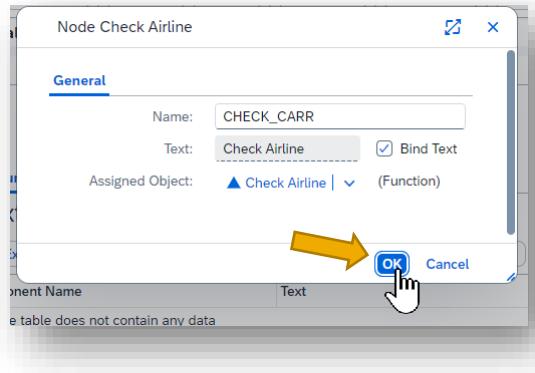
MDG provides you with a set of folders to organize the BRFplus objects. Be aware that the names of the folders do not influence the execution of the BRFplus functions. Only the name of the function is important! Select the folder *Check Entity* and open the context menu. Choose *Create Object Node > Create Function*.



Name the function **CHECK\_CARR** with short text *Check Airline*. **⚠ Note:** It is important to use the prefix **CHECK** and suffix **CARR** in this name as both define that this function is called during the check cycle (**CHECK**) for entity type **CARR** (Airline). This is a convention. Press *Create* and *Navigate to Object*.



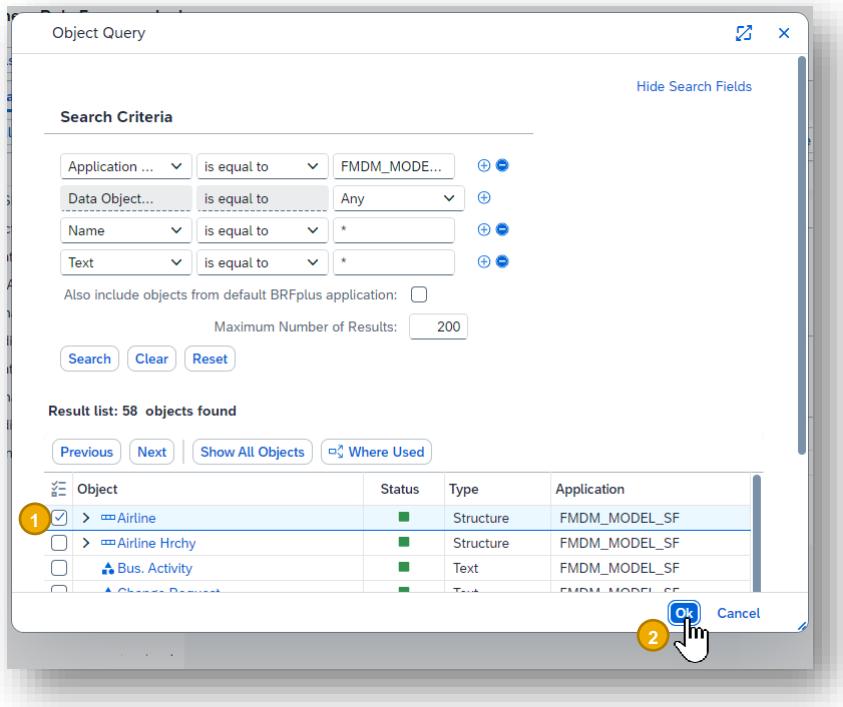
Confirm the name of the BRFplus catalog node.



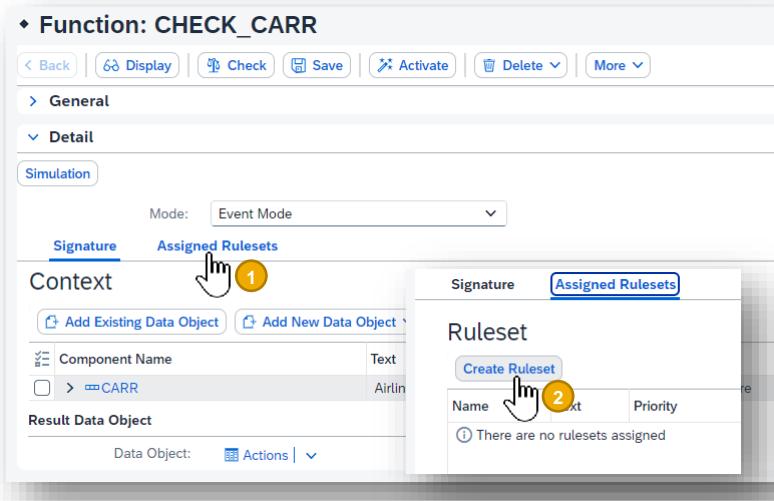
First you need to define the context of the function. Press *Add Existing Data Object*.

The screenshot shows the SAP Business Rule Framework plus interface. The left sidebar shows the Catalog structure under FMDDM\_MODEL\_SF, including nodes like Trigger Function, Check Entity, and Check Airline. The main area displays the 'Function: CHECK\_CARR' details. In the 'Context' section, there are two buttons: 'Add Existing Data Object' (highlighted with a yellow arrow) and 'Add New Data Object'. Below these buttons is a table with columns for Component Name, Text, and Type, which is currently empty.

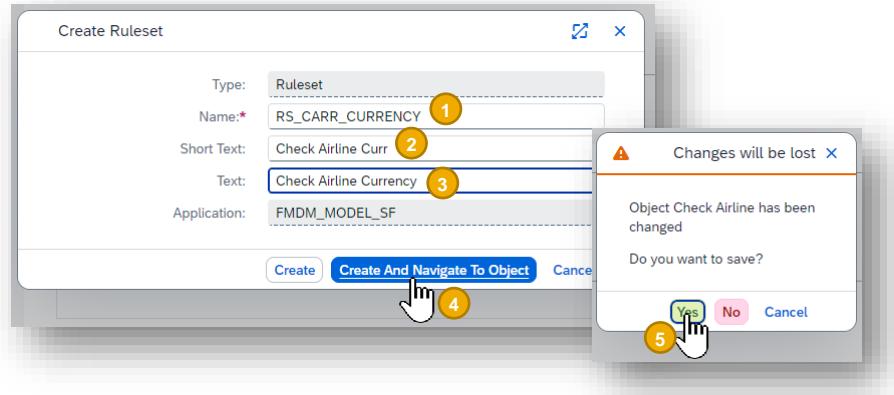
Since we want to check entities of type **CARR** (*Airline*), select the corresponding entry. **⚠ Note:** Even if you can add as many different data objects to the signature of the BRFplus function as you want, it depends on MDG whether these objects will contain data during runtime or not. For a **CHECK\_<ENTITY\_TYPE>** function, MDG will always provide the data of the respective entity type. For further information, please see the online documentation [Definition of Validations and Derivations in BRFplus](#).



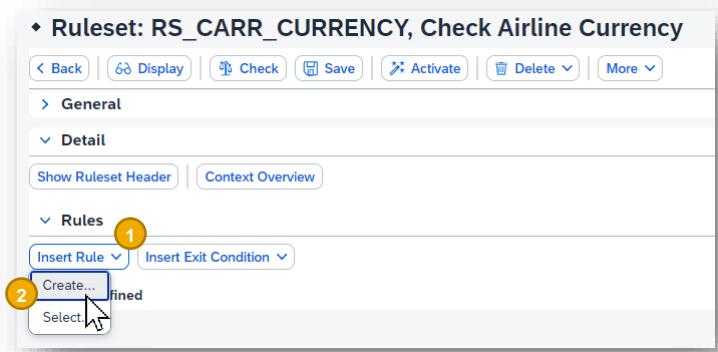
Switch to tab *Assigned Rulesets* and press *Create Ruleset*.



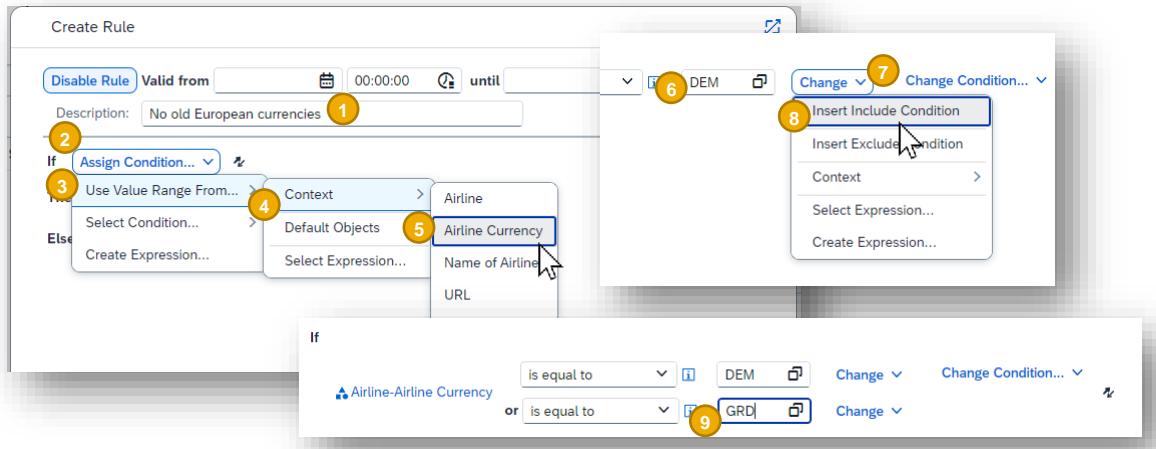
Choose any name for the ruleset, select *Create* and *Navigate to Object*. For this name there is no convention to be adhered to. Confirm save.



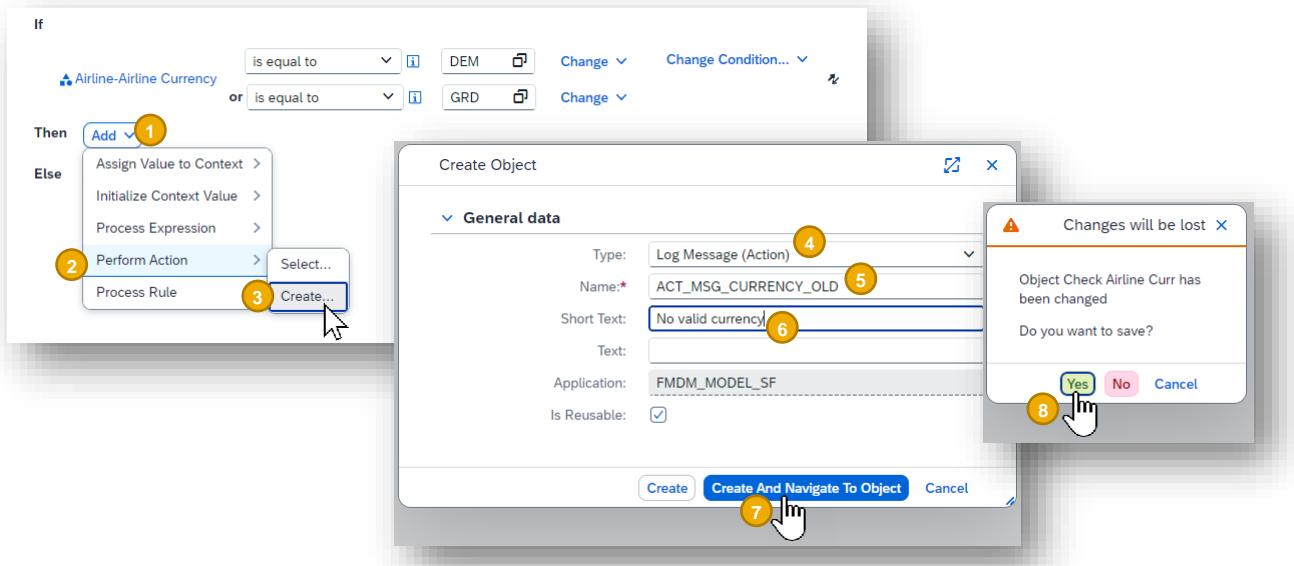
For the new ruleset, create a new rule, by selecting *Insert Rule > Create....*



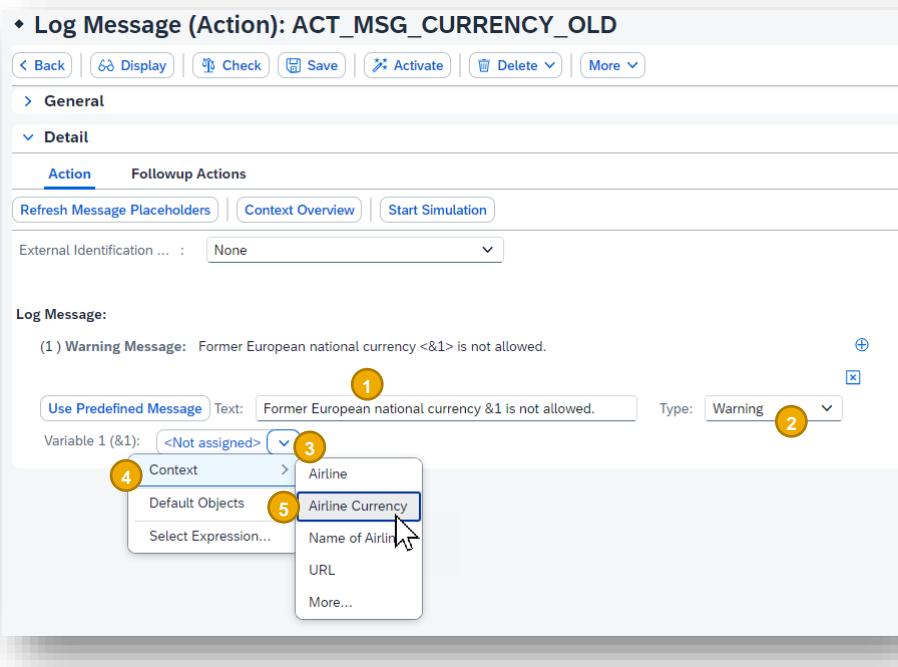
Start defining the rule using the BRFplus editor. Enter a description like *No old European currencies*. Press *Assign Condition... > Use Value Range From... > Context > Airline Currency*. Enter currency code **DEM**. Press *Change > Insert Include Condition*, by choosing the appropriate condition, then selecting *Add -> Perform Action -> Create*.



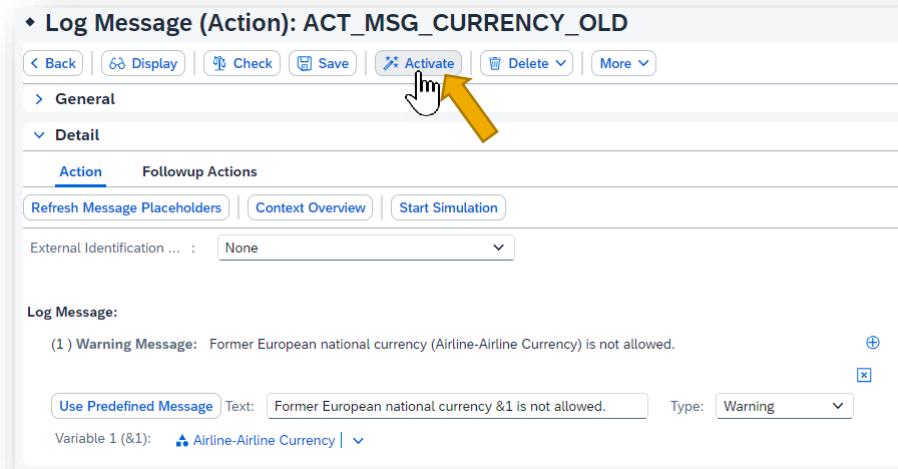
In the *Then-block* press *Add > Perform Action > Create....*. Select the type *Log Message (Action)* for the new action and choose any name, select *Create and Navigate to Object*. Confirm save.



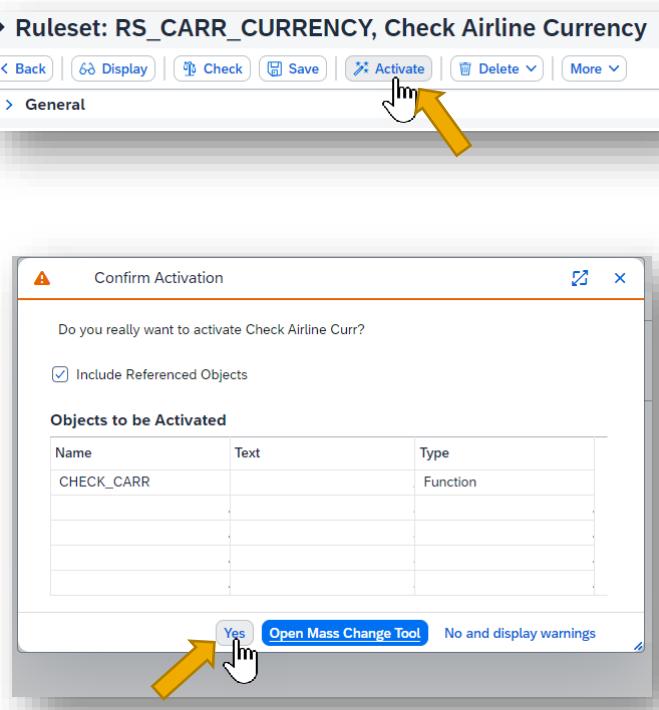
Enter a message text like *Former European national currency &1 is not allowed*. Please note that you can use variables with the '&n'-notation. Select message type *Warning*. After selecting the message type or when you hit enter, BRFplus will allow you to choose the values for the variables. Choose *Context > Airline Currency*.



Press *Activate*.

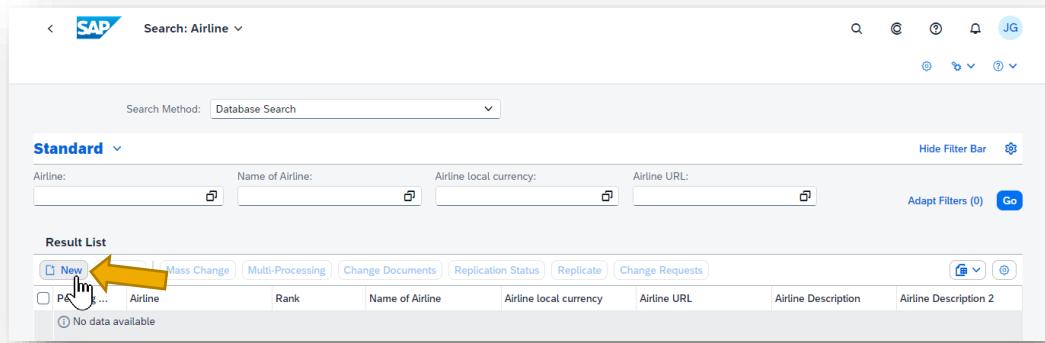


Navigate back to the ruleset and activate it. At pop-up *Confirm Activation* choose Yes.



## 2.3 Try out

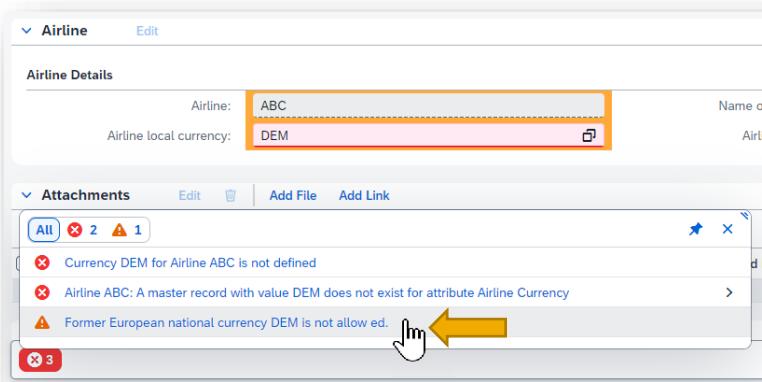
Open the Fiori Launchpad and start application *Manage Airlines – Governance*. Press *New*.



Enter **ABC** for Airline and some name and URL. In addition, enter *Airline local currency DEM*. Press *Enter*.

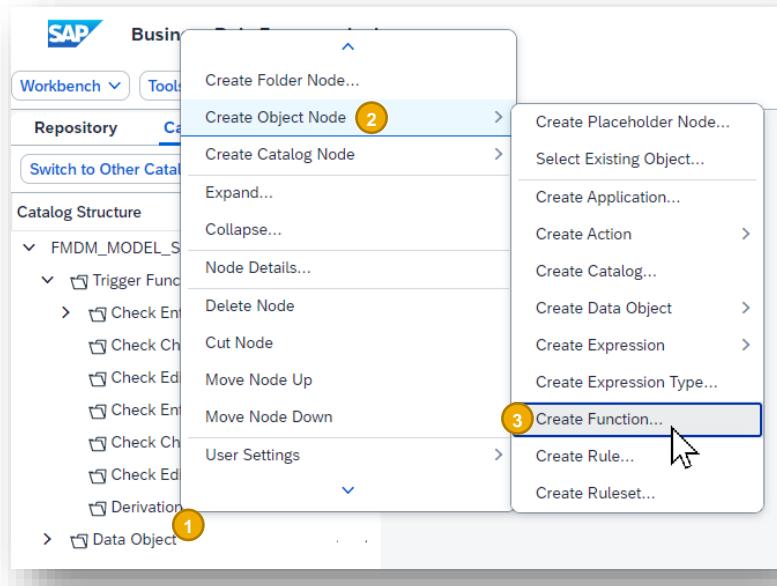
The screenshot shows the 'Process Airline' application. At the top, there's a search bar with 'Process Airline'. Below it, a toolbar with 'Refresh', 'Check', 'Undo', 'Redo', 'Side Panel', and other icons. The main area has tabs for 'Change Request' (selected) and 'Edit'. Under 'General', fields include 'Change Request' (49848), 'Description' (Create Airline), 'Priority' (Very High), 'Due Date' (04.04.2024), and 'Reason' (New Airline - Alliance member). Under 'Process Data', fields include 'Status' (Changes to Be Executed), 'Current Work Item' (New Change Request), 'Created On/By' (05.01.2024 15:09:17), and 'Type of Change Request' (SFC01). A large yellow arrow points to the 'Process Airline' button in the toolbar. Below, under 'Airline', 'Airline Details' show 'Airline' (ABC) and 'Name of Airline' (ABC Air). The 'Airline local currency' field (DEM) is highlighted with a yellow circle labeled 4. The 'Airline URL' field (https://abcair.com) is highlighted with a yellow circle labeled 3.

Multiple messages are displayed. Let's ignore the first two messages which indicate that **DEM** is no valid currency code anyways. The expected warning message is raised and displayed.

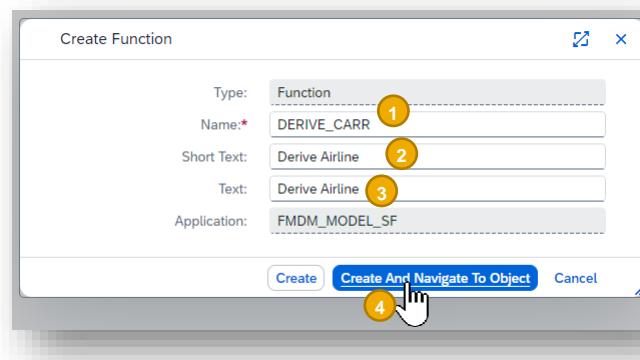


### 3 BRFplus-based derivation rule

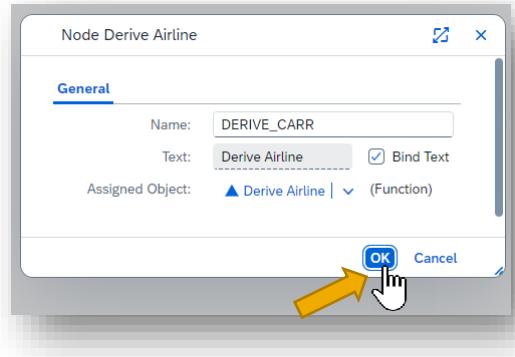
In a similar way like described in chapter 2.2 you can define rules to derive attribute values. To do so, create a new function below the *Derive Entity* folder. Open the context menu and choose *Create Object Node > Create Function...*.



Name the function **DERIVE\_CARR** with short text *Derive Airline*. **⚠ Note:** It is important to use the prefix **DERIVE** and suffix **CARR** in this name as both define that this function is called during the derive cycle (**DERIVE**) for entity type **CARR** (Airline). This is a convention. As for the check function (see chapter 2.2), only the name of the derivation function is important, not the folders where you locate the catalog object. Press *Create and Navigate to Object*.

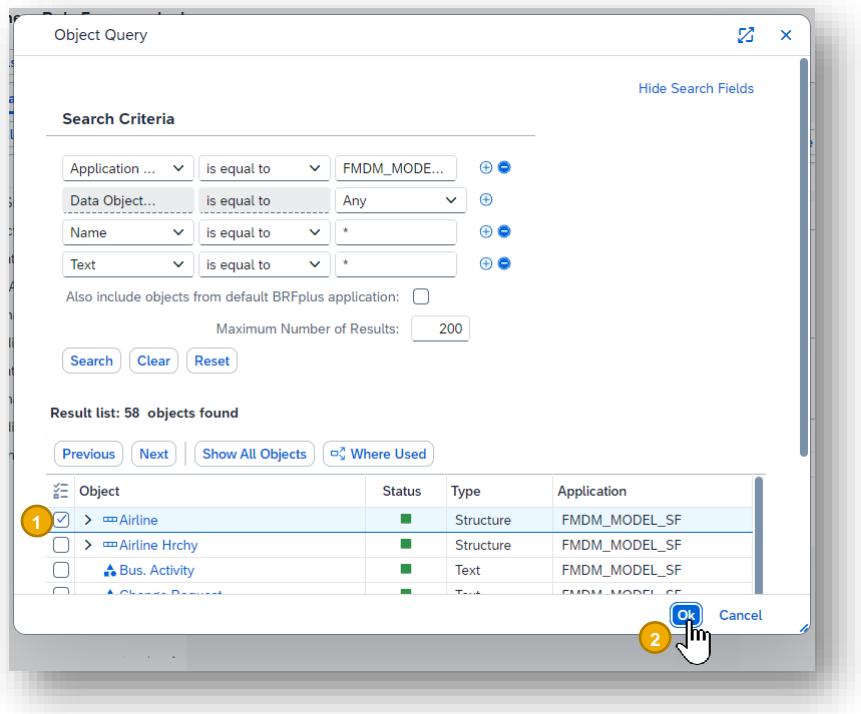


Confirm the name of the BRFplus catalog node.

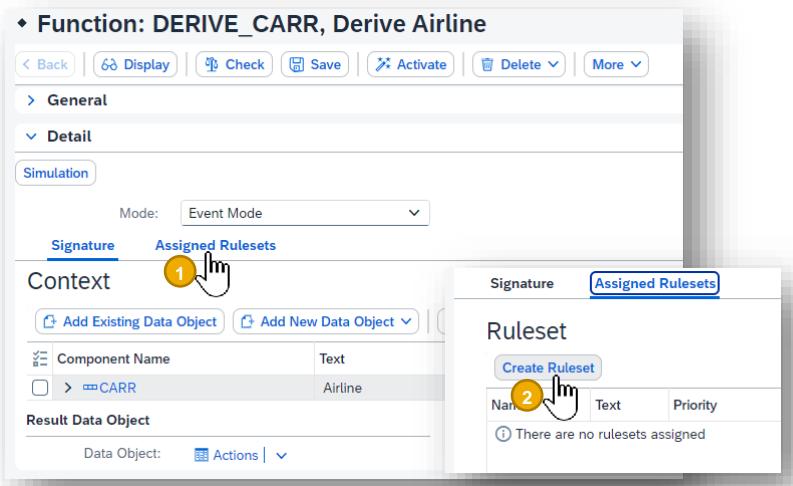


First you need to define the context of the function. Press *Add Existing Data Object*.

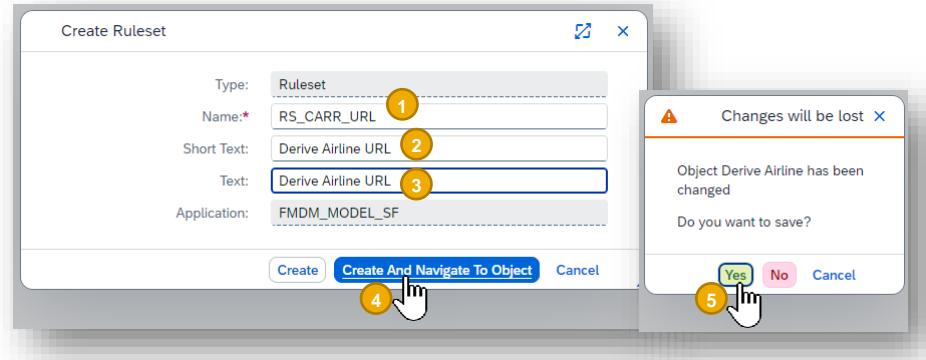
Since we want to check entities of type **CARR** (*Airline*), select the corresponding entry. **⚠ Note:** Even if you can add as many different data objects to the signature of the BRFplus function as you want, it depends on MDG whether these objects will contain data during runtime or not. For a **DERIVE\_<ENTITY\_TYPE>** function, MDG will always provide the data of the respective entity type. For further information, please see the online documentation [Definition of Validations and Derivations in BRFplus](#).



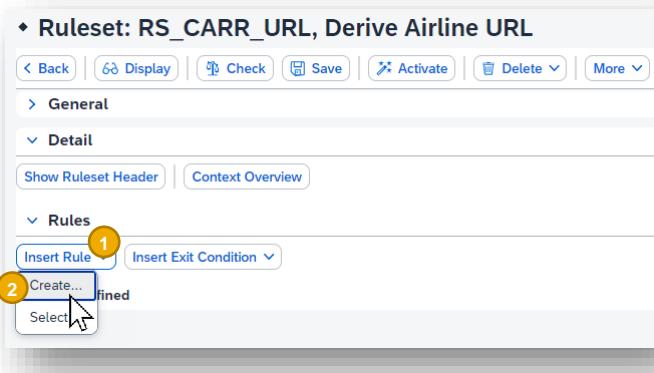
Switch to tab *Assigned Rulesets* and press *Create Ruleset*.



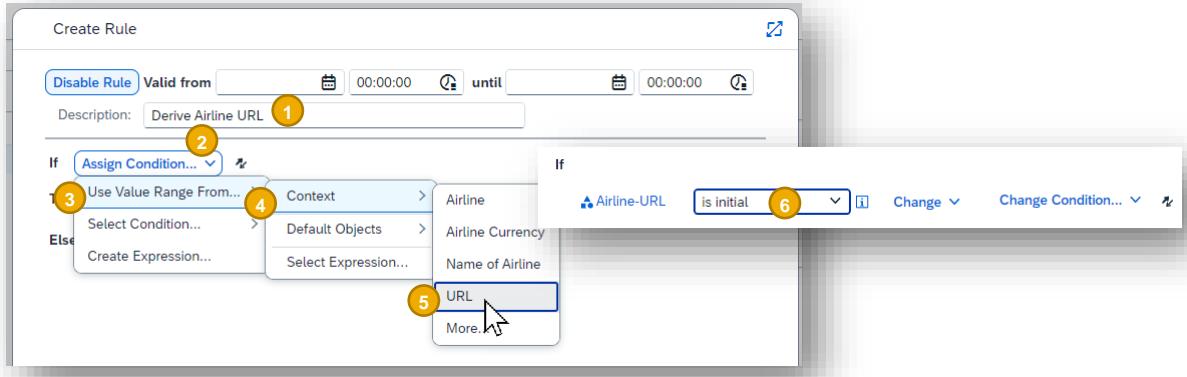
Choose any name for the ruleset, select *Create and Navigate to Object*. For this name there is no convention to be adhered to. Confirm save.



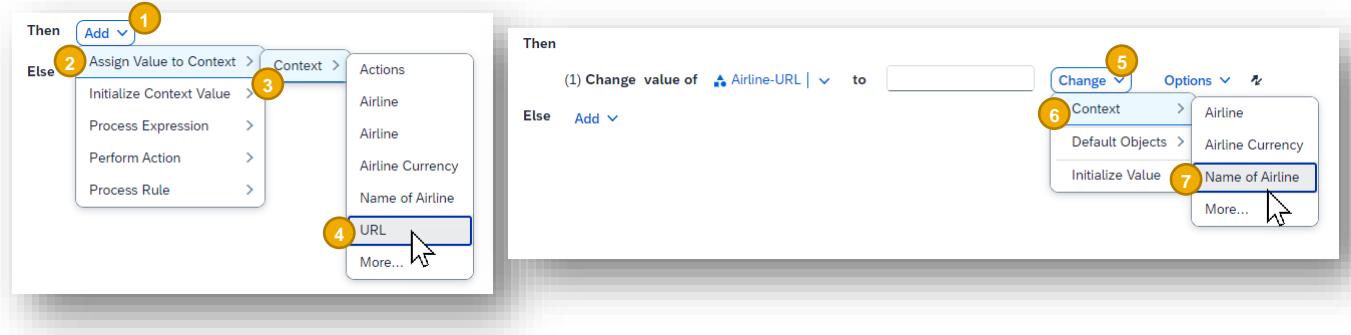
For the new ruleset, create a new rule, by selecting *Insert Rule > Create....*



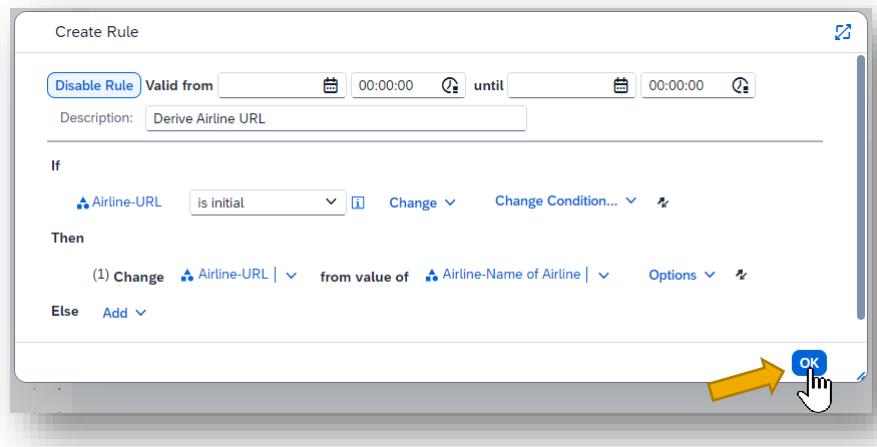
Start defining the rule using the BRFplus editor. Enter a description like *Derive Airline URL*. Press *Assign Condition... > Use Value Range From... > Context > URL*. Select *is initial*.



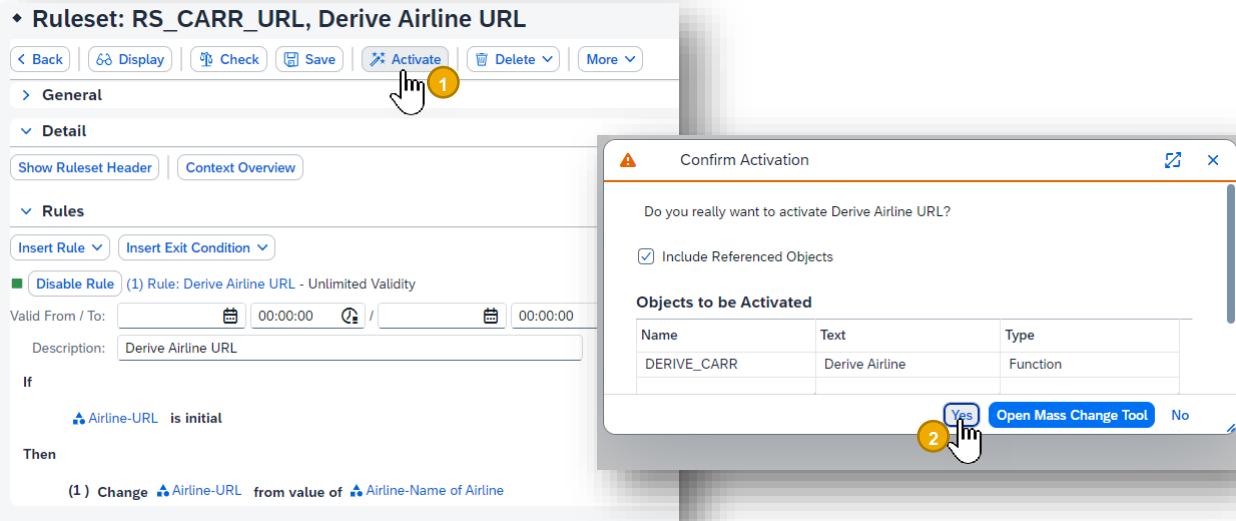
In the *Then-block* press *Add > Assign Value to Context > Context > URL*. Press *Change > Context > Name of Airline*.



Press OK.

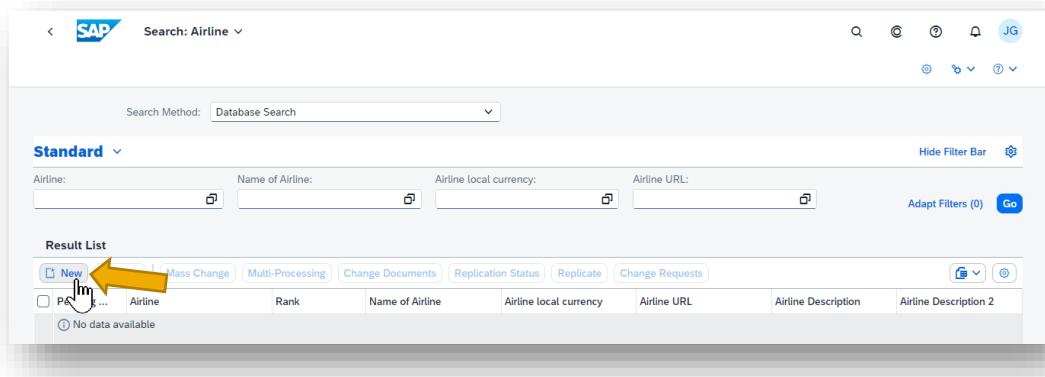


Active the ruleset and the function. At pop-up *Confirm Activation* choose Yes.

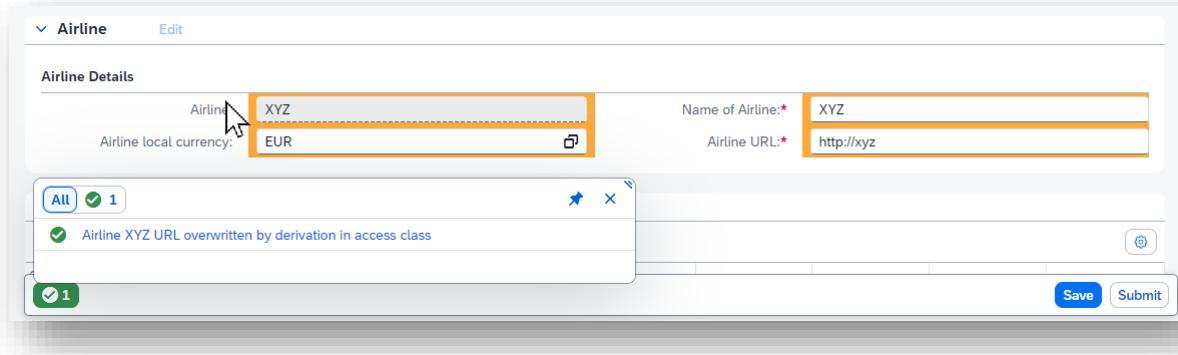


## 3.1 Try out

Open the Fiori Launchpad and start application *Manage Airlines – Governance*. Press *New*.



Enter **xyz** for Airline and its name as well as **EUR** for *Airline local currency*. Leave field *Airline URL* empty. Press *Enter* or *Check*. The BRFplus function **DERIVE\_CARR** will be triggered, and attribute *Airline URL* will be filled with the name of the airline. Because the access class of the example scenario also includes some check and derivation logic, this value will be converted to lower case and be prefixed with **http://**.



## 4 Hints for troubleshooting

When starting to work with BRFplus rules in MDG, it is a good idea to include a ruleset for debugging purposes to see whether the rule is actually triggered and to check values within the BRFplus context. In the following screenshot you can see a ruleset (RS\_DEBUG\_INFO). It can be enabled/ disabled on purpose. Insert a new rule. Press *Insert Rule > Create....*.

Ruleset: RS\_DEBUG\_INFO

General

Detail

Hide Ruleset Header | Context Overview

Enabled:  Number of Rules: 0

Function: ▲ Derive Airline | Number of Variables: 0

Precondition: <Not assigned> | Priority: 00

**Variables**

(1) <Not assigned> | Expressions Which Initialize the Variables

(1) <Not assigned> | v

**Rules** 1

Insert Rule | Insert Exit Condition | Create... | Defined | Select...

In the *Then-block* press *Add > Perform Action > Create...*.

Create Rule

Disable Rule Valid from 00:00:00 until 00:00:00

Description:

If Assign Condition... | Then 1 | Else

Then Add 1

Assign Value to Context | Initialize Context Value | Process Expression | Perform Action | Process Rule

Perform Action 2 | Select... | Create... 3

Choose *Log Message (Action)* as type, enter some name and short text. No convention applies here. Press *Create And Navigate To Object*.

Create Object

General data

Type: Log Message (Action) 1 | Name: DEBUG\_INFO 2 | Short Text: Debug Info 3 | Application: FMMD\_MODEL\_SF | Is Reusable:

Create | Create And Navigate To Object | Cancel 4

At the new action enter a message text using the '&n'-notation. Press *Enter*.

The screenshot shows the SAP Fiori interface for creating a log message. The title bar says 'Log Message (Action): DEBUG\_INFO'. The 'General' tab is selected under 'Detail'. In the 'Log Message' section, there is one information message with the text 'Name: &1, Curr: &2, URL: &3'. A yellow arrow points to the text input field, highlighting the placeholder '&1, &2, &3'.

After the roundtrip, there are three message variables displayed. From the context assign the corresponding attributes to all variables.

A context assignment dialog is open over the log message configuration. It shows three variables: Variable 1 (&1), Variable 2 (&2), and Variable 3 (&3). Variable 1 is set to '<Not assigned>'. A dropdown menu is open for Variable 1, showing options like 'Context', 'Default Objects', and 'Select Expression...'. The 'Context' option is highlighted with a yellow circle labeled '1'. The 'Select Expression...' option is highlighted with a yellow circle labeled '3'. A mouse cursor is pointing at the 'Name of Airline' option in the list, which is highlighted with a yellow circle labeled '2'.

As a result, all message variables are assigned with an attribute from the context.

The screenshot shows the log message configuration again. The message text is now '(1) Information Message: Name: (Airline-Name of Airline), Curr: (Airline-Airline Currency), URL: (Airline-URL)'. The context assignment dropdowns are now populated: Variable 1 (&1) is set to 'Airline-Name of Airline', Variable 2 (&2) is set to 'Airline-Airline Currency', and Variable 3 (&3) is set to 'Airline-URL'. These assignments are highlighted with a yellow border.

Activate the ruleset.

The screenshot shows two overlapping Fiori application windows. The top window is titled 'Ruleset: RS\_DEBUG\_INFO' and has a yellow circle with the number '1' in its top right corner. It contains tabs for 'General', 'Detail', and 'Rules'. Under 'Rules', there is a section for 'Insert Rule' and 'Insert Exit Condition'. A message box says '(1) Rule: No description is available - Un'. Below this, a button labeled '(1) Perform ⚡ Debug Info' is highlighted with a yellow box. The bottom window is titled 'Function: DERIVE\_CARR, Derive Airline' and also has a yellow circle with the number '1'. It has tabs for 'General' and 'Detail'. Under 'Detail', there is a 'Simulation' section with a dropdown for 'Mode' set to 'Event Mode'. A 'Signature' dropdown shows 'Assigned Rulesets'. At the bottom, there is a table titled 'Ruleset' with columns: Name, Text, Priority, Enabled, Precondition, Status, and Executable. Two rows are listed: 'RS\_CARR\_URL' (Derive Airline URL) and 'RS\_DEBUG\_INFO' (both columns show '[undefined]'), both with 'Yes' in the 'Enabled' column and checked 'Status' and 'Executable' boxes.

Open the Fiori Launchpad and start application *Manage Airlines – Governance*. Press *New*.

The screenshot shows the 'Manage Airlines – Governance' application in Fiori Launchpad. The title bar says 'Search: Airline'. The main area is titled 'Standard' and has fields for 'Airline', 'Name of Airline', 'Airline local currency', and 'Airline URL'. Below these is a 'Result List' table with columns: 'P...', 'Airline', 'Rank', 'Name of Airline', 'Airline local currency', 'Airline URL', 'Airline Description', and 'Airline Description 2'. A yellow arrow points to the 'New' button at the top left of the result list table.

Enter **RST** for Airline, name **RST Airline**, **EUR** for *Airline local currency* and <https://example.com> for *Airline URL*. Press *Enter* or *Check*. The just configured message is displayed.

The screenshot shows the 'Manage Airlines – Governance' application in edit mode for a new airline record. The 'Airline Details' section has fields: 'Airline' (RST), 'Name of Airline' (RST Airline), 'Airline local currency' (EUR), and 'Airline URL' (https://example.com). Below this is an 'Attachments' section with a table showing one attachment named 'Name: RST Airline, Curr: EUR, URL: https://example.com'. A yellow arrow points to this attachment entry. At the bottom right are 'Save' and 'Submit' buttons.

[WWW.sap.com](http://WWW.sap.com).

# 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

- [Validation and Derivation](#) (Configuration of MDG, Central Governance for Custom Objects)
- 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](#)

