How to Use Other Conditions in Decision Tables in Rules Composer

Applies to:
SAP NetWeaver Business Rules Management. For more information, visit the Business Rules Management homepage.

Summary
This tutorial helps you work with other conditions in Decision Tables in the Rules Composer. A business use case has been created for the purpose.

Given details such as connection provider, type of line/connection and connection destination, the long distance phone rate must be calculated.

This tutorial guides you add Other Conditions in a Decision Table to calculate the long distance phone rate given a set of criteria.

Authors: Arti Gopalan and Sarada Ustili
Company: SAP Labs, India
Created on: 27 May 2008
Table of Contents

Prerequisites .......................................................................................................................................................3
  Knowledge Required ........................................................................................................................................3
  Software Requirements .................................................................................................................................3

Procedure ...........................................................................................................................................................3
  Creating the Rules Composer DC ..................................................................................................................3
  Creating the Ruleset .......................................................................................................................................3
  Creating the XML Schema ...............................................................................................................................4
    Adding XSD Elements ...................................................................................................................................5
    Renaming the XSD Aliases .............................................................................................................................5
  Creating the Decision Table ...........................................................................................................................6
    Adding Condition and Action Values ...........................................................................................................7
  Creating the Variable Definition ....................................................................................................................8
  Creating the Decision Table with Other Condition ..............................................................................................9
    Adding Condition, Other Condition and Action Values ..................................................................................10
  Creating the Rule ......................................................................................................................................12
  Deploying the Rules ..................................................................................................................................18
  Executing the Rules ...................................................................................................................................19
    Creating the Web Module ............................................................................................................................19
    Adding Dependency to the Web Module .........................................................................................................19
    Creating the Enterprise Application .............................................................................................................20
    Adding Dependency to the Enterprise Application .......................................................................................20
    Creating application.xml ..............................................................................................................................20
    Building and Deploying ...............................................................................................................................21
    Running the Web Module .............................................................................................................................21

Related Content ............................................................................................................................................24

Copyright ..........................................................................................................................................................25
Prerequisites

Knowledge Required

• You have basic knowledge in rules modeling
• You are familiar with Business Rules Management System

Software Requirements

• You work in the SAP NetWeaver Developer Studio
• Your SAP NetWeaver Developer Studio version includes the Rules Composer perspective
• You should have a running instance of SAP AS, and should have configured the SAP NetWeaver Developer Studio with this instance

Note: In the SAP NetWeaver Developer Studio, choose Window -> Open Perspective -> Other. In the dialog box that appears, choose Rules Composer and choose OK.

Procedure

Creating the Rules Composer DC

1. In the SAP NetWeaver Developer Studio, choose File -> New -> Project.
2. In the wizard that appears, expand the Rules Composer node and choose Rules Composer Development Component. Choose Next.
3. In the screen that appears, choose the software component where you want to create the DC.
   For example the software component could be MyComponents [demo.sap.com] under the Local Development node. Choose Next.
4. In the screen that appears, enter callchargecalc in the Name field and choose Finish.
You should see the callchargecalc node in the Project Explorer view.

Creating the Ruleset

1. In the Project Explorer view, expand the callchargecalc node and in the context menu of the Rules Modeling node, choose New Ruleset.
2. In the dialog box that appears, enter SetCallCharge field. Choose OK.
You should see the ruleset: `SetCallCharge` under the `Rules Modeling` node as shown below:

Creating the XML Schema

1. In the `Project Explorer` view, expand the `src` node and in the context menu of the `wsdl` node, choose `New` -> `Other`.  
2. In the wizard that appears, expand the `XML` node and choose `XML Schema`. Choose `Next`.  
3. In the screen that appears, enter `callcharges.xsd` in the `File Name` field. Choose `Finish`.  
4. In the `Project Explorer` view, you should see the `callcharges.xsd` under the `wsdl` node.  
5. Double-click the `callcharges.xsd` and in the window that appears choose the `Source` tab at the bottom.  

In the tab page that appears delete all existing content and copy the following in the `Source` tab page:

```xml
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns="xmlsc.xsd" elementFormDefault="qualified"
    targetNamespace="xmlsc.xsd">
    <xsd:element name="CallCharges" type="CallChargesType" />

    <xsd:complexType name="CallChargesType">
        <xsd:sequence maxOccurs="unbounded">
            <xsd:element name="Charges" type="CallChargeType" />
        </xsd:sequence>
    </xsd:complexType>

    <xsd:complexType name="CallChargeType">
        <xsd:sequence>
            <xsd:element name="ConnectionProvider" type="xsd:string" />
            <xsd:element name="ConnectionRate" type="xsd:double" />
            <xsd:element name="DestinationCountry" type="xsd:string" />
            <xsd:element name="OriginCountry" type="xsd:string" />
            <xsd:element name="TypeOfLine" type="xsd:string" />
        </xsd:sequence>
    </xsd:complexType>
</xsd:schema>
```
Adding XSD Elements

1. In the Project Explorer view, expand the Rules Modeling node and double-click the Aliases node.

2. In the Project Aliases Editor that appears, choose the XSD Aliases tab and in the tab page that appears, choose the Add XSD Elements tab.

3. In the dialog box that appears, expand the http://www.example.org/CallCharges node and select Call Charges.


5. In the Alias Name table select all the XML schema element checkboxes.

Renaming the XSD Aliases

In the Alias Name table, click each of the aliases. The aliases become editable. Enter an alternative name for the alias.

<table>
<thead>
<tr>
<th>Alias Name</th>
<th>Rename as</th>
</tr>
</thead>
<tbody>
<tr>
<td>CallCharges/Charges/ConnectionProvider</td>
<td>Connection Provider</td>
</tr>
<tr>
<td>CallCharges/Charges/ConnectionRate = {double}</td>
<td>Connection Rate = {double}</td>
</tr>
<tr>
<td>CallCharges/Charges/ConnectionRate</td>
<td>Connection Rate</td>
</tr>
<tr>
<td>CallCharges/Charges/DestinationCountry</td>
<td>Destination Country</td>
</tr>
<tr>
<td>CallCharges/Charges/TypeofLine</td>
<td>TypeofLine</td>
</tr>
<tr>
<td>CallCharges/Charges/OriginCountry</td>
<td>Origin Country</td>
</tr>
</tbody>
</table>

The result must be as follows:
Creating the Decision Table

You can capture the criteria and the corresponding rates for long distance phone calls in a single table.

1. In the Project Explorer view, expand the callchargecalc node, the Rules Modeling node and in the context menu of the SetCallCharge node, choose New Decision Table.

2. In the Decision Table Creation Wizard that appears, enter CallChargeDT in the Decision Table Name field and choose Next.

3. On the Select the Conditions screen, double-click the following aliases in the Available Conditions section:
   - Destination Country
   - Origin Country

The aliases appear in the Selected Conditions section as follows:

4. Choose Next.
5. On the Select the Actions screen, double-click ConnectionRate = {double} in the Available Actions section.

The alias appears in the Selected Actions section as follows:

![Image of Decision Table Editor with selected action](image)

6. Choose Finish and save the changes.

Adding Condition and Action Values

1. In the Decision Table Editor, double-click each of the cells under the condition headers (Origin Country and Destination Country) and enter the values.

   **Note:** You can also choose Add Condition Value in the context menu of the cells under the condition headers and in the dialog box that appears enter all the values in the available rows.

2. Double-click each cell under the action header (ConnectionRate = {double}) in the Decision Table, enter a value and press Enter key.
Refer to the Decision Table below for the condition and action values you need to enter:

```
<table>
<thead>
<tr>
<th>Origin Country</th>
<th>Destination Country</th>
<th>Connection Rate = (double)</th>
</tr>
</thead>
<tbody>
<tr>
<td>India</td>
<td>USA</td>
<td>100</td>
</tr>
<tr>
<td>Germany</td>
<td>USA</td>
<td>150</td>
</tr>
<tr>
<td>USA</td>
<td>India</td>
<td>200</td>
</tr>
<tr>
<td>Germany</td>
<td>USA</td>
<td>250</td>
</tr>
<tr>
<td>India</td>
<td>Germany</td>
<td>50</td>
</tr>
<tr>
<td>Germany</td>
<td>USA</td>
<td>200</td>
</tr>
</tbody>
</table>
```

Creating the Variable Definition

1. In the Project Explorer view, expand the `callchargecalc` node, Rules Modeling node, and double-click the `SetCallCharge` ruleset.
2. In a Ruleset Editor that appears, choose the Definitions tab.
3. In the Definitions Editor, under the Variable Definitions section, choose the Add icon and in the drop down that appears, choose double.
4. In the dialog that appears, enter `discountPercentage` and choose OK.

The result must be as shown below:

```
Variable Definitions:

double discountPercentage = Q
```
Creating the Decision Table with Other Condition

1. In the Project Explorer view, expand the callchargecalc node, the Rules Modeling node and in the context menu of the SetCallCharge node, choose New Decision Table.

2. In the Decision Table Creation Wizard that appears, enter SetDiscountPercentage in the Decision Table Name field and choose Next.

3. On Select the Conditions screen, double-click Destination Country alias in the Available Conditions section.

   The alias appears in the Selected Conditions section as follows:

4. Select Table has Other Condition checkbox and choose Next.

5. On Select the Actions screen, double-click discountPercentage in the Available Actions section.
The variable definition appears in the *Selected Actions* section as follows:

6. Choose *Finish* and save the changes.

**Adding Condition, Other Condition and Action Values**

1. In the Decision Table Editor, double-click each of the cells under the condition header, *Destination Country* and enter the condition values. In the context menu of the cell under *Destination Country* condition header, enter the values: *India, USA* and *Germany*.

Note: You can also choose *Add Condition Value* in the context menu of the cells under the condition headers and in the dialog box that appears enter all the values in the available rows.

2. Double-click each cell under the *Other Condition* header in the Decision Table.

3. In the editor that appears, choose the icon that looks like a plus icon.
   The default condition: *System.currentTimeMillis()* Greater Than RValue appears.
4. To edit the default condition:
   a. Choose the LValue: `System.currentTimeMillis()` and in the drop down menu choose `TypeOfLine` as shown below:
   
   b. Once you select `TypeOfLine`, the comparator : Greater Than changes to Equals
   
   c. Choose the RValue and in the inline text box enter `Line1`, and press Alt + Enter.

5. Double-click each cell under the action header, `discountPercentage` in the Decision Table, enter a value and press Enter key.
Refer to the Decision Table below for the condition, other condition and action values you need to enter:

<table>
<thead>
<tr>
<th>Destination Country</th>
<th>Other Condition</th>
<th>discountPercentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>India</td>
<td>TypeOfLineRoutesInList</td>
<td>10</td>
</tr>
<tr>
<td>India</td>
<td>High Country To India, Germany</td>
<td>20</td>
</tr>
<tr>
<td>USA</td>
<td>Connection Rate Greater Than 100</td>
<td>5</td>
</tr>
<tr>
<td>Germany</td>
<td>Connection Rate Less Than 150</td>
<td>25</td>
</tr>
</tbody>
</table>

Creating the Rule
You need to create a rule called CallChargeRule that evaluates the CallChargeDT.

1. In the Project Explorer view, expand the callchargecalc node, the Rules Modeling node and in the context menu of the SetCallCharge node, choose New Rule.
2. In the dialog box that appears, enter CallChargeRule in the field and choose OK.
3. In the Rule Editor that appears, under If section, choose the Add icon.
   
   The default condition: Operation.isSuccessful() Equals true appears as shown below:
4. Edit the default condition as follows:
   
a. Choose the LValue: `Operation.isSuccessful()` and in the drop down menu choose `Connection Provider` as shown below:
   
   ![Diagram showing the selection of `Operation.isSuccessful()` and `Connection Provider`]

   b. Leave the comparator: `Equals` as it is
c. Choose the RValue: Default Value and in the inline textbox enter A.

The result must be as shown below:

5. Under Then section, choose the Add icon and in the drop down menu that appears, expand the Evaluate-Decision Table node and choose CallChargeDT as shown below:
The result must be as shown below:

6. Repeat step 5 to set and evaluate SetDiscountPercentage Decision Table. Choose \texttt{SetDiscountPercentage} as shown below:
The result must be as shown below:

7. In the Rule editor that appears, under *Then* section, choose the *Add* icon and in the drop down menu that appears, choose *Execute-<Method Name>*.
8. Choose `<Action Method>` and in the drop down menu choose *Connection Rate = {double}* as shown below:

The result must be as shown below:
9. Choose (double) and in the drop down menu choose *Connection Rate*, add a compound expressions and braces as shown below:

![Decision Table Screenshot](image)

10. Save the changes.

**Deploying the Rules**

1. In the *Project Explorer* view, in the context menu of the Rules Composer DC:*callchargecalc* node, choose *Development Component -> Build*.

2. In the dialog box that appears, select the *callchargecalc* checkbox and choose OK.

   **Note:** If the Infrastructure Console is not open, choose Window -> Show View -> Other and in the dialog box that appears, expand the Development Infrastructure node and choose Infrastructure Console and then choose OK.

3. In the context menu of the Rules Composer DC:*callchargecalc* node, choose *Development Component -> Deploy*.

4. In the dialog box that appears, select the *callchargecalc* checkbox and choose OK.

   **Note:** Open the Infrastructure Console, to check if the deployment has happened successfully.
Executing the Rules

Creating the Web Module

1. In the SAP NetWeaver Developer Studio, choose File -> New -> Project.
2. In the wizard that appears, expand the Development Infrastructure node and choose Development Component. Choose Next.
3. In the screen that appears, expand the J2EE node and choose Web Module. Choose Next.
4. In the screen that appears, choose the software component where you want to create the DCs. For example expand the Local Development node and choose MyComponents [demo.sap.com]. Choose Next.
5. In the screen that appears, enter callchargecalculator_wm in the Name field. Choose Finish.
6. The Java EE perspective opens and in the Project Explorer view, you should see the callchargecalculator_wm node.

Note: If the Project Explorer view does not open, choose Window -> Show View -> Other. In the dialog box that appears, expand the General node and choose Project Explorer.

Adding Dependency to the Web Module

1. Choose Window -> Open Perspective -> Other.
2. In the dialog that appears, choose Development Infrastructure. Choose OK.

Note: If the Component Browser view does not open, choose Window -> Show View -> Other. In the dialog box that appears, expand the Development Infrastructure node and choose Component Browser. Choose Ok.

3. In the Component Browser view, expand the ‘Local Development’ node, MyComponents [demo.sap.com] node and choose the callchargecalculator_wm node.

Note: If the Component Browser view does not open, choose Window -> Show View -> Other. In the dialog box that appears, expand the Development Infrastructure node and choose Component Properties. Choose Ok.

4. In the Component Properties view, choose Dependencies.
5. Choose the Add button and in the wizard that appears, expand the BRMS-FAÇADE [sap.com] node and select the tc/brms/facade checkbox. Choose Next.

Note: Make sure you are in the Java EE perspective. Unzip the project file and do the following

1. Expand the web module: callchargecalculator_wm node and in the context menu of the Java Resources: source node, choose New ->Other.
2. In the wizard that appears, expand the Java node and choose Package. Choose Next.
3. In the screen that appears, enter com.sap.helper in the Name field.
6. In the context menu of Web Content node copy the following files: CallCharges.jsp, index.jsp, invoker.jsp.
Creating the Enterprise Application

Make sure that you are in the Java EE perspective.

1. In the SAP NetWeaver Developer Studio, choose File -> New -> Project.
2. In the wizard that appears, expand the Development Infrastructure node and choose Development Component. Choose Next.
3. In the screen that appears, expand the J2EE node and choose Enterprise Application. Choose Next.
4. In the screen that appears, choose the software component where you want to create the DCs. For example, expand the Local Development node and choose MyComponents [demo.sap.com]. Choose Next.
5. In the screen that appears, enter callchargecalc_ear in the Name field. Choose Next.
6. Choose Next.
7. In the screen that appears, select the LocalDevelopment~LocalDevelopment~buyer_wm~demo.sap.com checkbox. Choose Finish

In the Project Explorer view, you should see the callchargecalc_ear node.

Adding Dependency to the Enterprise Application

Make sure that you are in the Development Infrastructure perspective.

1. In the Component Browser view, expand the MyComponents[demo.sap.com] node and choose the callchargecalc_ear node.
2. In the Component Properties view, choose Dependencies.
3. Choose the Add button and in the wizard that appears, expand the BRMS-FACADE[sap.com] node and select the tc/brms/facade checkbox. Choose Next.
4. In the screen that appears, select the Design Time, Deploy Time, Run Time checkboxes. Choose Finish.

Note: In the context menu of the callchargecalculator_wm and callchargecalc_ear nodes, choose Sync/Create Project -> Sync Used DCs.
In the dialog box that appears, choose Ok.

Creating application.xml

Make sure you are in the Java EE perspective.

1. Expand the enterprise application: buyer_ear node and in the context menu of the Deployment Descriptor: LocalDevelopment~LocalDevelopment~buyer_ear~demo.sap.com node, choose create application.xml.

Note: Expand the enterprise application: buyer_ear node, META-INF node double click the application.xml node.

You should see the application.xml window with the following lines:

```xml
<?xml version="1.0" encoding="ASCII"?>
  &lt;display-name&gt;LocalDevelopment~LocalDevelopment~callchargecalc_ear~demo.sap.com&lt;/display-name&gt;
</application>
```

**Building and Deploying**

1. In the context menu of the `callchargecalculator_wm` and `callchargecalc_ear` nodes, choose `Development Component -> Build`.
2. In the dialog box that appears, choose **OK**.
3. In the context menu of the `callchargecalc_ear` node, choose `Development Component -> Deploy`.
4. In the dialog box that appears, choose **OK**.
5. Check the **Infrastructure Console** for messages.

**Running the Web Module**

1. Open the browser and enter the Application Server Address followed by the port number and the application name: **CallChargeCalc**.
2. Enter the following data in the respective fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>User Entry</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connection Provider</td>
<td>A</td>
</tr>
<tr>
<td>Destination Country</td>
<td>India</td>
</tr>
<tr>
<td>Origin Country</td>
<td>Germany</td>
</tr>
<tr>
<td>Type of Line</td>
<td>Line2</td>
</tr>
</tbody>
</table>
3. Choose **Submit**.

You should get the Call Charges (Connection Rate) as 80.0

You can find below the explanation for executing the **CallChargeRule**:

When the **CallChargeRule** is invoked, in the **CallChargeRule**:

1. The **Connection Provider** is compared with the Line A, and as it is satisfied, the CallChargeDT is executed. As the call is made from Germany to India, and in the CallChargeDT the **Connection Rate** from Germany to India is set to **100**.
2. Evaluate-Decision Table line is called, and **SetDiscountPercentage** decision table is executed. In this decision table, the discount percentage for the call from Germany to India is set to **20**.
3. The new Connection Rate is calculated as per

   \[ \text{Connection Rate} = \text{Connection Rate} - \left( \frac{\text{Connection Rate} \times \text{Discount Percentage}}{100} \right) \]

   and you get the new Connection Rate / Call Charges after discount as **80**.
Here is the snapshot of the web module:

Also try this:

1. Enter the following data in the respective fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>User Entry</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connection Provider</td>
<td>A</td>
</tr>
<tr>
<td>Destination Country</td>
<td>USA</td>
</tr>
<tr>
<td>Origin Country</td>
<td>Germany</td>
</tr>
<tr>
<td>Type of Line</td>
<td>Line1</td>
</tr>
</tbody>
</table>

2. Choose Submit.

You should get the Call Charges as 190.
You can find below the explanation for executing the CallChargeRule:

When the CallChargeRule is invoked, in the CallChargeRule:

1. The Connection Provider is compared with the Line A, and as it is satisfied, the CallChargeDT is executed. As the call is made from Germany to USA, and in the CallChargeDT the Connection Rate from Germany to USA is set to 200.

2. In the CallChargeRule, Evaluate-Decision Table : : SetDiscountPercentage is called and SetDiscountPercentage decision table is executed. In this decision table, the discount percentage for the condition satisfying destination country USA and connection rate greater than 100 is set to 5.

3. The new Connection Rate is calculated as per
   Connection Rate = Connection Rate – (Connection Rate * Discount Percentage) / 100, and you get the new Connection Rate / Call Charges after discount as 190.

Here is the snapshot of the web module:
Related Content
For more information, visit the Business Rules Management homepage.
How to Use Other Conditions in Decision Tables in Rules Composer

Copyright

© 2008 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, System i, System i5, System p, System p5, System x, System z, System z9, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, Informix, i5/OS, POWER, POWER5, POWER5+, OpenPower and PowerPC are trademarks or registered trademarks of IBM Corporation.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

MaxDB is a trademark of MySQL AB, Sweden.

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are either registered trademarks or trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

These materials are provided “as is” without a warranty of any kind, either express or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.

SAP shall not be liable for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials.

SAP does not warrant the accuracy or completeness of the information, text, graphics, links or other items contained within these materials. SAP has no control over the information that you may access through the use of hot links contained in these materials and does not endorse your use of third party web pages nor provide any warranty whatsoever relating to third party web pages.

Any software coding and/or code lines/strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, except if such damages were caused by SAP intentionally or grossly negligent.