

Building a Tax Calculation Application



Applies to:

Business Rules Framework plus shipped with **SAP NetWeaver 7.0 Enhancement Package 1**.

Summary

In this tutorial, you learn to model an application for calculating the final tax amount to be paid by an employee. The application processes the user input and calculates the tax amount to be paid. The function uses a decision tree expression as the top expression to calculate the tax amount.

Author: Orenthung Ovung

Company: SAP Labs India

Created on: 12 August 2008

About the Author



Orenthung Ovung is an Information Developer in the BRFPplus team. He has been part of this team since March 2008.

Table of Contents

Prerequisites	3
Learning Objectives	3
Procedure	3
Creating the Application	3
Creating the Function	3
Creating Formula Expressions	4
Creating Value Range Expressions	6
Adding Values to the Decision Tree Expression	7
Activating the Application and the Function	9
Simulating the Function	9
Related Information	10
Copyright	11

Prerequisites

- You have a basic knowledge of BRFplus.

Learning Objectives

- How to create a function
- How to create a decision tree expression
- How to create value range expressions and use them as conditions in the decision tree node
- How to create formula expression and use them as results in the decision tree node

Procedure

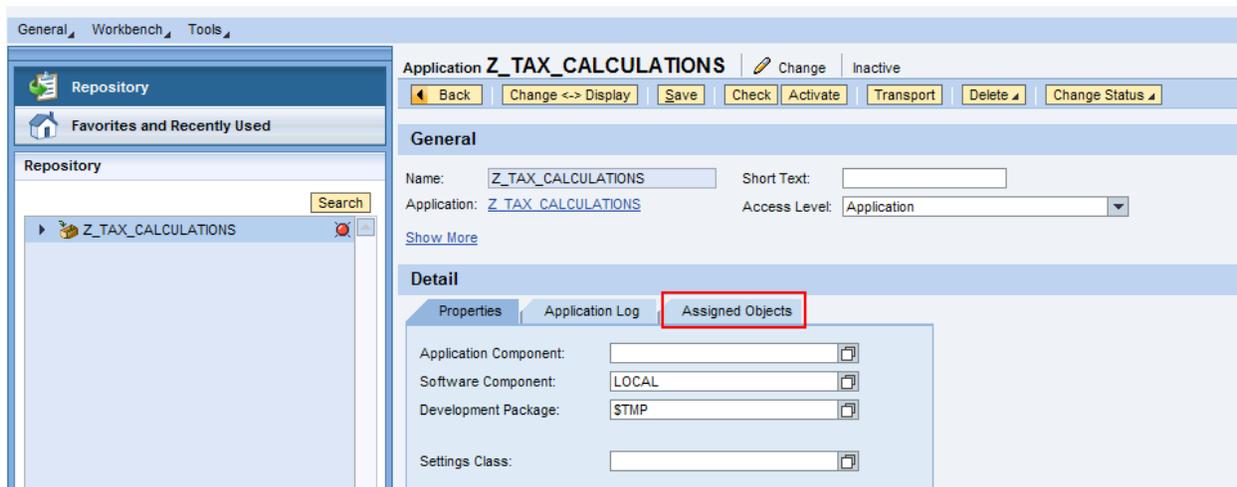
Creating the Application

- In the menu bar, choose *Workbench -> Create Application...*
- In the *Object Creation* dialog box that appears, enter **Z_Tax_Calculations** in the *Name* field.
- Choose *System* as the storage type and select the *Create Local Application* check box.
- Enter **\$TMP** in the *Development Package* field.
- Choose *Create & Navigate To Object*.
- Save the application.

Creating the Function

1. Create a Function

- In the *Object Manager* panel, under the *Detail* section, choose the *Assigned Objects* tab.



The *Assigned Objects* tab page appears.

- Choose *Function* from the *Type* field and choose *Create Object*.
- In the *Object Creation* dialog box that appears, enter **Tax_Calculator** in the *Name* field and choose *Create & Navigate To Object*.

The function is created and opens in the *Object Manager* panel.

2. Assign Expression to the Function

- a) Choose  (*Graphical Access*) next to the *Top Expression* field.
- b) In the context menu, choose *Create Expression...*
- c) In the *Object Creation* dialog box that appears, enter **Tax_Calculation** in the *Name* field and choose *Decision Tree* in the *Type* field.
- d) Choose *Create*.

3. Assign Context to the Function

- a) Choose the *Signature* tab.
The *Signature* tab page opens.
- b) Under *Context*, choose *Add new Data Object*.
- c) In the *Object Creation* dialog box that appears, enter **Gross_Salary** in the *Name* field.
- d) Choose *Amount* from the *Element Type* field.
- e) Choose *Create*.
The data object is added as the context.

4. Assign Result to the Function

- a) Choose  (*Graphical Access*).
- b) In the context menu, choose *Create Result Data Object...*
- c) In the *Object Creation* dialog box that appears, enter **Payable_Tax_Amount** in the *Name* field and **Amount to be paid** in the *Text* field.
- d) Choose *Amount* from the *Element Type* field.
- e) Choose *Create*.
The data object is added as the result data object.
- f) Save the function.

Creating Formula Expressions

You need to create formula expressions to calculate tax for different gross salaries. These formula expressions will be used in a decision tree.

1. To create the formula expressions, you have to navigate to the application in the *Object Manager* panel. To do so, choose the *Back* button.
2. Under the *Detail* section, choose the *Assigned Objects* tab.
The *Assigned Objects* tab page opens.
3. Choose *Expression* from the *Type* field and choose *Create Object*.
4. In the *Object Creation* dialog box that appears, enter **Calculate_tax_for_low_salary** in the *Name* field.
5. Enter **Amount to be paid** in the Short Text field.
6. Choose *Formula* from the *Type* field.
7. Choose *Create & Navigate to Object*.
You are navigated to the formula builder page.

Formula **CALCULATE_TAX_FOR_LOW_SALARY** Change Inactive

Back Change <-> Display Save Check Activate Transport Delete Change Status You Can Also

General

Name: Short Text:

This graphic is explained in the accompanying text.
 Application: Access Level:

[Show More](#)

Detail

Switch to Expert Mode Delete Formula Element

Result:

Formula:

Context		Functionals	
Name	Description	Functional	Functional Name
		ABS	Amount
		ARCCOS	Arc Cosinus
		ARCSIN	Arc Sinus
		ARCTAN	Arc Tangent
		CONCATENATE	Concatenate
		CONDENSE	Condense
		COS	Cosine
		COSH	Hyperbola Cosinus
		DIV	Quotient
		DT_DURATION_DIFF	Calculate duration between two TPs (returns Decimals)

8. Assign a result data object by choosing  (*Graphical Access*).
9. In the context menu, choose *Select Data Object*.
10. In the *Object Query* dialog box that appears, select *PAYABLE_TAX_AMOUNT* and choose *Select*.
It is added as the result data object.
11. Choose the (button.
12. In the context menu of the formula field, choose *Insert Data Object -> Select*.
13. In the *Object Query* dialog box that appears, select *GROSS_SALARY* and choose *Select*.
The data object gets added to the formula.
14. Choose the * button.
15. Choose *Number*.
16. In the *Insert* dialog box that appears, enter **0,1** and choose *Ok*.
The number appears in the *Formula* field.
17. Choose the) button.
18. Save the formula expression.

Create the following formula expressions using the above procedure.

Name of the Formula Expression	Result Data Object	Formula
Calculate_Tax_for_med_salary	Payable_Tax_Amount	25000 EUR * 0.1 + (Gross_Salary-25000 EUR)*0.2
Calculate_Tax_for_high_salary	Payable_Tax_Amount	25000 EUR*0.1+75000 EUR*0.2+((Gross_Salary-100000 \$EUR)*0.3

Creating Value Range Expressions

1. To create the value range expressions, you have to navigate to the application in the *Object Manager* panel. To do so, choose the *Back* button.
2. In the *Object Manager* panel, under the *Detail* section, choose the *Assigned Objects* tab.
The *Assigned Objects* tab page opens.
3. Choose *Expression* from the *Type* field and choose *Create Object*.
4. In the *Object Creation* dialog box that appears, enter **Gross_Salary_is_low** in the *Name* field.
5. Choose *Value Range* in the *Type* field and choose *Create & Navigate To Object*.
The value range expression appears in the *Object Manager* panel.
6. Define the incoming parameter by choosing  (*Graphical Access*).
7. In the context menu, choose *Select Context Data Object*.
8. In the *Object Query* dialog box that appears, choose *GROSS_SALARY* and choose *Select*.
The data object is added as the incoming parameter for the value range expression.

Detail

Show References

If **GROSS_SALARY**  (Type : Amount)

is equal to  

and unless it is... 

Then : **Return Value**  is true.Otherwise: **Return Value is false.**

9. Enter values for the incoming parameter.
Choose *is less than* as the range, enter **25000** as the amount and choose **EUR** as the currency.

Detail

Show References

If GROSS SALARY (Type : Amount)

is less than 25.000,00 EUR

and unless ...

then Return Value is true, otherwise **Return Value** is false.

10. Save the value range expression.

Create the following value range expressions using the above procedure.

Name of the Value Range Expression	Incoming Parameter	Values
Gross_Salary_is_Medium	Choose <i>Gross_Salary</i> data object.	Choose as value range <i>is between</i> and as the Amounts 25000 EUR and 100000 EUR

Adding Values to the Decision Tree Expression

- In the *Object Manager* panel, under the *Detail* section, choose the *Assigned Objects* tab.
- In the *Assigned Objects* tab page, choose *TAX_CALCULATOR*.
The function, *TAX_CALCULATOR*, opens in the *Object Manager* panel.
- Choose *TAX_CALCULATION* expression in the *Top Expression* field.
The decision tree opens in the *Object Manager* panel.

Detail

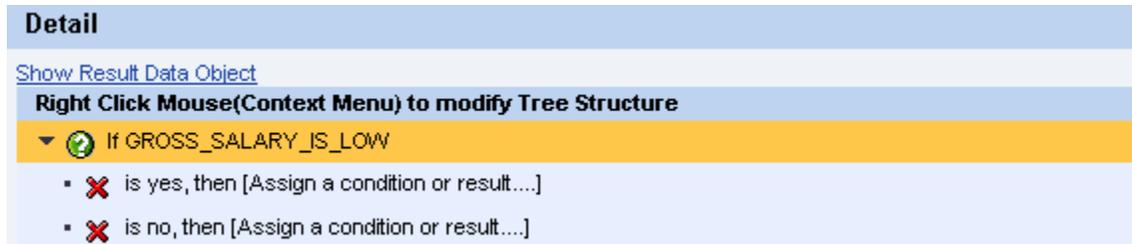
Result Data Object: <not assigned>

Right Click Mouse(Context Menu) to modify Tree Structure

If [Assign a condition...]

- Assign result data object.
 - Choose (Graphical Access).
 - In the context menu, choose *Select Result Data Objects -> More Result Data Objects...*
 - In the *Object Query* dialog box that appears, select *PAYABLE_TAX_AMOUNT* and choose *Select*.
The data object is added as a result data object.
- Assign a root node.
 - In the context menu of the node, choose *Node Actions -> Assign condition on node -> Select existing expression -> More Expression...*
 - In the *Object Query* dialog box that appears, select *GROSS_SALARY_IS_LOW* and choose *Select*.

The selected expression is set as the root node and two child nodes (positive and negative) appear under the root node as shown below.



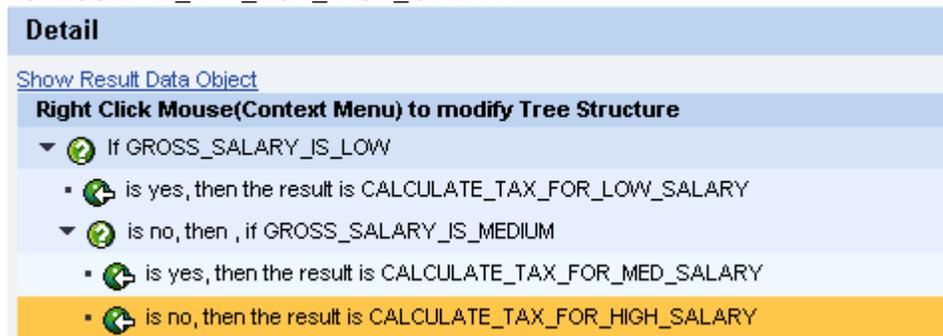
6. Add conditions and results to the child node.
 - a) In the context menu of the positive child node, choose *Node Actions -> Assign result on node -> Select existing expression -> More Expressions...*
 - b) In the *Object Query* dialog box that appears, choose *CALCULATE_TAX_FOR_LOW_SALARY* and choose *Select*.

The expression is added as the result of the child node.

- c) In the context menu of the negative child node, choose *Node Actions -> Assign condition on node -> Select existing expression -> More Expression...*
- d) In the *Object Query* dialog box that appears, choose *GROSS_SALARY_IS_MEDIUM*.
The expression is added as the condition of the child node.
- e) Similarly, add the following expressions as results to the decision tree.

CALCULATE_TAX_FOR_MED_SALARY

CALCULATE_TAX_FOR_HIGH_SALARY



7. Save the decision tree.

Activating the Application and the Function

1. Click the name of the application (**Z_Tax_Caculations**).
2. Choose *Activate* button.
3. In the *Confirmation of Activation* dialog box that appears, choose *OK*.
The application, *Z_TAX_CALCULATIONS*, gets activated.
4. In the *Object Manager* panel, under *Detail* section, choose *Assigned Objects* tab.
5. In the *Assigned Objects* tab page, choose *TAX_CALCULATOR*.
The function, *TAX_CALCULATOR*, opens in the *Object Manager* panel.
6. Choose *Activate* button.
7. In the *Confirmation of Activation* dialog box that appears, activate the *Include Referenced Objects* checkbox and choose *OK*.
The function gets activated.

Simulating the Function

1. Choose *Tools -> Simulation*. The simulation page is displayed.
2. Choose *Select Function*.
3. In the *Object Query* dialog box that appears, enter **Tax_Calculator** in the *Name* field and **Z_Tax_Calculations** in the *Application Name* field. Choose *Search*.
4. The function, *Tax Calculator*, appears in the table. Select the function and choose *Select*.
5. In the *Simulation* dialog box that appears, enter **35000** in the *Gross Salary* field and **EUR** in the *Currency* field.
6. Under *Simulation Mode* section, choose *Show only Result* and choose *Run Simulation*.
7. The amount to be paid as tax for the gross salary appears under the *Result* section.

Related Information

- BRFplus – The Very Basics
- Wikipedia, Business Rules: http://en.wikipedia.org/wiki/Business_rules
- Wikipedia, Business Rule Management System: http://en.wikipedia.org/wiki/Business_Rule_Management_System
- Carsten Ziegler, About Business Rules: <https://www.sdn.sap.com/irj/sdn/weblogs?blog=/pub/wlg/9713>
- Carsten Ziegler, BRFplus a Business Rule Engine written in ABAP, <https://www.sdn.sap.com/irj/sdn/weblogs?blog=/pub/wlg/8889>
- Carsten Ziegler, Important Information for Using BRFplus <https://www.sdn.sap.com/irj/sdn/weblogs?blog=/pub/wlg/11632>
- Rajagopalan Narayanan, Business Rules and Software Requirements, <https://www.sdn.sap.com/irj/sdn/go/portal/prtroot/docs/library/uuid/70c669d8-3ac2-2a10-0e96-c7c3786168f0>
- Rajagopalan Narayanan, Seven Tips for Your First Business Rules Project, <https://www.sdn.sap.com/irj/sdn/go/portal/prtroot/docs/library/uuid/201a9e3d-3ec2-2a10-85b2-ce56d276dd7a>
- Rajagopalan Narayanan, Real World Return of Investment Scenarios with Business Rules Management, <https://www.sdn.sap.com/irj/sdn/go/portal/prtroot/docs/library/uuid/b050905e-3cc2-2a10-979a-81a57a787f56>
- Rajagopalan Narayanan, Five Reasons to Build Agile Systems Using Business Rules Management Functionality, <https://www.sdn.sap.com/irj/sdn/go/portal/prtroot/docs/library/uuid/504486eb-43c2-2a10-f5a7-e84ef3fd45be>
- Rajagopalan Narayanan, How Business Rules Management Functionality Helps Tariff Plans Management in Transportation and Shipping, <https://www.sdn.sap.com/irj/sdn/go/portal/prtroot/docs/library/uuid/40a9cf69-40c2-2a10-8a8b-969fb311dd31>
- Rajagopalan Narayanan, Getting Started with Business Rules Management, <https://www.sdn.sap.com/irj/sdn/go/portal/prtroot/docs/library/uuid/70c669d8-3ac2-2a10-0e96-c7c3786168f0>

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 trademarks or registered 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.