Disclaimer

This presentation outlines our general product direction and should not be relied on in making a purchase decision. This presentation is not subject to your license agreement or any other agreement with SAP. SAP has no obligation to pursue any course of business outlined in this presentation or to develop or release any functionality mentioned in this presentation. This presentation and SAP's strategy and possible future developments are subject to change and may be changed by SAP at any time for any reason without notice. This document is provided 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 assumes no responsibility for errors or omissions in this document, except if such damages were caused by SAP intentionally or grossly negligent.
Agenda

- Key Trends and Issues
- SAP Solution Overview
- Exercises
Key Trends and Issues
Key Trends and Issues Driving Change in Application Development

**Flexibility at low cost**
- Fast time from initial idea to realized solution
- Decrease dependency on highly specialized IT resources
- Allow for ongoing optimization and adaption of business logic

**Transparency**
- Transparent business logic and repository for business rules
- Increased demand for governance, risk, and compliance / policy enforcement

**Business user empowerment**
- Enable business user to adapt application to constantly changing environment w/o IT expertise
- Role specific user experience
Problems with Conventional Methods

- Rules inside Database: Triggers, Stored Procedures
- Rules as Tacit Knowledge: Business User, Business Analyst, Business Expert
- Rules inside Code: Example code snippet
- Rules in Documents: Costs, archiving, backups, compliance

Agility, visibility for business, changes?
Business Applications, Business Processes, and Business Rules

Application
- Completeness check
- Data validation
- No-go criteria

Credit Score
- Calculation of credit score
- Enable/disable contract terms

Approval
- Automatic/manual approval
- Compliance rules
- Risk management

Business Application Release 1
Business Application Release 2

Process Version 1
Process Version 2
Process Version 3

Rules
Vers.
1
Vers.
2
Vers.
3
Vers.
4
Vers.
5
Vers.
6
Vers.
7

Business Application
- Banking

Business Process
- Loan Approval

Business Rule
- Calculation of credit score
Benefits of Using Business Rules Management

<table>
<thead>
<tr>
<th>Business Rules Management</th>
<th>Benefit</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Separation</strong> of business rules and code</td>
<td>Agile and maintainable systems</td>
</tr>
<tr>
<td><strong>Modeling</strong> of complex business logic</td>
<td>Automation of key business decisions</td>
</tr>
<tr>
<td>Central <strong>management</strong> of business rules</td>
<td>Consistent application of corporate policy and government regulations</td>
</tr>
<tr>
<td><strong>Business Experts</strong> can find, change and test critical business rules</td>
<td>Visibility, accountability and error-free business</td>
</tr>
</tbody>
</table>

**Business Rules Management**
- Flexible & agile business systems
- Faster turnaround times for changes
- Reduced cost for changes
- Precise & informed decision making
- Improved visibility for the business
SAP Solution Overview

SAP Business Rule Framework plus
BRFplus is a Best of Breed Rules Engine

**Business User Empowerment**
Graphical approach and business semantics for intuitive rule composition and management (modeling rather than programming)

**Rapid Time-to-Value**
Rapid prototyping with feedback-based build cycle (build, simulate, optimize) leading to reduced project durations (up to 60% reduction)

**Transparency**
Transparency of rule execution and tracing for audit

**Personalization**
Personalization and segmentation of rules
BRFplus can do Much Better Than Other Rules Engines

- **Business Semantics**
  - Business semantics out of the box

- **Part of ABAP Stack**
  - Seamless integration with ABAP applications and ABAP server including security and authorization

- **Low TCO**
  - BRFplus comes for free – it is covered by the general SAP license

- **No Hardware**
  - No additional hardware required: BRFplus is part of SAP NetWeaver and runs on the same server

- **High Performance**
  - High performance and nearly no footprint (<100 kB)

- **No Setup**
  - Zero initial setup effort
What is in from a Development Perspective?

- **Backend Call**
  
  Easy to call – 6 lines of ABAP or RFC or Webservice

- **Support**
  
  Supported together and in line with SAP NetWeaver

- **Extensibility**
  
  BRFplus can be extended by additional expression and action types

- **Integration**
  
  Fully integrated with DDIC, ABAP data base, backend routines, authorization, security, transportation management

- **User Interface**
  
  API allowing for role specific User Interface

- **No Setup**
  
  Zero initial setup effort

- **Timeless Software**
  
  Allows to build timeless, flexible, and customizable business applications with less code lines and efforts
Business Rule Services with BRFplus

Web Service 
RFC 

Code Exit 
BADI 

Application

Business Rules Service

BRFplus Function: Service Definition
Input (Context), Output (Result), Purpose
Can be called by ABAP API, RFC, or Web Service

BRFplus Rulesets: Service Implementation
Collection of rules
Preconditions, priorities, validity
Context enrichment (additionally needed data)

BRFplus Rule: step implementation
IF <condition(s)>
    THEN <operation(s)>
    ELSE <operation(s)>
Usage of expression types (decision tables, decision trees, formulas, table operations, …) to define complex conditions and value derivations/calculations
Usage of action types (message log, workflow, email, …) to trigger side effects
BRFplus Impressions

Function
BRFplus Impressions
Ruleset
BRFplus Impressions
Expression Type Decision Table
BRFplus Impressions
Expression Type Formula
BRFplus Impressions
Simulation
BRFplus Demo
Exercises
Exercises

In the exercises we want to set up a sample application (exercise 1) that derives the price of some product (exercise 2), adds shipping costs of 5%, but at least 2$, if the price is under $100 (exercise 3) and sends a notification mail if a product is not found in the price list (exercise 3) Then we test the application (exercise 4) Finally we implement the function call in backend (exercise 5)
Login Information

- Please log on to system M53, client 100
- Your user is BRFPLUS_<nn>, where nn is the number of your seat, e.g. BRFPLUS_07
- The password is “Hands-On” (case sensitive)
- That system is a SAP NetWeaver 7.31 system
- In Ramp-Up, starting in Q4/2011, we will ship a later SP level
- Be aware that that system is a preliminary version
- Please keep that in mind in case you experience any issues
System Information

- If system M53 is not yet available in SAP Logon, you need to create a respective connection.
  - To do so click the New icon → Connection
  - Select User Specified System
  - Click Next
  - Enter the following values
    - Description: M53
    - Message Server: wdflbmt0752.wdf.sap.corp
    - Group/Server: SPACE
    - Instance Number: 10
    - System ID: M53
  - Click Finish

- Alternatively you can access the BRFplus Workbench directly by entering the following URL to your internet browser:
Exercise 1

Create an Application, Function, and Ruleset
Exercise 1
Create an Application, Function, and Ruleset

- Call the BRFplus Workbench
- Create a new Application
- Create a Function assigned to that Application
- Define the Signature of the Function
  - Input parameter is the product name
  - Output parameter is the price
- Create a Ruleset assigned to the Function
Exercise 1
Walkthrough 1/6

- Call the BRFplus Workbench
  - Call transaction BRFPLUS
  - Click icon Personalize on the upper right of the screen and select Show in Change Mode as Viewing Mode. Then click Save. By doing so, you do not need to change to edit mode in every single screen

- Create a new Application
  - Click Workbench → Create Application …
  - Enter Name, Short Text, and Text, e.g. TechEd <your name> (you can use the same for all three fields, if the text is short enough)
  - Click Create Local Application
  - Click Create and Navigate to Object
  - Click Activate and confirm the popup window with Activate
Exercise 1
Walkthrough 2/6

- Create a Function assigned to that Application
  - Select tab *Contained Objects* and click *Create Object*

- Enter Name, Short Text, and Text, e.g. *Product Price*
- Click *Create and Navigate to Object*
Exercise 1
Walkthrough 3/6

- Define the Signature of the Function
  - Select the Signature tab
  - Click *Add New Data Object* → *Add New Data Object*

- Select Type *Element* and click *Create*
- Enter Name, Short Text, and Text, e.g. *Product*
- For Length select a reasonable text length for product names, e.g. 40
- Click Create and Navigate to Object and confirm the popup window with Yes to save your input
Exercise 1
Walkthrough 4/6

- Select tab Domain Values
- Click Create Values

- Enter Name, Short Text, Text, and Value, e.g. *Tennis Racket* and click Create
- Repeat that previous step for some more products
- Click *Back* and confirm the popup window with *Yes* to get back to your function
Exercise 1
Walkthrough 5/6

- Make sure you are in the Function editor
- At Result Data Object click on the icon right of the word Actions and select Create ...

- Select Type Element
- Enter Name, Short Text, and Text, e.g. Price
- Select Element Type Amount
- Enter a reasonable number of digits for a price including the defaulted 10 decimal places, e.g. 15
- Click Positive Values Only
- Click Create
Exercise 1
Walkthrough 6/6

- Create a Ruleset assigned to the Function
  - Select tab Assigned Ruleset
  - Click Create Ruleset

- Enter Name, Short Text, and Text, e.g. Product Price
- Click Create and Navigate to Object and confirm the popup window with Yes
- Click Save
- You have now created the framework of your business rule. What is left is to create the actual rules themselves in the following exercises
Exercise 2
Define a Simple Rule
Exercise 2
Define a Simple Rule

- In the Ruleset defined in exercise 1, create a Rule using a Decision Table
  - The decision table shall return the price for a specific product
  - In case there are several entries for the same product, the first one is taken into consideration
  - If no matching entry is available, an initial value is returned
Exercise 2  
Walkthrough 1/4

- In the Ruleset defined in exercise 1, create a Rule using a Decision Table
  - Ensure you are in the Ruleset editor for the Ruleset you created in the last exercise
  - Click Insert Rule → Create …

- Enter a Description, e.g. Look up product prices
- At the Then branch click Add → Process Expression → Create …
Exercise 2
Walkthrough 2/4

- Select type *Decision Table*
- Enter Name, Short Text, and Text, e.g. Product Prices
- Select Price (the name of the data element of type Amount that you created in Exercise 1 above)
- Click *Create and Navigate to Object* and confirm the popup window with *Yes*

- Click *Table Settings*
Exercise 2
Walkthrough 3/4

- Click *Insert Column* → *From Context Data Objects* …

- Select Product (the name of the data element of type Text that you created in Exercise 1 above) and click *Ok*
- Click *OK*
Exercise 2
Walkthrough 4/4

- Click Insert *New Row*
- Click on … in the first column of that new row
- Enter a product name in the empty field, e.g. Tennis Racket (you can select products you created before by clicking on the icon right of it)
- Click OK
- Click on … in the second column of the same row
- Enter a price and a currency, e.g. USD for US Dollars (you can select a valid currency by clicking on the icon right of the currency field) and click OK. We are using US Dollars in this example. If you are going for another currency you need to replace US Dollars with that currency at all later occurrences
- Repeat the last six steps (from “Click Insert New Row”) for several products. Also include at least one product more than once with different prices
- Click *Back* and confirm the popup window with *Yes* (now the Ruleset you created before should be selected)
- Click *Save*. You have now created a complete BRFplus application. In the next exercise you will extend that application
Exercise 3
Define More Complex Rules
Exercise 3
Define more Complex Rules

- Add other Rules to the existing Ruleset using Formulas and Actions (e.g. Sending an Email)
  - Derive the shipping costs (5%, but at least $2, if the price is under 100$, nothing if the price is 100$ or more) using a Formula and add them to the price
  - If the product was not found in the price list (price is set to initial value), trigger a notification mail to a fixed email address
Exercise 3
Walkthrough 1/5

- Add other Rules to the existing Ruleset using Formulas and Actions
  - In the Ruleset at the right side of the Rule we already created click *Other Operations → Insert Next Rule by ... → Create ...*
  - Enter a description, e.g. *Add Shipping Costs*
  - Click on *Assign Condition ... → Use Value Range From ... → Context → Price* (resp. the name of the result data object you created in exercise 1)
  - Select condition *is greater than*
  - Enter the currency USD for US Dollars (you can select a valid currency by clicking on the icon right of the currency field). The amount is 0, which is prefilled anyway.
Exercise 3
Walkthrough 2/5

■ At the Then branch click Add → Process Expression → Create …

■ Select type Formula
■ Enter Name, Short Text, and Text, e.g. Add Shipping Costs
■ As a result data object, select Price
■ Click Create and Navigate to Object and confirm the popup window with Yes
Exercise 3
Walkthrough 3/5

- By using context elements, operators, and formula functions offered at the lower part of the screen enter the formula
- Price + IF ( Price >= 100_USD , 0_USD , MAX ( 2_USD , Price * 0.05 ) )

- Note: a better way to implement that "IF" condition is to create a Rule with respective condition. In this example we put all that into a formula to show the capabilities of the formula editor
- Click the Check button to make sure that the formula is consistent
- Click Back and confirm the popup window with Yes (now the Ruleset should be selected)
Exercise 3
Walkthrough 4/5

- Click the Edit Rule icon (pencil) at the second rule

- At the Else branch click Add → Perform Action → Create …

- Select type Send Email (Act)
- Enter Name, Short Text, and Text, e.g. Notification Mail
- Click Create and Navigate to Object and confirm the popup window with Yes
At the Recipient(s) field enter an arbitrary email address (we are not going to actually send the mail. So it does not matter, if the address is valid or not)

Enter some Subject, e.g. **Notification: Product not available**

Enter the body text, where you replace the product name by &1, e.g. **The product &1 does not exist. Regards, your Sales Team**

Click **Refresh Message Placeholders**

Click on the icon right of Variable 1 and select **Context → Product**

Click **Save.** You have now finished modeling your business rule! The next exercise will guide you through activating, executing, and testing your rule.
Exercise 4
Run the Rules
Exercise 4
Run the Rules

- Activate all created objects
- Run Function in Simulation Mode
  - Enter different products, including products that are not listed or are listed more often than once, and watch the results
  - Check the intermediate steps of rules processing
Exercise 4
Walkthrough 1/4

- Activate all created objects
  - In the repository tree on the left select the Ruleset you created before
  - Click Activate

- Make sure that the checkbox *Include Referenced Objects* is set
- Confirm the popup window with *OK*

- On the upper left of the screen you should see a success message *Objects activated* and all objects you created should show a green icon in the repository tree, indicating they are active
Exercise 4
Walkthrough 2/4

- Run Function in Simulation Mode
  - In the repository tree on the left select the Function you created before
  - Click Start Simulation
  - Click Continue

- Enter a product (you can select products you created before by clicking on the icon right of the input field)
- Click Execute and Display Processing Steps
Exercise 4
Walkthrough 3/4

- Notice the resulting price at the top of the screen
Exercise 4
Walkthrough 4/4

- Notice the intermediate rules processing steps:
  - Looking up the price in the decision table
  - If a product does not exist, the price is set to the initial value zero
  - If a product is listed more often than once, the price of the first occurrence is returned
  - Adding shipping costs if the price is not zero
  - Triggering a notification mail otherwise

- Click Back to Simulation to run the simulation with another product
- Repeat the simulation with products that are not listed in the decision table
- Repeat the simulation with products that are listed in the decision table more often than once
- Congratulations! You have completed the standard part of the BRFplus Hands-On workshop! If you still have time left and are an ABAP programmer, please proceed to the next exercise.
Exercise 5
Call the Function
Exercise 5
Call the Application

- Call the Function from Backend
  - Implement an ABAP report that calls the BRFplus Function and displays the result

- That exercise is intended for people who are familiar with ABAP programming
- If you are not an ABAP programmer or don't have any time left please skip the exercise
Exercise 5
Walkthrough 1/2

- Call the Application from Backend
  - Call the ABAP editor (transaction SE38)
  - For your convenience we already created a report ZZ_CALL_TECHED_EXERCISE you can use as a template
  - Copy that report to a local object and name it ZZ_
  - In line 17 of the report replace the Function Id by the Id of the function of your application
  - You find that Id at the section General, tab General of the function in the BRFplus Workbench
  - If it is not expanded, click the respective icon at the upper right
  - Save and activate the report
  - Run the report
  - When calling the BRFplus Function for the first time or after changing it, an automated code generation is triggered
  - For all further calls that generated code will be executed
Exercise 5
Walkthrough 2/2

This is the coding of our template report:

```
REPORT  ZZ_CALL_TECHED_EXERCISE.

TYPES BEGIN OF price_type.
  TYPES number TYPE decfloat16.
  TYPES currency TYPE string.
TYPES END OF price_type.
DATA: lo_admin_data TYPE REF TO if_fdt_admin_data,
  lo_function TYPE REF TO if_fdt_function,
  lo_context TYPE REF TO if_fdt_context,
  lo_result TYPE REF TO if_fdt_result,
  lx_fdt TYPE REF TO cx_fdt,
  price TYPE price_type,
  product TYPE string.

product = 'Tennis Racket'.
cl_fdt_factory=>get_instance_generic(
  EXPORTING iv_id = '005056b4164a1ee0a28df43ad91f0553'
  IMPORTING eo_instance = lo_admin_data).
lo_function ?= lo_admin_data.
lo_context ?= lo_function->get_process_context().
lo_context->set_value( iv_name = 'PRODUCT'
  ia_value = product ).

TRY.
  lo_function->process( EXPORTING io_context = lo_context
    IMPORTING eo_result = lo_result ).
  lo_result->get_value( IMPORTING ea_value = price ).
  WRITE: 'The price of', product, 'is',
    (6) price-number DECIMALS 2,
    price-currency NO-GAP, '.
  CATCH cx_fdt INTO lx_fdt.
ENDTRY.
```
You Have Done It!

Congratulations!
You have completed the BRFplus Hands-On workshop!
Findings of the Exercises

Within one hour we set up an application
- that is business driven and is based on a business model
- that checks for product prices and availability
- that applies context sensitive logic
- that triggers a follow up action
- that can be adapted or extended at any time
- that may have role based authorization
- that is fully integrated with the backend
- without one line of coding
- modification free
- without any previous training or upskilling
- that can be called from backend very easily
Summary

**Business User Empowerment**
Graphical approach for intuitive rule composition and management (modeling rather than programming)
- Transparency of rule execution and tracing for audit
- Personalization and segmentation of rules (object catalogs, filters, personal configuration, etc.)

**Rapid Time-to-Value**
Rapid prototyping with feedback-based build cycle (build, simulate, optimize) leading to lower project cycle times (up to 60% reduction)
- Business users can work with the rules in business semantics while IT experts can technically modify the rules.
- Both business and IT can work in the same environment.

**Zero Administration**
Seamless integration with ABAP applications and ABAP server (ABAP, DDIC, Change and Transport System, Web Service and RFC Generator)
- High performance and nearly no footprint (<100 kB)
- Zero setup effort before rule modeling (no installations, no plug-ins, no business vocabulary setup, no additional sizing, browser-based UI)
Further Information

**SAP Public Web**

BRFplus in SDN: https://www.sdn.sap.com/irj/sdn/index?rid=/webcontent/uuid/90754865-f283-2b10-6d9f-b10f3c28c3a0

**Media**


**Related Workshops/Lectures at SAP TechEd 2011**

- PMC101 Overview of SAP NetWeaver BPM and SAP NetWeaver BRM
- PMC103 Business Rules Management with SAP: BRFplus and SAP NetWeaver BRM
- PMC265 Accelerating Business Rules with SAP NetWeaver BRM
- PMC-P06 Pod Business Rules Management with SAP
- EXP45/46 Expert Session TCO of Business Rules Management
- EXP57 Expert Session BRFplus - Preview on new features
- EXP59 Expert Session How to do a organize and deliver business rule project (with BRFplus)
Feedback

Please complete your session evaluation.
Be courteous — deposit your trash, and do not take the handouts for the following session.

THANK YOU!
Thank You!