Enhancement Technique: How-To-Guide on the usage of Validations

Applies to:
SAP 4.6C and higher

Summary
This article provides a step-by-step guide on the usage of Validations, an Enhancement technique in the Financial Accounting Module of the SAP R/3 system.

Author(s): Lakshman Tandra
Company: Intelligroup Asia Pvt Ltd
Created on: 24 April 2007

Author Bio
Lakshman Tandra has been working in Intelligroup from more than 10 years. He is member of the SAP NetWeaver/ESA core team and currently oversees competency building activities at Intelligroup.
Table of Contents

What are Validations ........................................................................................................................ 3
Basic Steps in Configuring Validations ............................................................................................ 3
Example Business Scenario for Validations .................................................................................... 7
Related Transaction Codes ........................................................................................................... 19
Disclaimer and Liability Notice ....................................................................................................... 20
What are Validations?
Validation allows you to define your own individual checks for specific fields when a business transaction is being processed.

A validation consists of three parts:
• Prerequisite
• Check
• Message

If the prerequisite is met (if it is TRUE), a check is performed. If the check statement is FALSE, the system returns a message.

Basic Steps in Configuring Validations
You must complete the following steps to create a validation:
1. Enter the application area in which the validation is called up.
2. Select the correct Callup point for the validation.
3. Define the validation.
4. Assign your validation to an organizational unit (for example, company code for FI, company code or global company for FI-SL, controlling area for CO) and
5. Activate the Validation. The fields that can be used in your validation or substitution are determined from the application area and the Callup point. When you define your rules for the prerequisite and check statements, you can choose the fields that you want to use.

Call Up Point:
Callup Point 1
• The Callup point tells you when the substitution is performed.
• Callup point 1 checks the fields in a document header.

Callup Point 2
• Callup point 2 is used to check the fields in a document line for a G/L account posting or a FI-SL.
• The fields that are available at Callup point 1 are also available for cross-checks.

Operands and Operators
The following elements are used to create rules for prerequisite statements in substitutions.
• Operands: Logical operators (Boolean logic) and Comparison operators.

User Exits in Validations
• User exits are form routines programmed by the user.
• If you want to use your own rule, you can use user exits to define prerequisites or checks for validating or substituting fields.
• Exit routines are created in a report. Each client has its own report. The form pools for the report are stored in table T80D.
  • User exits are user-defined FORM routines that are used to calculate and/or replace values within a validation, substitution, or rule.
  • User exits have the following format: U (for a user-defined user exit) or S (for a standard user exit)

The user exit number (three digits)
• For example, U301 is a user-defined user exit.
• You can configure the form pool name of the user exits and must store it in the table for client-dependent user exits (T80D) in Customizing. Table T80D contains the form pool
names for the user exits used in validations, substitutions, and rules. Each validation/substitution form pool is client-dependent. (For more information, refer to the Maintain Client-Dependent User Exits activity in the Implementation Guide (IMG) for Special Purpose Ledger.)

• Example form pools RGGBS000 and RGGBR000 for client 000 are delivered with the SAP R/3 System. You must copy these form pools and configure them in T80D. The new form pool name should conform to the customer naming convention (beginning with the letter Z or Y) so that it is not overwritten when the next SAP upgrade is installed (for example, YGGBR000).

• Example FORM routine for validation exit 003 (U301).

```java
IF SY-DATUM = BKPF-BUDAT.
  B_RESULT  = B_TRUE.
ELSE.
  B_RESULT  = B_FALSE.
ENDIF.
```

• Tables and structures should not be declared in the FORM routines so that the contents can be used together with the calling transaction.

• SAP exits are FORM routines that have been programmed by SAP. The name of the form pool for SAP exits is SAPFGBEB.

The following table shows the types of user exits that can be used in substitutions.

<table>
<thead>
<tr>
<th>User exit type</th>
<th>Description</th>
<th>Application</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>No parameters are defined for the user exit.</td>
<td>Rules, validations, and substitutions (prerequisite)</td>
<td>See form pool RGGBR000, parameter type C_EXIT_PARAM_NONE</td>
</tr>
<tr>
<td>2</td>
<td>Same as user exit type 1, except one parameter (the field to be substituted) is defined in the user exit. For example, you can create a substitution routine that analyzes the cost center irrespective of the used field.</td>
<td>Substitutions</td>
<td>See form pool RGGBS000, parameter type C_EXIT_PARAM_FIELD</td>
</tr>
<tr>
<td>3</td>
<td>All data is passed as one parameter; this exit type can only be used in matrix validations and substitutions.</td>
<td>Rules, validations, and substitutions (prerequisite)</td>
<td>See form pool RGGBR000, parameter type C_EXIT_PARAM_CLASS</td>
</tr>
</tbody>
</table>

• Validations and rules use exit numbers 1 and 3 from the above table.

• If you want to define a parameter for your user exit that is different from the result of a validation (B_RESULT), you must make an entry for your user exit in the FORM routine GET_EXIT_TITLES in the form pool you defined. It is recommended that you copy the SAP example form pool RGGBR000 for validation exits or RGGBS000 for substitution exits. These example form pools already contain entries in the FORM routine GET_EXIT_TITLES for the examples delivered by SAP.

For each validation step:

• Data is entered into the SAP R/3 System.

• Data is sent to the FI-SL Integration Manager and to validations. Validations are part of the Integration Manager along with substitutions, ledger selection rules, and summation rules.

• Data is then checked against a prerequisite statement.
If the prerequisite statement is true, the data is checked against the check statement. If the prerequisite statement is false, the data is valid.

- If the prerequisite statement is true and the check statement is:
  - True, then the data is valid
  - False, then the system displays a message

You need to correct E (error) type messages before the value you are validating can be posted. You should also correct W (warning) type messages. A (abend) type messages result in termination of validation processing.

- If there are additional steps in the validation, the system accesses the next step for validating the data until all validation steps have been processed.

**Formula Builder:**

The Formula Builder provides an easy-to-use interface for inputting arithmetical and logical expressions.

Depending on the context for the formulation required, you are given buttons to press which provide Operands and Operators for logical expressions which have to be inserted in the formula.

Your entry is checked step by step for correct syntax including parentheses.

Depending on your entries the system will only allow you to dispose of the correct logical formulation to guarantee a fully correct statement. Errors in entering your rules are therefore scored out.

The use of technical names, as was the case in the past (tables and field names) is still supported, but here the options you can use are delivered using press buttons and texts. This enhances the comfort of handling a logic which is intrinsically technical.

**Validation Maintenance:**

- Each validation can contain up to 999 steps that each consist of a
  - Prerequisite
  - Check
  - Message

- Prerequisites and checks contain the same fields and have the same field processing options.

- You can use a message that has already been defined or you can create a new message for the validation.

- You can include up to four fields in a message. The '&' characters in the message are placeholders for the fields to be output.

- The messages to be output can have different levels of severity:
  - I = Information
  - W = Warning
  - E = Error (the user must correct the entry)
  - C = Cancel
Validation Activation:

- A validation is activated for each of the following:
  - an ORGANIZATIONAL UNIT (in FI-SL: Company code or global company)
  - a CALLUP POINT (in FI-SL: 0001 (document header) or 0002 (document line))
  - A validation can be valid for several company codes / global companies at the same time.
Example Business Scenario for Validations

Company A would like to have a validation which should allow for creation of an accounting document with any document types except the document type SA and the Posting date should be equal to the syst-datum (Using Exit) in the FI Document when posting an accounting document using the tcode FB01.

Go to GGB0 transaction:

Click on the Documents Header of the Financial Accounting and Create a Validation
After that Create a Step by Clicking on the Step button on the toolbar
Double click on Prerequisite and give the prerequisite. For our example we will give the syst-uname: Double click on the structure SYST and select the syst-uname field.
Select the syst-uname field. Using the Formula builder, give a constant value for Username as 'NTWDEV'. This validation will get execute only if the prerequisite is True. (i.e., Validation will trigger only for the User ‘NTWDEV’)

Double click on Check under the Step created above and select the field from the structure BKPF this time. Say for our example use BKPF-BLART and click on the button ‘<>’ in the formula editor. After that click on the Constant button and give the value ‘SA’.
Double click on Message and give the required message variables. Say for our example BKPF-BLART

After we complete the Prerequisite, Check Condition and the Message under the validation the Screen should like below
After that go to OB28 (to assign the validation to the Company code and for activation) and create a new entry by specifying the Company Code, Call point, Validation and Activation Level and Lock in a transport request.
Create an Accounting document using FB01 with the Document type SA:

![Image of SAP GUI screen showing the creation of an Accounting document with Document type SA]

- **Document Date**: 04/18/2007
- **Company Code**: 1000
- **Document Number**: TEST - USE ONLY
- **Posting Date**: 04/18/2007
- **Reference**: TEST - USE ONLY
- **Doc. Header Text**: Test - Doc. Header Text

**Description of the Error**:

The system displays an error message indicating that the Document type SA is not allowed with standardizing entries.

**Message**: Document type SA not allowed with standardizing entries

**Message number**: Z1000
To use the Exits in validations we have to do the following:
Copy the program RGGBR000 into YGGBR000 and lock in a request.

Create an entry in the program we have created above in the form GET_EXIT_TITLES and then create a form with the same name. It should look like below

Say for our example we will write the code as below. It should check and allow to create an entry only if the Syst-datum and the Posting data (BKPF-BUDAT) are equal.
Create an entry in the table T80D using the view V_T80D using the Tcode SM30. Note: If already there is a program existing for the application we can create an exit in the same program. Since the application area is the key field in table T80D it will allow only one program per application area.

Save the entry and Lock it in a transport request.
After that again go back to the validation what we have created above:

![Change check: YVAL Step 001](image1)

Give the required message variables.

![Change Validation: YVAL - Step 001 - Message](image2)

After we complete the Prerequisite, Check Condition and the Message under the validation the Screen should like below:
Create an accounting document using the transaction code FB01 by giving posting date other than syst-datum. (Generally it depends on the business requirement. To make our exercise simple we will compare the Syst-datum with the posting date)
Enhancement Technique: How-To-Guide on the usage of Validations

Post Document for GL: Header Data

Document type 04/19/2007 not allowed with standardizing entries

Document type 04/19/2007 not allowed with standardizing entries

Message no. 21100
Related Transaction Codes

- OB28 : C FI Maintain Table T001D
- GGB0 : Validation Maintenance
- SM30 : Call View Maintenance
Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.