1156325 - BAdIs in the Logistics Invoice Verification environment

Version	3	Type	SAP Note
Language	Inglés	Master Language	Alemán
Component	MM-IV-LIV-BADI (BADI)	Released On	08.12.2008

Please find the original document at https://launchpad.support.sap.com/#/notes/1156325

Symptom

This note lists known Business Add-Ins (BAdIs) freely available to customers (at the time of creating this note) in the Logistics Invoice Verification area from SAP R/3 Enterprise Release 4.70 up to and including SAP ERP Enhancement Package 3 for SAP ERP 6.00.

Since new BAdIs may be developed at any time, we cannot make any guarantees about how current or complete this note is. This note is simply intended as an aid when searching for the relevant BAdI.

Other Terms

FAQ, MRMBADI, MRM_BADI, SE18, SE19

Reason and Prerequisites

Solution

BAdIs as of Release 4.70:

Application component: MM-IV
Package: MRM

Definition name: INVOICE_UPDATE

Usage
The BAdI 'INVOICE_UPDATE' was only designed to execute checks during document entry or document posting in transaction MIRO.

Therefore, it has only import parameters. There are no
export parameters. This means that the data changed in the BAdI
is not processed further in the standard program MIRO.

© 2016 SAP SE or an SAP affiliate company. All rights reserved

More information
Note that no system messages can be sent in the methods. The method CHANGE_AT_SAVE is an exception. A system message can be issued in this method if the exception ERROR_WITH_MESSAGE is triggered at the same time.

DO NOT ENTER THE STATEMENTS "COMMIT-WORK" or "ROLLBACK" SINCE THIS CAN CAUSE DATA INCONSISTENCIES ON THE DATABASE.

Application component: MM-IV-LIV
Package: MRM_BADI

Definition name: MRM_HEADER_CHECK

What does the BAdI do?
Along with the checks in the standard system when entering invoices in the Logistics Invoice Verification, you can use the BAdI MRM_HEADER_CHECK to execute customer-specific checks for header and item data of an invoice document.

In what business and technical context is this BAdI used?
You can use this BAdI to check the header data of an incoming invoice according to customer-specific criteria. Depending on the checks, the system may, for example, issue an error message or call other transactions.

When is the BAdI called?
The call occurs via the PAI module HEADERDATA_CHECK in the flow logic of screen 6000 in the program SAPLMR1M. This module contains the call of the auxiliary function module MRMBADI_HEADER_CHECK, which in turn contains the call of the BAdI MRM_HEADER_CHECK.

How does the BAdI affect the application?
The header data is provided for the BAdI. In addition, item data known at the time of the call can also be transferred. Note that you cannot change the data with this BAdI, rather you can only check it; that is, the BAdI receives the data but does not return data to the application.

Example
You can perform an enhanced check on duplicate invoices. Along with the standard criteria and the criteria that can be set in Customizing for the Logistics Invoice Verification that are used for the check on duplicate invoices, you can use this BAdI to check other header data.

More information
Documentation for the BAdI interface IF_EX_MRM_HEADER_CHECK

Definition name: MRM_HEADER_DEFAULT

What is the behavior like in the standard system?
In transaction MIRO "Enter Invoice", MIRA "Enter Invoice for Invoice Verification in the Background" and MIR7 "Park Invoice", you cannot determine the default values for the header data individually. For example, the document type can only be set in Customizing depending on the transaction code; only one setting per transaction is possible.

What does the BAdI do?
You can use the BAdI MRM_HEADER_DEFAULT to define different default values per transaction for the header fields of an incoming invoice.

In what business and technical context is this BAdI used?
You can use the BAdI MRM_HEADER_DEFAULT when you enter an incoming invoice, to define your own default values for header fields (for example, document type, header text, document date). In this way, you can specify the document type depending on the company code or transaction.

Within the module MRMBADI_HEADER_DEFAULT, the module SXC_EXIT_CHECK_ACTIVE is used to check whether there is an active implementation for the BAdI; if there is, the method HEADER_DEFAULT_SET is called.

The system performs various consistency checks after the header fields are returned from the method HEADER_DEFAULT_SET.

More information
Documentation for the BAdI interface IF_EX_MRM_HEADER_DEFAULT

Definition name: MRM_MRIS_HDAT_MODIFY

What does the BAdI do?
You can use the BAdI MRM_MRIS_HDAT_MODIFY to change different header fields in transaction MRIS "Invoicing Plan Settlement with Logistics Invoice Verification" when posting a document.

When is the BAdI called?
It is called in the report RMMR1MIS (MRIS). The auxiliary module MRMBADI_MRIS_HDAT_MODIFY is called, which in turn contains the actual call of the BAdI MRM_MRIS_HDAT_MODIFY. The system first checks whether the BAdI is active.

The following data is provided for the BAdI:
I_RBKPV (document header)
TI_FRSEG (document items)

The BAdI returns the following data:
E_RBKPV_IS_CHANGE (document header fields that can be changed)
E_CHANGE (must be set to 'X' for the changes to be copied)

© 2016 SAP SE or an SAP affiliate company. All rights reserved

More information
Documentation for the BAdI interface IF_EX_MRM_MRIS_HDAT_MODIFY

Definition name: MRM_MRIS_IDAT_MODIFY

What does the BAdI do?
You can use the BAdI MRM_MRIS_IDAT_MODIFY to change document lines in transaction MRIS "Invoicing Plan Settlement with Logistics Invoice Verification" when posting a document.

When is the BAdI called?
It is called in the report RMMR1MIS (MRIS). The auxiliary module MRMBADI_MRIS_IDAT_MODIFY is called, which in turn contains the actual call of the BAdI MRM_MRIS_IDAT_MODIFY. The system first checks whether the BAdI is active.

The following data is provided for the BAdI:
I_RBKPV (document header)
I_FRSEG (document items)

The BAdI returns the following data:
E_FRSEG_IS_CHANGE (is of the type mrm_frseg_ers_change)
E_CHANGE (must be set to 'X' for the changes to be copied .

More information
Documentation for the BAdI interface IF_EX_MRM_MRIS_IDAT_MODIFY

Definition name: MRM_MRKO_HDAT_MODIFY

What does the BAdI do?
You can use the BAdI MRM_MRKO_HDAT_MODIFY to change the vendor (invoicing party) and the document type (so that another number range is used).

When is the BAdI called?
The BAdI is called in the consignment report RMVKON00 (MRKO) at the start of the subprogram Block_abrechnen; the auxiliary module MRMBADI_MRKO_HDAT_MODIFY is called, which in turn calls the BAdI MRM_MRKO_HDAT_MODIFY. The system first checks whether the BAdI is active.

The following data is provided for the BAdI:
I_BUKRS (company code)
I_LIFNR (vendor)
I_BLART (document type)
TI_RKWA (consignment withdrawals)

You can use the BAdI to change the vendor and the document type if the following indicators are

© 2016 SAP SE or an SAP affiliate company. All rights reserved

filled with the value X:
E_CHANGE_LIFNR
E_CHANGE_BLART

More information
Documentation for the BAdI interface IF_EX_MRM_MRKO_HDAT_MODIFY

Definition name: MRM_PAYMENT_TERMS

What is the behavior like in the standard system?
The terms of payment are proposed during invoice entry with purchase order reference, during direct posting to the G/L account or material account from the accounting view of the vendor master record.

The terms of payment shown on the invoice may differ from the terms of payment originally agreed upon in the purchase order or in the vendor master record.

You can manually enter or overwrite the terms of payment in the header data on the payment tab page when entering the invoice; the terms of payment entered manually have priority here.

What does the BAdI do?
You can use the BAdI MRM_PAYMENT_TERM to check or possibly change the automatically determined or manually entered terms of payment.

In what business and technical context is this BAdI used?
You can use this BAdI to change the following fields of the terms of payment in the invoice document header:
ZFBDT: Baseline Date for Due Date Calculation
ZBD1T: Cash discount days 1
ZBD1P: Cash discount percentage 1
ZBD2T: Cash discount days 2
ZBD2P: Cash discount percentage 2
ZBD3T: Deadline for net conditions
ZLSPR: Payment block key

The following information is available for this in the BAdI at runtime depending on the processing status of the invoice:
RBKP: Invoice Document Header
RSEG: Invoice Items

You can use this information to determine differing terms of payment. The terms of payment that are determined are checked for consistency and are replaced in the corresponding fields of the invoice document header.

When is the BAdI called?

© 2016 SAP SE or an SAP affiliate company. All rights reserved

The BAdI is called after the assignment (availability of invoice item information -> purchase order data) and before further processing of terms of payment (calculation of cash discount).

The BAdI can be set for all invoice entry procedures within the Logistics Invoice Verification due to its central positioning in the function module MRM_FINAL_CHECK. You can use the origin of an invoice document (RBKP-IVTYP) to control for which transactions the BAdI is to be set.

How does the BAdI affect the application?
The (possibly) changed terms of payment are provided for the application. They form the basis for further processing in the Logistics Invoice Verification and in accounting.

Example
You want to carry out a determination of the terms of payment that differs from the standard; this could be, for example:

The baseline date for payment is to be derived from the goods receipt date.
The terms of payment entered by the user and available in the purchase orders are to be checked and the most favorable terms of payment are to be determined from this.
You want to set a payment block in the document header for certain transactions.

More information
Documentation for the BAdI interface IF_EX_MRM_PAYMENT_TERMS

Definition name: MRM_RELEASE_CHECK

What is the behavior like in the standard system?
The system checks invoice documents for set criteria in Customizing for Logistics Invoice Verification and may lock them, for example, due to date variance or amount variance. You can use transaction MRBR "Release Invoices" to display and release blocked invoice documents. Selection criteria are used to select and display blocked invoice documents; you can release the displayed invoice documents for the payment if you have the relevant authorization.

What does the BAdI do?
You can use the BAdI MRM_RELEASE_CHECK to perform additional checks before the blocked invoices are released. The list of documents to be released that was selected at the start is further restricted depending on the result of the additional checks.

In what business and technical context is this BAdI used?
The auxiliary module MRMBADI_RELEASE_CHECK calls the BAdI and checks whether the BAdI is active. A list of invoice documents to be released is transferred to this function module as the parameter tab_rbkp_blocked (TYPE mrm_tab_rbkp_blocked) for all calls. You can use this BAdI to further restrict this list.

The BAdI returns a new list of invoices to be released as a result. This list contains only those invoice documents which passed the additional checks successfully.

When is the BAdI called?
The function module MRM_INVOICE_RELEASE_UPDATE is called when invoices are released. The following calls are taken into account here:

1. LMRBRU08 (validation check of blocked price items and release invoice (Workflow)
2. LMRBRU14 (delete blocking reason 'price' due to list and release invoice (Workflow)
3. LMRM_BAPIF0D (INVOICE_RELEASE_UPDATE (BAPI))
4. RM08RELEASE_F0I (DOCUMENT_CHANGE_UPDATE (MRBR))

The auxiliary function module MRMBADI_RELEASE_CHECK is called immediately before the function module MRM_INVOICE_RELEASE_UPDATE for calls 1 to 3.
The call that occurs with transaction MRBR is executed for call 4. In this case, the auxiliary function module is called directly after the invoice documents to be released have been selected (in the report RM08RELEASE). As a result, with regard to the online release of invoices, you avoid a situation whereby invoices that in reality were excluded from the release are displayed as released.

How does the BAdI affect the application?
Calls listed for the auxiliary function module and thus the BAdI prevent the related accounting document of an invoice from being released without the invoice document being released.

The data records of the original list are reconciled with the list returned by the BAdI and compiled into a result list that forms the basis for the further release process.

Example
In the Public Sector area, the system checks whether the indicator PPA (Prompt Payment Act) is active for the corresponding company code before an invoice is released. The system checks whether there have been goods accepted for the invoice document by checking a further indicator. If this is not the case, the invoice is excluded from the current release process.
You can define check criteria, for example, an enhanced authorization check for the release of invoices.
You can find sample coding for this under Goto -> Sample Coding -> Display by double-clicking the method IF_EX_MRM_RELEASE_CHECK~DOCUMENT_CHECK_RELEASE.

More information
Documentation for the BAdI interface IF_EX_MRM_RELEASE_CHECK

Definition name: MRM_TOLERANCE_GROUP

What is the behavior like in the standard system?
Only one tolerance group can be assigned to a vendor within a company code. The following can be defined for each tolerance group:

the amount of differences (from the default value) in an invoice that are accepted by the system

whether invoices can be automatically reduced
You define the tolerance groups in the table T169L in Customizing for the Logistics Invoice Verification (Incoming Invoice -> Vendor-Specific Tolerances) and assign them to the vendors in the vendor master.

What does the BAdI do?
You can use the BAdI MRM_TOLERANCE_GROUP to determine different tolerance groups according to defined criteria and assign them to the vendor within a company code during invoice entry.

In what business and technical context is this BAdI used?
You can use this BAdI to change the tolerance group of the vendor (field TOGRR: Tolerance Group; Invoice Verification).

The following information is available for this in the BAdI at runtime:

TOGRR: Tolerance group from vendor master
RBKP: Invoice Document Header
RSEG: Invoice Items

You can use this information to determine a different tolerance group from the vendor master.

When is the BAdI called?
The BAdI is called within the differential calculation before the respective tolerance group is read from the table T169L.

The BAdI can be set for the invoice entry procedures within the Logistics Invoice Verification due to its central positioning in the function module MRM_FINAL_CHECK. You can use the origin of an invoice document (RBKP-IVTYP) to control for which transactions the BAdI is to be set in the Logistics Invoice Verification.

How does the BAdI affect the application?
Depending on the criteria defined in the BAdI, different tolerance groups can be assigned to a vendor within a company code at runtime during invoice entry. The system takes these into account and the invoice is blocked or reduced when differences occur.

Example
You want to define tolerance limits for vendors differently within a company code:
Tolerance limits are to be defined depending on the invoice amount
Tolerance limits are to be defined depending on the plants stored in the invoice item.

More information
Documentation for the BAdI interface IF_EX_MRM_TOLERANCE_GROUP

Definition name: MRM_TRANSACT_DEFAULT

What is the behavior like in the standard system?
When you call transaction MIRO "Enter Invoice", MIRA "Enter Invoice for Invoice Verification in the Background" and MIR7 "Park Invoice", different fields, such as transaction (Invoice, Credit memo, Subsequent debit, Subsequent credit), reference document type (Purchase Order/Scheduling Agreement, Delivery Note, Bill of Lading, Service Entry Sheet, Vendor, Transportation Service Agent) have the same values as the document that was just posted. These values are saved in the database table ESDUS for each user.

What does the BAdI do?
You can use the BAdI MRM_TRANSACT_DEFAULT to define proposals for the following settings user-dependently, for example:

- Transaction: If you enter credit memos only, choose the transaction "Credit memo".
- Reference document type: If you create your settlement for each vendor, choose "Vendor" as the reference here.
- Display variant for invoice items: If you have to change account assignments for each invoice, choose "Acct Assignment - Cost Center" as the display variant.
- PO structure: If you want to display business processes and transactions for the entered purchase order item, display the PO structure.
- Worklist: If you use the parking worklist, for example, to process selected documents further, display the worklist.
- Display variant for G/L account items: You can define a display variant for processing G/L account items.

When is the BAdI called?
Within the module MRMBADI_TRANSACTION_DEFAULT, the system checks whether there is an active implementation for the BAdI; if there is, the method TRANSACTION_DEFAULT_SET is called.

The system performs various consistency checks after the values are returned from the method TRANSACTION_DEFAULT_SET.

How does the BAdI affect the application?
The settings that were last used are not proposed for each user, rather the settings defined in the BAdI are proposed.

Prerequisites:
Only existing transactions (1 = Invoice, 2 = Credit memo, 3 = Subsequent debit, 4 = Subsequent credit) or existing reference document types (1 = Purchase Order/Scheduling Agreement, 2 = Delivery Note, 3 = Bill of Lading, 4 = Service Entry Sheet, 5 = Vendor, 6 = Transportation Service

Agent) can be used.

If you choose a non-existing transaction or reference document type, the system chooses the value that was saved last.

More information
Documentation for the BAdI interface IF_EX_MRM_TRANSACT_DEFAULT

Definition name: MRM_UDC_DISTRIBUTE

What is the behavior like in the standard system?
The distribution of the unplanned delivery costs is carried out according to a fixed schema in the standard system.

What does the BAdI do?
You can use the BAdI MRM_UDC_DISTRIBUTE to change the distribution of shares of delivery costs to individual invoice items.

In what business and technical context is this BAdI used?
Provide the BAdI with the same data that is provided for the standard module MRM_DISTRIBUTION_BNK for determining delivery costs. The BAdI returns the table (t_drseg_udc) with the shares of delivery costs in the document currency as a result. The shares of delivery costs determined by the BAdI are transferred to the related item data records.

When is the BAdI called?
In the subprogram DISTRIBUTION_BNK, the standard module MRM_DISTRIBUTION_BNK is first called for the distribution of unplanned delivery costs. The auxiliary module MRMBADI_UDC_DISTRIBUTION is called at the end of the subprogram. The BAdI is called at the same time as with the conventional calculation for the online transactions.

How does the BAdI affect the application?
It changes the field BNKAN_FW (share of delivery costs in foreign currency) in the item data for each invoice item that is taken into account in the distribution.

The corresponding field BNKAN_HW (share of delivery costs in local currency) is also changed accordingly if the document currency differs from the local currency (using the module CONVERT_TO_LOCAL_CURRENCY).

Prerequisites:
Check in Customizing for the Logistics Invoice Verification (table T169P) whether the setting for delivery costs distribution has been made.

Example
You want to distribute the delivery costs incurred to invoice items in accordance with your own procedure. Examples of the distribution are:

© 2016 SAP SE or an SAP affiliate company. All rights reserved

- Exclusive look at items that meet certain criteria
- Distribution on the basis of quantities
- Look at items with planned delivery costs during the distribution

You can find sample coding for this under Goto -> Sample Coding -> Display by double-clicking the method IF_EX_MRM_UDC_DISTRIBUTE~AMOUNTS_CALCULATE.

More information
Documentation for the BAdI interface IF_EX_MRM_UDC_DISTRIBUTE

Definition name: MRM_WT_SPLIT_UPDATE

What does the BAdI do?
In the Logistics Invoice Verification, you can use transaction MIRO "Enter Invoice", MIR7 "Park Invoice" and MIRA "Enter Invoice for Invoice Verification in the Background", as well as the automatic settlements, to change the data for the withholding tax (table RBWS) or for the amount split (table RBVS) based on certain characteristics.

In what business and technical context is this BAdI used?
Change withholding tax data
The invoice item data is transferred to the BAdI MRM_WT_SPLIT_UPDATE. The BAdI checks, based on certain characteristics, whether invoice items are subject to withholding tax; if this is the case, it adjusts the data to the withholding tax accordingly. If an invoice item is subject to withholding tax, the entire document is subject to withholding tax.
Change amount split data
The BAdI checks, based on certain characteristics, whether a portion of the invoice amount must be posted to a blocked account. The amount split data (table RBVS) can be changed for this.
When is the BAdI called?

This function is called from the subprogram move_data_to_fi in the online transactions MIRO, MIR7 and MIRA.

How does the BAdI affect the application?
Changed values are returned to the application.
An active implementation of this BAdI excludes the manual entry of the withholding tax base and amount. You define the settings for the calculation in Financial Accounting Customizing under the following activity: Define Withholding Tax Type for Invoice Posting

More information
Documentation for the BAdI interface IF_EX_MRM_WT_SPLIT_UPDATE

BAdIs as of Release 6.00:

****
Definition name:INVOICE_UPDATE

The BAdI INVOICE_UPDATE was migrated for Release 6.00 from a classic BAdI to the enhancement spot INVOICE_UPDATE.
Your implementations of the BAdI INVOICE_UPDATE may no longer be called at runtime after an upgrade to Release 6.00 or higher. In this case, proceed as follows and regenerate the enhancement spot INVOICE_UPDATE (see Note 538897 also):
1. Call transaction SE18.

2. Enter the enhancement spot INVOICE_UPDATE.

3. In the menu bar choose, "Utilities (M)" -> "Regeneration".

Information about the migration of classic BAdIs within the framework of the enhancement concept are available in the SAP Library for SAP NetWeaver:
SAP Library under SAP NetWeaver Library -> SAP NetWeaver by Key Capability -> Application Platform by Key Capability -> ABAP Technology -> ABAP Workbench -> Enhancement Framework -> Enhancement Technologies -> Business Add-Ins (BAdIs) -> Migrating Classic BAdIs

****

Application component: MM-IV-LIV
Package: MRM_BADI

Definition name: MRM_ITEM_CUSTFIELDS

Usage
This BAdI is used in the component MM-IV.
It is used to change customer-specific fields in vendor invoices.

You can use the BAdI MRM_ITEM_CUSTFIELDS to integrate your own subscreen at item level in order to edit customer-specific data. The customer include CI_DRSEG_CUST is available for customer-specific fields. The structure DRSEG_CI contains the fields C_RBLGP (Document Item in Invoice Document), C_KOART (Account Type) and the customer include CI_DRSEG_CUST. The structure DRSEG_CI is already contained in the structure DRSEG (Invoice Item (online processing)), MMCR_FRSEG and MMCR_RBMA. If you want to save customer-specific fields on the database, you must add these fields in an APPEND structure to the corresponding database table (RSEG, RBDRSEG, RBMA). You add an APPEND structure with the customer-specific fields to the structure COBL_MRM for G/L account lines. The customer-specific fields are then

© 2016 SAP SE or an SAP affiliate company. All rights reserved

also available in the database table RBCO.

You can use the BAdI MRM_ITEM_CUSTFIELDS to edit the customer-specific fields on your own subscreen during the online invoice check. The following data is available for this:

- Transaction Type (Add, Change...)
- Invoice Document Header
- Invoice Document Items (Items with PO Reference, Items with G/L Account Posting, Items with Material Posting)

Note:
The customer-specific fields cannot be filled for automatic transactions.

Prerequisites:
To implement the BAdI, you need to have ABAP knowledge for subscreen and flow logic creation.

Example:
The BAdI implementation WKA_ITEM_CUSTFIELDS is available as an example.

More information
Documentation for the BAdI methods:

- TABPAGE_LABEL_SET
- CUSTOMDATA_MODIFY
- INVOICE_DATA_TRANSFER
- CUSTOM_DATA_TRANSFER
- CUSTOM_DATA_GET
- INVOICE_DATA_GET

Definition name: MRM_ERS_HDAT_MODIFY

Usage
This BAdI is used in the component MM-IV.
It is used to change different document header data from vendor invoices in the automatic settlement of planned delivery costs (ERS).

It is called in the report RMMR1MDC (MRDC). The auxiliary module MRMBADI_ERS_HDAT_MODIFY is called, which in turn contains the actual call of the BAdI

MRM_ERS_HDAT_MODIFY. The system first checks whether the BAdI is active.

The following data is provided for the BAdI:
I_RBKPV (document header)
TI_ERS_DC (delivery costs to be settled)

The BAdI returns the following data:
E_RBKPV_IS_CHANGE (document header fields that can be changed)
E_CHANGE (must be set to 'X' for the changes to be copied)

Definition name: MRM_ERS_IDAT_MODIFY

Usage
This BAdI is used in the component MM-IV.

It is used to change the document lines from vendor invoices in the automatic settlement of planned delivery costs (ERS).

It is called in the report RMMR1MRS (MRRL) and in the report RMMR1MDC (MRDC). The auxiliary module MRMBADI_ERS_IDAT_MODIFY is called, which in turn contains the actual call of the BAdI MRM_ERS_IDAT_MODIFY. The system first checks whether the BAdI is active.

The following data is provided for the BAdI:
I_SEGTAB (header and item data for settling planned delivery costs)

The BAdI returns the following data:
E_SEGTAB_DC_CHANGE (document fields that can be changed)
E_CHANGE must be set to 'X' for the changes to be copied.

Definition name: MRM_VARIANCE_TYPE

Usage
You can use the BAdI MRM_VARIANCE_TYPE in transaction MIRO "Enter Invoice", MIR7 "Park Invoice" and MIR6 "Invoice Overview".

What is the behavior like in the standard system?
The "Variance Type" field at item level of an invoice is filled only with the values SPACE, 'A' und 'B'.

What does the BAdI do?
You can use the method DETERMINE to place your own values in the "Variance Type" field.

In what technical context is this BAdI used?
The method DETERMINE is called at 3 different stages:

© 2016 SAP SE or an SAP affiliate company. All rights reserved

- when loading the invoice
- when reassigning an invoice
- when executing the PAI of the items (screen 6310 for example)

Application component: MM-IV
Package: WRF_INVOICEVERIFICATION_ASSIGN

Definition name: WRF_MRM_ASSIGN_TEST

What is the behavior like in the standard system?
Using the settings stored under "Incoming Invoice" -> "Assignment Test Group" in Customizing for the Logistics Invoice Verification, the system determines the first assignment test date, the end of the assignment test and the gaps between the assignment tests.

What does the BAdI do?
The BAdI WRF_MRM_ASSIGN_TEST provides two methods that can be used to adjust the assignment test procedure to suit your needs.

You can use the method DATES_REDETERMINE to change the date of the first assignment test and the end of the assignment test.

The method INVOICE_SAVE is called when saving the invoice within the assignment test. Here you can change the status of the assignment test, the date for the next assignment test and other data.

In what business and technical context is this BAdI used?
You can use the methods DATES_REDETERMINE and INVOICE_SAVE to set different data for the assignment test.

When are the BAdIs called?
The method DATES_REDETERMINE is called after the assignment test date is determined and before the invoice is saved. The method INVOICE_SAVE is called when saving the invoice within the assignment test.

Application component: MM-IV
Package: WRF_INVOICEVERIFICATION_PREPAY

Definition name: WRF_PREPAY_INVOICE

What is the behavior like in the standard system?
The prepayment relevance of a logistics invoice is determined from prepayment-specific Customizing. A prepayment document may be posted with the posting date of the logistics invoice.

What does the BAdI do?
The BAdI WRF_PREPAY_INVOICE provides two methods.

You can change the posting date of the prepayment document in the method SET_POSTING_DATE. The posting and document date are transferred from the logistics invoice document.
If no prepayment document is posted, you can use the method PREPAYMENT_RELEVANCE_CHANGE to override the "Prepayment status" field from the header of the logistics invoice currently processed. Thus, a prepayment can be prevented or triggered.
In what business and technical context is this BAdI used?

You can use the method SET_POSTING_DATE to select another posting date for the prepayment document.

You can use the method PREPAYMENT_RELEVANCE_CHANGE to implement an extended logic for controlling the prepayment relevance (request or reject). The invoice document header is provided in the BAdI at runtime (RBKP) for this purpose.

When are the BAdIs called?
The call of both BAdI methods is carried out before the logistics invoice is saved in the method INVOICE_SAVE_PREPARE of the class CL_WRF_PREPAYMENT.

How do the BAdIs affect the application?
Note that you can only change the posting date in the method SET_POSTING_DATE and the prepayment relevance in the method PREPAYMENT_RELEVANCE_CHANGE if no prepayment document was created.

Application component: BW-BCT-MM-IV
Package: MRM

Definition name: INVOICE_BW

What does the BAdI do?
You can use the BAdI to adjust the communication structure data (MCRBHD - header data, MCRBITM - item data, MCRBCO - controlling data, MCRBMA - material items, MCRBKEY - key fields at account assignment level).

© 2016 SAP SE or an SAP affiliate company. All rights reserved

The BAdI also allows additional fields to be filled that are not contained in the delivered extract structures and are only added later.

When are the BAdIs called?
The BAdI is called from the function module MRM_BW_EXTRACT_PREPARE before the extract structure MC06M_0ITM is set up.

How do the BAdIs affect the application?
Information, warning and error messages are saved in an error log (table MRMRBBW_ERR_EXTR).

Definition name: MRM_INVOICE_UPDATE

Usage
This BAdI is used in various components, for example, Logistics Invoice Verification (MM-IV).

You can use it to implement other process steps or checks when deleting or saving documents. The BAdI is called, for example, when scheduling an invoice for background processing, when entering incorrect invoices via EDI and when deleting documents from background processing or documents that were created via EDI.

When are the BAdIs called?
The BAdI interface provides two methods that are called depending on the transaction:

IF_EX_MRM_INVOICE_UPDATE PROCESS_AT_SAVE is processed when saving an invoice

IF_EX_MRM_INVOICE_UPDATE PROCESS_AT_DELETE is processed when deleting an invoice

The calls of the methods are encapsulated in the modules MRMBADI_INVOICE_SAVE and MRMBADI_INVOICE_DELETE and are called from the modules MRM_INVOICE_DOCUMENT_SAVE and MRM_INVOICE_DOCUMENT_DELETE that are used for saving and deleting invoices.

BAdIs as of Release 6.03:

Application component: MM-IV-LIV
Package: MRM

Definition name: ES_MRM_BAPI_COMPLAIN

© 2016 SAP SE or an SAP affiliate company. All rights reserved

Usage
This BAdI is used in the component "Invoice Verification" (MM-IV).

You can use this BAdI to change the complaint data for a reduced invoice before the display.

headerdata: Header data for the incoming invoice
addressdata: Address data and tax number(s) of the vendor
itemdata: Table with the reduced invoice items
taxdata: Table with tax information of invoice reduction
extensionout: Enhancement fields of BAPI
return: Confirmations

The BAdI is called at the end of the function moduel BAPI BAPI_INCOMINGINVOICE_COMPLAIN.

Application component: MM-IV-LIV
Package: VPACK_MRM

Definition name: MRM_BLOCKREASON_DELETE_CUST

Usage
This BAdI is used in the component "Invoice Verification" (MM-IV).

You can use this BAdI to define your own checks to determine invalid blocking reasons in blocked invoices and to set them to invalid. The system removes blocks that are no longer valid for a large number of invoices and releases them for payment. The blocking indicator is deselected in the accounting document of the invoice.

You can use this check in invoices with and without order reference (direct posting to G/L account or material account) and choose between automatic and manual release during processing.

The BAdI is called in transaction MRBR after you select the blocked invoices (via selection criteria) that you want to release.

Prerequisites
The validity check takes into account only blocks at item level and is only executed if you have selected the blocking procedure "Blocked Due to Variances".

Also refer to:
Methods

Check Blocks in Invoices with Purchase Order Reference
Check Blocks in Invoices with Manual Posting to G/L Account
Check Blocks in Invoices with Manual Posting to Material

Interface
Check Blocking Reasons in Invoices for Validity

Activities
In order to activate the BAdI, you must create an active implementation. To do so, go to the SAP menu and select Tools -> ABAP Workbench -> Utilities -> Business Add-Ins -> Implementation (transaction SE19).

1. In the "Implementation" field, enter the name of the BAdI implementation in the following dialog box and choose "Copy".
2. If implementation have already been created for the BAdI, a dialog box appears with the existing implementations.
3. Choose "Create" in this dialog box.
4. Enter a short text for the implementation in the corresponding field.
5. Insert filter specifications in the "Properties" tab page if this is a filter-dependent BAdI.
6. Choose the "Interface" tab page.
7. The system generates the name of the implementing class based on the name of your implementation. You can change the name.
8. Save your entries and assign a package.
9. Place the cursor on a method. Double-click to call the class builder.
10. Enter your required source code for the implementation between the statements method <Interface-Name>~<Name of the Method>. and endmethod..
11. Save and activate your source code and return to the "Change Implementation" screen.
12. Save your changes on the "Change Implementation" screen.
13. Note: You can first create an implemenation for a BAdI and later activate it. In this case, conclude your work at this point.
14. Choose "Activate".
15. The source code you created is run when the application program is executed.

You can find further information about the procedure in the SAP Library under Basis -> ABAP Workbench -> Changes to the SAP Standard -> Business Add-Ins -> Implementation of a Business Add-In.

Information about the implementation of BAdIs within the framework of the enhancement concept is available in the SAP Library for SAP NetWeaver:
SAP Library under SAP NetWeaver Library -> SAP NetWeaver by Key Capability -> Application

© 2016 SAP SE or an SAP affiliate company. All rights reserved

Platform by Key Capability -> ABAP Technology -> ABAP Workbench -> Enhancement Framework -> Enhancement Technologies -> Business Add-Ins (BAdIs) -> BAdI Concept -> BAdIs - Embedded in the Enhancement Concept

© 2016 SAP SE or an SAP affiliate company. All rights reserved