



## **PROCESS DIRECTOR 7.3**

---

# Reference Guide

September 2016 – ReadSoft AB

The contents of this document are subject to change without notice.

Questions or comments about this document may be emailed to [documentation@readsoft.com](mailto:documentation@readsoft.com).

02092016.1236

# Contents

<b>Introduction</b> .....	<b>4</b>
About this guide .....	4
Typographical conventions .....	4
<b>General reference</b> .....	<b>5</b>
Available query classes .....	5
Available mapping conversions.....	7
Available preset IDs .....	16
Available preset classes .....	17
Available checks .....	18
Available determinations.....	20
Available process parameters .....	21
Mechanisms for populating fields .....	28
Workflow control patterns and step types.....	30
Standard documentation objects.....	31
Placeholders for documentation objects .....	35
User profile parameters.....	38
Programs .....	39
User Exits/BAdIs .....	48
<b>Process types reference</b> .....	<b>81</b>
Accounts Payable (IV) .....	81
Asset Acquisitions (AA) .....	89
Asset Retirement (AR).....	94
Asset Transfer (AT) .....	99
Customer Orders (SO).....	105
Electronic Bank Statements (ES) .....	132
Financial Postings (FI).....	137
Generic archiving (ARCH_BUS) .....	152
Goods Receipts (DN) .....	156
Invoice Block and Cancelation (IB) .....	169
Master Data Maintenance (MD).....	175
Order Confirmations (OR) .....	202
Payment Advices (PA) .....	213
Payment Approvals (PR) .....	226
Requisitions (PO) .....	234
<b>Index</b> .....	<b>260</b>

# Introduction

## About this guide

This guide is intended for implementation consultants who are installing and configuring the SAP/ABAP component of PROCESS DIRECTOR.

The guide provides reference information for PROCESS DIRECTOR.

For information on installing and configuring PROCESS DIRECTOR, see the following guides:

- *Importing PROCESS DIRECTOR into SAP*
- *PROCESS DIRECTOR SAP Configuration Guide*
- *PROCESS DIRECTOR SAP Advanced Configuration Guide*
- *PROCESS DIRECTOR Troubleshooting Guide*

## Typographical conventions

The following typographical conventions are used:

- Menu names, commands, and dialog names appear in **bold**.
- Names of keys on your keyboard appear in **SMALL CAPITAL LETTERS**.
- Names of files, folders, and settings appear in `Courier` font.
- SAP transaction codes and program names appear in `ARIAL MONOSPACE`.
- Variables that have to be replaced by an actual value are *italicized* or appear in `<>`.

## General reference

This section provides reference information that applies to all process types. It covers the following subjects:

- [Query classes](#)
- [Mapping conversions](#)
- [Preset IDs](#)
- [Preset classes](#)
- [Checks](#)
- [Determinations](#)
- [Mechanisms for populating fields](#)
- [Workflow control patterns and step types](#)
- [Standard documentation objects](#)
- [Placeholders for documentation objects](#)
- [User profile parameters](#)
- [Programs](#)

See the [Process types reference section](#) for process type specific reference information.

## Available query classes

Query classes are used in the ranges configuration of semi-dynamic and dynamic Worklist nodes to supply parts of the query at runtime. For example, the query class /EBY/CL\_PDBO\_WL\_QRY\_PD\_USER determines the name of the currently logged on user and displays all documents assigned to that user in the Worklist node.

The following query classes are available in the PROCESS DIRECTOR standard:

Query class name	Description
/EBY/CL_PDBO_WL_QRY_ALPHA_CONV	<p><b>Alpha conversion</b></p> <p>To convert the query result from internal to external format. This query class is useful for dynamic nodes, where the field content forms the node name and therefore may not be easily readable.</p> <p>For example, if you build dynamic nodes for individual vendors, without alpha conversion the node names would be something like 0000001000, 0000002000, etc. If you use the alpha conversion query, the node descriptions would be 1000, 2000, etc.</p>
/EBY/CL_PDBO_WL_QRY_CUR_DAY	<p><b>Current day (sy-datum) +/- days</b></p> <p>To include documents whose date is within a specific time frame, for example, documents from the last week or the last month.</p>
/EBY/CL_PDBO_WL_QRY_CUR_DAY_TS	<p><b>Time stamp of the current day +/- days</b></p> <p>To include documents whose timestamp is within a specific time frame.</p>
/EBY/CL_PDBO_WL_QRY_PD_SUB_ACT	<p><b>Active substitutes query</b></p> <p>To include documents of active and passive substitutes.</p> <p><b>Note:</b> When you use this query, the fields WC_PROCESSER and WC_PROC_TYPE must be included in the node's view.</p>
/EBY/CL_PDBO_WL_QRY_PD_SUBST	<p><b>PD user substitutions</b></p> <p>To include documents of active substitutes only.</p> <p><b>Note:</b> When you use this query, the fields WC_PROCESSER and WC_PROC_TYPE must be included in the node's view.</p>

Query class name	Description
/EBY/CL_PDBO_WL_QRY_PD_USER	<p><b>PD user name</b></p> <p>To include documents of the currently logged on user.</p>
/EBY/CL_PDBO_WL_QRY_PD_USER_TY	<p><b>PD user type</b></p> <p>To include documents of the currently logged on user type:</p> <ul style="list-style-type: none"> <li>S - SAP user</li> <li>I - Internet user</li> <li>L - LDAP user</li> </ul>
/EBY/CL_PDBO_WL_QRY_STATE_DESC	<p><b>State description</b></p> <p>To create dynamic nodes for individual document statuses and display status descriptions (<b>New</b>, <b>Processed</b>, etc.) instead of the status value (<b>NN</b>, <b>OK</b>, etc.).</p>

## Available mapping conversions

Mapping conversions enable you to perform functions on fields when they are imported into PROCESS DIRECTOR. These mapping conversions are available in the PROCESS DIRECTOR standard:

Mapping function module	Description
/EBY/PDBO_MAF_ADD	<p><b>Add parameter to value</b></p> <p>For numeric fields, adds the value specified in the parameters to the field value.</p> <p><b>Example:</b> An original field value of 1000 and a parameter value of 100 results in a value of 1100 after conversion.</p>

Mapping function module	Description
/EBY/PDBO_MAF_ADD_TEXT	<p><b>Write input to a text defined by text ID</b></p> <p>Saves an external string to a PROCESS DIRECTOR text.</p> <ol style="list-style-type: none"> <li>1. Map the external field containing the string to any internal field of the appropriate PROCESS DIRECTOR logical level (HEADER or ITEMS). Note that this internal field will not be filled.</li> <li>2. Create a mapping conversion using the /EBY/PDBO_MAF_ADD_TEXT function. In the parameters, specify the text type and ID, for example, EKKO for purchasing document header texts, or /EBY/PD for PROCESS DIRECTOR notes. You can use any of the defined supported text types.</li> <li>3. Assign the mapping conversion to the external field.</li> </ol>
/EBY/PDBO_MAF_ARITHMETIC	<p><b>Process arithmetic functions on input</b></p> <p>Performs the arithmetic function defined in the parameters on the field value.</p> <p><b>Example:</b> Use the TRUNC function to remove decimal places, for example, to truncate 1.4142 to 1. (Note that this function performs no rounding, so 1.999 is also truncated to 1).</p> <p>See the SAP documentation for detailed information on the available arithmetic functions.</p>

Mapping function module	Description
/EBY/PDBO_MAF_ASSIGN	<p><b>Assign a variable to output</b></p> <p>Assigns an SAP system variable or other interface variable to the output. Enter the variable in the parameters.</p> <p>The following interface variables are currently supported:</p> <p>IR_MAPPER</p> <p>IS_CURRENT_DMO</p> <p>IS_CURRENT_MAP</p> <p>IN_SUBNODE</p> <p>IN_PARENTSUBNODE</p> <p>IR_CURRENT_PARENT</p> <p>IR_CURRENT_OBJECT</p> <p><b>Example:</b> Enter IN_SUBNODE in the parameters to sequentially number the line items: 1, 2, 3.</p> <p><b>Tip:</b> To number the generated line item numbers in a 10, 20, 30 sequence instead of 1, 2, 3, additionally enter the /EBY/PDBO_MAF_MULTIPLY mapping function with a parameter of 10, followed by the /EBY/PDBO_MAF_ARITHMETIC mapping function with the parameter TRUNC.</p>

Mapping function module	Description
/EBY/PDBO_MAF_CLEAR	<p><b>Initialize the value</b></p> <p>Removes the value from the field.</p> <p>Generally speaking, if you do not want a value in a specific field in the PROCESS DIRECTOR document, you should not add the field to the mapping table. However, there may be cases where you want to capture a value, but remove the value if it does not meet certain conditions.</p> <p><b>Example:</b> Your PROCESS DIRECTOR field is limited to 5 characters, while the captured field may contain up to 20 characters. In this case, only the first 5 characters of the captured value are inserted in the field, which may not make sense, so you may want to clear these characters from the output field and set the field value later in the PROCESS DIRECTOR document.</p> <p>To achieve this, first add the mapping conversion <b>Skip next function unless condition is met</b> (/EBY/PDBO_MAF_IF_LEN) and in the parameters, select the operator <code>GT</code> and enter 5 in the next field (this specifies the condition "field contains more than 5 characters"). Assign the order 1 to this conversion.</p> <p>Next, add the mapping conversion <b>Initialize the value</b> and assign the order 2 to this conversion.</p> <p>Result: The mapping conversion <b>Initialize the value</b> is skipped if the captured field contains 5 characters or less.</p>
/EBY/PDBO_MAF_DATEADJUST	<p><b>Adjust date (add/subtract working days)</b></p> <p>Changes the date by adding or subtracting the number of working days specified in the parameters.</p>

Mapping function module	Description
/EBY/PDBO_MAF_DATECONVERT	<p><b>Check/convert date to internal format</b></p> <p>Checks whether a date is in SAP internal format or converts the date to SAP internal format.</p> <p><b>Example:</b> A date in the format MM/DD/YYYY (for example, 01/23/2012) is converted to the internal format YYYYMMDD (20120123).</p>
/EBY/PDBO_MAF_DELIST	<p><b>Delete first occurrence of parameter from output</b></p> <p>Deletes the first occurrence of the string specified in the parameters from the output.</p> <p><b>Example:</b> A captured order number contains two leading zeroes, which should not be present in the PROCESS DIRECTOR field. Enter 00 in the parameters. 00 at the beginning of the field is removed by the mapping conversion, but 00 within the field is retained.</p>
/EBY/PDBO_MAF_DELETE	<p><b>Delete all occurrences of a parameter from input</b></p> <p>Replaces all occurrences of the string specified in the parameters.</p>
/EBY/PDBO_MAF_DIVIDE	<p><b>Divide input with parameter</b></p> <p>For numeric fields, divides the field value by the value specified in the parameters.</p> <p><b>Example:</b> An original field value of 1000 and a parameter value of 100 results in a value of 10 after conversion.</p>

Mapping function module	Description
/EBY/PDBO_MAF_FILTER_ALLOWED	<p><b>Remove all characters which do not appear in parameter list</b></p> <p>Removes all characters that are not specified in the parameter list from the field.</p> <p><b>Example:</b> Enter 0123456789 in the parameters to remove non-numerical characters such as . / - from date fields.</p>
/EBY/PDBO_MAF_FILTER_FORBIDDEN	<p><b>Remove all characters which do appear in parameter list</b></p> <p>Removes all characters specified in the parameter list from the field.</p> <p><b>Example:</b> Enter characters such as - _ ( ) [ ] in the parameters to remove these characters from telephone numbers.</p>
/EBY/PDBO_MAF_FIXEDVALUE	<p><b>Use fixed value from parameters</b></p> <p>Inserts the value specified in the parameters into the field.</p> <p>This is useful for entering values into fields that are not captured during mapping. It can be used as an alternative to presets.</p> <p><b>Example:</b> If your Customer Orders are always of the type <code>OR Standard Order</code>, you can automatically populate the <b>Sales document type</b> field with the value <code>OR</code> by entering this value in the parameters.</p>

Mapping function module	Description
/EBY/PDBO_MAF_IF	<p><b>Skip next function unless condition is met</b></p> <p>Skips the next mapping conversion unless the condition specified in the parameters is met.</p> <p>You specify the sequence in which mapping conversions are performed by entering the appropriate number in the <b>Order</b> column of the mapping conversion configuration.</p> <p>This mapping conversion is useful, for example, for determining whether a value is already in a field and, if it is not, entering a predefined value.</p> <p><b>Example:</b> Suppose you want to ensure that a date field is filled. If the user enters a date, this value is not changed; if the user does not enter a date, the current system date is inserted.</p> <p>To achieve this, first add the mapping conversion <b>Skip next function unless condition is met</b> (/EBY/_PDBO_MAF_IF) and in the parameters, select the option EQ and leave the <b>Text</b> field blank (this specifies the condition "field = empty"). Assign the order 1 to this conversion.</p> <p>Next, add the mapping conversion <b>Get a system value</b> and enter DATLO in the parameters (this adds the system date to the field). Assign the order 2 to this conversion.</p> <p>Result: The mapping conversion <b>Get a system value</b> is skipped if the field is filled by the user.</p>
/EBY/PDBO_MAF_IF_LEN	<p><b>Skip next function unless condition is met</b></p> <p>Skips the next mapping conversion unless the condition specified in the parameters is met. This function checks only the field length. To check field contents, use /EBY/PDBO_MAF_IF.</p> <p><b>Example:</b> To skip the next function if the field length is less than 10, enter the operator GT and the value 10 in the parameters.</p>

Mapping function module	Description
/EBY/PDBO_MAF_LOWERCASE	<p><b>Change the value to lower case</b></p> <p>Changes the value of the field to lower case characters.</p> <p><b>Example:</b> Changes the field value TEXT or Text to text.</p>
/EBY/PDBO_MAF_MULTIPLY	<p><b>Multiply input with parameter</b></p> <p>For numeric fields, multiplies the field value by the value specified in the parameters.</p> <p><b>Example:</b> An original field value of 1000 and a parameter value of 100 results in a value of 100,000 after conversion.</p>
/EBY/PDBO_MAF_NOSPACE	<p><b>Change the value by removing all space characters</b></p> <p>Removes all spaces from the field value.</p> <p><b>Example:</b> Converts the telephone number 001 555 21 21 to 0015552121.</p>
/EBY/PDBO_MAF_OTHER_FIELD	<p><b>Map different field from the same level</b></p> <p>In the parameters of this mapping function you can specify an external field from the same data level as the field that is to be mapped. Using this approach, you can fill the field with the value from a different field if the original field is empty.</p>
/EBY/PDBO_MAF_POSTFIX	<p><b>Append parameter to value</b></p> <p>Adds the string specified in the parameters at the end of the field value.</p>
/EBY/PDBO_MAF_PREFIX	<p><b>Prepend parameter to value</b></p> <p>Adds the string specified in the parameters at the beginning of the field value.</p>

Mapping function module	Description
/EBY/PDBO_MAF_REPLACE	<p><b>Replace all occurrences of parameter A with B</b></p> <p>Replaces all occurrences of the first specified parameter with the second specified parameter.</p> <p><b>Example:</b> Replaces all commas (first parameter) with a decimal point (second parameter).</p>
/EBY/PDBO_MAF_SAMPLE	<p><b>Sample mapping function</b></p> <p>This function is a template that you can use to create your own mapping functions.</p> <p><b>Note:</b> Do not assign this mapping function to a conversion. Instead, copy the template to create your own function and assign that function to the conversion.</p>
/EBY/PDBO_MAF_SPACE_BUC	<p><b>Insert space before first upper case character</b></p> <p>Inserts a space before the first upper case character found in the field.</p>
/EBY/PDBO_MAF_SUBTRACT	<p><b>Subtract parameter from value</b></p> <p>For numeric fields, subtracts the value specified in the parameters from the field value.</p> <p><b>Example:</b> An original field value of 1000 and a parameter value of 100 results in a value of 900 after conversion.</p>
/EBY/PDBO_MAF_SYST	<p><b>Get a system value</b></p> <p>Enters the system value specified in the parameter list in the field.</p> <p><b>Example:</b> Enters the current system date when DATLO is entered in the parameter list.</p>

Mapping function module	Description
/EBY/PDBO_MAF_UOM	<p><b>Unit of measurement ISO or string to SAP</b></p> <p>Converts an ISO unit of measurement or a text string to the SAP internal format required by PROCESS DIRECTOR.</p> <p><b>Example:</b> Converts the text string <code>Piece(s)</code> or the ISO code <code>PCE</code> to the SAP value <code>PC</code>.</p> <p><b>Note:</b> This mapping conversion uses a number of checks to find the correct SAP format. If all checks fail, the field is cleared.</p>
/EBY/PDBO_MAF_UPPERCASE	<p><b>Change the value to upper case</b></p> <p>Changes the value of the field to upper case characters.</p> <p><b>Example:</b> Changes the field value <code>text</code> or <code>Text</code> to <code>TEXT</code>.</p>

## Available preset IDs

A preset ID determines at what point in the life cycle of a document the preset value is applied. These preset IDs are available in the PROCESS DIRECTOR standard and should usually be enough for most usage scenarios:

Preset ID	Applies presets
APPROVE	When a document is approved in a workflow step.
CHECK	Before checks and determinations run.
CREATE	When a new document or a new line item or other sub-object (accounts, partners, etc.) is created.
INSERT	When a newly created document is saved for the first time (that is, inserted in the database).
POST_SUCCESS	When a document is successfully posted.

Preset ID	Applies presets
RECALL	When a workflow is recalled. <b>Note:</b> This applies only to workflow recall; it does not apply to the removal of individual processors from a workflow step.
REJECT	When a document is rejected in a workflow step.
UPDATE	When a document is saved.

## Available preset classes

A preset class enables you to dynamically compute a value which can then be applied to a field as a preset. For example, a preset class can determine the company code entered in a document and populate the **Currency** field with the currency defined for that company code.

The following general preset classes are available for all process types in the PROCESS DIRECTOR standard:

Preset class name	Description
/EBY/CL_PDBO_PRESET_CURR_AMNT	<b>Convert a currency amount to SAP internal format</b>  Uses the field <code>CURRENCY</code> from the same level or the first field of type <code>WAERS</code> to identify the currency and converts the input value to SAP internal format.  This preset must be used for all currency value fields which are imported using the preset ID <code>CREATE</code> .
/EBY/CL_PDBO_PRESET_CURR_OF_CC	<b>Preset currency from company code</b>

Preset class name	Description
/EBY/CL_PDBO_PRESET_TEMPL	<p><b>Preset template</b></p> <p>This preset class is provided as a template that you can use to configure your own custom presets.</p> <p><b>Note:</b> Do not add this preset class to the presets configuration. Instead, create your own preset class based on the template and add this custom preset class to the configuration.</p>

See the following sections for information on process type specific preset classes:

[Preset classes for Asset Retirement](#)

[Preset classes for Asset Transfer](#)

[Preset classes for Electronic Bank Statements](#)

[Preset classes for Financial Postings](#)

[Preset classes for Goods Receipts](#)

[Preset classes for Master Data Maintenance](#)

[Preset classes for Payment Advices](#)

[Preset classes for Requisitions](#)

## Available checks

A check verifies the completeness, accuracy and consistency of data in a business document.

The following general checks are available for all process types in the PROCESS DIRECTOR standard.

Check event name	Description
/EBY/CL_PDBO_EVT_DFC_CHK	<p><b>Duplicate value check</b></p> <p>Checks if a defined field value or combination of field values entered by the user is already available in the database. This prevents duplicate records being added to the database.</p> <p>You define the field values that should be checked in the duplicates configuration.</p>
/EBY/CL_PDBO_EVT_CHK_ATTACHMNT	<p><b>Check if the requested attachments exist</b></p> <p>Checks if the document has an attachment in the format specified in the initialization parameters (for example, a PDF file).</p> <p>Add this check if users are required to add an attachment to the document.</p>
/EBY/CL_PDBO_EVT_CHK_TEMPL	<p><b>Check template</b></p> <p>This check is provided as a template that you can use to configure your own custom checks.</p> <p><b>Note:</b> Do not add this check to the checks configuration. Instead, create your own check using the template and add this check to the configuration.</p>
/EBY/CL_PDBO_EVT_APE_CHK	<p><b>Check if document is excluded from autoposting</b></p> <p>For document-driven process types, checks whether an error-free document can be automatically posted or not depending on specific field values. You specify the field values in the check's initialization parameters.</p> <p><b>Note:</b> You can also define the field values that should be checked in the autoposting configuration.</p>

Check event name	Description
/EBY/CL_PDWC_EVT_CHK_FLDST_MND	<p><b>Mandatory field check based on field status</b></p> <p>Checks the mandatory fields from the field status and returns messages if these fields have not been filled.</p>

See the following sections for information on process type specific checks:

[Check for Asset Acquisition](#)

[Checks for Asset Retirement](#)

[Check for Asset Transfer](#)

[Checks for Customer Orders](#)

[Checks for Financial Postings](#)

[Check for Goods Receipts](#)

[Checks for Invoice Block and Cancelation](#)

[Checks for Master Data Maintenance](#)

[Checks for Order Confirmations](#)

[Checks for Payment Advices](#)

[Checks for Payment Approvals](#)

[Checks for Requisitions](#)

## Available determinations

A determination attempts to infer the correct value for a given document field from the other fields in the document, or from other sources such as mapping tables.

There are no determinations that apply for all process types. See the following sections for information on the available determinations per process type in the PROCESS DIRECTOR standard:

[Determinations for Customer Orders](#)

[Determinations for Goods Receipts](#)

[Determinations for Order Confirmations](#)

[Determinations for Payment Advices](#)

[Determinations for Requisitions](#)

## Available process parameters

PROCESS DIRECTOR provides an easy way for you to control the behavior of the system for the following processes:

- [Archiving](#)
- [Checks](#)
- [Posting](#)
- [Workflow](#)
- [... and more](#)

No knowledge of PROCESS DIRECTOR actions and events (which control these processes) is required; all you have to do is set the appropriate parameters for the process.

See the *PROCESS DIRECTOR SAP Configuration Guide* for detailed information on configuring process parameters.

The following general parameters are available for all process types. See the [process types reference section](#) for information on process parameters for specific process types.

### Archiving

/EBY/PDBO\_EPC\_ARCHIV

Initial settings > Process parameters

Process	Description
<b>Add only the current workflow to the Smart Form</b>	In the parameters, activate <b>Process current workflow only</b> to include only the details of the current workflow in the workflow log. If this option is not activated, details of all workflows to which the document was previously sent, as well as the current workflow, are listed in the workflow log.

Process	Description
<b>Archiving object parameters</b>	<p>Automatically creates a log of all activities that took place on a document during a workflow and adds this as an attachment to the PROCESS DIRECTOR document and the corresponding SAP document. In the parameters, enter the archiving document type and the name of the Smart Form that should be used to create the workflow log.</p> <p>Select a status and/or substatus to create the workflow log only when the document has this status/substatus. See also Workflow log creation.</p>
<b>Archiving object type assignment to MIME type</b>	
<b>Archive settings for EDI data</b>	<p>Automatically creates an attachment that visualizes documents received via EDI (similar to the images of scanned documents that may be used with other process types). In the parameters, enter the archiving document type and the name of the Smart Form that should be used to create the attachment. See also Configuring IDoc image attachments.</p>
<b>Configure attachment deletion in WebApp</b>	<p>Specifies whether the archive document itself, or only the link, should be removed when users delete attachments in the Web Application.</p> <p><b>Note:</b> Do not use options 0 and 1, they are not applicable.</p>
<b>Cover sheet properties</b>	<p>Automatically creates a cover sheet for request-driven documents when these are saved in the Web Application. In the parameters, enter the archiving document type and the name of the Smart Form that should be used to create the cover sheet. See also Configuring cover sheets.</p>
<b>Exclude document types for upload</b>	<p>In the parameters, add document types to prevent Web Application users from adding these document types as attachments to PROCESS DIRECTOR documents.</p>

Process	Description
<b>Smart Form and document type for archiving after posting</b>	<p>Automatically adds the workflow log as an attachment to the PROCESS DIRECTOR document and the corresponding SAP document when the document is posted. In the parameters, enter the archiving document type and the name of the Smart Form that should be used to create the workflow log.</p> <p>Select a status and/or substatus to create the workflow log only when the document has this status/substatus.</p>
<b>Wait x seconds for audit log archiving to finish</b>	<p>Enables you to delay archiving for a specified number of seconds to ensure that the PROCESS DIRECTOR workflow log and notes are archived correctly and available in the corresponding SAP documents. Enter a value here if you are experiencing problems with missing attachments in the SAP documents.</p>
<b>Workflow log creation after approval</b>	<p>Activate the <b>Turn on</b> parameter to create the workflow log when a document is approved.</p>
<b>Workflow log creation after recall</b>	<p>Activate the <b>Turn on</b> parameter to create the workflow log when a document is recalled.</p>
<b>Workflow log creation after reject</b>	<p>Activate the <b>Turn on</b> parameter to create the workflow log when a document is rejected.</p>

## Checks

/EBY/PDBO\_EPC\_CHECKS

### Initial settings > Process parameters

Process	Description
<b>Automatic workflow start after all checks succeed</b>	<p>Automatically starts the specified workflow on a document when the checks return no errors. In the parameters, enter the workflow ID(s) and activate <b>Turn on</b>.</p>

Process	Description
<b>Automatic workflow start after check failed</b>	Automatically starts the specified workflow on a document when the checks return an error. In the parameters, enter the workflow ID(s) and activate <b>Turn on</b> .

## Posting

/EBY/PDBO\_EPC\_POST

Initial settings > Process parameters

Process	Description
<b>Add only the current workflow to the Smart Form</b>	In the parameters, activate <b>Process current workflow only</b> to include only the details of the current workflow in the workflow log. If this option is not activated, details of all workflows to which the document was previously sent, as well as the current workflow, are listed in the workflow log.
<b>Automatic post after workflow approval</b>	Automatically posts the document when the specified workflow(s) is approved. In the parameters, enter the workflow ID(s) and activate <b>Turn on</b> .
<b>Automatic workflow start after post failed</b>	Automatically starts the specified workflow(s) on a document when posting of the document fails. In the parameters, enter a workflow ID and activate <b>Turn on</b> .
<b>Automatic workflow start after post succeeded</b>	Automatically starts the specified workflow(s) on a document when posting of the document is successful. In the parameters, enter a workflow ID and activate <b>Turn on</b> .

Process	Description
<b>Smartform and document type for archiving after posting</b>	<p>Automatically adds the workflow log as an attachment to the PROCESS DIRECTOR document and the corresponding SAP document when the document is posted. In the parameters, enter the archiving document type and the name of the Smart Form that should be used to create the workflow log.</p> <p>Select a status and/or substatus to create the workflow log only when the document has this status/substatus.</p>
<b>Text ID of the email</b>	
<b>Wait x seconds for audit log archiving to finish</b>	<p>Enables you to delay archiving for a specified number of seconds to ensure that the PROCESS DIRECTOR workflow log and notes are archived correctly and available in the corresponding SAP documents. Enter a value here if you are experiencing problems with missing attachments in the SAP documents.</p>

## Workflow

/EBY/PDBO\_EPC\_WORKFL

Initial settings > Process parameters

Process	Description
<b>Add only the current workflow to the Smart Form</b>	<p>In the parameters, activate <b>Process current workflow only</b> to include only the details of the current workflow in the workflow log. If this option is not activated, details of all workflows to which the document was previously sent, as well as the current workflow, are listed in the workflow log.</p>

Process	Description
<p><b>Archiving object parameters</b></p>	<p>Automatically creates a log of all activities that took place on a document during a workflow and adds this as an attachment to the PROCESS DIRECTOR document and the corresponding SAP document. In the parameters, enter the archiving document type and the name of the Smart Form that should be used to create the workflow log.</p> <p>Select a status and/or substatus to create the workflow log only when the document has this status/substatus. See also Workflow log creation.</p>
<p><b>Automatic post after workflow approval</b></p>	<p>Automatically posts the document when the specified workflow(s) is approved. In the parameters, enter the workflow ID(s) and activate <b>Turn on</b>.</p>
<p><b>Automatic workflow start after all checks succeed</b></p>	<p>Automatically starts the specified workflow on a document when the checks return no errors. In the parameters, enter the workflow ID(s) and activate <b>Turn on</b>.</p>
<p><b>Automatic workflow start after check failed</b></p>	<p>Automatically starts the specified workflow on a document when the checks return an error. In the parameters, enter the workflow ID(s) and activate <b>Turn on</b>.</p>
<p><b>Automatic workflow start after post failed</b></p>	<p>Automatically starts the specified workflow(s) on a document when posting of the document fails. In the parameters, enter a workflow ID and activate <b>Turn on</b>.</p>
<p><b>Automatic workflow start after post succeeded</b></p>	<p>Automatically starts the specified workflow(s) on a document when posting of the document is successful. In the parameters, enter a workflow ID and activate <b>Turn on</b>.</p>

Process	Description
<b>Automatic workflow start after transfer failed</b>	Automatically starts the specified workflow(s) on a document when transfer of the document from the capture software to PROCESS DIRECTOR fails. In the parameters, enter a workflow ID and activate <b>Turn on</b> .
<b>Automatic workflow start after transfer succeeded</b>	Automatically starts the specified workflow(s) on a document when transfer of the document from the capture software to PROCESS DIRECTOR is successful. In the parameters, enter a workflow ID and activate <b>Turn on</b> .
<b>Smart Form and document type for archiving after posting</b>	<p>Automatically adds the workflow log as an attachment to the PROCESS DIRECTOR document and the corresponding SAP document when the document is posted. In the parameters, enter the archiving document type and the name of the Smart Form that should be used to create the workflow log.</p> <p>Select a status and/or substatus to create the workflow log only when the document has this status/substatus.</p>
<b>Where to go after workflow approve? (advanced)</b>	
<b>Where to go after workflow recall? (advanced)</b>	
<b>Where to go after workflow reject? (advanced)</b>	
<b>Workflow log creation after approval</b>	Activate the <b>Turn on</b> parameter to create the workflow log when a document is approved.
<b>Workflow log creation after recall</b>	Activate the <b>Turn on</b> parameter to create the workflow log when a document is recalled.

Process	Description
<b>Workflow log creation after reject</b>	Activate the <b>Turn on</b> parameter to create the workflow log when a document is rejected.

## Other

/EBY/PDBO\_EPC\_OTHER

Initial settings > Process parameters

Process	Description
<b>Document rejection: note/reason required</b>	In the parameters, activate <b>Rejection note required</b> or/and <b>Rejection reason required</b> to ensure that users enter a predefined rejection reason, a note, or both when they reject a document. See also <a href="#">Configuring rejection reasons</a> .
<b>Parameters regarding uploaded files</b>	These parameters are required to enable line item upload or document creation from a file.
<b>Unlock original object?</b>	

## Mechanisms for populating fields

PROCESS DIRECTOR offers these mechanisms for populating business document fields with values:

Mechanism	When to use
Conversion functions	To populate display fields. Note that these values will <i>not</i> be persisted.

Mechanism	When to use
Presets	<p>To automatically initialize document data.</p> <p>Presets are a key mechanism in processes driven by internal requests, such as managing Requisitions. However, they can also be applied during document-driven processes, for example when an external document is received.</p> <p>Use presets if you want certain document fields to be automatically populated with default values on document creation or update. The most important types are:</p> <ul style="list-style-type: none"> <li>▪ fixed value presets</li> <li>▪ dynamic presets (developed as customer coding)</li> </ul> <p>Each preset can only populate a single field. Apply them only to document model fields—for populating display fields, use conversion functions.</p> <p>Presets are always defined or coded as per individual customer requirement.</p>
User exits / BAdIs	<p>Customer implementations of the PROCESS DIRECTOR user exit / BAdI <a href="#">Initialize values</a> can automatically initialize document data, as an alternative to dynamic presets.</p> <p>However, there are also other user exits / BAdIs that can manipulate the document model.</p>
Determinations	<p>To automatically complete document data.</p> <p>Determinations are a key mechanism in processes driven by incoming documents.</p> <p>Use determinations if you want the values of certain document fields to be calculated during a human task, for example, when checking or updating a document.</p> <p>A determination can populate a single field or multiple fields, depending on the determination.</p> <p>Although it is possible to develop customer determinations, usually you can accommodate the customer's business scenario by configuring the determinations already available out-of-the-box for the given PROCESS DIRECTOR process type.</p>

Mechanism	When to use
Entry templates	<p>To manually enter document data.</p> <p>Use entry templates if you want to let users define their own sets of default values that they can insert during manual processing to speed up data entry. You can also define system templates which will be available to all users. The most important types are:</p> <ul style="list-style-type: none"> <li>▪ line item templates</li> <li>▪ account templates</li> </ul> <p>An entry template usually populates several fields.</p> <p>Entry templates are always defined as per individual customer requirement.</p>
External data	This is the main source of field input for document-driven process types.
Customer events	Although customer events are not supported in the standard configuration mode, they are mentioned here for the sake of completeness. Among other capabilities, customer events can manipulate a business document's data in arbitrary ways.

## Workflow control patterns and step types

### Control patterns

#### Parallel split (AND-split)

A parallel split may occur when several workflow steps are configured with the same previous step, that is, the previous step has several successors, or when the workflow has several start steps. Valid successors proceed in parallel and may or may not be synchronized at some future time. A parallel split may cause the workflow to have more than one final step.

#### Synchronization (AND-join)

A synchronization may occur when a workflow step is configured with several previous steps. All valid previous steps must be approved for the successor to proceed.

### Step types

#### Valid step

A step is valid by default, but can be invalidated at runtime in case any specified step conditions are not satisfied.

- ❗ **Note:** The validity status of a step can be overridden by an implementation of the user exit / BAdI [Workflow steps handling](#) interface method CHECK\_STEP\_VALIDITY.

### Start step

A valid step not configured with a previous step is a start step. In other words, a successor step can never become a start step by way of an invalid previous step. If several start steps exist, a parallel split occurs.

If *all* start steps of a workflow have the processor assignment `In background (automatic, without manual interaction)`, it is possible for an automatic workflow start to occur.

If it is possible for a step to be a start step, you *must* ensure that the step's **Reject** setting is set to `1 Cancel the workflow`.

If a workflow doesn't have at least one start step, it cannot be started.

### Final step

A valid step is final if it is not defined as a previous step or if none of its successors are valid.

For a workflow to be approved, *all* of its final steps must be approved.

## Standard documentation objects

The following documentation objects are available in the PROCESS DIRECTOR standard for use in messages and workflow emails. Most of these documentation objects include [placeholders](#), which are replaced by the relevant values when the message or email that contains the documentation object is sent. Documentation objects can be created or edited in transaction SE61.

Use	Documentation object name	Purpose
Messages	/EBY/PDBO_DISPATCH_CUSTOMER	Insert customer address data
	/EBY/PDBO_DISPATCH_SAMPLE	Sample data for messages
	/EBY/PDBO_DISPATCH_USER	Insert user address data
Posting notification	/EBY/PDBO_DISPATCH_POST	Inform the creator of a document that the document has been posted.

Use	Documentation object name	Purpose
	/EBY/PDMDVM_DISPATCH_POST	Inform the creator of a vendor master data maintenance document that the document has been posted.
Collective emails	/EBY/PDWC_COLLECTIVEREMINDER	Text for collective reminder mails. A collective email is sent when the program <a href="#">/EBY/PDWC DUE DATE CHECK</a> is run. The collective email contains all workflow step notifications for that processor since the last program run.
	/EBY/PDWC_COLLECTIVE_HTML	Same as /EBY/PDWC_COLLECTIVEREMINDER, but in HTML format
Workflow reminder emails	Use objects beginning with /EBY/ICWC for Accounts Payable documents and objects beginning with /EBY/PDWC for all other document types.	
	/EBY/PDWC_REMINDER /EBY/ICWC_REMINDER	Remind a workflow processor that a workflow step is overdue.
	/EBY/PDWC_REMINDER_HTML /EBY/ICWC_REMINDER_HTML	Same as /EBY/PDWC_REMINDER / /EBY/ICWC_REMINDER, but in HTML format
	/EBY/PDWC_SHORT_REMINDER /EBY/ICWC_SHORT_REMINDER	Short text included in a collective mail to remind a workflow processor that a workflow step is overdue.
	/EBY/PDWC_SHORT_REMINDER_HTM /EBY/ICWC_SHORT_REMINDER_HTM	Same as /EBY/PDWC_SHORT_REMINDER / /EBY/PDWC_SHORT_REMINDER, but in HTML format
Workflow receipt emails	Use objects beginning with /EBY/ICWC for Accounts Payable documents and objects beginning with /EBY/PDWC for all other document types.	
	/EBY/PDWC_SENT /EBY/ICWC_SENT	Inform a workflow processor that they have received a document for processing.

Use	Documentation object name	Purpose
	/EBY/PDWC_SENT_HTML /EBY/ICWC_SENT_HTML	Same as /EBY/PDWC_SENT / /EBY/ICWC_SENT, but in HTML format
	/EBY/PDWC_SHORT_SENT_HTML /EBY/ICWC_SHORT_SENT_HTML	Short text included in a collective email to inform the user that a document has been sent to a workflow
Workflow query emails *	/EBY/PDWC_QUERY	Inform the recipient of a query that a query has been sent to them.
	/EBY/PCWC_QUERY_HTML	Same as /EBY/PDWC_QUERY, but in HTML format.
Workflow approval emails *	/EBY/PDWC_WF_APPROVED	Inform the workflow initiator that a workflow has been approved.
	/EBY/PDWC_WF_APPROVED_HTML	Same as /EBY/PDWC_WF_APPROVED, but in HTML format.
	/EBY/PDWC_SHORT_WF_APPROVED	Short text included in a collective mail to inform the workflow initiator that a workflow has been approved.
	/EBY/PDWC_SHORT_WF_APPR_HTML	Same as /EBY/PDWC_SHORT_WF_APPROVED, but in HTML format.
Workflow recall emails *	/EBY/PDWC_WF_RECALLED	Inform the workflow processors that a workflow has been recalled.
	/EBY/PDWC_WF_RECALLED_HTML	Same as /EBY/PDWC_WF_RECALLED, but in HTML format.
	/EBY/PDWC_SHORT_WF_RECALLED	Short text included in a collective mail to inform the workflow processors that a workflow has been recalled.

Use	Documentation object name	Purpose
	/EBY/PDWC_SHORT_WF_RECL_HTML	Same as /EBY/PDWC_SHORT_WF_RECALLED, but in HTML format.
Workflow cancellation emails *	/EBY/PDWC_WF_CANCELLED	Inform the workflow initiator that a workflow has been canceled as the result of a step being rejected ( <b>Reject</b> setting is set to <b>Cancel the workflow</b> in the step settings).
	/EBY/PDWC_WF_CANCELLED_HTML	Same as /EBY/PDWC_CANCELLED, but in HTML format
	/EBY/PDWC_SHORT_WF_CANCELLED	Short text included in a collective email to inform the workflow initiator that a workflow has been canceled as the result of a step being rejected.
	/EBY/PDWC_SHORT_WF_CANC_HTML	Same as /EBY/PDWC_SHORT_WF_CANCELLED, but in HTML format.
MOBILE APPROVAL emails*	/EBY/PDWC_SENT_MA	Inform a MOBILE APPROVAL workflow processor that they have received a document for processing.
	/EBY/PDWC_SENT_HTML_MA	Same as /EBY/PDWC_SENT_MA, but in HTML format
	/EBY/PDWC_QUERY_MA	Inform a MOBILE APPROVAL user that a query has been sent to them.
	/EBY/PDWC_QUERY_HTML_MA	Same as /EBY/PDWC_QUERY_MA, but in HTML format
	/EBY/MOBILE_APPR_OK	Inform a MOBILE APPROVAL workflow processor that their action (approve, reject, add note) was successful.

Use	Documentation object name	Purpose
	/EBY/MOBILE_APPR_ERR	Inform the workflow processor that their action (approve, reject, add note) could not be performed.
Other	/EBY/PDFI_GLPR_REMINDER	Inform the creator of an FI provision document that the validity date of the posted document has been exceeded.

\* Not available for Accounts Payable documents.

## Placeholders for documentation objects

Placeholders can be used to insert variable data. The placeholder is replaced with the relevant value. For example, in a message text, the placeholder `&SENDER-ADDRESS-FIRSTNAME&` will be replaced in the output by the first name of the user that creates the message and the placeholder `&SENDER-ADDRESS-LASTNAME&` will be replaced by the last name of the user.

These placeholders are available for use in workflow and message texts:

Use	Placeholder	Inserts in the email
User data	<code>&amp;SENDER-ADDRESS-[FIELD]&amp;</code>	<p>You can use any field provided in the SAP structure <code>BAPIADDR3</code> with this placeholder. Replace <code>[FIELD]</code> with the name of the field.</p> <p><b>Examples:</b></p> <p><code>&amp;SENDER-ADDRESS-FIRSTNAME&amp;</code></p> <p><code>&amp;SENDER-ADDRESS-LASTNAME&amp;</code></p> <p><code>&amp;SENDER-ADDRESS-E_MAIL&amp;</code></p> <p><code>&amp;SENDER-ADDRESS-TELE_NUMBR&amp;</code></p> <p><code>&amp;SENDER-ADDRESS-TELE_EXT&amp;</code></p> <p><code>&amp;SENDER-ADDRESS-FAX_NUMBR&amp;</code></p> <p><b>Note:</b> For workflow texts, the sender will be the current step's processor or initiator.</p>

Use	Placeholder	Inserts in the email
	&RECIPIENT-ADDRESS-[FIELD]&	<p>You can use any field provided in the SAP structure BAPIADDR3 with this placeholder. Replace [FIELD] with the name of the field.</p> <p><b>Examples:</b></p> <p>&amp;RECIPIENT-ADDRESS-TITLE_P&amp;            &amp;RECIPIENT-ADDRESS-FIRSTNAME&amp;            &amp;RECIPIENT-ADDRESS-LASTNAME&amp;            &amp;RECIPIENT-ADDRESS-DEPARTMENT&amp;</p>
Document data	&HEADER-[FIELD]&	<p>You can use any field provided in the header interface structure of the document type with this placeholder. For example, for Requisitions, use any field in the structure /EBY/PDPO_SHDR_IF, for Customer Orders the structure /EBY/PDSO_SHDR_IF, etc. Replace [FIELD] with the name of the field.</p> <p><b>Examples:</b></p> <p>&amp;HEADER-OBJ-TXT&amp; (document description)            &amp;HEADER-NUMBR&amp; (document number)            &amp;HEADER-GUID&amp; (unique document ID)</p> <p>See the <a href="#">Process types reference</a> (<i>Document model customization</i>) for more information on process type specific structures.</p>
Workflow data	&WORKFLOW-WC_DESCR&	Description of the current workflow.
	&WORKFLOWSTEP-WC_STEP_DESCR&	Description of the current workflow step.
	&WORKFLOWSTEP-WC_STEP_DUEDATE&	Due date of the current workflow step.

Use	Placeholder	Inserts in the email
	&RECEIPT_LIST&	List of workflows received for processing.
	&REMINDER_LIST&	List of overdue workflows.
	&APPROVED_WORKFLOW_LIST&	List of approved workflows.
	&APPROVED_WORKFLOW_LIST-COUNTER&	Number of approved workflows.
	&RECALLED_WORKFLOW_LIST&	List of recalled workflows.
	&RECALLED_WORKFLOW_LIST-COUNTER&	Number of recalled workflows.
	&CANCELLED_WORKFLOW_LIST&	List of workflows canceled after rejection of a step.
	&CANCELLED_WORKFLOW_LIST-COUNTER&	Number of workflows canceled after rejection of a step.
	&LAST_NOTE&	Text of the last note that was added to the document by the current workflow processor.
MOBILE APPROVAL data	&MOBILE-APPROVE&	Link that the MOBILE APPROVAL user clicks to approve the document.
	&MOBILE-APPROVE_HTML&	Same as &MOBILE-APPROVE&, but in HTML format. May be displayed as a button instead of a link.
	&MOBILE-REJECT&	Link that the MOBILE APPROVAL user clicks to reject the document.
	&MOBILE-REJECT_HTML&	Same as &MOBILE-REJECT&, but in HTML format. May be displayed as a button instead of a link.

Use	Placeholder	Inserts in the email
	&MOBILE-NOTE&	Link that the MOBILE APPROVAL user clicks to add a note to the document.
	&MOBILE-NOTE_HTML&	Same as &MOBILE-NOTE&, but in HTML format. May be displayed as a button instead of a link.
Other	&URL&	The Web Application URL specified in Workflow General Settings.
	&HTML_EMAIL_CSS_STYLES&	Add this alias to the <body> section of your own HTML email documentation objects to use your own stylesheet for HTML emails. See Customizing HTML emails for more information.  <b>Note:</b> This placeholder applies only to HTML emails.

To include a single document link in a text, use this syntax:

- For Accounts Payable documents: &URL&&&GUID=&HEADER-WEBGUID&
- For all other document types: &URL&&ACTION=OTHER~&GUID=&HEADER-GUID&

To change the default Worklist node in which the document will appear when the single document link is followed, you can create a customer implementation of the user exit / BAdI [Set Worklist node of a single document link](#).

## User profile parameters

You can add the following PROCESS DIRECTOR parameters to an SAP user profile:

Parameter	Description
/EBY/PDBO_DEBUG_MSG	Adds debugging messages to the system messages. See the <i>PROCESS DIRECTOR Troubleshooting Guide</i> for more information.

Parameter	Description
/EBY/PDBO_CALLSTACK	Displays message call stacks. See the <i>PROCESS DIRECTOR Troubleshooting Guide</i> for more information.
/EBY/PDBO_PROJECT	Assigns a user to a project. The project configuration will be used instead of the standard configuration for this user. See the <i>PROCESS DIRECTOR SAP Configuration Guide</i> for information on working with projects.
/EBY/PDBO_CHK_SPLITV	Set to a value of X, this parameter splits the checks and determinations configuration screen vertically so that the check details and message parameters are displayed on the right instead of at the bottom of the screen. This is convenient when using a high resolution screen.

## Programs

The following programs are available for all PROCESS DIRECTOR process types. Use the standard SAP transaction SE38 to run a program once, or SM36 to schedule it as a background job.

- [/EBY/PDWC DUE DATE CHECK](#)

Sends receipt emails when users receive documents in a workflow and sends reminder emails for overdue workflows. Emails can also be sent when workflows are approved, recalled, or canceled after rejection.

- [/EBY/PDBO REPETITOR](#)

Automatically checks and posts PROCESS DIRECTOR documents.

- [/EBY/PDBO CONFIG DUMP](#)

Use this program to download and upload PROCESS DIRECTOR project configurations to XML files and compare configurations.

- [/EBY/PDBO CONFIG GEN WORKLIST](#)

Generates the Worklist.

- [/EBY/PDBO CHECK NESTEDSET NEW](#)

Regenerates the mapping of documents to static Worklist nodes.

- [/EBY/PDBO GEN FAILED NSTDSETS](#)

Generates the mapping of skipped documents to static Worklist nodes.

- [/EBY/PDBO CHECK WL DOC COUNT](#)

Checks or restores Worklist document counters for static nodes.

- [/EBY/PDBO CONFIG WRKL ENV CONV](#)

Sets the Worklist node **Not visible in** option for nodes created prior to PROCESS DIRECTOR 7.3.

- [/EBY/PDBO CHANGE ARC LINKS](#)

Changes the archiving link of attachments to a different process type.

See the following sections for information on process type specific programs:

[Programs for Accounts Payable](#)

[Programs for Financial Postings](#)

[Programs for Payment Advices](#)

[Programs for Payment Approvals](#)

## **[/EBY/PDWC\\_DUE\\_DATE\\_CHECK](#)**

Use the /EBY/PDWC\_DUE\_DATE\_CHECK program to send receipt emails when users receive documents in a workflow and to send reminder emails for overdue workflows. Emails can also be sent when workflows are approved, recalled, or canceled after rejection.

**Note:** Customers using ReadSoft's WORK CYCLE for processing Accounts Payable documents can assign email users to workflow steps. Currently, PROCESS DIRECTOR does not support email users, so these types of users are not processed by the /EBY/PDWC\_DUE\_DATE\_CHECK program and will not receive email notifications.

Setting	Description
<b>Object type</b>	Process type for which emails will be sent. Select the blank entry at the end of the list to include all process types.
<b>PD document number</b>	Range of PD documents to be processed.

Setting	Description
<b>Approved workflows</b>	Select to send emails to the workflow initiator when a workflow is approved.
<b>Rejected workflows</b>	Select to send emails to the workflow initiator when a workflow step is rejected, with the result that the entire workflow is canceled ( <b>Reject</b> setting is set to <b>Cancel the workflow</b> in the step settings).
<b>Recalled workflows</b>	Select to send emails to workflow processors when a workflow is recalled.
<b>Receipt emails</b>	Select to send emails when users receive a document for processing in a workflow.
<b>Reminder emails</b>	Select to send emails when a workflow step is overdue.
<b>Step overdue from</b>	Enter a date. Emails will be sent for workflow steps that become overdue on this date or between this date and the date the program is run.
<b>Reminder interval</b>	Interval, in days, at which further reminder mails should be sent. For example, if you enter 3, reminder mails are sent every three days.
<b>Ignore next reminder</b>	Select if the next due reminder email should not be sent. For example, if the next reminder email is due on Friday and you run the /EBY/PDWC_DUE_DATE_CHECK program on Thursday, no reminder will be sent on the Friday. However, a reminder email is sent directly after the program has run.
<b>Collective email</b>	One email containing all workflow step notifications for the processor since the last program run is sent.
<b>Email per document</b>	An individual email for each document is sent. <b>Note:</b> This setting only applies to receipt and reminder emails.

## /EBY/PDBO\_REPETITOR

Use the /EBY/PDBO\_REPETITOR program to automatically check and post documents at regular intervals. The document status is automatically changed after checking/posting takes place. The program can also be called with the transaction /EBY/PDBO\_REPETITOR.

### General selection criteria

Setting	Description
<b>Object type</b>	Process type for which documents will be checked/posted.
<b>PD document number</b>	Document or range of documents to be checked/posted.
<b>Document status</b>	Document statuses to be checked. For example, it makes no sense to include documents that have already been posted.
<b>Substate</b>	Document substatuses to be checked.
<b>Workflow</b>	Workflow statuses of documents to be checked. For example, you may want to check only approved documents.
<b>Created on</b>	Date on which the document was created.
<b>Changed on</b>	Date on which the document was last changed.

### Actions

Setting	Description
<b>Check and post</b>	Select to check the documents and post them if there are no errors.
<b>Check only</b>	Select to check the documents only.

### Messages and application log

Setting	Description
<b>Display messages</b>	Select to display the messages generated during the program run.

Setting	Description
<b>Create application log</b>	Select to create an application log that you can view in transaction SLG1. Enter the object /EBY/PD and the subobject /EBY/PD_REPETITOR in SLG1.

## /EBY/PDBO\_CONFIG\_DUMP

Use the /EBY/PDBO\_CONFIG\_DUMP program to download the entire configuration of a PROCESS DIRECTOR project and store it in an XML file. The saved XML file can be uploaded to the same system or to other systems. You can also compare two project configurations, irrespective of whether these are stored in the system or in XML files. The program can also be run with transaction /EBY/PDBO\_CONF\_DUMP.

### Task to perform

Select whether you want download a configuration to an XML file, upload from an XML file, or compare one configuration with another.

In the **Set of tables** field you can specify which tables should be processed (customizing and system tables) during upload, download and comparison.

Variant	Description
<b>CUS&amp;EPD_FLDCAT</b>	Exclude the field catalog from comparison
<b>CUS&amp;EPD_MENU</b>	Exclude the menu entries from comparison
<b>CUS&amp;EPD_WRKLST</b>	Exclude the Worklist entries from download
<b>CUS&amp;IPD_WRKLST</b>	Download only the Worklist entries
<b>CUS&amp;I_COCKPIT</b>	Download only PROCESS DIRECTOR Accounts Payable tables

### Download XML file

Setting	Description
<b>To file</b>	Name of the XML file in which the configuration will be stored.

Setting	Description
<b>Of document type</b>	<p>The document type to be downloaded. It includes independent tables of all the document types. When a table is document dependent:</p> <p>It contains all the entries where OBJ is empty.</p> <p>It contains all the entries where OBJ is equal to that specified on the selection screen.</p>
<b>From project</b>	Project containing the configuration that will be stored.
<b>Include system tables</b>	<p>Activate this option to include not only customized tables but also standard PROCESS DIRECTOR system tables in the downloaded XML file.</p> <p>This can be useful, for example, to check whether data in system tables has changed, or whether there are differences in configuration between two systems. Use the <a href="#">Compare</a> function to compare system tables.</p>

### Upload XML file

Only data from customized tables is uploaded. If system tables were included when the configuration was downloaded to XML, the data from these tables will not be uploaded.

**Note:** Uploading a Worklist configuration to a project in which the Worklist has already been configured is not recommended.

Setting	Description
<b>Request/Task</b>	Name of a customizing request. All uploaded data is included in this request, which can then be released and transported to another system.
<b>From file</b>	Name of the XML file from which the configuration will be uploaded.
<b>To project</b>	Name of the project to which the configuration will be uploaded.

Setting	Description
<b>Remove before upload</b>	<p>Activate this option to remove existing entries in any PD or PDAP customizing table stored in the uploaded file.</p> <p>As it is possible to selectively download and upload parts of the configuration, only entries in those tables that are available in the XML file are removed before the upload; other tables remain untouched. You can also download and upload PROCESS DIRECTOR Accounts Payable tables.</p> <p>If you deactivate this option:</p> <ul style="list-style-type: none"> <li>▪ Entries that are present in the system database, but not in the XML file, are retained.</li> <li>▪ Entries that are present in the XML file, but not in the system, are added to the database.</li> <li>▪ Entries in the XML file overwrite identical entries in the database.</li> </ul>

## Compare

Setting	Description
<b>Ignore field 'project'</b>	<p>Activate this option to exclude the project ID from the comparison (recommended).</p> <p>If this option is deactivated, the compared project configurations are considered to be different even if they are identical apart from the project ID.</p>
<b>Include system tables</b>	<p>Activate this option to include system tables as well as customized tables in the comparison.</p> <p>If you are comparing XML files, the system tables must have been included when the configuration was downloaded to the XML file (see <a href="#">Include system tables</a>).</p>
<b>Element 1, Element 2</b>	<p>Select the elements that you want to compare. For example, to compare a stored XML file with a project configuration in the system:</p> <p>In the <b>Element 1</b> section, select <b>Local file</b> and enter the file path and name.</p> <p>In the <b>Element 2</b> section, select <b>Customizing on server</b> and select a project.</p>

When you compare configurations, the differences are displayed in the ABAP Splitscreen Editor. See the SAP ABAP documentation for information on working with this editor.

Configuration of project:		Configuration of project: HT	
4323		4323	0000000041 0000000036 000005 S0  50
4324		4324	
4325	/EBY/PDBO_CWLSCT	4325	/EBY/PDBO_CWLSCT
4326	0000000001 E	4326	0000000001 E
4327	0000000002 E  With error	4327	0000000002 E  Requisitions
4328	0000000003 E  Sales Order	4328	0000000003 E  Goods Receipts
4329	0000000004 E  Processed	4329	0000000004 E  Ordered
4330	0000000025 E  Processed requests	4330	0000000005 E  To approve
4331	0000000026 E  New requests	4331	0000000006 E  My Requisitions
4332	0000000027 E  Vendor attachments	4332	0000000007 E  Drafts
4333	0000000028 E  Accounting View Maintenance	4333	0000000008 E  Approved
4334	0000000029 E  Purchasing View Maintenance	4334	0000000009 E  In approval
4335	0000000030 E  Other	4335	0000000010 E  Not posted
4336	0000000031 E  Travel Expense, Old Prototype	4336	0000000011 E  Posted
4337	0000000032 E  Submitted	4337	0000000012 E  With errors
4338	0000000033 E  New	4338	0000000013 E  In workflow

## /EBY/PDBO\_CONFIG\_GEN\_WORKLIST

Use the /EBY/PDBO\_CONFIG\_GEN\_WORKLIST program to generate the PROCESS DIRECTOR Worklist. This can also be done manually in the Worklist configuration screen.

Parameter	Description
<b>Document batch size</b>	The number of documents that will be processed in one batch.
<b>Remove orphan entries</b>	Removes orphan entries from the table that contains the document assignment.

## /EBY/PDBO\_CHECK\_NESTEDSET\_NEW

Use the /EBY/PDBO\_CHECK\_NESTEDSET\_NEW program to regenerate the mapping of documents to static Worklist nodes.

This program maps documents to the correct Worklist nodes, according to the nodes' ranges configuration. For example, if a node's ranges configuration specifies `STATE = 'OK'` (document status = posted), all documents with this status will be mapped to and therefore displayed in the node.

Normally, you only need to run this program if you are having problems with Worklist generation.

## **/EBY/PDBO\_GEN\_FAILED\_NSTDSETS**

Use the /EBY/PDBO\_GEN\_FAILED\_NSTDSETS program to generate the mapping of skipped documents to static Worklist nodes.

When the Worklist is generated, documents that are currently locked because they are being edited by users are skipped by the Worklist generation program, which means that they may be displayed in the wrong node, or may not be displayed at all. These documents are written to the /EBY/PDBO\_TWLF table. The /EBY/PDBO\_GEN\_FAILED\_NSTDSETS program reads this table and generates the mapping to the correct Worklist nodes for these documents, provided the documents are no longer locked.

Normally, you only need to run this program if the Worklist generation messages indicate that documents were skipped. The program should be run repeatedly until all skipped documents are processed successfully.

## **/EBY/PDBO\_CHECK\_WL\_DOC\_COUNT**

Use the /EBY/PDBO\_CHECK\_WL\_DOC\_COUNT program to check or restore Worklist document counters for static nodes.

This program calculates the current number of documents assigned to static nodes, checks these numbers against the database and displays any discrepancy errors. Check the **Repair** box before running the program if the new values should replace the old ones in the database.

Normally, you only need to run this program if you suspect that Worklist node counters are incorrect.

## **/EBY/PDBO\_CONFIG\_WRKL\_ENV\_CONV**

In PROCESS DIRECTOR 7.3, by default the Worklist node **Not visible in** option is activated for all environments except SAP GUI and Web Application (such as the **Mobile Application** environment), so that the Worklist node is not displayed for those environments. If a node has been created prior to 7.3, this option is not activated by default and must be set retrospectively for all nodes by running the /EBY/PDBO\_CONFIG\_WRKL\_ENV\_CONV program. The option can then be deactivated individually for nodes that should be visible in that environment.

## **/EBY/PDBO\_CHANGE\_ARC\_LINKS**

Use the /EBY/PDBO\_CHANGE\_ARC\_LINKS program to assign archiving links for attachments to a different PROCESS DIRECTOR object type.

## User Exits/BAdIs

PROCESS DIRECTOR provides the following predefined user exits / BAdIs as extension points for customer coding. In addition to these standard platform user exits / BAdIs, process type specific user exits / BAdIs are available for individual process types. See the *PROCESS DIRECTOR Advanced Configuration Guide* for more information on developing BAdI implementations.

BAdI	Description
<b>Basic</b>	
<a href="#"><u>Screen PBO/PAI</u></a>	Add custom logic for the screen display. For example, you can use this BAdI to display or verify values.
<a href="#"><u>Modify Worklist</u></a>	Add custom logic for the PROCESS DIRECTOR Worklist configuration.
<a href="#"><u>Modify field profiles</u></a>	Add custom logic when field statuses are applied.
<a href="#"><u>Set Worklist node of a single document link</u></a>	Add custom logic to open a specific Worklist node when a user clicks a single document link in a workflow notification email or SAP Business Workflow work item.
<a href="#"><u>Determination results</u></a>	Add custom logic after determinations have been performed.
<a href="#"><u>Rejection reason</u></a>	Add custom logic before and after a rejection reason is entered.
<b>Web Application</b>	
<a href="#"><u>Set messages</u></a>	Add custom logic to modify messages sent to the Web Application.
<a href="#"><u>Set actions</u></a>	Add custom logic to modify which actions are available to the Web Application.
<a href="#"><u>User logon</u></a>	Add custom logic to modify the behavior of the application during logon to the Web Application.

BAdI	Description
<b>Workflow</b>	
<a href="#">Workflow email handling</a>	Add custom logic for handling workflow emails.
<a href="#">Workflow handling</a>	Add custom logic for handling workflows.
<a href="#">Workflow steps handling</a>	Add custom logic for handling workflow steps.

## BAdI Screen PBO/PAI

### User exits / BAdIs > Platform > Basic

This platform user exit / BAdI currently provides the interface methods [PBO](#), [PAI](#), [SET\\_PA\\_I\\_OKCODE](#) and [EXCLUDE\\_FUNCTION\\_CODES](#) that you can implement to add your own customer logic for the screen display. For example, you can use this BAdI to display or verify values.

### PBO method

#### Calling sequence

This method is called in the SAP GUI when the screen is refreshed after an action is processed.

#### Method signature

Type	Parameter	Description
	IR_SELOBJECT	Selected object
	IR_MAINSCREEN	Main screen object
	IC_EDITMODE	GUI edit mode
	ERROR_OCCURRED	

## PAI method

### Calling sequence

This method is called in the SAP GUI before an action is processed.

### Method signature

Type	Parameter	Description
	IR_SELOBJECT	Selected object
	IR_MAINSCREEN	Main screen object
	CC_OKCODE	Screens, function code triggered by PAI
	ERROR_OCCURRED	

## SET\_PA\_I\_OKCODE method

### Calling sequence

This action is called in the SAP GUI before an action is processed.

### Method signature

Type	Parameter	Description
	IR_SELOBJECT	Selected object
	IR_MAINSCREEN	Main screen object
	CC_OKCODE	Screens, function code triggered by PAI
	ERROR_OCCURRED	

## EXCLUDE\_FUNCTION\_CODES method

### Calling sequence

This method is called in the SAP GUI when the screen is refreshed after an action is processed.

### Method signature

Type	Parameter	Remarks
	IT_UCOMM_MAPPING	Map GUI commands to actions
	IT_SELECTED_OBJECTS	Table of /EBY/CL_PDBO (any PD document)
	IR_MAINSCREEN	Main screen object
	CT_EXCLUDED_FCODES	<p>This parameter is pre-populated with the list of function codes excluded for any of these reasons:</p> <ul style="list-style-type: none"> <li>▪ not assigned to an action</li> <li>▪ it is the function code of a button that calls the detail screen, but we are already in the detail screen</li> <li>▪ because of state</li> <li>▪ not authorized</li> </ul> <p>It is possible to re-enable a disabled action by deleting its function code from CT_EXCLUDED_FCODES. To do this, you need to know the function code mapping of the action.</p>
	ERROR_OCCURRED	

## BAdI Modify Worklist

### User exits / BAdIs > Platform > Basic

This platform user exit / BAdI currently provides the interface method [MODIFY\\_WORKLIST](#) that you can implement to add your own customer logic for the PROCESS DIRECTOR Worklist configuration.

Typical usages of this user exit / BAdI are to:

- [change the currently selected Worklist node](#)

- [change the Worklist construction](#), for example, by disabling some nodes (greying them out) or removing them entirely

## MODIFY\_WORKLIST method

### Calling sequence

In the SAP GUI environment, this user exit is called during PROCESS DIRECTOR startup and when the **Refresh** toolbar button  is clicked (action REFRESH\_ALL\_OBJECTS Refresh list of objects).

1. At PROCESS DIRECTOR startup when the Worklist tree is built.
2. At PROCESS DIRECTOR startup when the user's previous Worklist node is restored (action=STARTPROCESSDIRECTOR, /EBY/CL\_PDVI\_FORW\_REFRESH does RAISE EVENT refresh\_data which triggers /EBY/CL\_PDVI\_WORKLIST\_TREE->EXPEND\_NODES\_FOR\_SELECTED which calls REFRESH\_TREE\_FULL).
3. After pressing refresh button (fcode=REFRESH, action=REFRESH\_TREE\_AND\_OBJ, /EBY/CL\_PDVI\_FORW\_RSH\_WRK\_TREE does RAISE EVENT refresh\_tree which triggers /EBY/CL\_PDVI\_MAINSCREEN->ON\_REFRESH\_TREE which calls REFRESH\_TREE\_FULL).

REFRESH\_TREE\_FULL calls the BAdI.

In the Web Application environment, this user exit is called from the action WA\_GET\_TREE\_DATA Retrieve Worklist data XML, immediately after retrieving the last Worklist node visited by the user.

### Method signature

Type	Parameter	Remarks
	IR_CONTEXT	Reference to the user session context
	IR_WORKLIST	
	CR_SELECTED_NODE	
	ERROR_OCCURRED	

### Changing the currently selected Worklist node

To change the currently selected node, simply assign a Worklist node to the CHANGING parameter CR\_SELECTED\_NODE.

- ❗ **Note:** Under certain conditions, such reassignments of the currently selected Worklist node can be overridden by an implementation of the user exit / BAdI [Set Worklist node of a single document link](#).

One way to obtain an existing Worklist node is to search for it by node ID via the method `IR_WORKLIST->FIND_NODE2`. This only works for static nodes and semi-dynamic nodes which are not under dynamic nodes, because this method uses the static node ID taken from the Worklist configuration.

## Changing the Worklist construction

Removing nodes completely is not recommended, since you can lose access to a document in case a change in its properties causes it to be assigned to a removed node.

- ❗ **Warning:** Document access can be lost even if a removed node becomes visible again after the BAdI implementation is deactivated, because the document won't be visible in that node until the next update action is performed on it—which can never happen unless document is visible in some other node.
- ❗ **Note:** Since this user exit / BAdI is called *after* Worklist node authorization checks have been performed, any nodes added to the Worklist by a **Modify Worklist** implementation are *not* subject to authorization checks (unless you explicitly perform authorization checks in your coding).

To make a Worklist node invisible, Call `IR_WORKLIST->GET_NODE_FOR_EXT_ID` to find a node using its external ID, and then set `MB_INVISIBLE = ABAP_TRUE` on that node.

## BAdI Modify field profiles

User exits / BAdIs > Platform > Basic > Modify field profiles

This user exit / BAdI currently provides the interface method [MODIFY\\_FIELD\\_STATES](#) that you can implement to add your own customer logic when field statuses are applied.

### MODIFY\_FIELD\_STATES method

#### Method signature

Type	Parameter	Remarks
▶	IC_OBJ	Object type
▶	IC_COMP_TYPE	Screen component type
▶▶	IC_GRID_NO	Grid number

Type	Parameter	Remarks
▶▶	IC_DMILEVEL	Logical level of the internal data model
▶▶	IC_GRID_CONF	Grid configuration
▶▶	IT_OBJECTS	Table of objects
▶▶	IR_CONTEXT	Reference to the user session context
▶▶	CR_STATES	Field states / field profile

## BAdI Set Worklist node of a single document link

### User exits / BAdIs > Platform > Basic

When a user clicks a single document link in a workflow notification email or SAP Business Workflow work item, the document will open up in a default Worklist node that is calculated from the document GUID.

This platform user exit / BAdI currently provides the interface method [SET\\_NODE](#) that you can implement to add your own calculation routine for determining this default Worklist node.

You can change to any node that the document belongs to.

**Note:** The document will not be displayed in the overview list if you change to a node that does not contain the document as part of its document selection.

### SET\_NODE method

#### Calling sequence

This method is only called when resolving SAP Business Workplace work items and single document links in workflow notification emails. This method is called *after* the user exit / BAdI [Modify Worklist](#).

#### Method signature

Type	Parameter	Remarks
▶	IR_CONTEXT	Reference to the user session context

Type	Parameter	Remarks
▶□	VALUE ( FLT_VAL )	Filter values passed implicitly to the BAdI and explicitly to the method.
▶▶	CR_NODE	Worklist node

## BAdI Determination results

### User exits / BAdIs > Platform > Basic > Determination results

This user exit / BAdI currently provides the interface method `PROCESS` that you can implement to add your own customer logic after determinations have been performed.

### PROCESS method

#### Calling sequence

This method is called after determinations have been performed, but before checks run.

#### Method signature

Type	Parameter	Remarks
▶□	IC_OBJ_TYPE	Document type
▶□	IS_COORDINATES	Structure containing information about the current action chain, action and envelope
▶▶	IR_CONTEXT	Action chain context
▶▶	IS_BO_HEADER	Basic document header
▶▶	CR_OBJECT	Document object

## BAdI Rejection reason

### User exits / BAdIs > Platform > Basic > Rejection reason

This user exit / BAdI currently provides the interface methods [BEFORE\\_REJECT](#) and [AFTER\\_REJECT](#) that you can implement to add your own customer logic before and after a rejection reason is entered.

## BEFORE\_REJECT method

This method can be used, for example, to exclude mandatory rejection reasons/notes for a specific company code, or change the rejection reason descriptions.

### Calling sequence

This method is called before the **Rejection reason** popup is displayed.

### Method signature

Type	Parameter	Remarks
▶□	IR_OBJECT	Abstract PROCESS DIRECTOR basis object
▶▶	CS_REJECT_REQ	Document rejection requirements
▶▶	CT_REASON	Document rejection reason descriptions

## AFTER\_REJECT method

This method can be used, for example, to change values in the document or create a PDF note after document rejection.

### Calling sequence

This method is called after the user selects a rejection reason.

### Method signature

Type	Parameter	Remarks
▶□	IR_OBJECT	Abstract PROCESS DIRECTOR basis object

## BAdI Set messages

### User exits / BAdIs > Platform > Web Application

This user exit / BAdI currently provides the interface method `SET_MESSAGES` that you can implement to add your own customer logic to modify messages sent to the Web Application.

## SET\_MESSAGES method

### Method signature

Type	Parameter	Description
▶□	IR_CONTEXT	Reference to the user session context
▶□	IR_WA_CONTEXT	Web Application session context
▶□	VALUE ( FLT_VAL )	Filter values passed implicitly to the BAdI and explicitly to the method.
▶▶	CT_MESSAGES	Messages with localized text

## BAdI Set actions

### User exits / BAdIs > Platform > Web Application

This user exit / BAdI currently provides the interface method [SET\\_ACTIONS](#) that you can implement to add your own customer logic to modify which actions are available to the Web Application. You can use this method to enable any action, even if it is not in the Web Application's base action pool.

## SET\_ACTIONS method

### Method signature

Type	Parameter	Description
▶□	IR_CONTEXT	Reference to the user session context
▶□	IR_WA_CONTEXT	Web Application session context
▶▶	CT_ACTIONS	List of actions for the Web Application
▶□	VALUE ( FLT_VAL )	Filter values passed implicitly to the BAdI and explicitly to the method.

## BAdI User logon

### User exits / BAdIs > Platform > Web Application

This user exit / BAdI currently provides the interface methods [SSO\\_USER\\_CHECK](#), [USER\\_CHECK](#) and [USER\\_PW\\_CHANGE](#) that you can implement to add your own customer logic to modify the behavior of the application during logon to the Web Application.

PROCESS DIRECTOR provides a standard implementation for Accounts Payable, /EBY/ICWC\_USER\_LOGON, which should be activated if PROCESS DIRECTOR should call User Exit 923, User check during logon to the Web Application, when users log on to the Web Application or change their password (either the initial password or the password in the user profile).

### SSO\_USER\_CHECK method

#### Calling sequence

This method is called during user logon to the Web Application from a mobile device. It can be used to extend the search logic for the user ID, and indicates whether or not a user can access the Web Application from a mobile device.

#### Method signature

Type	Parameter	Description
▶	VALUE( IC_SSO_ID )	SSO ID used for logon to the Web Application
▶	VALUE( IC_USER_ID )	ID used for logon to the Web Application
▶	VALUE( IC_USER_TYPE )	PROCESS DIRECTOR user type
▶	VALUE( IT_TMAL )	Table of logons with SSO ID
▶▶	CB_ALLOWED	

### USER\_CHECK method

#### Calling sequence

This method is called during user logon to the Web Application. It can be used to provide additional logic for authenticating the user, or to replace the user who logs in with an

alternative user. In the latter case, checks such as checking the password, or checking whether the account is locked, are performed with the logon user ID, but authorization checks are performed against the data of the alternative user. For example: Dave has authorization to display the Financial Postings Worklist node, but Roger does not. Dave logs on to the Web Application, but the Financial Postings node is not displayed because Roger (the alternative user) does not have authorization to display it.

### Method signature

Type	Parameter	Description
▶□	VALUE( IC_USER_ID )	Current processor
▶□	VALUE( IC_USER_TYPE )	PROCESS DIRECTOR user type
▶□	VALUE( IC_PASSWORD )	PROCESS DIRECTOR user password type
▶▶	CS_USER_DATA	Structure containing user data
▶▶	CS_MESSAGE	Return parameter
▶▶	CB_NO_STANDARD	

### USER\_PW\_CHANGE method

#### Calling sequence

This method is called when users change their password in the Web Application. It can be used to extend the logic for authenticating the user.

#### Method signature

Type	Parameter	Description
▶□	VALUE( IC_USER_ID )	Current processor
▶□	VALUE( IC_USER_TYPE )	PROCESS DIRECTOR user type

Type	Parameter	Description
▶□	VALUE ( IC_PASSWORD )	PROCESS DIRECTOR user password type
▶□	VALUE ( IC_NEW_PASSWORD )	PROCESS DIRECTOR user password type for new password
▶▶	CS_MESSAGE	Return parameter
▶▶	CB_NO_STANDARD	

## BAI Workflow email handling

### User exits / BAdIs > Platform > Workflow

This user exit / BAdI currently provides the interface method [BEFORE\\_CREATE\\_MAIL](#) that you can implement to add your own customer logic for handling workflow emails.

### BEFORE\_CREATE\_MAIL method

#### Calling sequence

This method is called before a workflow email is created. Workflow emails are sent out by the [/EBY/PDWC DUE DATE CHECK](#) program.

#### Method signature

Type	Parameter	Description
▶□	IS_COORDINATES	Structure containing information about the current action chain, action and envelope
▶□	IC_WC_ID	Workflow ID
▶□	IS_WC_HDR	Structure containing workflow header data
▶□	IS_WC_CONF	Structure containing workflow configuration

Type	Parameter	Description
▶□	IT_WC_ALL_STEPS	Table containing all workflow steps and their data
▶□	IC_STEP_ID	Workflow step ID. This parameter refers to the step that is being processed by the BAdI method, not by the user.
▶□	IS_STEP_CONF	Structure containing workflow step configuration. This parameter refers to the step that is being processed by the BAdI method, not by the user.
▶□	IS_STEP_DATA	Structure containing workflow step data. This parameter refers to the step that is being processed by the BAdI method, by the user.
▶□	IN_STEP_POSITION	This value represents the position of the step that is currently being processed by the user.  <b>Note:</b> When the workflow is in the start phase but has not yet started, the value is set to 0. The same value is set when the workflow has already finished.
▶□	IT_STEP_USER_TYPES	User types available for the step. This parameter refers to the step that is being processed by the BAdI method, not by the user.
▶□	IT_STEP_PROCESSORS	List containing all processors of the current and next steps
▶□	IT_STEP_APPROVERS	List containing all step approvers of the current and previous steps
▶□	IT_NEXT_STEPS	List containing all next steps with their details
▶□	IT_CURR_STEPS	List containing all current steps with their details
▶□	IT_PREV_STEPS	List containing all previous steps with their details

Type	Parameter	Description
▶□	IR_OBJECT	Base class of all ABAP objects
▶▶	CB_CANCEL_MAIL	Indicates if the email should be created
▶▶	CS_SENDER	PROCESS DIRECTOR user and user type
▶▶	CT_RECIPIENTS	List of user IDs of email recipients
▶▶	CT_CC_RECIPIENTS	List of user IDs of email CC recipients
▶▶	CC_SUBJECT	Email subject
▶▶	CT_TEXT	Email text
▶▶	CC_LANGU	Email language
▶▶	CT_ALIASES	Aliases and their replacements
▶▶	CC_MAIL_PURPOSE	Email purpose
▶▶	CC_EMAIL_TYPE	Email type

## BApI Workflow handling

### User exits / BAdIs > Platform > Workflow

This user exit / BAdI currently provides the interface methods [BEFORE\\_START](#), [AFTER\\_START](#) and [AFTER\\_FINISH](#) that you can implement to add your own customer logic for handling workflows.

This user exit / BAdI is filter-dependent and allows you to filter on object (process type) and workflow.

### BEFORE\_START method

This method can be used, for example, to determine whether the workflow is valid or not.

## Calling sequence

This method is called before a workflow starts.

## Method signature

Type	Parameter	Description
▶□	IS_COORDINATES	Structure containing information about the current action chain, action and envelope
▶□	IC_WC_ID	Workflow ID
▶□	IS_WC_HDR	Structure containing workflow header data
▶□	IS_WC_CONF	Structure containing workflow configuration
▶□	IT_WC_ALL_STEPS	Table containing all workflow steps and their data
▶□	IR_WORKFLOW	Reference to the workflow object
▶□	IR_CURRENT_DOCUMENT	Reference to the current document
▶□	IR_CONTEXT	Reference to the user session context
▶□	VALUE ( FLT_VAL )	Filter values passed implicitly to the BAdI and explicitly to the method. Structure contains object type and workflow ID.
▶▶	CB_INVALID	To render the workflow invalid, set this parameter to ABAP_TRUE.

## AFTER\_START method

### Calling sequence

This method is executed for all workflows that are pending for the document.

## Method signature

Type	Parameter	Description
▶□	IS_COORDINATES	Structure containing information about the current action chain, action and envelope
▶□	IC_WC_ID	Workflow ID
▶□	IS_WC_HDR	Structure containing workflow header data
▶□	IS_WC_CONF	Structure containing workflow configuration
▶□	IT_WC_ALL_STEPS	Table containing all workflow steps and their data
▶□	IR_WORKFLOW	Reference to the workflow object
▶□	IR_CURRENT_DOCUMENT	Reference to the current document
▶□	IR_CONTEXT	Reference to the user session context
▶□	VALUE (FLT_VAL)	Filter values passed implicitly to the BAdI and explicitly to the method. Structure contains object type and workflow ID.

## AFTER\_FINISH method

### Calling sequence

This method is only called when the workflow is finished (that is, the last step of the workflow has been approved).

## Method signature

Type	Parameter	Description
▶	IS_COORDINATES	Structure containing information about the current action chain, action and envelope
▶	IC_WC_ID	Workflow ID
▶	IS_WC_HDR	Structure containing workflow header data
▶	IS_WC_CONF	Structure containing workflow configuration
▶	IT_WC_ALL_STEPS	Table containing all workflow steps and their data
▶	IR_WORKFLOW	Reference to the workflow object
▶	IR_CURRENT_DOCUMENT	Reference to the current document
▶	IR_CONTEXT	Reference to the user session context
▶	VALUE (FLT_VAL)	Filter values passed implicitly to the BAdI and explicitly to the method. Structure contains object type and workflow ID.

## BAdI Workflow steps handling

### User exits / BAdIs > Platform > Workflow

This user exit / BAdI currently provides the interface methods [CHECK\\_RECEIVER\\_VALIDITY](#), [CHECK\\_STEP\\_VALIDITY](#), [USER\\_DETERMINATION](#), [AFTER\\_FINISHED\\_STEP](#), [GET\\_ITEM\\_APPROVER](#) and [CHECK\\_ACTION](#) that you can implement to add your own customer logic for handling workflow steps.

This user exit / BAdI is filter-dependent and allows you to filter on object (process type), workflow and workflow step.

## Explanation of terms

Term	Explanation
Current steps	Steps for which the state is <b>Active</b> , that is, a processor is currently working on them.
Next steps	Steps whose position in the workflow definition is the same or greater than the position of the current step. In the case of steps with the same position as the current step, only those steps for which step processing has not yet started are taken into account. If the workflow has not yet started, all steps are considered as next steps.
Previous step(s)	Steps for which the state is <b>Done</b> . Invalid steps are not considered as previous steps.
Step being processed by the user / Steps being processed by the BAdI	<p>This differentiation is due to the architecture used for the BAdI implementation.</p> <p>The <i>step being processed by the user</i> refers to the step that the user is currently processing in the PROCESS DIRECTOR application.</p> <p>When the user performs an action that triggers the BAdI, for some methods all steps of the workflow must be processed. This is performed in a loop, one step at a time. In this case, the PROCESS DIRECTOR application processes not only steps that are being processed by the user, but also other steps in the workflow - these are then <i>steps being processed by the BAdI</i>.</p>

## CHECK\_RECEIVER\_VALIDITY method

This method can be used, for example, to determine whether a user is valid for a specific workflow action or not.

### Calling sequence

This method is called in the following cases:

- For the APPROVE action – for next steps only
- For the CONTINUE action – for current and next steps
- For the WORKFLOW START action – for current and next steps
- For the FORWARD action – only for the step currently being processed by the user
- For the QUERY action – only for the step currently being processed by the user

## Method signature

Type	Parameter	Remarks
▶	IS_COORDINATES	Structure containing information about the current action chain, action and envelope
▶	IC_WC_ID	Workflow ID
▶	IS_WC_HDR	Structure containing workflow header data
▶	IS_WC_CONF	Structure containing workflow configuration
▶	IT_WC_ALL_STEPS	Table containing all workflow steps and their data
▶	IC_STEP_ID	Workflow step ID. This parameter refers to the step that is being processed by the BAdI method, not by the user.
▶	IS_STEP_CONF	Structure containing workflow step configuration. This parameter refers to the step that is being processed by the BAdI method, not by the user.
▶	IS_STEP_DATA	Structure containing workflow step data. This parameter refers to the step that is being processed by the BAdI method, by the user.
▶	IN_STEP_POSITION	This value represents the position of the step that is currently being processed by the user.  <b>Note:</b> When the workflow is in the start phase but has not yet started, the value is set to 0. The same value is set when the workflow has already finished.
▶	IT_STEP_USER_TYPES	User types available for the step. This parameter refers to the step that is being processed by the BAdI method, not by the user.

Type	Parameter	Remarks
▶□	IT_STEP_PROCESSORS	List containing all processors of the current and next steps
▶□	IT_STEP_APPROVERS	List containing all step approvers of the current and previous steps
▶□	IT_NEXT_STEPS	List containing all next steps with their details
▶□	IT_CURR_STEPS	List containing all current steps with their details
▶□	IT_PREV_STEPS	List containing all previous steps with their details
▶□	IS_RECEIVER	The step recipient is passed here
▶□	IR_STEP	Reference to the step object
▶□	IR_RECEIVER	Reference to the step receiver object
▶□	IR_CURRENT_DOCUMENT	Reference to the current document
▶□	IR_CONTEXT	Reference to the user session context
▶□	VALUE ( FLT_VAL )	Filter values passed implicitly to the BAdI and explicitly to the method. Structure contains object type, workflow ID and step ID.
▶▶	CB_INVALID	To render the recipient invalid, set this parameter to ABAP_TRUE.

### **CHECK\_STEP\_VALIDITY method**

This method can be used, for example, to determine whether a step is valid or not.

## Calling sequence

This method is called in the following cases:

- For the APPROVE action – for next steps only
- For the CONTINUE action – for current and next steps
- For the WORKFLOW START action – for current and next steps

This method is called *after* workflow step conditions (if any) have been evaluated for the given step.

## Method signature

Type	Parameter	Remarks
▶	IS_COORDINATES	Structure containing information about the current action chain, action and envelope
▶	IC_WC_ID	Workflow ID
▶	IS_WC_HDR	Structure containing workflow header data
▶	IS_WC_CONF	Structure containing workflow configuration
▶	IT_WC_ALL_STEPS	Table containing all workflow steps and their data
▶	IC_STEP_ID	Workflow step ID. This parameter refers to the step that is being processed by the BAdI method, not by the user.
▶	IS_STEP_CONF	Structure containing workflow step configuration. This parameter refers to the step that is being processed by the BAdI method, not by the user.
▶	IS_STEP_DATA	Structure containing workflow step data. This parameter refers to the step that is being processed by the BAdI method, by the user.

Type	Parameter	Remarks
▶	IN_STEP_POSITION	This value represents the position of the step that is currently being processed by the user.  <b>Note:</b> When the workflow is in the start phase but has not yet started, the value is set to 0. The same value is set when the workflow has already finished.
▶	IT_STEP_USER_TYPES	User types available for the step. This parameter refers to the step that is being processed by the BAdI method, not by the user.
▶	IT_STEP_PROCESSORS	List containing all processors of the current and next steps
▶	IT_STEP_APPROVERS	List containing all step approvers of the current and previous steps
▶	IT_NEXT_STEPS	List containing all next steps with their details
▶	IT_CURR_STEPS	List containing all current steps with their details
▶	IT_PREV_STEPS	List containing all previous steps with their details
▶	IS_RECEIVER	The step recipient is passed here
▶	IR_STEP	Reference to the step object
▶	IR_RECEIVER	Reference to the step receiver object
▶	IR_CURRENT_DOCUMENT	Reference to the current document
▶	IR_CONTEXT	Reference to the user session context

Type	Parameter	Remarks
▶□	VALUE ( FLT_VAL )	Filter values passed implicitly to the BAdI and explicitly to the method. Structure contains object type, workflow ID and step ID.
▶▶	CB_INVALID	To render the step invalid, set this parameter to ABAP_TRUE.

## USER\_DETERMINATION method

This method can be used, for example, to add or remove workflow step processors.

### Calling sequence

This method is only called for valid steps. It is called in the following cases:

- For the APPROVE action – for next steps only
- For the CONTINUE action – for current and next steps
- For the WORKFLOW START action – for current and next steps
- For the FORWARD action – only for the step currently being processed by the user
- For the QUERY action – only for the step currently being processed by the user

### Method signature

Type	Parameter	Description
▶□	IS_COORDINATES	Structure containing information about the current action chain, action and envelope
▶□	IC_WC_ID	Workflow ID
▶□	IS_WC_HDR	Structure containing workflow header data
▶□	IS_WC_CONF	Structure containing workflow configuration
▶□	IT_WC_ALL_STEPS	Table containing all workflow steps and their data

Type	Parameter	Description
▶	IC_STEP_ID	Workflow step ID. This parameter refers to the step that is being processed by the BAdl method, not by the user.
▶	IS_STEP_CONF	Structure containing workflow step configuration. This parameter refers to the step that is being processed by the BAdl method, not by the user.
▶	IS_STEP_DATA	Structure containing workflow step data. This parameter refers to the step that is being processed by the BAdl method, by the user.
▶	IN_STEP_POSITION	This value represents the position of the step that is currently being processed by the user.  <b>Note:</b> When the workflow is in the start phase but has not yet started, the value is set to 0. The same value is set when the workflow has already finished.
▶	IT_STEP_USER_TYPES	User types available for the step. This parameter refers to the step that is being processed by the BAdl method, not by the user.
▶	IT_STEP_PROCESSORS	List containing all processors of the current and next steps
▶	IT_STEP_APPROVERS	List containing all step approvers of the current and previous steps
▶	IT_NEXT_STEPS	List containing all next steps with their details
▶	IT_CURR_STEPS	List containing all current steps with their details
▶	IT_PREV_STEPS	List containing all previous steps with their details
▶	IR_STEP	Reference to the step object

Type	Parameter	Description
▶□	IR_CURRENT_DOCUMENT	Reference to the current document
▶□	IR_CONTEXT	Reference to the user session context
▶□	VALUE ( FLT_VAL )	Filter values passed implicitly to the BAdI and explicitly to the method. Structure contains object type, workflow ID and step ID.
▶▶	CT_USERS	List of user IDs

## AFTER\_FINISHED\_STEP method

### Calling sequence

This method is called for all steps whose status is **Done** for the actions APPROVE, RECALL COMPLETE, RECALL PROCESSOR, REJECT, REPLY.

This method is called *after* a step is approved, rejected or recalled. If the step is a [final step](#), this method is called before the [Workflow handling](#) user exit / BAdI method [AFTER\\_FINISH](#).

### Method signature

Type	Parameter	Remarks
▶□	IS_COORDINATES	Structure containing information about the current action chain, action and envelope
▶□	IC_WC_ID	Workflow ID
▶□	IS_WC_HDR	Structure containing workflow header data
▶□	IS_WC_CONF	Structure containing workflow configuration
▶□	IT_WC_ALL_STEPS	Table containing all workflow steps and their data

Type	Parameter	Remarks
▶	IC_STEP_ID	Workflow step ID. This parameter refers to the step that is being processed by the BAdl method, not by the user.
▶	IS_STEP_CONF	Structure containing workflow step configuration. This parameter refers to the step that is being processed by the BAdl method, not by the user.
▶	IS_STEP_DATA	Structure containing workflow step data. This parameter refers to the step that is being processed by the BAdl method, by the user.
▶	IN_STEP_POSITION	This value represents the position of the step that is currently being processed by the user.  <b>Note:</b> When the workflow is in the start phase but has not yet started, the value is set to 0. The same value is set when the workflow has already finished.
▶	IT_STEP_USER_TYPES	User types available for the step. This parameter refers to the step that is being processed by the BAdl method, not by the user.
▶	IT_STEP_PROCESSORS	List containing all processors of the current and next steps
▶	IT_STEP_APPROVERS	List containing all step approvers of the current and previous steps
▶	IT_NEXT_STEPS	List containing all next steps with their details
▶	IT_CURR_STEPS	List containing all current steps with their details
▶	IT_PREV_STEPS	List containing all previous steps with their details

Type	Parameter	Remarks
▶	IR_STEP	Reference to the step object
▶	IR_CURRENT_DOCUMENT	Reference to the current document
▶	IR_CONTEXT	Reference to the user session context
▶	VALUE (FLT_VAL)	Filter values passed implicitly to the BAdI and explicitly to the method. Structure contains object type, workflow ID and step ID.

### GET\_ITEM\_APPROVER method

This method can be used, for example, to assign processors for line item approval. The approver is returned using the `CS_APPROVER` parameter and should be one of the step processors, who are available in the `IT_LINE_PROCESSORS` table.

### Calling sequence

This method is called for active steps for which user assignment is necessary and line item approval is enabled. Each item of the document is processed in a loop. This method is called on workflow start and step approval.

### Method signature

Type	Parameter	Description
▶	IS_COORDINATES	Structure containing information about the current action chain, action and envelope
▶	IC_WC_ID	Workflow ID
▶	IS_WC_HDR	Structure containing workflow header data
▶	IS_WC_CONF	Structure containing workflow configuration
▶	IT_WC_ALL_STEPS	Table containing all workflow steps and their data

Type	Parameter	Description
▶	IC_STEP_ID	Workflow step ID. This parameter refers to the step that is being processed by the BAdl method, not by the user.
▶	IS_STEP_CONF	Structure containing workflow step configuration. This parameter refers to the step that is being processed by the BAdl method, not by the user.
▶	IS_STEP_DATA	Structure containing workflow step data. This parameter refers to the step that is being processed by the BAdl method, by the user.
▶	IN_STEP_POSITION	This value represents the position of the step that is currently being processed by the user.  <b>Note:</b> When the workflow is in the start phase but has not yet started, the value is set to 0. The same value is set when the workflow has already finished.
▶	IT_STEP_USER_TYPES	User types available for the step. This parameter refers to the step that is being processed by the BAdl method, not by the user.
▶	IT_STEP_PROCESSORS	List containing all processors of the current and next steps
▶	IT_STEP_APPROVERS	List containing all step approvers of the current and previous steps
▶	IT_NEXT_STEPS	List containing all next steps with their details
▶	IT_CURR_STEPS	List containing all current steps with their details
▶	IT_PREV_STEPS	List containing all previous steps with their details

Type	Parameter	Description
▶□	IT_LINE_PROCESSORS	Possible processors for the step
▶□	IR_STEP	Reference to the step object
▶□	IR_CURRENT_DOCUMENT	Reference to the current document
▶□	IR_CONTEXT	Reference to the user session context
▶□	VALUE ( FLT_VAL )	Filter values passed implicitly to the BAdI and explicitly to the method. Structure contains object type, workflow ID and step ID.
▶▶	CS_APPROVER	Approver data
▶▶	CB_PRESET_APPROVED	Preset the <b>Approved</b> flag

## CHECK\_ACTION method

### Calling sequence

This method is called before any other workflow BAdI method. It is called during the approve, reject, forward and query actions.

## Method signature

Type	Parameter	Remarks
▶	IC_ACTION	Action for which the BAdI is executed (domain /EBY/PDWC_MACTIVITY): <ul style="list-style-type: none"> <li>▪ APR - approve</li> <li>▪ QRY - query</li> <li>▪ FWD - forward</li> <li>▪ RJC - reject</li> </ul>
▶	IS_COORDINATES	Structure containing information about the current action chain, action and envelope
▶	IC_WC_ID	Workflow ID
▶	IS_WC_HDR	Structure containing workflow header data
▶	IS_WC_CONF	Structure containing workflow configuration
▶	IT_WC_ALL_STEPS	Table containing all workflow steps and their data
▶	IC_STEP_ID	Workflow step ID. This parameter refers to the step that is being processed by the BAdI method, not by the user.
▶	IS_STEP_CONF	Structure containing workflow step configuration. This parameter refers to the step that is being processed by the BAdI method, not by the user.
▶	IS_STEP_DATA	Structure containing workflow step data. This parameter refers to the step that is being processed by the BAdI method, by the user.

Type	Parameter	Remarks
▶	IN_STEP_POSITION	This value represents the position of the step that is currently being processed by the user.  <b>Note:</b> When the workflow is in the start phase but has not yet started, the value is set to 0. The same value is set when the workflow has already finished.
▶	IT_STEP_USER_TYPES	User types available for the step. This parameter refers to the step that is being processed by the BAdI method, not by the user.
▶	IT_STEP_PROCESSORS	List containing all processors of the current and next steps
▶	IT_STEP_APPROVERS	List containing all step approvers of the current and previous steps
▶	IT_NEXT_STEPS	List containing all next steps with their details
▶	IT_CURR_STEPS	List containing all current steps with their details
▶	IT_PREV_STEPS	List containing all previous steps with their details
▶	IR_STEP	Reference to the step object
▶	IR_CURRENT_DOCUMENT	Reference to the current document
▶	IR_CONTEXT	Reference to the user session context
▶	VALUE ( FLT_VAL )	Filter values passed implicitly to the BAdI and explicitly to the method. Structure contains object type, workflow ID and step ID.

Type	Parameter	Remarks
	EB_CANCEL	To render the action on a step invalid, you can set this parameter to <code>ABAP_TRUE</code> .
	ET_MESSAGES	List of messages to be added to the queue. Message of type 'E' (error) has the same meaning as the parameter <code>EB_CANCEL</code> set to <code>ABAP_TRUE</code> .

# Process types reference

This section provides reference information for the following process types:

- [Accounts Payable \(IV\)](#)
- [Asset Acquisitions \(AA\)](#)
- [Asset Retirement \(AR\)](#)
- [Asset Transfer \(AT\)](#)
- [Customer Orders \(SO\)](#)
- [Electronic Bank Statements \(ES\)](#)
- [Financial Postings \(FI\)](#)
- [Generic Archiving \(ARCH\\_BUS\)](#)
- [Goods Receipts \(DN\)](#)
- [Invoice Block and Cancellation \(IB\)](#)
- [Master Data Maintenance \(MD\)](#)
- [Order Confirmations \(OR\)](#)
- [Payment Advices \(PA\)](#)
- [Payment Approvals \(PR\)](#)
- [Requisitions \(PO\)](#)

## Accounts Payable (IV)

SAP Object type: /EBY/PDIV

Package: /EBY/ICIV

This chapter provides reference information for Accounts Payable. For detailed information on configuring Accounts Payable in PROCESS DIRECTOR, see the *PROCESS DIRECTOR Accounts Payable Configuration Guide*.

[Document model customization](#)

[View model customization](#)

[Document statuses](#)

[Text types](#)

[Programs](#)

## Document model customization

The document model provides an interface structure for each logical level containing all fields that are available for that level. Using the customer includes (CIs) provided in this structure, you can extend the document model to include customer-specific fields. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

- Use the include ending in `_DATA` if the custom field should be automatically persisted in the database.
  - Use the include ending in `_DISP` if you only want to automatically calculate a field at runtime and display it.
- ⓘ Important:** Never add customer-specific fields directly to the customer include, but use your own customer-specific structure instead. For example, first create a customer-specific structure corresponding to each used CI, include that structure in the CI and put all customer-specific fields in the customer-specific structure. **Do not use an APPEND structure.** Set the enhancement category of the customer-specific structure to **Can Be Enhanced, Character-Type or Numeric-Type** (menu **Extras**). Never use the enhancement category **Can Be Enhanced (Deep)**. See the SAP documentation for more information.

Logical level	Package	Interface structure	Customer Includes
Header data	/EBY/ICIV_MODEL	/EBY/ICIV_SHDR_IF	CI_COCKPIT_HDR
Item data	/EBY/ICIV_MODEL	/EBY/ICIV_SITM_IF	CI_COCKPIT_ITEM
Account assignment data	/EBY/ICIV_MODEL	/EBY/ICIV_SPOACCASS_IF	
Tax data	/EBY/ICIV_MODEL	/EBY/ICIV_STAX_IF	CI_COCKPIT_TAX

## View model customization

This section provides detailed information about which settings to enter when creating a customer view model for Accounts Payable documents. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

### By GUI element

- SAP document overview list:

**Display Incoming Invoice #10000000000001 (1 of 100)**

PROCESS DIRECTOR by ReadSoft

Worklist	Total
Requisitions	100/1.317
Goods Receipts	100/276
Financial Postings	100/308
Customer Orders	100/1.627
Order Confirmations	43/43
<b>Incoming Invoices</b>	<b>100/1.148</b>
With purchase order	100/470
General ledger	100/677
Posted	100/283
With errors	100/428
Payment Advices	69/69
Payment Approvals	100/174
Cost center maintenance	41/41
Profit Center Maintenance	57/57
G/L Account Maintenance	100/127
Asset Master Maintenance	100/178
Vendor Master Maintenance	100/239

Status	WF status	Follow-...	PD doc number	FI / MM	Doc..Date	Vendor	Partner name	Purch.Doc.
			10000000000001	MM	04.04.2012	1060	ReadSoft Ebydos AG	4500015376

It...	Purch.Doc.	Item	AccAss	POAccAss	Net	Items grid	Component type: Blank, Grid No.: 2, Logical level ITEMS
1	4500015376	10					uction test 1
2	4500015376	20			40,00 0,00	1 PC	Invoice reduction test 2

G/L Acct	Net amount	Gross	Text	Cost Ctr	Ord	Accounts grid	Component type: Blank, Grid No.: 3, Logical level ACCOUNT	Tax	Jur.
191120	80,00	0,00						VW	0,00

Purch.Doc.	Item	Material	Quantity	OUIn	Net	PO Items grid	Component type: Blank, Grid No.: 4, Logical level POITEM	Plant
4500015376	10		100	PC	120,0			test 1 1000
4500015376	20		100	PC	80,0000	PC 8.000,0000		Invoice reduction test 2 1000

- SAP document detail:

To maintain the SAP GUI detail screen, use SAP transaction /EBY/PDVI\_VDSTC.

- Web Application document overview list:

Worklist		Incoming Invoice (61)						
Refresh worklist		<b>Status</b>	<b>Doc. no.</b>	<b>FI / MM</b>	<b>Doc..Date</b>	<b>Vendor</b>	<b>Purch.Doc.</b>	<b>Reference</b>
<input type="checkbox"/> Requisition (156)		✉	5035	MM	04/14/10	35500	4500016075	56-1234
<input type="checkbox"/> Goods Receipt (344)		✉	5041				16051	6007432
<input type="checkbox"/> Financial Posting (65)		✉	5057				16075	56-1234
<input type="checkbox"/> Payment Advice (124)		✉	5062				16051	6007432
<input type="checkbox"/> Customer Order (367)		✉	5218	MM	09/27/10	20010	4500016046	271947
<b>Incoming Invoice (61)</b>		✉	5229	MM	09/27/10	20010	4500016046	271947
<input type="checkbox"/> Document Approval (12)								
<input type="checkbox"/> Payment Approval (154)								

- Web Application document detail:

Worklist

- Requisition (156)
- Goods Receipt (344)
- Financial Posting (65)
- Payment Advice (124)
- Customer Order (367)
- Incoming Invoice (61)**
- Document Approval (12)
- Payment Approval (154)
- Order Confirmations (2)
- Cost Center Maintenance (14)
- Vendor Master Maintenance (2)
- GL Account Maintenance (1)
- Profitcenter Maintenance (1)
- Asset Master Maintenance (1)
- All Workflows (150)
- My Workflow Inbox (31)

Document 5035

Save Accept Reject Forward Query Upload Status Back

Header

General	Invoice	Taxes (1)	Notes (0)	Attachments (1)
Document number	5035			
FI / MM Doc	MM			
Purchasing Document	4500016075			
Company Code	3000			

**Header data grid:**  
Component type: WA Document Detail, Grid No.: 1, Logical level: HEADER

Current messages

Task: Check price mismatch

Items (2)

Item	Purch.Doc.	Item	Net amount	Quantity	Order Unit	Text
000001	4500016075	000010	30.00	1.000	PC	

**Items grid:** Component type: WA Document Detail, Grid No.: 2, Logical level: ITEMS

Planned account assignments (1)

Purch.Doc.	Item	Quantity	Net amount	GL Acct
4500016075	00010	100,000.000	3,000,000.00	400000

**Planned account assignments grid:** Component type: WA Document Detail, Grid No.: 7, Logical level: ACCASS

PO data (3)

Item	Material	Quantity	Order Unit	Net price	Short text	Currency
00010	40-200R	0.000	PC	0.00	Bulb A 40/3x128 red EMG	USD
00020	ALT-PPV1	0.000	PC	0.00	ALT Ball bearing	USD
00030	400-151	0.000	PC	0.00	Temperature sensor	USD

**PO items grid:** Component type: WA Document Detail, Grid No.: 6, Logical level: POITEM

Accounts (1)

GL Acct	Net amount	Text	Cost Ctr
	0.00		

**Accounts grid:** Component type: WA Document Detail, Grid No.: 4, Logical level: ACCOUNT

- Workflow status:

**Workflow selection**

Workflow	Approve and create PO (2200)
Status description	Running
Started by	Blake Ev
Created on	08/22/2011
Time created	17:11:21
Workflow due date	08/30/2011

**Workflow status header grid:**  
Component type: Workflow status, Grid No.: 1,  
Logical level: HEADER

**Current messages**

Next

Status	Step	Date	Time	Processor	Overdue
	Create PO			Karina Kunz	

Current

Status	Step	Date	Time	Processor	Overdue
	Approve PO	08/22/2011	17:11:42	Roger Tillman	08/30/2011

Continue

### By view model component type

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
	1	HEADER	ZXY_PDIV_SGRID1 ZXY_PDIV_LGRID1	Header Header data
	2	ITEMS	ZXY_PDIV_SGRID2 ZXY_PDIV_LGRID2	Items Items
	3	ACCOUNT	ZXY_PDIV_SGRID4 ZXY_PDIV_LGRID4	Accounts Accounts
	4	POITEM	ZXY_PDIV_SGRID6 ZXY_PDIV_LGRID6	PO items PO items

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
AI Additional grid 01	1	TAX	ZXY_PDIV_SGRID3 ZXY_PDIV_LGRID3	Taxes Taxes
AI Additional grid 01	2	ACCASS	ZXY_PDIV_SGRID5 ZXY_PDIV_LGRID5	Account assignments Account assignments
AI Additional grid 01	4	PO_ACCASS	ZXY_PDIV_SAI_GRID5 ZXY_PDIV_LAI_GRID5	Planned account assignments Planned account assignments
WA Document Detail	1	HEADER	ZXY_PDIV_SWA_DOC_GRID1 ZXY_PDIV_LWA_DOC_GRID1	Remittance Advice Remittance Advice
WA Document Detail	2	ITEM	ZXY_PDIV_SWA_DOC_GRID2 ZXY_PDIV_LWA_DOC_GRID2	Items Items
WA Document Detail	3	TAX	ZXY_PDIV_SWA_DOC_GRID3 ZXY_PDIV_LWA_DOC_GRID3	Items Items
WA Document Detail	4	ACCOUNT	ZXY_PDIV_SWA_DOC_GRID4 ZXY_PDIV_LWA_DOC_GRID4	Items Items
WA Document Detail	5	ACCASS	ZXY_PDIV_SWA_DOC_GRID5 ZXY_PDIV_LWA_DOC_GRID5	Items Items
WA Document Detail	6	SORDER	ZXY_PDIV_SWA_DOC_GRID6 ZXY_PDIV_LWA_DOC_GRID6	Items Items
WA Document Detail	7	PO_ACCASS	ZXY_PDIV_SWA_DOC_GRID8 ZXY_PDIV_LWA_DOC_GRID8	Items Items
WA List Header	1	HEADER	ZXY_PDIV_SWA_LIST_GRID1 ZXY_PDIV_LWA_LIST_GRID1	Header Header data
Workflow status	1	HEADER	ZXY_PDIV_SWA_STAT_GRID1 ZXY_PDIV_LWA_STAT_GRID1	Header Header data
Header data for archive log	1	HEADER	ZXY_PDIV_SARC_LOG_GRID1 ZXY_PDIV_LARC_LOG_GRID1	Header Header data

## Document statuses

PROCESS DIRECTOR ships with these system document statuses for Accounts Payable:

Status	Status Description
01	Unprocessed
02	Erroneous
03	Posted
04	Deleted
05	Partially posted
06	In workflow
07	WEB CYCLE for information
08	WEB CYCLE on posted documents
09	Paid
10	Collective invoice completed
11	Collective invoice uncompleted
13	Rejected
23	Parked
B1	New document in WEB BOARD
B2	Deleted document in WEB BOARD
F1	Fast entry

## Text types

PROCESS DIRECTOR ships with this system text type for Accounts Payable:

Logical level	Text object	Description	ID	Description	Multiple
HEADER	/EBY/PD	PROCESS DIRECTOR texts	PDN	PROCESS DIRECTOR note	X

## Programs

The following programs are available for use with Accounts Payable:

- [/EBY/ICIV\\_DOC\\_SYNCHRONIZE](#)  
Registers existing Accounts Payable (former INVOICE COCKPIT) documents in PROCESS DIRECTOR.
- [/EBY/ICIV\\_ACCRUAL\\_REPORT](#)  
Collects aggregated accrual information from all un-posted PROCESS DIRECTOR Accounts Payable documents.

### **[/EBY/ICIV\\_DOC\\_SYNCHRONIZE](#)**

Use the /EBY/ICIV\_DOC\_SYNCHRONIZE program to register existing Accounts Payable (former INVOICE COCKPIT) documents in PROCESS DIRECTOR. It is necessary to run this program in order to be able to view and access Accounts Payable documents in PROCESS DIRECTOR.

You can specify which documents should be registered. For example, by default, documents that have already been completed in PROCESS DIRECTOR are excluded from registration, but you may want to include them.

Setting	Description
<b>Identification</b>	Unique document identifier (GUID).
<b>Document number</b>	PROCESS DIRECTOR document number.
<b>PROCESS DIRECTOR doc. type</b>	PROCESS DIRECTOR document type (defined in /COCKPIT/C11).

Setting	Description
<b>Document date</b>	Document date of the document (DOC_DATE).
<b>FI / MM doc</b>	Whether the document is an FI document or an MM document.
<b>Include finished documents</b>	Check to include documents that have already been processed and completed in PROCESS DIRECTOR (posted, parked or rejected documents and completed collective invoices).
<b>System ID</b>	When using the umbrella solution, specify the system ID of the remote system for which the documents should be synchronized to the umbrella system.

## **/EBY/ICIV\_ACCRUAL\_REPORT**

Use this report to collect aggregated accrual information from all un-posted PROCESS DIRECTOR Accounts Payable documents. The report displays all PROCESS DIRECTOR Accounts Payable line items that have not yet been posted. You can select line items in the report and create a PROCESS DIRECTOR Financial Posting document consisting of all selected line items and one aggregated offsetting / balancing position. The report can also be run with transaction /EBY/ICIV\_ACCR\_REP.

See the *PROCESS DIRECTOR Accounts Payable User Guide* for detailed information on using this report.

## **Asset Acquisitions (AA)**

SAP Object type: /EBY/PDAA

Package: /EBY/PDAA

This chapter provides reference information for Asset Acquisitions.

### **Checks for Asset Acquisitions**

The following check is available for Asset Acquisitions in the PROCESS DIRECTOR standard. You can also use [general checks](#) that are available for all process types.

#### **Check Asset Acquisition posting**

/EBY/CL\_PDAA\_EVT\_CHK\_POST

Simulates the creation of an asset acquisition posting in SAP and returns all SAP messages generated during the simulation. Stores the result of the simulation for further analysis.

## Document model customization

The document model provides an interface structure for each logical level containing all fields that are available for that level. Using the customer includes (CIs) provided in this structure, you can extend the document model to include customer-specific fields. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

- Use the include ending in `_DATA` if the custom field should be automatically persisted in the database.
  - Use the include ending in `_DISP` if you only want to automatically calculate a field at runtime and display it.
- 🔴 **Important:** Never add customer-specific fields directly to the customer include, but use your own customer-specific structure instead. For example, first create a customer-specific structure corresponding to each used CI, include that structure in the CI and put all customer-specific fields in the customer-specific structure. **Do not use an APPEND structure.** Set the enhancement category of the customer-specific structure to **Can Be Enhanced, Character-Type or Numeric-Type** (menu **Extras**). Never use the enhancement category **Can Be Enhanced (Deep)**. See the SAP documentation for more information.

Logical level	Package	Interface structure	Customer Includes
Header data	/EBY/PDAA_MODEL	/EBY/PDAA_SHDR_IF	CI_EBY_PDAA_SHDR_DATA CI_EBY_PDAA_SHDR_DISP
List of Assets data	/EBY/PDAA_MODEL	/EBY/PDAA_SLOA_IF	CI_EBY_PDAA_SLOA_DATA CI_EBY_PDAA_SLOA_DISP

## View model customization

This section provides detailed information about which settings to enter when creating a customer view model for Asset Acquisitions. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

## By GUI element

- SAP document overview list:

The screenshot displays the SAP PROCESS DIRECTOR interface. On the left is a 'Worklist' sidebar with categories like Purchase Order, Sales Order, Goods Receipt, etc. The main area shows a 'Header data grid' table with columns: St..., Wf..., PD doc no, Doc..Date, Posting Date, CoCd, Refer.Date, Text, and Att. ind. Below this is a 'List of Assets grid' table with columns: Asset, SNo., Amount, CrCy, Asset description, Quantity, Unit, CoCd, Year, and DocumentNo. Callouts identify the grid types: 'Header data grid: Component type: Blank, Grid No.: 1, Logical level: HEADER' and 'List of Assets grid: Component type: Blank, Grid No.: 2, Logical level: ITEMS'.

St...	Wf...	PD doc no	Doc..Date	Posting Date	CoCd	Refer.Date	Text	Att. ind.
59			18.04.2012	14.08.2012	1000			
58			17.04.2012	17.04.2012	1000	19.04.2012	acquisition	
57			17.04.2012	17.04.2012	1000	17.04.2012	Car acquisition consultants	
56			17.04.2012	17.04.2012	1000	17.05.2012	Car acquisition consultants	
55			16.04.2012	16.04.2012	1000			
54			08.03.2012	08.03.2012	1000	08.03.2012	Audi TT RS	
53								
52								
51								
50								
49			22.02.2012	22.02.2012	1000	22.02.2012	Audi Lieferung	
48	AF		01.02.2012	01.02.2012	2000	01.02.2012	Test	
47			08.01.2012	08.01.2012	2000	01.01.2012	2012 test posting	
46			08.02.2012	08.02.2012	1000	01.01.2012	New computer	
45	AF				2000			
44			03.04.2002	03.04.2002	1000	03.04.2002	VW Fuhrpark	
43			02.01.2002	15.01.2002	1000	02.01.2002	Valuation for Car 1234	
42			16.01.2002	17.01.2002	1000	17.01.2002	Asset acquisition	
41			17.01.2002	17.01.2002	1000	17.01.2002	New Laptops for Accounting	
40			16.01.2002	16.01.2002	1000			

Asset	SNo.	Amount	CrCy	Asset description	Quantity	Unit	CoCd	Year	DocumentNo
33	0	229,...	EUR	Audi A4	5	EA	1000	20	100000254
3390	0	98,5...	EUR	Audi A5	2	EA	1000	20	100000255

- SAP document detail:

To maintain the SAP GUI detail screen, use SAP transaction /EBY/PDVI\_VDSTC.

- Web Application document overview list:

The screenshot shows the SAP Web Application interface. On the left is a 'Worklist S47' sidebar with categories like Purchase Order, Sales Order, Goods Receipt, etc. The main area shows a 'Header data grid' table with columns: Status, Wf status, PD doc no, Type, Doc..Date, Postg Date, Period, CoCode, Trans.Type, and Ast.val.dt. Below this is a 'List of Assets grid' table with columns: Asset, SNo., Amount, CrCy, Asset description, Quantity, Unit, CoCd, Year, and DocumentNo. Callouts identify the grid types: 'Header data grid: Component type: WA List Header, Grid No.: 1, Logical level: HEADER' and 'List of Assets grid: Component type: Blank, Grid No.: 2, Logical level: ITEMS'.

Status	Wf status	PD doc no	Type	Doc..Date	Postg Date	Period	CoCode	Trans.Type	Ast.val.dt
		59			08/14/2012	0	1000		
		58		04/18/2012	04/17/2012	0	1000		04/19/2012
		57		04/17/2012	04/17/2012	0	1000		04/17/2012
		56		04/17/2012	04/17/2012	0	1000		05/17/2012
		55		04/16/2012	04/16/2012	0	1000		
		54		03/08/2012	03/08/2012	0	1000	120	03/08/2012
		51	AA		03/02/2012	0	1000	020	03/01/2012
		50		02/01/2002	02/01/2012	0	5000		02/01/2002
		49		02/22/2012	02/22/2012	0	1000	100	02/22/2012
		48	AF	02/01/2012	02/01/2012	0	2000	060	02/01/2012
		47		01/08/2012	01/08/2012	0	2000		01/01/2012
		46		02/08/2012	02/08/2012	0	1000		01/01/2012
		45	AF			0	2000	060	

Asset	SNo.	Amount	CrCy	Asset description	Quantity	Unit	CoCd	Year	DocumentNo
33	0	229,...	EUR	Audi A4	5	EA	1000	20	100000254
3390	0	98,5...	EUR	Audi A5	2	EA	1000	20	100000255

■ Web Application document detail:

The screenshot shows the 'Asset Acquisition 57' document detail page. The sidebar on the left lists various document types and counts, with 'Asset Acquisition' selected. The main content area is divided into several sections:

- Transaction data grid:**

Document status	●	
PD document number	57	
Workflow status		
Company Code	1000	IDES AG
Document Date	04/17/2012	
Posting Date	04/17/2012	
Reference Date	04/17/2012	
- Header data grid:**

Component type: WA Document Detail, Grid No.: 1, Logical level: HEADER
- List of Assets grid:**

Asset	Subnumber	Amount	Currency	Descript.	Quantity	Unit	CoCode	Fisc. Year	DocumentNo
3388	0	28,569.53	EUR	Audi A3	1.000	EA	1000	2012	100000248
- List of Assets grid (bottom):**

Component type: WA Document Detail, Grid No.: 2, Logical level: ITEMS
- Current messages:** For messages from previous steps use the "Audit trail" button

■ Workflow status:

The screenshot shows the 'Workflow status' page. The main content area includes:

- Workflow selection table:**

Workflow	Approve and create PO (2200)
Status description	Running
Started by	Blake Ev
Created on	08/22/2011
Time created	17:11:21
Workflow due date	08/30/2011
- Workflow status header grid:**

Component type: Workflow status, Grid No.: 1, Logical level: HEADER
- Current messages:** (Empty)
- Next section:**

Status	Step	Date	Time	Processor	Overdue
📄	Create PO			Karina Kunz	
- Current section:**

Status	Step	Date	Time	Processor	Overdue
✉	Approve PO	08/22/2011	17:11:42	Roger Tillman	08/30/2011
- Continue button:** (Checked)

## By view model component type

GUI area	Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming suggestions)	Btn. label Grid title (naming suggestions)
SAP document list		1	HEADER	ZXY_PDAA_SGRID1 ZXY_PDAA_LGRID1	Header Header data
SAP List of Assets grid		2	List of Assets	ZXY_PDAA_SGRID2 ZXY_PDAA_LGRID2	List of Assets List of Assets
Web Application document detail header data	WA Document Detail	1	HEADER	ZXY_PDAA_SWA_DOC_GRID1 ZXY_PDAA_LWA_DOC_GRID1	Header Header data
Web Application document detail List of Assets grid	WA Document Detail	2	List of Assets	ZXY_PDAA_SWA_DOC_GRID2 ZXY_PDAA_LWA_DOC_GRID2	List of Assets List of Assets
Web Application document list	WA List Header	1	HEADER	ZXY_PDAA_SWA_LIST_GRID1 ZXY_PDAA_LWA_LIST_GRID1	Header Header data
Workflow status popup	Workflow status	1	HEADER	ZXY_PDAA_SWA_STAT_GRID1 ZXY_PDAA_LWC_STAT_GRID1	Header Header data
Archive log	Header data for archive log	1	HEADER	ZXY_PDAA_SARC_LOG_GRID1 ZXY_PDAA_LARC_LOG_GRID1	Header Header data

## Document statuses

PROCESS DIRECTOR ships with these system document statuses/substatuses for Asset Acquisitions:

Status	Status Description
ER	In error or incomplete
NN	New or ready for processing
OK	Processed
XX	Rejected

## Text types

PROCESS DIRECTOR ships with this system text type for Asset Acquisitions:

Logical level	Text object	Description	ID	Description	Multiple
HEADER	/EBY/PD	PROCESS DIRECTOR texts	PDN	PROCESS DIRECTOR note	X

## Asset Retirement (AR)

SAP Object type: /EBY/PDAR

Package: /EBY/PDAR

This chapter provides reference information for Asset Retirement.

### Preset classes for Asset Retirement

The following preset class is available for Asset Retirement in the PROCESS DIRECTOR standard. You can also use [general preset classes](#) that are available for all process types.

#### Set substate based on transaction type

/EBY/CL\_PDAR\_PRESET\_SUBSTATE

Sets the document substatus based on the transaction type.

### Checks for Asset Retirement

The following check is available for asset retirement in the PROCESS DIRECTOR standard. You can also use [general checks](#) that are available for all process types.

#### Check Asset Retirement posting

/EBY/CL\_PDAR\_EVT\_CHK\_POST

Simulates the creation of an asset retirement posting in SAP and returns all SAP messages generated during the simulation. Stores the result of the simulation for further analysis.

### Document model customization

The document model provides an interface structure for each logical level containing all fields that are available for that level. Using the customer includes (CIs) provided in this structure,

you can extend the document model to include customer-specific fields. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

- Use the include ending in `_DATA` if the custom field should be automatically persisted in the database.
  - Use the include ending in `_DISP` if you only want to automatically calculate a field at runtime and display it.
- 🔴 **Important:** Never add customer-specific fields directly to the customer include, but use your own customer-specific structure instead. For example, first create a customer-specific structure corresponding to each used CI, include that structure in the CI and put all customer-specific fields in the customer-specific structure. **Do not use an APPEND structure.** Set the enhancement category of the customer-specific structure to **Can Be Enhanced, Character-Type or Numeric-Type** (menu **Extras**). Never use the enhancement category **Can Be Enhanced (Deep)**. See the SAP documentation for more information.

Logical level	Package	Interface structure	Customer Includes
Header data	/EBY/PDAR_MODEL	/EBY/PDAR_SHDR_IF	CI_EBY_PDAR_SHDR_DATA CI_EBY_PDAR_SHDR_DISP
List of Assets data	/EBY/PDAR_MODEL	/EBY/PDAR_SLOA_IF	CI_EBY_PDAR_SLOA_DATA CI_EBY_PDAR_SLOA_DISP

## View model customization

This section provides detailed information about which settings to enter when creating a customer view model for Asset Retirement. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

## By GUI element

- SAP document overview list:

**Display Asset Retirement #31 (2 of 32)**

Worklist: Total

- Purchase Order: 1,546/1,546
- Sales Order: 1,627/1,627
- Goods Receipt: 358/358
- Payment Advice: 50/69
- Financial postings: 722/722
- All documents in Web Cycle: 50/3,235
- Incoming Invoice: 1,418/1,407
- Quick Travel Expense: 71/71
- Payment Approval: 40/177
- Vendor Master Maintenance: 293/293
- Customer Master maintenance: 8/8
- G/L Account Maintenance: 133/133
- Cost Center Maintenance: 44/44
- Profitcenter Maintenance: 58/58
- Asset Master Maintenance: 195/195
- Asset Acquisition: 50/99
- Asset Transfer Posting: 21/21
- Asset Retirement: 32/32
  - New request: 22/22
  - Finished request: 10/10
- Order Confirmations: 65/65
- Vendor attachments: 24/24
- Invoice Blocking/Unblocking: 28/28
- Electronic Bank Statement: 34/34

Status	Document type	Wf status	PD doc no	CoCode	Text	Type	Posting Date	Transaction	Att. L...
◇			32	1000			06.03.2012	ABAV	
◇			31	1000	Sell Audi A4		17.04.2012	ABAO	
◇			30	1000			06.03.2012	ABAV	
◇			29	5000				ABAO	
◇			28	2000	TEST	AA	02.04.2002	ABAV	
◇			27	2000	Out of lifetime sale	AB	08.02.2012	ABAO	
◇			26	1000				ABAO	
◇			25	10			01.05.2002	ABAO	
◇			24	10			01.05.2002	ABAV	
◇			23	20			14.03.2002		
◇			22	20			30.03.2002		
◇			21						
◇			20	2000	Outdated computer		17.04.2002		
◇			19	1000	no item test		11.12.2002		
◇			18	1000			06.09.2011		
◇			17	1000	Scrap me if you can		02.09.2002		
◇			16	1000	New Asset retirement		31.08.2002		
◇			15	1000			29.08.2011		
◇			14	1000	Scrap me if you can		29.08.2002		
◇			13				31.08.2011		
◇			12				31.08.2011		

**Header data grid**  
Component type: Blank, Grid No.: 1, Logical level: HEADER

As...	SNo.	Retmt.rev.	Crcy	Descript.	CoCd	Year	DocumentNo
3	0	25,478.23	EUR	Audi A4	1000	2012	100000256

**List of Assets grid**  
Component type: Blank, Grid No.: 2, Logical level: ITEMS

- SAP document detail:

To maintain the SAP GUI detail screen, use SAP transaction /EBY/PDVI\_VDSTC.

- Web Application document overview list:

**Worklist: S47**

- Purchase Order: 1548
- Sales Order: 1630
- Goods Receipt: 363
- Payment Advice: 50
- Financial postings: 724
- All documents in Web Cycle: 50
- Incoming Invoice
- Quick Travel Expense: 71
- Payment Approval: 40
- Vendor Master Maintenance: 293
- Customer Master maintenance: 15
- G/L Account Maintenance: 133
- Cost Center Maintenance: 44
- Profitcenter Maintenance: 58
- Asset Master Maintenance: 195
- Asset Acquisition: 50
- Asset Transfer Posting: 21
- Asset Retirement: 32
  - New request: 22
  - Finished request: 10

**Asset Retirement (32)**

Status	Wf status	PD doc no	Type	Doc.Date	Postg Date	Period	Transact.	CoCode	Asset	Subnumber	Trans.Type
		32			03/06/2012	0	Asset Retirement by Scrapping	1000			
		31		04/17/2012	04/17/2012	0	Asset Sale Without Customer	1000			
		30			03/06/2012	0	Asset Retirement by Scrapping	1000			
		29				0	Asset Sale Without Customer	5000			
		28	AA	01/02/2002	04/02/2002	0	Asset Retirement by Scrapping	2000			101
		27	AB	02/08/2012	02/08/2012	0	Asset Sale Without Customer	2000			060
		26									
		25									
		24									
		23		03/02/2002	03/14/2002	0		2000			
		22		03/30/2002	03/30/2002	0		2000			
		21				0					
		20		04/17/2002	04/17/2002	0		2000			
		19		12/11/2002	12/11/2002	1		1000			
		18		09/06/2011	09/06/2011	0		1000			
		17		09/02/2002	09/02/2002	0		1000			
		16		08/31/2002	08/31/2002	0		1000			

**Header data grid**  
Component type: WA List Header, Grid No.: 1, Logical level: HEADER

■ Web Application document detail:

The screenshot shows the 'Asset Retirement 31' document detail page. The main content area contains a 'Transaction data' grid with the following information:

Document status	●	
PD document number	31	
Workflow status		
Company Code	1000	IDES AG
Document Date	04/17/2012	
Posting Date	04/17/2012	
Reference Date	06/17/2012	

A tooltip titled 'Header data grid' is displayed over the 'Transaction data' grid, showing: Component type: 'WA Document Detail, Grid No.: 1, Logical level: HEADER'.

At the bottom of the page, there is a 'List of Assets' grid:

Asset	Subnumber	Amount	Currency	Descript.	Co.Code	Fisc. Year	DocumentNo
3389	0	25,478.23	EUR	Audi A4	1000	2012	100000256

A tooltip titled 'List of Assets grid' is displayed over the 'List of Assets' grid, showing: Component type: 'WA Document Detail, Grid No.: 2, Logical level: ITEMS'.

■ Workflow status:

The screenshot shows the 'Workflow status' window. The 'Workflow selection' dropdown is set to 'Approve and create PO (2200)'. The 'Workflow status header grid' displays the following information:

Workflow	Approve and create PO (2200)
Status description	Running
Started by	Blake Ev
Created on	08/22/2011
Time created	17:11:21
Workflow due date	08/30/2011

A tooltip titled 'Workflow status header grid:' is displayed over the 'Workflow status header grid', showing: Component type: 'Workflow status, Grid No.: 1, Logical level: HEADER'.

The 'Current messages' panel is empty.

Below the header grid, there are sections for 'Next' and 'Current' workflow steps:

**Next**

Status	Step	Date	Time	Processor	Overdue
📄	Create PO			Karina Kunz	

**Current**

Status	Step	Date	Time	Processor	Overdue
✉️	Approve PO	08/22/2011	17:11:42	Roger Tillman	08/30/2011

A 'Continue' button is located at the bottom left of the window.

## By view model component type

GUI area	Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming suggestions)	Btn. label Grid title (naming suggestions)
SAP document list		1	HEADER	ZXY_PDAR_SGRID1 ZXY_PDAR_LGRID1	Header Header data
SAP List of Assets grid		2	List of Assets	ZXY_PDAR_SGRID2 ZXY_PDAR_LGRID2	List of Assets List of Assets
Web Application document detail header data	WA Document Detail	1	HEADER	ZXY_PDAR_SWA_DOC_GRID1 ZXY_PDAR_LWA_DOC_GRID1	Header Header data
Web Application document detail List of Assets grid	WA Document Detail	2	List of Assets	ZXY_PDAR_SWA_DOC_GRID2 ZXY_PDAR_LWA_DOC_GRID2	List of Assets List of Assets
Web Application document list	WA List Header	1	HEADER	ZXY_PDAR_SWA_LIST_GRID1 ZXY_PDAR_LWA_LIST_GRID1	Header Header data
Workflow status popup	Workflow status	1	HEADER	ZXY_PDAR_SWA_STAT_GRID1 ZXY_PDAR_LWC_STAT_GRID1	Header Header data
Archive log	Header data for archive log	1	HEADER	ZXY_PDAR_SARC_LOG_GRID1 ZXY_PDAR_LARC_LOG_GRID1	Header Header data

## Document statuses

PROCESS DIRECTOR ships with these system document statuses/substatuses for Asset Retirement:

### Statuses

Status	Status Description
ER	In error or incomplete
NN	New or ready for processing
OK	Processed
XX	Rejected

The following system document substatuses are available for asset retirement:

Substatus	Description
BS	Asset Retirement by Scrapping
SW	Asset Sale Without Customer

## Text types

PROCESS DIRECTOR ships with this system text type for Asset Retirement:

Logical level	Text object	Description	ID	Description	Multiple
HEADER	/EBY/PD	PROCESS DIRECTOR texts	PDN	PROCESS DIRECTOR note	X

## Asset Transfer (AT)

SAP Object type: /EBY/PDAT

Package: /EBY/PDAT

This chapter provides reference information for Asset Transfer.

### Preset classes for Asset Transfer

The following preset class is available for Asset Transfer in the PROCESS DIRECTOR standard. You can also use [general preset classes](#) that are available for all process types.

#### Set substate based on transaction type

/EBY/CL\_PDAT\_PRESET\_SUBSTATE

Sets the document substatus based on the transaction type.

### Checks for Asset Transfer

The following check is available for Asset Transfer in the PROCESS DIRECTOR standard. You can also use [general checks](#) that are available for all process types.

## Check Asset Transfer posting

/EBY/CL\_PDAT\_EVT\_CHK\_POST

Simulates the creation of an asset transfer posting in SAP and returns all SAP messages generated during the simulation. Stores the result of the simulation for further analysis.

## Document model customization

The document model provides an interface structure for each logical level containing all fields that are available for that level. Using the customer includes (CIs) provided in this structure, you can extend the document model to include customer-specific fields. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

- Use the include ending in `_DATA` if the custom field should be automatically persisted in the database.
  - Use the include ending in `_DISP` if you only want to automatically calculate a field at runtime and display it.
- ⓘ Important:** Never add customer-specific fields directly to the customer include, but use your own customer-specific structure instead. For example, first create a customer-specific structure corresponding to each used CI, include that structure in the CI and put all customer-specific fields in the customer-specific structure. **Do not use an APPEND structure.** Set the enhancement category of the customer-specific structure to **Can Be Enhanced, Character-Type or Numeric-Type** (menu **Extras**). Never use the enhancement category **Can Be Enhanced (Deep)**. See the SAP documentation for more information.

Logical level	Package	Interface structure	Customer Includes
Header data	/EBY/PDAT_MODEL	/EBY/PDAT_SHDR_IF	CI_EBY_PDAT_SHDR_DATA CI_EBY_PDAT_SHDR_DISP
List of Assets data	/EBY/PDAT_MODEL	/EBY/PDAT_SLOA_IF	CI_EBY_PDAT_SLOA_DATA CI_EBY_PDAT_SLOA_DISP

## View model customization

This section provides detailed information about which settings to enter when creating a customer view model for Asset Transfer. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

**By GUI element**

- SAP document overview list:

**Display Asset Transfer Posting #6 (17 of 21)**

Worklist

Worklist	Total
Purchase Order	1,546/1,546
Sales Order	1,627/1,627
Goods Receipt	358/358
Payment Advice	50/69
Financial postings	722/722
All documents in Web Cycle	50/3,235
Incoming Invoice	1,418/1,407
Quick Travel Expense	71/71
Payment Approval	40/177
Vendor Master Maintenance	293/293
Customer Master maintenance	8/8
G/L Account Maintenance	133/133
Cost Center Maintenance	44/44
Profitcenter Maintenance	58/58
Asset Master Maintenance	195/195
Asset Acquisition	50/59
Asset Transfer Posting	21/21
New request	7/7
Finished request	14/14
Asset Retirement	32/32
Order Confirmations	65/65
Vendor attachments	24/24
Invoice Blocking/Unblocking	28/28
Electronic Bank Statement	34/34

**Header data grid:**  
Component type: Blank, Grid No.: 1, Logical level: HEADER

Status	Wf status	PD doc no	CoCode	Type	Posting Date	Text	Att. ind.
◇		25	1000	AA	08.04.2012		
◇		24	1000		17.04.2012	intercompany transfer consultant	
◇		23	1000	AA	08.03.2012	BLART ship	
◇		22	1000		06.03.2012		
◇		21	5000		28.02.2012		
◇		20	1000		22.02.2012		
◇		19	1000		22.02.2012		
◇		18	1000		22.02.2012		
◇		17	1000		22.02.2012		
◇		16	1000		22.02.2012		
◇		15	1000		22.02.2012		
◇		14	1000		22.02.2012		
◇		13	1000		17.03.2002	New computer owner in another company	
◇		12	1000		17.02.2002	New computer owner	
◇		11	1000		16.01.2002		
◇		10	1000		01.05.2002	PI - check asset error message from TT	
◇		6	1000		01.03.2002	PI Test -cross comp - multi	
◇		5	1000	AA	01.03.2002	PI Test - intercomp - multi	
◇		3	1000	AA	01.03.2002	PI Test - cross	
◇		2	1000	AA	01.03.2002	PI Test	

**List of Assets grid:**  
Component type: Blank, Grid No.: 2, Logical level: ITEMS

CoCode	Asset	SNo.	Descript.	Manua
1000	1290	0	PI test	
1000	1289	0	PI test	

- SAP document detail:

To maintain the SAP GUI detail screen, use SAP transaction /EBY/PDVI\_VDSTC.

▪ Web Application document overview list:

Worklist S47		Asset Transfer Posting (21)							
Refresh worklist		Create	Check	Post	Status				
Purchase Order	1548	<input type="checkbox"/>	<input type="checkbox"/>		25	1000	AA	04/08/2012	Transfer within Company Code
Sales Order	1630	<input type="checkbox"/>	<input checked="" type="checkbox"/>		24	1000		04/17/2012	Intercompany Asset Transfer
Goods Receipt	363	<input type="checkbox"/>	<input type="checkbox"/>		23	1000	AA	03/08/2012	Transfer within Company Code
Payment Advice	50	<input type="checkbox"/>	<input type="checkbox"/>		22	1000		03/06/2012	Transfer within Company Code
Financial postings	724	<input type="checkbox"/>	<input type="checkbox"/>		21	5000		02/28/2012	Intercompany Asset Transfer
All documents in Web Cycle	50	<input type="checkbox"/>	<input type="checkbox"/>		20	1000		02/22/2012	Transfer within Company Code
Incoming Invoice		<input type="checkbox"/>	<input type="checkbox"/>		19	1000		02/22/2012	Transfer within Company Code
Quick Travel Expense	71	<input type="checkbox"/>	<input type="checkbox"/>						Company Code
Payment Approval	40	<input type="checkbox"/>	<input type="checkbox"/>						Asset Transfer
Vendor Master Maintenance	293	<input type="checkbox"/>	<input checked="" type="checkbox"/>		16	1000		02/08/2012	Transfer within Company Code
Customer Master maintenance	15	<input type="checkbox"/>	<input checked="" type="checkbox"/>		15	1000		02/02/2002	Intercompany Asset Transfer
G/L Account Maintenance	133	<input type="checkbox"/>	<input checked="" type="checkbox"/>		14	1000		02/17/2002	Intercompany Asset Transfer
Cost Center Maintenance	44	<input type="checkbox"/>	<input checked="" type="checkbox"/>		13	1000		03/17/2002	Intercompany Asset Transfer
Profitcenter Maintenance	58	<input type="checkbox"/>	<input checked="" type="checkbox"/>		12	1000		02/17/2002	Transfer within Company Code
Asset Master Maintenance	195	<input type="checkbox"/>	<input checked="" type="checkbox"/>		11	1000		01/16/2002	Transfer within Company Code
Asset Acquisition	50	<input type="checkbox"/>	<input checked="" type="checkbox"/>		10	1000		05/01/2002	Transfer within Company Code
<b>Asset Transfer Posting</b>	<b>21</b>								
New request	7								
Finished request	14								

Header data grid:  
Component type: WA List Header, Grid No.: 1, Logical level: HEADER

▪ Web Application document detail:

Worklist S47		Asset Transfer Posting 24												
Refresh worklist		Upload	Copy	Save	Back	Audit trail						Current messages		
Purchase Order	1548	Header											? For messages from previous steps use the "Audit trail" button	
Sales Order	1630	Header   Transaction data   Additional detail												
Goods Receipt	363	Document status <input checked="" type="checkbox"/>												
Payment Advice	50	Workflow status												
Financial postings	724	PD document number 24												
All documents in Web Cycle	50	Header data grid:												
Incoming Invoice		Component type: WA Document Detail, Grid No.: 1, Logical level: HEADER												
Quick Travel Expense	71	List of Assets (1)												
Payment Approval	40	CoCode	Asset	SNo.	Descrpt.	Amount	Crcy	PartCoCd	PrtrnAsset	P. sub-no.	Descrpt.	Fisc. Year	Docu	List of Assets grid:
Vendor Master Maintenance	293	1000	3388	0	Audi A3	28,569.53	EUR	2000	3005	0	Audi A3	2012	1000	Component type: WA Document Detail, Grid No.: 2, Logical level: ITEMS
Customer Master maintenance	15													
G/L Account Maintenance	133													
Cost Center Maintenance	44													
Profitcenter Maintenance	58													
Asset Master Maintenance	195													
Asset Acquisition	50													
<b>Asset Transfer Posting</b>	<b>21</b>													
New request	7													
Finished request	14													

- Workflow status:

### By view model component type

GUI area	Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming suggestions)	Btn. label Grid title (naming suggestions)
SAP document list		1	HEADER	ZXY_PDAT_SGRID1 ZXY_PDAT_LGRID1	Header Header data
SAP List of Assets grid		2	List of Assets	ZXY_PDAT_SGRID2 ZXY_PDAT_LGRID2	List of Assets List of Assets
Web Application document detail header data	WA Document Detail	1	HEADER	ZXY_PDAT_SWA_DOC_GRID1 ZXY_PDAT_LWA_DOC_GRID1	Header Header data

GUI area	Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming suggestions)	Btn. label Grid title (naming suggestions)
Web Application document detail List of Assets grid	WA Document Detail	2	List of Assets	ZXY_PDAT_SWA_DOC_GRID2 ZXY_PDAT_LWA_DOC_GRID2	List of Assets List of Assets
Web Application document list	WA List Header	1	HEADER	ZXY_PDAT_SWA_LIST_GRID1 ZXY_PDAT_LWA_LIST_GRID1	Header Header data
Workflow status popup	Workflow status	1	HEADER	ZXY_PDAT_SWA_STAT_GRID1 ZXY_PDAT_LWC_STAT_GRID1	Header Header data
Archive log	Header data for archive log	1	HEADER	ZXY_PDAT_SARC_LOG_GRID1 ZXY_PDAT_LARC_LOG_GRID1	Header Header data

## Document statuses

PROCESS DIRECTOR ships with these system document statuses/substatuses for Asset Transfer:

Status	Status Description
ER	In error or incomplete
NN	New or ready for processing
OK	Processed
XX	Rejected

The following system document substatures are available for asset transfer:

Substatus	Description
IC	Intercompany Asset Transfer
WI	Transfer within Company Code

## Text types

PROCESS DIRECTOR ships with this system text type for Asset Transfer:

Logical level	Text object	Description	ID	Description	Multiple
HEADER	/EBY/PD	PROCESS DIRECTOR texts	PDN	PROCESS DIRECTOR note	X

## Customer Orders (SO)

SAP Object type: /EBY/PDSO

Package: /EBY/PDSO

This chapter provides reference information for Customer Orders.

### Checks for Customer Orders

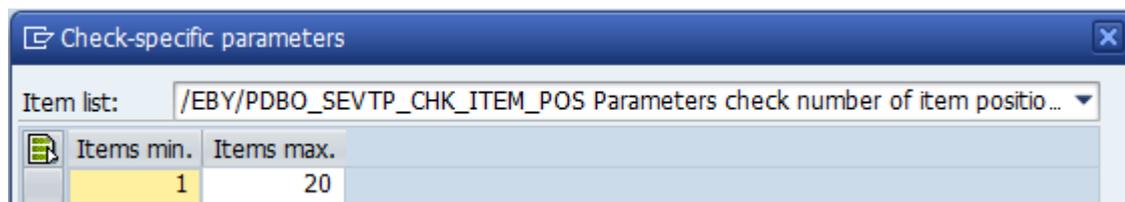
The following checks are available for Customer Orders in the PROCESS DIRECTOR standard. You can also use [general checks](#) that are available for all process types.

#### Check count of order item positions

/EBY/CL\_PDSO\_EVT\_CHK\_ITEMS

Checks for a required number of order line items and/or a maximum permitted number of line items. Also checks whether a material number has been entered for all line items.

Enter the minimum and maximum values in the initialization parameters.



For example, you can specify that a sales order must contain at least one line item and may have a maximum of 20 line items.

If one of these rules is broken, or if the material number is missing in any of the line items, a message is returned.

## Check if order is in reference to another document

/EBY/CL\_PDSO\_EVT\_CHK\_REFDOC

Checks whether the order should be created in reference to another document because a reference document is mentioned.

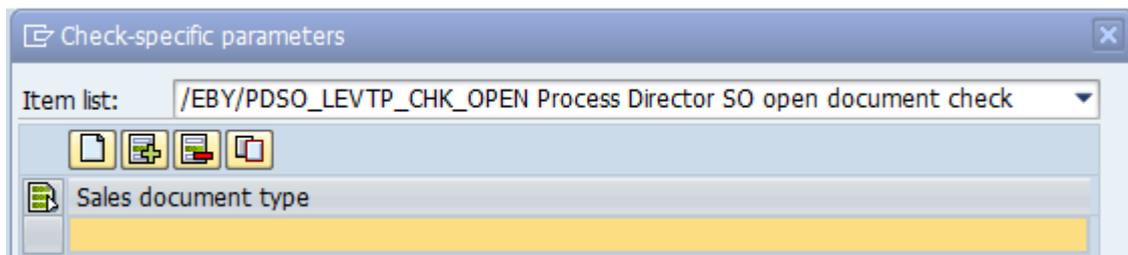
- ❗ **Note:** If a reference could be a contract, ensure that the [contract relation determination](#) has been called to verify the contract reference.

## Check if sold-to party has open documents

/EBY/CL\_PDSO\_EVT\_CHK\_OPEN

Checks whether an open contract or quotation exists for the given sold-to party and whether the **Reference doc.** field in the PROCESS DIRECTOR document is empty.

You can specify one or more sales document types in the initialization parameters.



This check is used to prevent automatic posting in systems where it is common to work with quotations, but the quotation number for the sales order has not been captured.

## Check incompleteness log

/EBY/CL\_PDSO\_EVT\_CHK\_INCOMP

Lists all entries from the incompleteness log.

This check is dependent on the order simulation, so the **Check whether an SAP sales order can be created** check must have run successfully first.

## Check incompleteness log of created/simulated order for price

/EBY/CL\_PDSO\_EVT\_CHK\_EDICOND

Checks the incompleteness log for entries specific to problems with the expected price and the simulated price.

This check is dependent on the order simulation, so the **Check whether an SAP sales order can be created** check must have run successfully first.

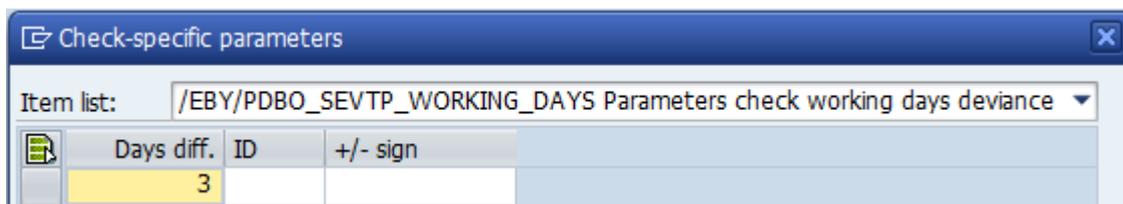
This check is only necessary if the condition types EDI1 or EDI2 are used in the calculation scheme and generate entries in the incompleteness log.

### Check item schedules regarding availability

/EBY/CL\_PDSO\_EVT\_CHK\_SCHEDULES

Checks calculated schedule dates on item level to indicate availability problems.

In the initialization parameters you can define allowed deviations from the target delivery date. If you also enter a factory calendar ID, the schedule dates calculation takes this calendar into account.

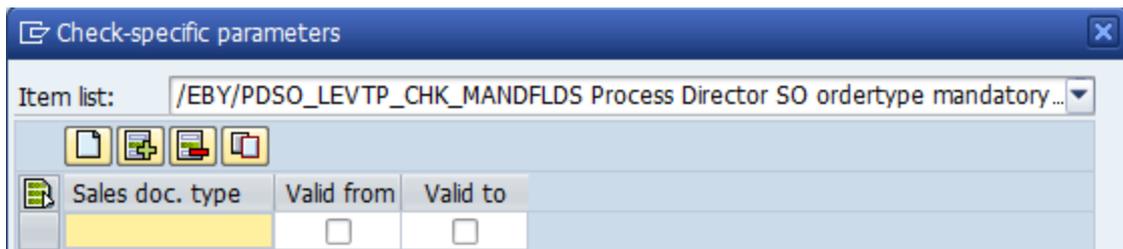


This check is dependent on the order simulation, so the **Check whether an SAP sales order can be created** check must have run successfully first.

### Check mandatory fields for order type

/EBY/CL\_PDSO\_EVT\_CHK\_MANDFLDS

Checks mandatory fields for special order types. Specify the document types to check in the initialization parameters.



#### Examples:

- Quotations need a **Valid to** date.
- Scheduling agreements need a **Valid from** and a **Valid to** date.

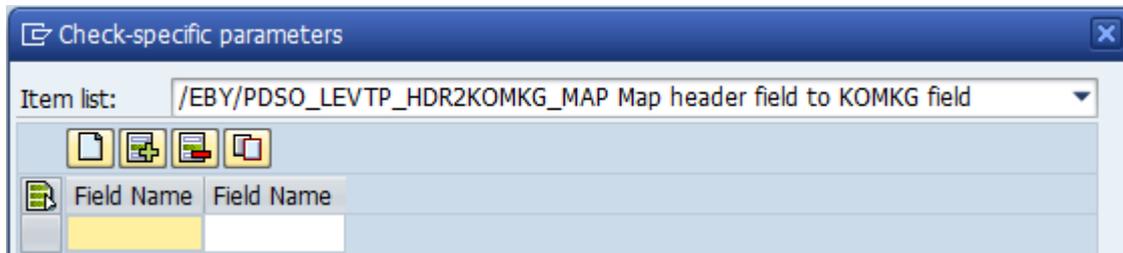
## Check material of items against exclusion/list logic

/EBY/CL\_PDSO\_EVT\_CHK\_MAT\_EXCL

Checks the Listing/Exclusion rules configured in SAP, which make it possible to allow or disallow the sale of specific materials to a specific customer.

This check returns an error message if a material that is excluded for the sold-to party is entered, or if the material is not included in the listed materials for the sold-to party.

By default, PROCESS DIRECTOR maps the sales organization, distribution channel, division, sales document type and partner number to the KOMKG structure. If additional header fields for Listing/Exclusion (such as customer-specific fields) are used in the KOMKG structure and should also be used for the check, these must be mapped to the corresponding PROCESS DIRECTOR fields. You can define this field mapping in the check's initialization parameters.



## Check minimum order quantity for all items

/EBY/CL\_PDSO\_EVT\_CHK\_MIN\_ORD\_Q

Checks whether the minimum order quantity is reached for every line item.

Different order units are taken into account.

## Check price discrepancies on item level

/EBY/CL\_PDSO\_EVT\_CHK\_ITEMPRICE

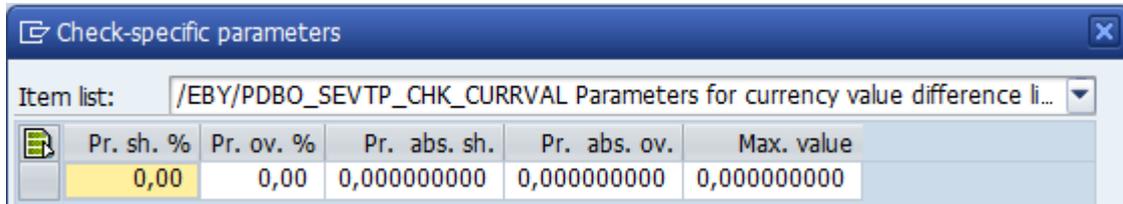
Checks price discrepancies on item level against allowed limits. Only possible for order items where a customer expected price is given.

In the initialization parameters, you can define the following rules:

- value shortfall in %
- value overrun in %
- absolute value shortfall
- absolute value overrun

- limit for standard line item value

If one of the defined rules is broken, a message is generated.



This check is dependent on the order simulation, so the **Check whether an SAP sales order can be created** check must have run successfully first.

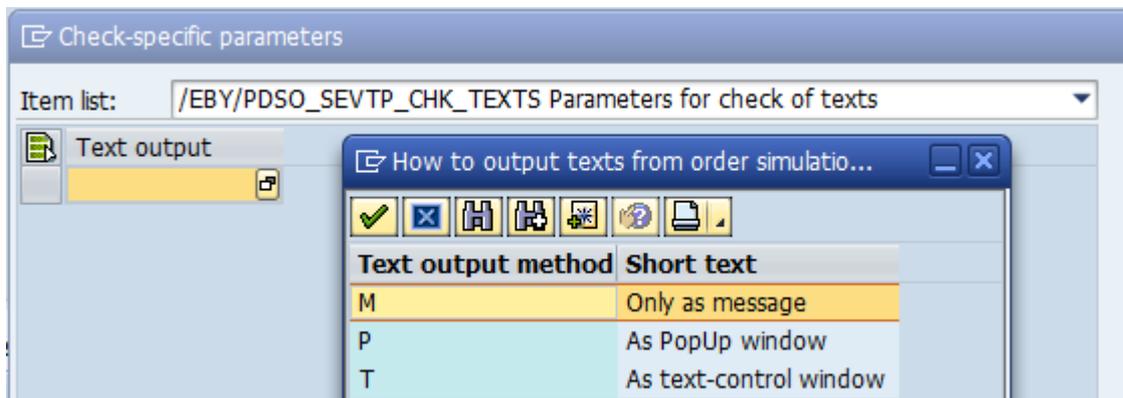
### Check sales related texts

/EBY/CL\_PDSO\_EVT\_CHK\_TEXTS

Checks for sales-related texts, such as:

- customer information
- material information
- order type related information

In the initialization parameters you can define how the texts are displayed:



Initialization parameter	Description
<b>M Only as message</b>	Displays the text as a message. Suitable for background processing, but with the limitation that only 200 characters are displayed and there are no formatting options. This is the default setting.

Initialization parameter	Description
<b>P As popup window</b>	Displays the text in a popup, with the limitation that only the first 4 lines of the text are displayed.
<b>T As text control window</b>	Displays the entire text in the text editor.

### Check whether an SAP sales order can be created

/EBY/CL\_PDSO\_EVT\_CHK\_CREATE

Simulates the creation of the sales order in SAP and returns all SAP messages generated during the simulation. Stores the result of the simulation for further analysis.

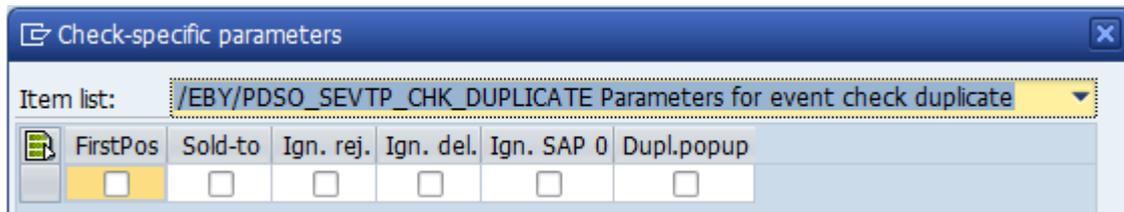
### Check whether PO number has been already used

/EBY/CL\_PDSO\_EVT\_CHK\_DUPLICATE

Checks whether the number in the **PO number** field has already been used.

First all PROCESS DIRECTOR documents, then all posted SAP orders are checked.

You can specify these initialization parameters:



Initialization parameter	Description
<b>FirstPos</b>	Select to compare the PO number and also the first line item of the order with existing orders (item number, material and quantity are compared).
<b>Sold-to</b>	Select to compare the PO number and also the sold-to party customer number with existing orders.

Initialization parameter	Description
<b>Ign. rej.</b>	Select to ignore PROCESS DIRECTOR documents that have been rejected. These are not considered as duplicates even if the PO number is the same as the checked document.
<b>Ign. del.</b>	Select to ignore PROCESS DIRECTOR documents that have been deleted.
<b>Ign. SAP 0</b>	Select to ignore SAP documents that have a total header value equal to zero. These are not considered as duplicates even if the PO number is the same as the checked document.
<b>Dupl.popup</b>	Select to display a popup if a purchase order number that already exists is captured or entered, and the duplicate check is configured accordingly. The user can then view the documents that have the same PO number, and choose to reject the current or original document, manually update the sales order in SAP, or delete the PROCESS DIRECTOR document. See the <i>PROCESS DIRECTOR SAP User Guide</i> for more information.

## Compare header net values of document with simulated order

/EBY/CL\_PDSO\_EVT\_CHK\_HDRNETVAL

If an expected order value is given, this can be checked against defined ranges.

In the initialization parameters, you can define the following rules:

- value shortfall in %
- value overrun in %
- absolute value shortfall
- absolute value overrun
- limit for standard line item value

If one of the defined rules is broken, a message is generated.

Pr. sh. %	Pr. ov. %	Pr. abs. sh.	Pr. abs. ov.	Max. value
0,00	0,00	0,000000000	0,000000000	0,000000000

This check is dependent on the order simulation, so the **Check whether an SAP sales order can be created** check must have run successfully first.

### Test availability for every item position

/EBY/CL\_PDSO\_EVT\_CHK\_ATP\_QTY

Checks the availability of every ordered line item by calling the SAP function module BAPI\_MATERIAL\_AVAILABILITY for the defined delivery plant.

## Determinations for Customer Orders

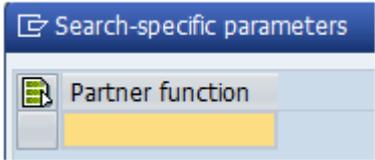
These determinations are available for Customer Orders in the PROCESS DIRECTOR standard:

### Sales order BillTo determination

/EBY/CL\_PDSO\_DET\_BILL\_TO\_INIT

Determines the Bill-To party of a customer order by searching for the following known information in customer master records (depending on the defined searches):

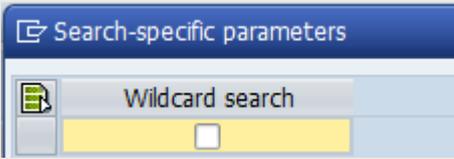
Search area	Search name
Customer number	/EBY/CL_PDSO_DET_PARTNER_KNA1
Customer account number	/EBY/CL_PDSO_DET_PARTNER_KNB1 (finance) /EBY/CL_PDSO_DET_PARTNER_KNVV (sales)
Name or partial name	/EBY/CL_PDSO_DET_PARTNER_NAME

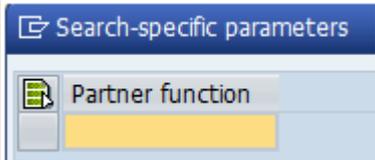
Search area	Search name
<p>Address and contact data</p>	<p>/EBY/CL_PDSO_DET_PARTNER_ZIP (postal code)            /EBY/CL_PDSO_DET_PARTNER_ZIP1 (first part postal code)            /EBY/CL_PDSO_DET_PARTNER_ZIP (city)            /EBY/CL_PDSO_DET_PARTNER_STR (street name or partial street name)            /EBY/CL_PDSO_DET_PARTNER_TEL (telephone number)            /EBY/CL_PDSO_DET_PARTNER_FAX (fax number)            /EBY/CL_PDSO_DET_PARTNER_MAIL (email address)</p>
<p>Tax numbers</p>	<p>/EBY/CL_PDSO_DET_PARTNER_VAT (VAT number)            /EBY/CL_PDSO_DET_PARTNER_TAX (Tax number)            /EBY/CL_PDSO_DET_PARTNER_SIREN (Tax number 2)</p>
<p>Business partners</p> <p>Specify the partner function in the search parameters:</p> 	<p>/EBY/CL_PDSO_DET_PARTNER_SP (Sold-To party)            /EBY/CL_PDSO_DET_PARTNER_SH (Ship-To party)            /EBY/CL_PDSO_DET_PARTNER_CP (Contact person)</p>

## Sales order Contact determination

/EBY/CL\_PDSO\_DET\_CONTACT\_INIT

Determines the contact person of a customer order by searching for the following known information in customer master records (depending on the defined searches):

Search area	Search name
<p>Name or partial name</p> <p>In the search parameters, you can specify that wild cards can be used in the search.</p> 	<p>/EBY/CL_PDSO_DET_CONTACT_NAME</p>

Search area	Search name
<p>Name from the list of all maintained contacts of a given business partner (Sold-To, Ship-To, Bill-To).</p> <p>Specify the partner function in the search parameters:</p> 	/EBY/CL_PDSO_DET_CONTACT_P_N
Contact data	/EBY/CL_PDSO_DET_CONTACT_TEL (telephone number) /EBY/CL_PDSO_DET_CONTACT_FAX (fax number) /EBY/CL_PDSO_DET_CONTACT_MAIL (email address)
Business partners	/EBY/CL_PDSO_DET_CONTACT_SD (Sold-To party) /EBY/CL_PDSO_DET_CONTACT_SH (Ship-To party)

### Sales order contract relation determination

/EBY/CL\_PDSO\_DET\_CONTRACT\_INIT

Evaluates whether there are existing contracts for a sales order by (depending on the defined searches):

Search type	Search name
Reading all valid contracts of the customer	/EBY/CL_PDSO_DET_CONTRACT_CUST
Checking whether the customer purchase order is a contract reference	/EBY/CL_PDSO_DET_CONTRACT_PO

Search type	Search name
Checking whether the reference document is a contract reference	/EBY/CL_PDSO_DET_CONTRACT_REF

If the determination is successful, the sales order document type is switched to a contract and the contract is mentioned as the reference document on item level as well.

### Sales order doc type determination

/EBY/CL\_PDSO\_DET\_SO\_TYP\_INIT

Determines the correct sales document type using the search

/EBY/CL\_PDSO\_DET\_SO\_TYP\_DUMMY.

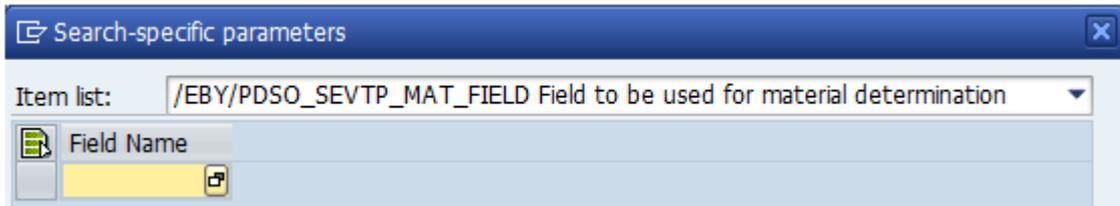
### Sales order Material determination

/EBY/CL\_PDSO\_DET\_MAT\_INIT

Determines the material number by checking whether the material number given on the customer purchase order is (depending on the defined searches):

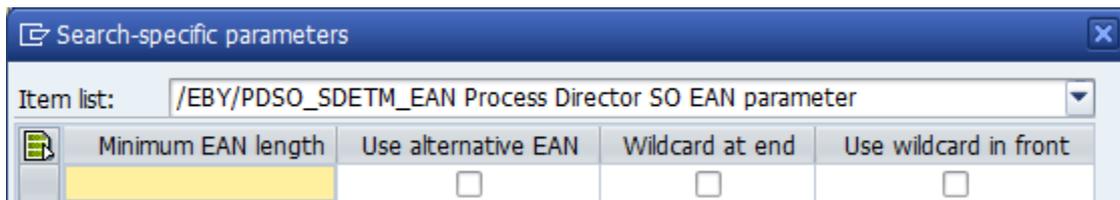
Search type	Search name
The material number from the material master data	/EBY/CL_PDSO_DET_MAT_MARA
A manufacturer part number maintained in the material master data	/EBY/CL_PDSO_DET_MAT_MANU
An EAN maintained in the material master data	/EBY/CL_PDSO_DET_MAT_EAN
A customer part number maintained in SAP table KNMT (transactions VD51/VD52/VD53)	/EBY/CL_PDSO_DET_MAT_CUST
The material number of a substitute material	/EBY/CL_PDSO_DET_MAT_SUBST

In the search parameters, you can specify which field should be used for material determination:



When searching by manufacturer or customer part number or EAN, you can specify a minimum length in the search parameters. The search will not be executed if the length of the material number given on the customer order is below this value.

For EANs, you can activate the search parameter **Alt. EAN** to search for alternative EANs (MEAN table) instead of main EAN numbers (MARA table).



Materials that are not valid for the given sales organization and materials that are blocked for sales are automatically eliminated from the search results.

If an exact match is not found, the searches use wild cards, first at the end and then at the beginning of the material number. Thus, a search for 2345 will find 12345 and 23456.

## Sales order Plant determination

/EBY/CL\_PDSO\_DET\_PLANT\_INIT

Determines the correct delivery plant for a customer order line item by searching for this information in (depending on the defined searches):

Search type	Search name
The material master data	/EBY/CL_PDSO_DET_PLANT_MAT
The customer master data	/EBY/CL_PDSO_DET_PLANT_SORG

## Sales order sales organization determination

/EBY/CL\_PDSO\_DET\_SORG\_INIT

Uses the search /EBY/CL\_PDSO\_DET\_SORG\_SP to determine the sales organization data for a customer order by retrieving it from the customer master data of the Sold-To party.

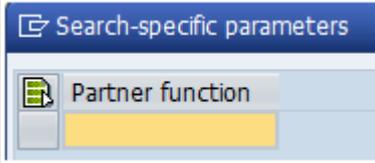
This determination can only succeed if the Sold-To party is known.

## Sales order ShipTo determination

EBY/CL\_PDSO\_DET\_SHIP\_TO\_INIT

Determines the Ship-To party of a customer order by searching for the following known information in customer master records (depending on the defined searches):

Search area	Search name
Customer number	/EBY/CL_PDSO_DET_PARTNER_KNA1
Customer account number	/EBY/CL_PDSO_DET_PARTNER_KNB1 (finance) /EBY/CL_PDSO_DET_PARTNER_KNVV (sales)
Name or partial name	/EBY/CL_PDSO_DET_PARTNER_NAME
Address and contact data	/EBY/CL_PDSO_DET_PARTNER_ZIP (postal code) /EBY/CL_PDSO_DET_PARTNER_ZIP1 (first part postal code) /EBY/CL_PDSO_DET_PARTNER_ZIP (city) /EBY/CL_PDSO_DET_PARTNER_STR (street name or partial street name) /EBY/CL_PDSO_DET_PARTNER_TEL (telephone number) /EBY/CL_PDSO_DET_PARTNER_FAX (fax number) /EBY/CL_PDSO_DET_PARTNER_MAIL (email address)
Tax numbers	/EBY/CL_PDSO_DET_PARTNER_VAT (VAT number) /EBY/CL_PDSO_DET_PARTNER_TAX (Tax number) /EBY/CL_PDSO_DET_PARTNER_SIREN (Tax number 2)

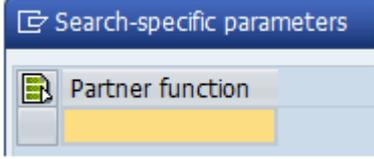
Search area	Search name
<p><b>Business partners</b></p> <p>Specify the partner function in the search parameters:</p> 	<p>/EBY/CL_PDSO_DET_PARTNER_SP (Sold-To party)</p> <p>/EBY/CL_PDSO_DET_PARTNER_BP (Bill-To party)</p> <p>/EBY/CL_PDSO_DET_PARTNER_CP (Contact person)</p> <p>/EBY/CL_PDSO_DET_PARTNER_DEFSH (finds the default Ship-To party for the given Sold-To party)</p>

### Sales order SoldTo determination

/EBY/CL\_PDSO\_DET\_SOLD\_TO\_INIT

Determines the Sold-To party of a customer order by searching for the following known information in customer master records (depending on the defined searches):

Search area	Search name
Customer number	/EBY/CL_PDSO_DET_PARTNER_KNA1
Customer account number	/EBY/CL_PDSO_DET_PARTNER_KNB1 (finance) /EBY/CL_PDSO_DET_PARTNER_KNVV (sales)
Name or partial name	/EBY/CL_PDSO_DET_PARTNER_NAME
Address and contact data	/EBY/CL_PDSO_DET_PARTNER_ZIP (postal code) /EBY/CL_PDSO_DET_PARTNER_ZIP1 (first part postal code) /EBY/CL_PDSO_DET_PARTNER_ZIP (city) /EBY/CL_PDSO_DET_PARTNER_STR (street name or partial street name) /EBY/CL_PDSO_DET_PARTNER_TEL (telephone number) /EBY/CL_PDSO_DET_PARTNER_FAX (fax number) /EBY/CL_PDSO_DET_PARTNER_MAIL (email address)
Tax numbers	/EBY/CL_PDSO_DET_PARTNER_VAT (VAT number) /EBY/CL_PDSO_DET_PARTNER_TAX (Tax number) /EBY/CL_PDSO_DET_PARTNER_SIREN (Tax number 2)

Search area	Search name
<p><b>Business partners</b></p> <p>Specify the partner function in the search parameters:</p> 	<p>/EBY/CL_PDSO_DET_PARTNER_SH (Ship-To party)</p> <p>/EBY/CL_PDSO_DET_PARTNER_BP (Bill-To party)</p> <p>/EBY/CL_PDSO_DET_PARTNER_CP (Contact person)</p>

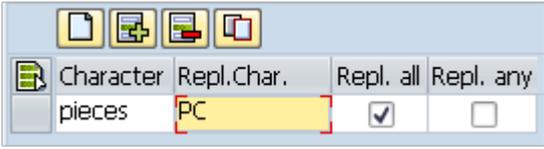
### Sales order UOM determination

/EBY/CL\_PDSO\_DET\_UOM\_INIT

Determines the correct sales unit for a customer order line item by searching for this information in (depending on the defined searches):

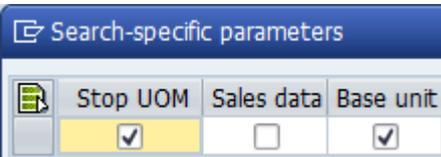
Search type	Search name
The UoM description captured from the customer order	/EBY/CL_PDSO_DET_UOM_EXT
The material master data	/EBY/CL_PDSO_DET_UOM_MASTER

If a unit of measurement is captured, enter in the search parameters which captured UoM description should be converted to which SAP sales unit. For example, if the captured UoM description is `pieces`, enter this together with the SAP sales unit `PC` in the search parameters:



Character	Repl.Char.	Repl. all	Repl. any
pieces	PC	<input checked="" type="checkbox"/>	<input type="checkbox"/>

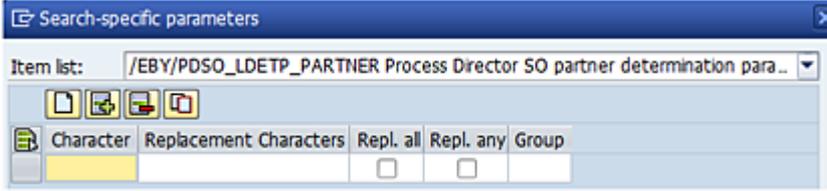
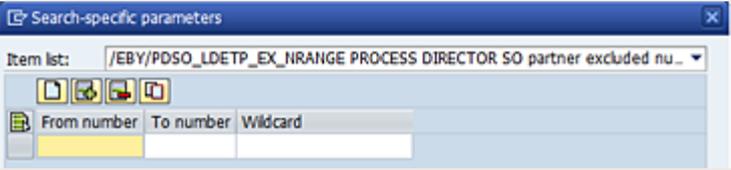
When searching the material master data, you can specify in the parameters whether to search only the base UoM in the material master (`MARA` table), only the UoM from the material sales data (`MVKE` table), or both. If you check the **Stop UOM** parameter, this search is not carried out if the unit of measurement has been captured.



Stop UOM	Sales data	Base unit
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

## Search parameters for partner searches

When performing partner searches (Bill-To, Ship-To, Sold-To determination), you can specify these parameters:

Parameter type	Description
/EBY/PDSO_LDETP_PARTNER	<p><b>PROCESS DIRECTOR SO partner determination parameters</b></p> <p>Enables you to replace characters while searching.</p> 
/EBY/PDSO_LDETP_EX_NRANGE	<p><b>PROCESS DIRECTOR SO partner excluded number range</b></p> <p>Enables you to exclude a specific number range from the search.</p> 

## Document model customization

The document model provides an interface structure for each logical level containing all fields that are available for that level. Using the customer includes (CIs) provided in this structure, you can extend the document model to include customer-specific fields. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

- Use the include ending in `_DATA` if the custom field should be automatically persisted in the database.
- Use the include ending in `_DISP` if you only want to automatically calculate a field at runtime and display it.

🔴 **Important:** Never add customer-specific fields directly to the customer include, but use your own customer-specific structure instead. For example, first create a customer-specific structure corresponding to each used CI, include that structure in the CI and put all customer-specific fields in the customer-specific structure. **Do not use an APPEND structure.** Set the enhancement category of the customer-specific structure to **Can Be Enhanced, Character-Type or Numeric-Type** (menu **Extras**). Never use the enhancement category **Can Be Enhanced (Deep)**. See the SAP documentation for more information.

Logical level	Package	Interface structure	Customer Includes
Header data	/EBY/PDSO_MODEL	/EBY/PDSO_SHDR_IF	CI_EBY_PDSO_SHDR_DATA CI_EBY_PDPO_SHDR_DISP
Item data	/EBY/PDSO_MODEL	/EBY/PDSO_SITM_IF	CI_EBY_PDSO_SITM_DATA CI_EBY_PDPO_SITM_DISP
Conditions data	/EBY/PDSO_MODEL	/EBY/PDSO_SCON_IF	CI_EBY_PDSO_SCON_DATA CI_EBY_PDPO_SCON_DISP
Partner data	/EBY/PDSO_MODEL	/EBY/PDSO_SPAR_IF	CI_EBY_PDPO_SPAR_DATA CI_EBY_PDPO_SPAR_DISP
Schedule data	/EBY/PDSO_MODEL	/EBY/PDSO_SSDL_IF	CI_EBY_PDPO_SSCH_DATA CI_EBY_PDPO_SSCH_DISP

### BAPI structure fields

Some document data models include BAPI structures. These structures can contain a large number of fields, many of which are not used. Therefore, most document data model BAPI structure fields are transient by default.

However, a customer may require that one or more of these fields be made persistent. PROCESS DIRECTOR offers an easy way to fulfill this requirement. In SAP transaction SE80, simply add the BAPI structure field(s) to the appropriate customer include.

Logical level	Package	Structure	Customer Include
Header data	/EBY/PDSO_PERSISTENCE	/EBY/PDSO_THDR	CI_EBY_PDSO_SHDR_BAPI

Logical level	Package	Structure	Customer Include
Item data	/EBY/PDSO_PERSISTENCE	/EBY/PDSO_TITM	CI_EBY_PDSO_SITM_BAPI
Conditions data	/EBY/PDSO_PERSISTENCE	/EBY/PDSO_TCON	CI_EBY_PDSO_SCON_BAPI
Partner data	/EBY/PDSO_PERSISTENCE	/EBY/PDSO_TPAR	CI_EBY_PDSO_SPAR_BAPI
Schedule data	/EBY/PDSO_PERSISTENCE	/EBY/PDSO_TSDL	CI_EBY_PDSO_SSDL_BAPI

## View model customization

This section provides detailed information about which settings to enter when creating a customer view model for Customer Orders. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

## By GUI element

- SAP document overview list:

The screenshot displays the SAP GUI interface for 'Display Customer Order #247 (295 of 404)'. The main window is divided into several sections, each with a callout box describing a specific grid:

- Header conditions grid:** Component type: Additional grid 01, Grid No.:1, Logical level: HEADER\_COND
- Header data grid:** Component type: Blank, Grid No.: 1, Logical level: HEADER
- Items grid:** Component type: Blank, Grid No.: 2, Logical level: ITEMS
- Partners grid:** Component type: Blank, Grid No.: 3, Logical level: HEADER\_PARTNERS
- Price conditions grid:** Component type: Additional grid 01, Grid No.:2, Logical level: ITEM\_COND
- Schedules grid:** Component type: Additional grid 01, Grid No.:3, Logical level: SCHEDULES

The main window also shows a 'PROCESS DIRECTOR by ReadSoft' sidebar on the left and a 'Header' table at the top with columns: Stat., PO doc., Alt. I., Sales Doc., StaTy, SOrg, DChn, Dv, SGrp, SOE, Req.dv:dt, PO date, PO number, Net value, Test., Crpy. The 'Items' table below has columns: Item, Material, Material number, Description, Indicat, Sta, Net value, Crpy, Con, Sch., Tests, Plant.

- SAP document detail:

To maintain the SAP GUI detail screen, use SAP transaction /EBY/PDVI\_VDSTC.

- Web Application document overview list:

Worklist		Customer Order (367)					
Refresh worklist		Create					
Requisition (156)		Status	WF status	PD doc no ↓	Sales Doc.	Sales org.	Req.dlv.dt
Goods Receipt (344)		■		424	10295	1000	12/09/10
Financial Posting (65)		■					12/06/10
Payment Advice (124)		■					12/06/10
<b>Customer Order (367)</b>		■		420		1000	09/25/10
Orders to be created (274)		■		419		1000	12/09/10
Order created (64)							
Incoming Invoice							

**Header data grid:**  
Component type: WA List Header, Grid No.: 1,  
Logical level: HEADER

- Web Application document detail:

Worklist		Customer Order 424									
Refresh worklist		Upload Copy Save Start Delete Back Audit trail									
Requisition (156)		Sales Order									
Goods Receipt (344)		General Org. Data Attachments (1)									
Financial Posting (65)		Document status									
Payment Advice (124)		Workflow status									
<b>Customer Order (367)</b>		PROCESS DIRECTOR document number 424									
Orders to be created (274)		Sales Order									
Order created (64)		Sales Order									
Incoming Invoice		Sales Order									
Document Approval (12)		Sales Order									
Payment Approval (154)		Sales Order									
Order Confirmations (2)		Sales Order									
Cost Center Maintenance (14)		Sales Order									
Vendor Master Maintenance (2)		Sales Order									
GL Account Maintenance (1)		Sales Order									
Profitcenter Maintenance (1)		Sales Order									
Asset Master Maintenance (1)		Sales Order									
All Workflows (150)		Sales Order									
My Workflow Inbox (31)		Sales Order									
		Current messages									
		For messages from previous steps use the "Audit trail" button									
		Items (1)									
		Item	Material	Descript.	MatEntered	Order qty	SU	Net price	Net value	Cur	
		000001	615-0482	Fan drive resistor	34R1057	20.000	PC	1.99	39.80	EUR	
		Header partners (4)									
		Partfunct.	Customer	Acct at ct	Name	Country	Post.Code	City	Telephone1	Fax Number	
		SP	1261		OPTARE	GB	S66 8HR	Rotherham			
		BP	1261		OPTARE	GB	S66 8HR	Rotherham			
		PY	1261		OPTARE	GB	S66 8HR	Rotherham			
		SH	1261		OPTARE	GB	S66 8HR	Rotherham			

**Header data grid:**  
Component type: WA Document Detail, Grid No.: 1,  
Logical level: HEADER

**Items grid:** Component type: WA Document Detail,  
Grid No.: 2, Logical level: ITEMS

**Partners grid:** Component type: WA Document Detail,  
Grid No.: 3, Logical level: HEADER\_PARTNERS

- Workflow status:

### By view model component type

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
	1	HEADER	ZXY_PDSO_SGRID1 ZXY_PDSO_LGRID1	Header Header data
	2	ITEMS	ZXY_PDSO_SGRID2 ZXY_PDSO_LGRID2	Items Items
	3	HEADER_PARTNERS	ZXY_PDSO_SGRID3 ZXY_PDSO_LGRID3	Partners Partners

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
Additional grid 01	1	HEADER_COND	ZXY_PDSO_SGRID4 ZXY_PDSO_LGRID4	Conditions Header conditions
Additional grid 01	2	ITEM_COND	ZXY_PDSO_SGRID6 ZXY_PDSO_SGRID6	Conditions Price conditions
Additional grid 01	3	SCHEDULES	ZXY_PDSO_SGRID5 ZXY_PDSO_LGRID5	Schedules Schedules
WA Document Detail	1	HEADER	ZXY_PDSO_SWC_DOC_GRID1 ZXY_PDSO_LWC_DOC_GRID1	Sales Order Sales Order
WA Document Detail	2	ITEMS	ZXY_PDSO_SWC_DOC_GRID2 ZXY_PDSO_LWC_DOC_GRID2	Items Items
WA Document Detail	3	HEADER_PARTNERS	ZXY_PDSO_SWC_DOC_GRID3 ZXY_PDSO_LWC_DOC_GRID3	Partners Partners
WA List Header	1	HEADER	ZXY_PDSO_SWC_LIST_GRID1 ZXY_PDSO_LWC_LIST_GRID1	Sales Order Sales Order
Workflow status	1	HEADER	ZXY_PDSO_SWC_STAT_GRID1 ZXY_PDSO_LWC_STAT_GRID1	Sales Order Sales Order
Header data for archive log	1	HEADER	ZXY_PDSO_SARC_LOG_GRID1 ZXY_PDSO_LARC_LOG_GRID1	Sales Order Sales Order

## Document statuses

PROCESS DIRECTOR ships with these system document statuses for Customer Orders:

Status	Status Description
BG	In background process
DL	Deleted
ER	In error - no sales order
NN	No sales order created

Status	Status Description
OK	Sales order created
XX	Rejected

## Text types

PROCESS DIRECTOR ships with these system text types for Customer Orders:

Logical level	Text object	Description	ID	Description	Multiple
HEADER	/EBY/PD	PROCESS DIRECTOR texts	PDN	PROCESS DIRECTOR note	X
HEADER	VBBK	Sales header texts	0002	Header note I	
HEADER	VBBK	Sales header texts	V002	Sales text	
ITEMS	VBBP	Sales item texts	0001	Material sales text	

## User exits / BAdIs

### BAdI Posting (Customer Orders)

This user exit / BAdI currently provides the interface methods [BEFORE\\_POST](#) and [AFTER\\_POST](#) that you can implement to a your own customer logic before or after a Customer Order document is posted.

#### BEFORE\_POST method

#### Calling sequence

This method is called from the action `POST_DOC` Post document. It runs immediately *before* document posting occurs.

## Method signature

Type	Parameter	Description
▶□	IR_OBJECT	Sales order object
▶□	IS_BO_HDR	Data fields of a PD base object
▶□	IS_SO_HDR	Persistent sales order header data
▶□	IT_ITEMS	Customer order items
▶□	IT_HDR_CONDITIONS	Customer order header conditions
▶□	IT_HDR_PARTNERS	Customer order header partners
▶□	IT_ITEM_CONDITIONS	Customer order line item conditions data
▶□	IT_ITEM_PARTNERS	Customer order line item partner data
▶□	IT_ITEM_SCHEDULES	Customer order line item schedules data
▶□	IS_EVENT	Configuration of processes
▶▶	CT_CONTRACTS	Table of BAPICTR
▶▶	CS_HEADER	Sales order BAPI header
▶▶	CT_ITEMS	Sales order BAPI items
▶▶	CT_PARTNERS	Sales order BAPI partners
▶▶	CT_CONDITIONS	Sales order BAPI conditions
▶▶	CT_SCHEDULES	Sales order BAPI schedules
▶▶	CT_TEXTS	Sales order BAPI texts

Type	Parameter	Description
	CT_EXTENSIONS	Sales order BAPI extension
	CT_ITEMXS	Table of BAPISDITMX
	CT_SCHEDULESX	Table of BAPISDITMX
	CT_CONDITIONSX	Table of BAPICONDX
	CT_BATCH	Table of BAPIBTSEL
	CT_SCHEDCONF	Table of BAPISCHDL2
	CT_ADRESSES	Table of BAPIADDR1
	CT_CONTRACTSX	Table of BAPICTRX
	CT_KEYS	Table of BAPISDKEY
	CT_CCARD	Table of BAPICCARD
	CT_CONF_REFINST	Table of BAPICUREF
	CT_CONF_VK	Table of BAPICUVK
	CT_CONF_BLOB	Table of BAPICUBLB

Type	Parameter	Description
	CT_CONF_VALUE	Table of BAPICUVAL
	CT_CONF_PARTOF	Table of BAPICUPRT
	CT_CONF_INST	Table of BAPICUINS
	CT_CONF_REF	Table of BAPICUCFG
	CS_HEADERX	Checkbox fields for Sales and Distribution Document Header
	CC_SALESDOCNO	Sales Document
	CS_SENDER	Transfer structure for logical systems
	CC_BINTYPE	Relationship type
	CB_INTNUMBER	
	CC_OBJ	Object type
	CB_CONVERT	
	BC_REFRESHBUFFER	
	CC_WHEN_ERROR	Single-character flag

Type	Parameter	Description
▶▶	CC_CALL_ACTIVE	Not more closely defined area, possibly used for patch levels
▶▶	CT_CMPB_ASGN	Table of BAPIBTSEL

## AFTER\_POST method

### Calling sequence

This method is called from the action `POST_SUCCESS` Posting succeeded. It runs after the document status `Posted` has been set.

**Note:** If document posting fails, `AFTER_POST` is *not* called.

### Method signature

Type	Parameter	Description
▶	IS_HEADER	Sales order BAPI header
▶	IT_ITEMS	Sales order BAPI items
▶	IT_PARTNERS	Sales order BAPI partners
▶	IT_CONDITIONS	Sales order BAPI conditions
▶	IT_SCHEDULES	Sales order BAPI schedules
▶	IT_TEXTS	Sales order BAPI texts
▶	IT_EXTENSIONS	Sales order BAPI extension
▶▶	CS_BO_HDR	Data fields of a PD base object
▶▶	CS_SO_HDR	Persistent sales order header data
▶▶	CT_HDR_CONDITIONS	Customer order header conditions

Type	Parameter	Description
	CT_HDR_PARTNERS	Customer order header partners
	CT_ITEMS	Customer order items
	CT_ITEM_CONDITIONS	Customer order item conditions
	CT_ITEM_PARTNERS	Customer order item partners
	CT_ITEM_SCHEDULES	Customer order item schedules

## Electronic Bank Statements (ES)

SAP Object type: /EBY/PDES

Package: /EBY/PDES

This chapter provides reference information for Electronic Bank Statements.

### Preset classes for Electronic Bank Statements

The following preset class is available for Electronic Bank Statements in the PROCESS DIRECTOR standard. You can also use [general preset classes](#) that are available for all process types.

#### Updates items in the Electronic Bank Statement

/EBY/CL\_PDES\_PRESET\_ITEM

Adds the Payment Advice number and customer number to line items in the Electronic Bank Statement.

- Note:** Although this preset is a PDES class, it must be added to the Payment Advices presets configuration, not the Electronic Bank Statement presets configuration.

### Determinations for Electronic Bank Statements

This determination is available for Electronic Bank Statements in the PROCESS DIRECTOR standard:

## ES payment advice determination

/EBY/CL\_PDES\_DET\_PA\_INIT

Determines the correct payment advice for an Electronic Bank Statement item by searching for this information in (depending on the defined searches):

Search type	Search name
Search by value	/EBY/CL_PDES_DET_PA_VALUE
Search by reference	/EBY/CL_PDES_DET_PA_REF

In the initialization parameters, you can define the following rules:

- value shortfall in %
- value overrun in %
- absolute value shortfall
- absolute value overrun
- limit for standard line item value

If one of the defined rules is broken, a message is generated.

Search-specific parameters				
Pr. sh. %	Pr. ov. %	Pr. abs. sh.	Pr. abs. ov.	Max. value
0,00	0,00	0,000000000	0,000000000	0,000000000

## Document model customization

The document model provides an interface structure for each logical level containing all fields that are available for that level. Using the customer includes (CIs) provided in this structure, you can extend the document model to include customer-specific fields. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

- Use the include ending in `_DATA` if the custom field should be automatically persisted in the database.
- Use the include ending in `_DISP` if you only want to automatically calculate a field at runtime and display it.

- ❗ **Important:** Never add customer-specific fields directly to the customer include, but use your own customer-specific structure instead. For example, first create a customer-specific structure corresponding to each used CI, include that structure in the CI and put all customer-specific fields in the customer-specific structure. **Do not use an APPEND structure.** Set the enhancement category of the customer-specific structure to **Can Be Enhanced, Character-Type or Numeric-Type** (menu **Extras**). Never use the enhancement category **Can Be Enhanced (Deep)**. See the SAP documentation for more information.

Logical level	Package	Interface structure	Customer Includes
Header data	/EBY/PDES_MODEL	/EBY/PDES_SHDR_IF	CI_EBY_PDES_SHDR_DATA CI_EBY_PDES_SHDR_DISP
Item data	/EBY/PDES_MODEL	/EBY/PDES_SITM_IF	CI_EBY_PDES_SITM_DATA CI_EBY_PDES_SITM_DISP

### BAPI structure fields

Some document data models include BAPI structures. These structures can contain a large number of fields, many of which are not used. Therefore, most document data model BAPI structure fields are transient by default.

However, a customer may require that one or more of these fields be made persistent. PROCESS DIRECTOR offers an easy way to fulfill this requirement. In SAP transaction SE80, simply add the BAPI structure field(s) to the appropriate customer include.

Logical level	Package	Structure	Customer Include
Header data	/EBY/PDES_PERSISTENCE	/EBY/PDES_THDR	CI_EBY_PDES_SHDR_BAPI
Item data	/EBY/PDES_PERSISTENCE	/EBY/PDES_TITM	CI_EBY_PDES_SITM_BAPI

### View model customization

This section provides detailed information about which settings to enter when creating a customer view model for Electronic Bank Statements. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

## By GUI element

- SAP document overview list:
- SAP document detail:

To maintain the SAP GUI detail screen, use SAP transaction /EBY/PDVI\_VDSTC.

- Web Application document overview list:
- Web Application document detail:
- Workflow status:

**Workflow selection**

Workflow	Approve and create PO (2200)
Status description	Running
Started by	Blake Ev
Created on	08/22/2011
Time created	17:11:21
Workflow due date	08/30/2011

**Workflow status header grid:**  
Component type: Workflow status, Grid No.: 1,  
Logical level: HEADER

**Current messages**

Next

Status	Step	Date	Time	Processor	Overdue
	Create PO			Karina Kunz	

Current

Status	Step	Date	Time	Processor	Overdue
	Approve PO	08/22/2011	17:11:42	Roger Tillman	08/30/2011

Continue

## By view model component type

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
	1	HEADER	ZXY_PDES_SGRID1 ZXY_PDES_LGRID1	Header Header data
	2	ITEMS	ZXY_PDES_SGRID2 ZXY_PDES_LGRID2	Items Items
	4	PA_ITEMS	ZXY_PDES_SGRID4 ZXY_PDES_LGRID4	PA Items PA Items
WA Document Detail	1	HEADER	ZXY_PDES_SWC_DOC_GRID1 ZXY_PDES_LWC_DOC_GRID1	Sales Order Sales Order
WA Document Detail	2	ITEMS	ZXY_PDES_SWC_DOC_GRID2 ZXY_PDES_LWC_DOC_GRID2	Items Items
WA Document Detail	4	PA_ITEMS	ZXY_PDES_SWC_DOC_GRID4 ZXY_PDES_LWC_DOC_GRID4	PA Items PA Items
WA List Header	1	HEADER	ZXY_PDES_SWC_LIST_GRID1 ZXY_PDES_LWC_LIST_GRID1	Header Header data
Workflow status	1	HEADER	ZXY_PDES_SWC_STAT_GRID1 ZXY_PDES_LWC_STAT_GRID1	Electronic Bank Statement Electronic Bank Statement
Header data for archive log	1	HEADER	ZXY_PDES_SARC_LOG_GRID1 ZXY_PDES_LARC_LOG_GRID1	Electronic Bank Statement Electronic Bank Statement

## Document statuses

PROCESS DIRECTOR ships with these system document statuses for Electronic Bank Statements:

### Statuses

Status	Status Description
ER	In error or incomplete

Status	Status Description
NN	New or ready for processing
OK	Processed

## Text types

PROCESS DIRECTOR ships with this system text type for Electronic Bank Statements:

Logical level	Text object	Description	ID	Description	Multiple
HEADER	/EBY/PD	PROCESS DIRECTOR texts	PDN	PROCESS DIRECTOR note	X

## Financial Postings (FI)

SAP Object type: /EBY/PDFI (G/L Account Postings), /EBY/PDFIC (Customer Postings)

Package: /EBY/PDFI

This chapter provides reference information for Financial Postings.

### Preset classes for Financial Postings

The following preset classes are available for Financial Postings in the PROCESS DIRECTOR standard. You can also use [general preset classes](#) that are available for all process types.

#### Preset for Credit/Debit Indicator based on posting type

/EBY/CL\_PDFI\_PRESET\_SHKZG

Automatically populates the debit/credit indicator on newly added positions.

**Note:** This preset is only for Customer Postings.

#### Preset for payment terms

/EBY/CL\_PDFI\_PRESETS\_PAY\_TERMS

Automatically populates the details of the payment terms. The payment terms are determined from the specified customer and company code.

**Note:** This preset is only for Customer Postings.

## Checks for Financial Postings

The following checks are available for Financial Postings in the PROCESS DIRECTOR standard. You can also use [general checks](#) that are available for all process types.

### Check against check tables of the popup fields

/EBY/CL\_PDFI\_EVT\_CHK\_CHECKTABL

Checks whether the original document values entered in the popup for entering initial data exist and are valid.

This check is pre-configured in the FI document creation process and does not have to be added to the checks configuration.

### Checks second popup for Adjustment Postings

/EBY/CL\_PDFI\_EVT\_CHK\_GLAP\_DATA

For Adjustment Postings (process type G/L Account Postings), checks whether source and target cost elements have been entered in the popup for entering initial data.

This check is pre-configured in the FI document creation process and does not have to be added to the checks configuration.

### Checks whether data can be posted

/EBY/CL\_PDFI\_EVT\_CHK\_CREATE

Simulates the creation of the FI document in SAP using the BAPI and returns all SAP messages generated during the simulation.

### Checks whether data can be posted via posting interface

/EBY/CL\_PDFI\_EVT\_CHK\_POST\_IF

Simulates the creation of the FI document in SAP using the posting interface and returns all SAP messages generated during the simulation.

### Checks whether recurring schedule fields are correct

/EBY/CL\_PDFI\_EVT\_CHK\_SCHEDULE

For recurring templates, checks whether the entered **Last run date** is later than the entered **First run date**.

## Process parameters for Financial Postings

The following process parameters are available for Financial Postings. You can also use the general process parameters that are available for all process types.

Process	Description												
<b>Archiving object types to be attached to the provision</b> (Archiving)													
<b>Posting interface parameters</b> (Checks, Posting)	<table border="1"> <thead> <tr> <th>Parameter</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>No authorization check</td> <td></td> </tr> <tr> <td>Only check</td> <td></td> </tr> <tr> <td>Update mode</td> <td></td> </tr> <tr> <td>Display mode of the BDC interface</td> <td></td> </tr> <tr> <td>Background</td> <td></td> </tr> </tbody> </table>	Parameter	Description	No authorization check		Only check		Update mode		Display mode of the BDC interface		Background	
Parameter	Description												
No authorization check													
Only check													
Update mode													
Display mode of the BDC interface													
Background													
<b>Prepare batch input</b> (Checks, Posting)													
<b>Set posting type for external data reception</b> (Posting)													
<b>Set posting key for write off determination</b> (Posting)													

## Document model customization

The document model provides an interface structure for each logical level containing all fields that are available for that level. Using the customer includes (CIs) provided in this structure, you can extend the document model to include customer-specific fields. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

- Use the include ending in `_DATA` if the custom field should be automatically persisted in the database.
  - Use the include ending in `_DISP` if you only want to automatically calculate a field at runtime and display it.
- 🔴 **Important:** Never add customer-specific fields directly to the customer include, but use your own customer-specific structure instead. For example, first create a customer-specific structure corresponding to each used CI, include that structure in the CI and put all customer-specific fields in the customer-specific structure. **Do not use an APPEND structure.** Set the enhancement category of the customer-specific structure to **Can Be Enhanced, Character-Type or Numeric-Type** (menu **Extras**). Never use the enhancement category **Can Be Enhanced (Deep)**. See the SAP documentation for more information.

### FI G/L Account Postings

Logical level	Package	Interface structure	Customer Includes
Header data	/EBY/PDFI_MODEL	/EBY/PDFI_SHDR_IF	CI_EBY_PDFI_SHDR_DATA CI_EBY_PDFI_SHDR_DISP
Item data	/EBY/PDFI_MODEL	/EBY/PDFI_SITM_IF	CI_EBY_PDFI_SITM_DATA CI_EBY_PDFI_SITM_DISP

### FIC Customer Postings

Logical level	Package	Interface structure	Customer Includes
Header data	/EBY/PDFI_MODEL	/EBY/PDFIC_SHDR_IF	CI_EBY_PDFIC_SHDR_DATA CI_EBY_PDFIC_SHDR_DISP
Item data	/EBY/PDFI_MODEL	/EBY/PDFIC_SITM_IF	CI_EBY_PDFIC_SITM_DATA CI_EBY_PDFIC_SITM_DISP

## BAPI structure fields

Some document data models include BAPI structures. These structures can contain a large number of fields, many of which are not used. Therefore, most document data model BAPI structure fields are transient by default.

However, a customer may require that one or more of these fields be made persistent. PROCESS DIRECTOR offers an easy way to fulfill this requirement. In SAP transaction SE80, simply add the BAPI structure field(s) to the appropriate customer include.

## FI G/L Account Postings

Logical level	Package	Structure	Customer Include
Header data	/EBY/PDFI_PERSISTENCE	/EBY/PDFI_THDR	CI_EBY_PDFI_SHDR_BAPI
Item data	/EBY/PDFI_PERSISTENCE	/EBY/PDFI_TITM	CI_EBY_PDFI_SITM_BAPI

## FIC Customer Postings

Logical level	Package	Structure	Customer Include
Header data	/EBY/PDFI_PERSISTENCE	/EBY/PDFIC_THDR	CI_EBY_PDFIC_SHDR_BAPI
Item data	/EBY/PDFI_PERSISTENCE	/EBY/PDFIC_TITM	CI_EBY_PDFIC_SITM_BAPI

## View model customization

This section provides detailed information about which settings to enter when creating a customer view model for Financial Postings. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

## By GUI element

- SAP document overview list:

The screenshot shows the SAP 'Display Finance Postings #20 (1 of 17)' interface. It features a left-hand navigation pane with a 'Worklist' section containing various document types like Requisitions, Goods Receipt, and Financial postings. The main area displays two grids:

- Header data grid:** A table with columns for document status, year, posting date, currency, and document header text. A tooltip indicates: 'Header data grid: Component type: Blank, Grid No.: 1, Logical level: HEADER'.
- Items grid:** A table with columns for item number, amount, currency, and cost center. A tooltip indicates: 'Items grid: Component type: Blank, Grid No.: 2, Logical level: ITEMS'.

- SAP document detail:

To maintain the SAP GUI detail screen, use SAP transaction /EBY/PDVI\_VDSTC.

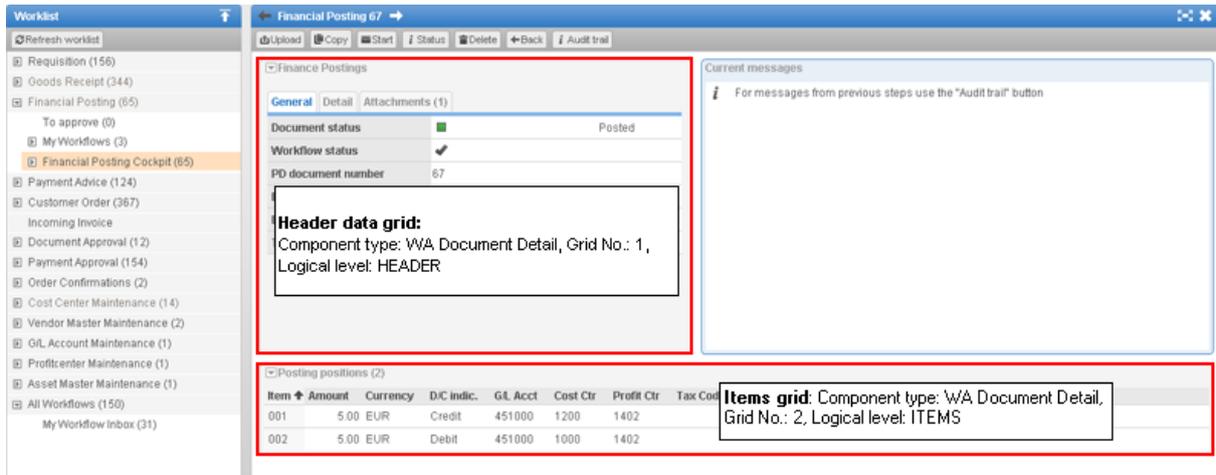
- Web Application document overview list:

The screenshot shows a web application interface with a 'Worklist' on the left and a 'Financial Posting Cockpit (65)' on the right. The cockpit contains a table of document entries:

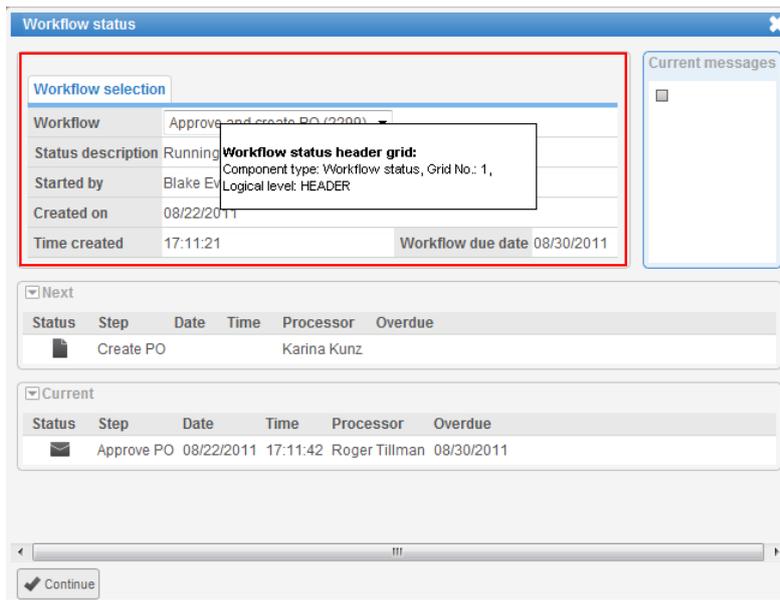
Status	Wf status	PD doc no ↓	Type	Reference	CoCode
Green square	Checkmark	67	AB		1000
Grey square		66	AB		1000
Green square					1000
Red square	Checkmark	63	AB		1000
Grey square		61	AB		1000
Grey square		60	AB		1000

A tooltip over the table indicates: 'Header data grid: Component type: WVA List Header, Grid No.: 1, Logical level: HEADER'.

■ Web Application document detail:



■ Workflow status:



**By view model component type**

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
	1	HEADER	ZXY_PDFI_SGRID1 ZXY_PDFI_LGRID1	Header Header data
	2	ITEMS	ZXY_PDFI_SGRID2 ZXY_PDFI_LGRID2	Positions Posting positions

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
WA Document Detail	1	HEADER	ZXY_PDFI_SGRID3 ZXY_PDFI_LGRID3	Finance Postings Header
WA Document Detail	2	ITEMS	ZXY_PDFI_SGRID4 ZXY_PDFI_LGRID4	Positions Posting positions
WA List Header	1	HEADER	ZXY_PDFI_SWA_LIST_GRID1 ZXY_PDFI_LWA_LIST_GRID1	Header Header data
Workflow status	1	HEADER	ZXY_PDFI_SWA_STAT_GRID1 ZXY_PDFI_LWA_STAT_GRID1	Header Header data
Header data for archive log	1	HEADER	ZXY_PDFI_SARC_LOG_GRID1 ZXY_PDFI_LARC_LOG_GRID1	Header Header data

## Document statuses

PROCESS DIRECTOR ships with these system document statuses for Financial Postings:

### Statuses

Status	Status Description
ER	Error
NN	New or ready for posting
OK	Posted
RE	Reversed
XX	Rejected

### Substatuses

FI substatures are used primarily to assign the appropriate field statuses to FI documents.

Status	Status Description
01	GLSP Regular posting
02	GLPR Provision
03	GLAP Adjustment posting
04	GLRE Recurring entry
05	GLAR Accrual with reversal
06	GLRA Recurring accrual with reversal
07	GLAA Aggregated accrual from PDAP

## Text types

PROCESS DIRECTOR ships with this system text type for Financial Postings:

Logical level	Text object	Description	ID	Description	Multiple
HEADER	/EBY/PD	PROCESS DIRECTOR texts	PDN	PROCESS DIRECTOR note	X

## User exits / BAdIs

### BAdI Posting (Financial Postings)

This user exit / BAdI currently provides the interface methods [BEFORE\\_POST](#), [BEFORE\\_POST\\_PI](#), [AFTER\\_POST](#) and [AFTER\\_POST\\_PI](#) that you can implement to add your own customer logic before or after a Financial Posting document is posted, either via the BADI or the posting interface (PI).

## BEFORE\_POST method

### Calling sequence

This method is called from the action `POST_DOC` Post document. It runs immediately *before* document posting occurs.

### Method signature

Type	Parameter	Description
▶	IR_OBJECT	Financial postings object
▶	IS_BO_HDR	Data fields of a PD base object
▶	IS_FI_HDR	Financial postings header data
▶	IT_ITEMS	Financial postings item data
▶▶	CS_HEADER	Header
▶▶	CS_CUSTOMERCPD	Posting in accounting: Partner billing doc (load receivable)
▶▶	CS_CONTRACTHEADER	Add. contract Accounts Receivable and Accounts Payable header line
▶▶	CT_ACCOUNTGL	G/L account item
▶▶	CT_ACCOUNTRECEIVABLE	Customer item
▶▶	CT_ACCOUNTPAYABLE	Vendor item
▶▶	CT_ACCOUNTTAX	Tax item
▶▶	CT_CURRENCYAMOUNT	Currency items
▶▶	CT_CRITERIA	CO-PA account assignment characteristics

Type	Parameter	Description
▶▶	CT_VALUEFIELD	CO-PA account assignment value fields
▶▶	CT_EXTENSION1	Container for 'Customer Exit' parameter
▶▶	CT_EXTENSION2	BAPI Extension

## BEFORE\_POST\_PI method

### Method signature

Type	Parameter	Description
▶	IR_OBJECT	Financial postings object
▶	IS_BO_HDR	Data fields of a PD base object
▶	IS_FI_HDR	Financial postings header data
▶	IT_ITEMS	Financial postings item data
▶▶	CC_TCODE	Transaction code
▶▶	CB_SGFUNCT	FI posting interface alternative function
▶▶	CT_BLNTAB	Document number table for financial accounting
▶▶	CT_FTPOST	Document header and items for internal posting interface
▶▶	CT_FTTAX	Taxes for internal posting interface
▶▶	CT_RETURN	Return parameter table
▶▶	CB_NO_AUTH	Deactivate authorization check

Type	Parameter	Description
▶▶	CB_CHECK	Only check
▶▶	CC_UPDATEMODE	Update mode for call transaction
▶▶	CC_CTU_MODE	Batch input processing mode
▶▶	CB_DARK	Transactions to be processed in the background

## AFTER\_POST method

### Calling sequence

In the PROCESS DIRECTOR standard, this user exit is called from the action `POST_SUCCESS` Posting succeeded. It runs after the document status has been set.

**Note:** If document posting fails, `AFTER_POST` is *not* called.

### Method signature

Type	Parameter	Description
▶	IR_OBJECT	Financial postings object
▶	IS_HEADER	Header
▶	IS_CUSTOMERCPD	Posting in accounting: Partner billing doc (load receivable)
▶	IT_CONTRACTHEADER	Add. contract Accounts Receivable and Payable Header Line
▶	IT_ACCOUNTGL	G/L account item
▶	IT_ACCOUNTRECEIVABLE	Customer item
▶	IT_ACCOUNTPAYABLE	Vendor item

Type	Parameter	Description
▶□	IT_ACCOUNTTAX	Tax item
▶□	IT_CURRENCYAMOUNT	Currency Items
▶□	IT_CRITERIA	CO-PA account assignment characteristics
▶□	IT_VALUEFIELD	CO-PA account assignment value fields
▶□	IT_EXTENSION1	Container for 'Customer Exit' Parameter
▶▶	CS_BO_HDR	Data fields of a PD base object
▶▶	CS_FI_HDR	Financial postings header data
▶▶	CT_ITEMS	Financial postings item data

## AFTER\_POST\_PI method

### Method signature

Type	Parameter	Description
▶□	IR_OBJECT	Financial postings object
▶□	IC_TCODE	Transaction code
▶□	IB_SGFUNCT	FI posting interface alternative function
▶□	IT_BLNTAB	Document number table for financial accounting
▶□	IT_FTPOST	Document header and items for internal posting interface
▶□	IT_FTTAX	Taxes for internal posting interface

Type	Parameter	Description
▶□	IT_RETURN	Return parameter table
▶□	IB_NO_AUTH	Deactivate authorization check
▶□	IB_CHECK	Only check
▶□	IC_UPDATEMODE	Update mode for call transaction
▶□	IC_CTU_MODE	Batch input processing mode
▶□	IB_DARK	Transactions to be processed in the background
▶▶	CS_BO_HDR	Data fields of a PD base object
▶▶	CS_FI_HDR	Financial postings header data
▶▶	CT_ITEMS	Financial postings item data

## Programs

The following programs are available for use with Financial Postings:

- [/EBY/PDFI\\_GEN\\_RECURRING\\_DOCS](#)

Automatically generates FI documents from recurring templates.

- [/EBY/PDFI\\_PROVISION\\_REMINDER](#)

Sends an email to the person who created the provision document when the validity date of the posted document is exceeded.

### **[/EBY/PDFI\\_GEN\\_RECURRING\\_DOCS](#)**

Use the [/EBY/PDFI\\_GEN\\_RECURRING\\_DOCS](#) program to automatically generate PROCESS DIRECTOR Financial Posting documents from recurring templates.

Setting	Description
<b>Company code</b>	Company codes for which documents should be created.
<b>Posting type</b>	PROCESS DIRECTOR posting types for which documents should be created.
<b>Only test general sel. options</b>	Check this option if you do not want to create any documents, but only need information about the number of documents that would be created if the program was run.
<b>Current processor</b>	The name specified here will be inserted as the creator of the generated documents.
<b>PROCESS DIRECTOR user type</b>	Enter the user type of the specified current processor: Internet, LDAP or SAP.

### **/EBY/PDFI\_PROVISION\_REMINDER**

Use the /EBY/PDFI\_PROVISION\_REMINDER program to send an email to the person who created the provision document when the validity date of the posted document is exceeded. PROCESS DIRECTOR provides a [standard email text](#) for this purpose.

Setting	Description
<b>Company code</b>	Company codes for which provision emails should be created.
<b>Posting type</b>	Select the posting type GLPR.
<b>Only test general sel. options</b>	Check this option if you do not want to send any emails, but only need information about the number of documents for which emails would be sent if the program was run.
<b>Current processor</b>	The name specified here will be inserted as the creator of the generated documents.
<b>PROCESS DIRECTOR user type</b>	Enter the user type of the specified current processor: Internet, LDAP or SAP.

## Generic archiving (ARCH\_BUS)

SAP Object type: /EBY/PDXY

Package: /EBY/PDAB

This chapter provides reference information for Generic Archiving.

### Document model customization

The document model provides an interface structure for each logical level containing all fields that are available for that level. Using the customer includes (CIs) provided in this structure, you can extend the document model to include customer-specific fields. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

- Use the include ending in `_DATA` if the custom field should be automatically persisted in the database.
  - Use the include ending in `_DISP` if you only want to automatically calculate a field at runtime and display it.
- ⓘ Important:** Never add customer-specific fields directly to the customer include, but use your own customer-specific structure instead. For example, first create a customer-specific structure corresponding to each used CI, include that structure in the CI and put all customer-specific fields in the customer-specific structure. **Do not use an APPEND structure.** Set the enhancement category of the customer-specific structure to **Can Be Enhanced, Character-Type or Numeric-Type** (menu **Extras**). Never use the enhancement category **Can Be Enhanced (Deep)**. See the SAP documentation for more information.

Logical level	Package	Interface structure	Customer Includes
Header data	/EBY/PDAB_MODEL	/EBY/PDAB_SHDR_IF	CI_EBY_PDAB_SHDR_DATA CI_EBY_PDAB_SHDR_DISP

### View model customization

This section provides detailed information about which settings to enter when creating a customer view model for Generic Archiving. See the *PROCESS DIRECTOR Configuration Guide* for more information.

## By GUI element

- SAP document overview list:

Status	Wf status	Att. ind.	PD doc no	Object Type	Description	Key
			672	/EBY/PDMDC	PROCESS DIRECTOR Cost Center	SDFG
			671	/EBY/PDMDC	PROCESS DIRECTOR Cost Center	SDFG
	⚠		670	/COCKPIT/W	WEB CYCLE workflow step	FRYGA SIMPLE1
	⚠		669	/COCKPIT/W	WEB CYCLE workflow step	FRYGA SIMPLE1
			668	LFA1	Vendor	0000001060
			667	/COCKPIT/I	INVOICE COCKPIT invoice document	987654
	✉		667	/COCKPIT/I	COCKPIT invoice document	45678
			667	/COCKPIT/I	document INVOICE COCKPIT	123456
	⚠		663	LFA1	Vendor	0000004100
			662	/COCKPIT/B	Base document INVOICE COCKPIT	123456
	⚠		661	/COCKPIT/B	Base document INVOICE COCKPIT	123456
	⚠		660	LFA1	Vendor	0000004100
	⚠		659	BUS2040	Asset Account	1000
	⚠		658	LFA1	Vendor	0000004100

- Web Application document overview list:

Status	Wf status	PD doc no	Obj. Type	Descript.	Key
		672	/EBY/PDMDC	PROCESS DIRECTOR Cost Center	SDFG
		671	/EBY/PDMDC	PROCESS DIRECTOR Cost Center	SDFG
	⚠	670	/COCKPIT/W	WEB CYCLE workflow step	FRYGA SIMPLE
		669	/COCKPIT/W	WEB CYCLE workflow step	FRYGA SIMPLE
	⚠	668	LFA1	Vendor	0000001060
		667	/COCKPIT/I	invoice document	987654
	✉	666	/COCKPIT/I	invoice document	45678
		665	/COCKPIT/B	Base document INVOICE COCKPIT	123456
	⚠	664	LFA1	Vendor	0000004100
		663	LFA1	Vendor	0000004100
		662	/COCKPIT/B	Base document INVOICE COCKPIT	123456
	⚠	661	/COCKPIT/B	Base document INVOICE COCKPIT	123456
	⚠	660	LFA1	Vendor	0000004100

- Web Application document detail:

Worklist S47HT

Generic archiving solution 663

Refresh worklist

Copy Save Check Change key data Post Delete Back

Header

General

Header data grid:  
Component type: WA Document Detail, Grid No.: 1,  
Logical level: HEADER

Document status	
Workflow status	
PD document number	663
Object Type	LFA1 Vendor
Key	0000004100

- Workflow status:

Workflow status

Workflow selection

Workflow	Approve and create PO (2200)
Status description	Running
Started by	Blake Ev
Created on	08/22/2011
Time created	17:11:21

Workflow due date 08/30/2011

Current messages

Workflow status header grid:  
Component type: Workflow status, Grid No.: 1,  
Logical level: HEADER

Next

Status	Step	Date	Time	Processor	Overdue
	Create PO			Karina Kunz	

Current

Status	Step	Date	Time	Processor	Overdue
	Approve PO	08/22/2011	17:11:42	Roger Tillman	08/30/2011

Continue

## By view model component type

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
	I	HEADER	ZXY_PDAB_SGRID I ZXY_PDAB_LGRID I	Header Header data
WA Document Detail	I	HEADER	ZXY_PDAB_SWA_DOC_GRID I ZXY_PDAB_LWA_DOC_GRID I	Header Header data
WA List Header	I	HEADER	ZXY_PDAB_SWA_LIST_GRID I ZXY_PDAB_LWA_LIST_GRID I	Header Header
Workflow status	I	HEADER	ZXY_PDAB_SWA_STAT_GRID I ZXY_PDAB_LWA_STAT_GRID I	Header Header
Header data for archive log	I	HEADER	ZXY_PDAB_SARC_LOG_GRID I ZXY_PDAB_LARC_LOG_GRID I	Header Header

## Document statuses

PROCESS DIRECTOR ships with these system document statuses/substatuses for Generic Archiving:

### Statuses

Status	Status Description
ER	In error or incomplete
NN	New or ready for processing
OK	Processed
XX	Rejected

## Text types

PROCESS DIRECTOR ships with this system text type for Generic Archiving:

Logical level	Text object	Description	ID	Description	Multiple
HEADER	/EBY/PD	PROCESS DIRECTOR texts	PDN	PROCESS DIRECTOR note	X

## Goods Receipts (DN)

SAP Object type: /EBY/PDDN

Package: /EBY/PDDN

This chapter provides reference information for Goods Receipts.

### Preset classes for Goods Receipts

The following preset classes are available for Goods Receipts in the PROCESS DIRECTOR standard. You can also use [general preset classes](#) that are available for all process types.

#### Copy the PO from header to item level

/EBY/CL\_PDDN\_PRESET\_PO2ITM

Copies the purchasing document number from the header to the line items.

#### Preset DN header from user settings

/EBY/CL\_PDDN\_PRESET\_HEADER

Presets values in the goods receipt document using the values contained in the user profile.

#### Preset DN vendor from PO or name from LIFNR

/EBY/CL\_PDDN\_PRESET\_VENDOR

Presets the vendor number using the value given in the purchase order, or presets the vendor name using the value from the **Vendor** field.

### Checks for Goods Receipts

The following checks are available for goods receipts in the PROCESS DIRECTOR standard. You can also use [general checks](#) that are available for all process types.

## Check whether a goods movement can be posted

/EBY/CL\_PDDN\_EVT\_CHK\_CREATE

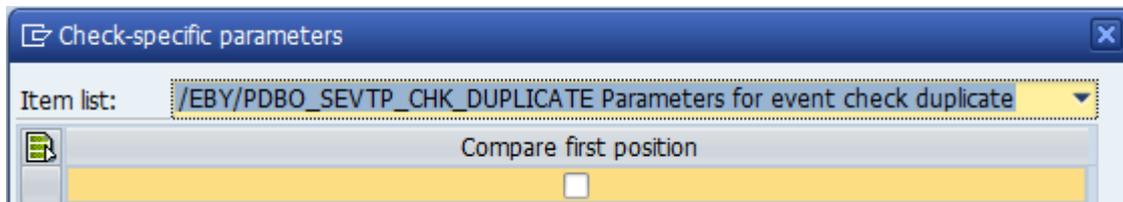
Simulates the creation of the goods movement in SAP and returns all SAP messages generated during the simulation.

## Check whether goods movement has already been posted

/EBY/CL\_PDDN\_EVT\_CHK\_DUPLICATE

Checks whether the same customer has already ordered with the given PO number.

In the initialization parameters, you can specify that the PO number and also the first line item of the PO should be used (item number, material and quantity are compared).



## Check vendor of added PO items

/EBY/CL\_PDDN\_EVT\_CHK\_VEND\_ITM

Checks whether new goods receipt line items added using the **Add PO items** function have the same vendor as existing line items.

## Determinations for Goods Receipts

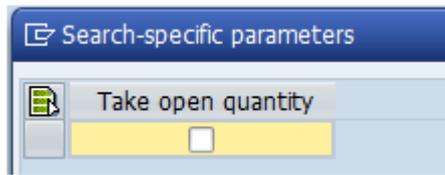
These determinations are available for Goods Receipts in the PROCESS DIRECTOR standard:

### Delivery Note item proposal as determination

/EBY/CL\_PDDN\_DET\_ITEMPROP\_INIT

Adds line items from the purchase order to the goods receipt.

If only open items (that is, items for which the full quantity has not yet been delivered) should be proposed, check the **Take open quantity** checkbox in the search parameters of the **Delivery Notes item proposal from PO** search (/EBY/CL\_PDDN\_DET\_ITEMPROP). Only the open quantity is transferred to the goods receipt line items.



## Delivery Notes Material determination

/EBY/CL\_PDDN\_DET\_MAT\_INIT

Determines the material number by checking whether the material number given on the delivery note is (depending on the defined searches):

Search area	Search name
The material number from the material master data	/EBY/CL_PDDN_DET_MAT_MARA
A manufacturer part number maintained in the material master data	/EBY/CL_PDDN_DET_MAT_MANU
An EAN maintained in the material master data	/EBY/CL_PDDN_DET_MAT_EAN
A vendor material number maintained in the table EINA	/EBY/CL_PDDN_DET_MAT_EINA
The vendor material number given on the purchase order	/EBY/CL_PDDN_DET_MAT_VENDOR

## Delivery Note Plant determination

/EBY/CL\_PDDN\_DET\_PLANT\_INIT

Determines the correct plant and PO unit for a goods receipt line item by searching for this information in (depending on the defined searches):

Search area	Search name
The material master data	/EBY/CL_PDDN_DET_PLANT_MAT
The purchase order	/EBY/CL_PDDN_DET_PLANT_PO

## Delivery Notes Purchase Order determination

/EBY/CL\_PDDN\_DET\_PO\_INIT

Verifies the purchase order number given on the delivery note by searching the EKKO (purchasing document header) table for this number.

## Delivery Note purchase order line item determination

/EBY/CL\_PDDN\_DET\_POPOS\_INIT

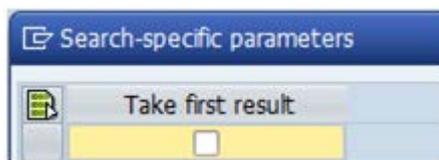
Determines the purchase order line item number by comparing the material and delivery quantity given on the delivery note with the material and undelivered order quantity on the purchase order.

An exact match takes preference over a partial delivery.

The PO line item number can be determined for (depending on the defined searches):

Search area	Search name
The material	/EBY/CL_PDDN_DET_POPOS_MAT
The customer material	/EBY/CL_PDDN_DET_POPOS_IDNLF
The quantity	/EBY/CL_PDDN_DET_POPOS_QTY

In the search parameters, you can specify that the system should use the first match found.



## Delivery Note vendor determination

/EBY/CL\_PDDN\_DET\_VENDOR\_INIT

Verifies the vendor based on (depending on the defined searches):

Search area	Search name
The account number given on the delivery note	/EBY/CL_PDDN_DET_VENDOR_ACC

Search area	Search name
The account number given on the purchase order	/EBY/CL_PDDN_DET_VENDOR_ACCPO
The vendor number given on the delivery note	/EBY/CL_PDDN_DET_VENDOR_OK
The vendor number given on the purchase order	/EBY/CL_PDDN_DET_VENDOR_PO

## Process parameters for Goods Receipts

The following process parameters are available for Goods Receipts. You can also use the general process parameters that are available for all process types.

Process	Description
<b>Save substate of document</b> (Checks)	Turn on Field name Too early Too late Early/late
<b>Posting parameters</b> (Posting)	Automatic serial numbers

## Document model customization

The document model provides an interface structure for each logical level containing all fields that are available for that level. Using the customer includes (CIs) provided in this structure, you can extend the document model to include customer-specific fields. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

- Use the include ending in `_DATA` if the custom field should be automatically persisted in the database.
- Use the include ending in `_DISP` if you only want to automatically calculate a field at runtime and display it.

- ❗ **Important:** Never add customer-specific fields directly to the customer include, but use your own customer-specific structure instead. For example, first create a customer-specific structure corresponding to each used CI, include that structure in the CI and put all customer-specific fields in the customer-specific structure. **Do not use an APPEND structure.** Set the enhancement category of the customer-specific structure to **Can Be Enhanced, Character-Type or Numeric-Type** (menu **Extras**). Never use the enhancement category **Can Be Enhanced (Deep)**. See the SAP documentation for more information.

Logical level	Package	Interface structure	Customer Includes
Header data	/EBY/PDDN_MODEL	/EBY/PDDN_SHDR_IF	CI_EBY_PDDN_SHDR_DATA CI_EBY_PDFI_SHDR_DISP
Item data	/EBY/PDDN_MODEL	/EBY/PDDN_SITM_IF	CI_EBY_PDDN_SITM_DATA CI_EBY_PDDN_SITM_DISP
PO line items	/EBY/PDDN_MODEL	/EBY/PDDN_SEKPO_IF	CI_EBY_PDDN_SEKPO_DATA CI_EBY_PDDN_SEKPO_DISP
Item serial numbers	/EBY/PDDN_MODEL	/EBY/PDDN_SSER_IF	CI_EBY_PDDN_SSER_DATA CI_EBY_PDDN_SSER_DISP

### BAPI structure fields

Some document data models include BAPI structures. These structures can contain a large number of fields, many of which are not used. Therefore, most document data model BAPI structure fields are transient by default.

However, a customer may require that one or more of these fields be made persistent. PROCESS DIRECTOR offers an easy way to fulfill this requirement. In SAP transaction SE80, simply add the BAPI structure field(s) to the appropriate customer include.

Logical level	Package	Structure	Customer Include
Header data	/EBY/PDDN_PERSISTENCE	/EBY/PDDN_THDR	CI_EBY_PDDN_SHDR_BAPI
Item data	/EBY/PDDN_PERSISTENCE	/EBY/PDDN_TITM	CI_EBY_PDDN_SITM_BAPI

Logical level	Package	Structure	Customer Include
Item serial number data	/EBY/PDDN_PERSISTENCE	/EBY/PDDN_TSER	CI_EBY_PDDN_SSER_BAPI

## View model customization

This section provides detailed information about which settings to enter when creating a customer view model for Goods Receipts. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

### By GUI element

- SAP document overview list:

**Header data grid:**  
Component type: Blank, Grid No.: 1, Logical level: HEADER

Stat.	WF	Alt. I.	PO doc.	Created on	Mat.Doc.	Matry	Posting Date	Doc. Date	Purch. Doc.	Vendor	Vendor	Reference	DocHeader Text
			140	21.04.2010	5000007106	2010	21.04.2010	21.04.2010	4500015029				
			139	15.04.2010			10.02.2008	15.04.2010		73025		33005810	Herr Tauss
			138	15.04.2010	5000007105	2010	15.04.2010	15.04.2010	4500015062				
			137	13.04.2010			14.02.2008	13.04.2010	4500014642	12905	12905	531334	Frauenhofer Chri...
			136	13.04.2010			14.02.2008	13.04.2010	4500014644	12905	12905	531355	Dellenbeck Beate
			135	13.04.2010			14.02.2008	13.04.2010	4500014643	12905	12905	531358	Frauenhofer Chri...
			134	13.04.2010			14.02.2008	13.04.2010	4500014642	12905	12905	531334	Frauenhofer Chri...
			133	13.04.2010			12.02.2008	13.04.2010		73025		33005829	Herr Tonn
			132	13.04.2010			11.02.2008	13.04.2010		73025		33005826	Herr Tauss
			131	13.04.2010			10.02.2008	13.04.2010		73025		33005812	Herr Tauss

**Items grid:** Component type: Blank, Grid No.: 2, Logical level: ITEMS

Material	Material Number	Qty in Unit	EU	PC	PO	Item	Plant
AGS92984-02	AGS92984-02	10	PC	30	4500014629		1000

**Purchase order items grid:** Component type: Blank, Grid No.: 3, Logical level: PO\_ITEMS

Purch. Doc.	Item	Plant	Material	Plant SLoc
4500014	10	1000	AGS92984-02	0001

**Serial numbers grid:**  
Component type: Additional grid 01, Grid No.: 1, Logical level: SERIALS

Purchasing Doc.	Item	Serial number

- SAP document detail:

To maintain the SAP GUI detail screen, use SAP transaction /EBY/PDVI\_VDSTC.

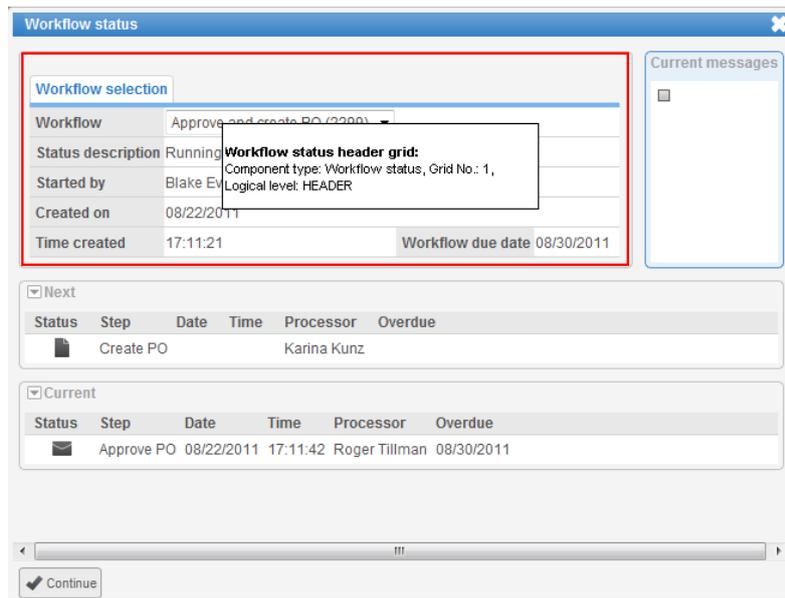
- Web Application document overview list:

Worklist		Goods Receipts Cockpit (344)																																													
Refresh worklist		Create																																													
<ul style="list-style-type: none"> <li>Requisition (156)</li> <li>Goods Receipt (344)                             <ul style="list-style-type: none"> <li>To approve (0)</li> <li>My Workflows (14)</li> <li><b>Goods Receipts Cockpit (344)</b></li> </ul> </li> <li>Financial Posting (65)</li> <li>Payment Advice (124)</li> <li>Customer Order (367)</li> </ul>	<table border="1"> <thead> <tr> <th>Status</th> <th>Wf status</th> <th>PD doc no ↓</th> <th>Mat.Doc.</th> <th>Postg Date</th> <th>Purch.Doc.</th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/></td> <td></td> <td>374</td> <td></td> <td></td> <td>4500016091</td> </tr> <tr> <td><input type="checkbox"/></td> <td></td> <td></td> <td></td> <td></td> <td>4500016091</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td></td> <td></td> <td></td> <td></td> <td>4500016099</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td></td> <td></td> <td></td> <td></td> <td>4500016098</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td></td> <td>370</td> <td></td> <td>01/07/11</td> <td>4500016097</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td></td> <td>369</td> <td></td> <td>01/07/11</td> <td>4500016096</td> </tr> </tbody> </table>	Status	Wf status	PD doc no ↓	Mat.Doc.	Postg Date	Purch.Doc.	<input type="checkbox"/>		374			4500016091	<input type="checkbox"/>					4500016091	<input checked="" type="checkbox"/>					4500016099	<input checked="" type="checkbox"/>					4500016098	<input checked="" type="checkbox"/>		370		01/07/11	4500016097	<input checked="" type="checkbox"/>		369		01/07/11	4500016096	<div style="border: 1px solid black; padding: 5px;"> <p><b>Header data grid:</b> Component type: WVA List Header, Grid No.: 1, Logical level: HEADER</p> </div>			
Status	Wf status	PD doc no ↓	Mat.Doc.	Postg Date	Purch.Doc.																																										
<input type="checkbox"/>		374			4500016091																																										
<input type="checkbox"/>					4500016091																																										
<input checked="" type="checkbox"/>					4500016099																																										
<input checked="" type="checkbox"/>					4500016098																																										
<input checked="" type="checkbox"/>		370		01/07/11	4500016097																																										
<input checked="" type="checkbox"/>		369		01/07/11	4500016096																																										

- Web Application document detail:

Worklist		Goods Receipt 374																										
Refresh worklist		Upload Add PO Items Copy Check Save Start Post Delete Back Audit trail																										
<ul style="list-style-type: none"> <li>Requisition (156)</li> <li>Goods Receipt (344)                             <ul style="list-style-type: none"> <li>To approve (0)</li> <li>MyWorkflows (14)</li> <li><b>Goods Receipts Cockpit (344)</b></li> </ul> </li> <li>Financial Posting (65)</li> <li>Payment Advice (124)</li> <li>Customer Order (367)</li> <li>Incoming Invoice</li> <li>Document Approval (12)</li> <li>Payment Approval (154)</li> <li>Order Confirmations (2)</li> <li>Cost Center Maintenance (14)</li> <li>Vendor Master Maintenance (2)</li> <li>GL Account Maintenance (1)</li> <li>Profitcenter Maintenance (1)</li> <li>Asset Master Maintenance (1)</li> <li>All Workflows (150)</li> </ul>	<table border="1"> <thead> <tr> <th colspan="2">Goods Receipt - Header data</th> </tr> <tr> <th>General</th> <th>Reference</th> <th>Notes (0)</th> </tr> </thead> <tbody> <tr> <td>Document status</td> <td><input type="checkbox"/></td> <td></td> </tr> <tr> <td>Workflow status</td> <td></td> <td></td> </tr> <tr> <td>PD document number</td> <td>374</td> <td></td> </tr> </tbody> </table> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p><b>Header data grid:</b> Component type: WVA Document Detail, Grid No.: 1, Logical level: HEADER</p> </div>	Goods Receipt - Header data		General	Reference	Notes (0)	Document status	<input type="checkbox"/>		Workflow status			PD document number	374		<div style="border: 1px solid black; padding: 5px;"> <p>Current messages</p> <p><i>i</i> For messages from previous steps use the "Audit trail" button</p> </div>												
Goods Receipt - Header data																												
General	Reference	Notes (0)																										
Document status	<input type="checkbox"/>																											
Workflow status																												
PD document number	374																											
	<table border="1"> <thead> <tr> <th colspan="2">Items (2)</th> </tr> <tr> <th>PO</th> <th>Item</th> <th>OK</th> <th>Quantity</th> <th>EUn</th> <th>Text</th> <th></th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/></td> <td>4500016091 00010</td> <td><input checked="" type="checkbox"/></td> <td>1.000</td> <td>PC</td> <td>Black n' Red - Poly</td> <td></td> </tr> <tr> <td><input type="checkbox"/></td> <td>4500016091 00020</td> <td><input checked="" type="checkbox"/></td> <td>1.000</td> <td>PC</td> <td>Rd Rubber S</td> <td>1000</td> </tr> </tbody> </table> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p><b>Items grid:</b> Component type: WVA Document Detail, Grid No.: 2, Logical level: ITEMS</p> </div>	Items (2)		PO	Item	OK	Quantity	EUn	Text		<input type="checkbox"/>	4500016091 00010	<input checked="" type="checkbox"/>	1.000	PC	Black n' Red - Poly		<input type="checkbox"/>	4500016091 00020	<input checked="" type="checkbox"/>	1.000	PC	Rd Rubber S	1000				
Items (2)																												
PO	Item	OK	Quantity	EUn	Text																							
<input type="checkbox"/>	4500016091 00010	<input checked="" type="checkbox"/>	1.000	PC	Black n' Red - Poly																							
<input type="checkbox"/>	4500016091 00020	<input checked="" type="checkbox"/>	1.000	PC	Rd Rubber S	1000																						
	<table border="1"> <thead> <tr> <th colspan="2">PO Items (2)</th> </tr> <tr> <th>Purch.Doc.</th> <th>Item</th> <th>Short text</th> <th>Plant</th> <th>Stor. Loc.</th> <th></th> </tr> </thead> <tbody> <tr> <td>4500016091</td> <td>00010</td> <td>Black n' Red - Polypropylene Twinstore No</td> <td>1000</td> <td></td> <td></td> </tr> <tr> <td>4500016091</td> <td>00020</td> <td>Rd Rubber S</td> <td>1000</td> <td></td> <td>1.000 PC 38.95 38.95 EUR</td> </tr> </tbody> </table> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p><b>PO Items grid:</b> Component type: WVA Document Detail, Grid No.: 3, Logical level: PO_ITEMS</p> </div>	PO Items (2)		Purch.Doc.	Item	Short text	Plant	Stor. Loc.		4500016091	00010	Black n' Red - Polypropylene Twinstore No	1000			4500016091	00020	Rd Rubber S	1000		1.000 PC 38.95 38.95 EUR							
PO Items (2)																												
Purch.Doc.	Item	Short text	Plant	Stor. Loc.																								
4500016091	00010	Black n' Red - Polypropylene Twinstore No	1000																									
4500016091	00020	Rd Rubber S	1000		1.000 PC 38.95 38.95 EUR																							

▪ Workflow status:



**By view model component type**

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
	1	HEADER	ZXY_PDDN_SGRID1 ZXY_PDDN_LGRID1	Header Header data
	2	ITEMS	ZXY_PDDN_SGRID2 ZXY_PDDN_LGRID2	Items Items
	3	PO_ITEMS	ZXY_PDDN_SGRID3 ZXY_PDDN_LGRID3	PO items PO items
Additional grid 01	1	SERIALS	ZXY_PDDN_SGRID4 ZXY_PDDN_LGRID4	Serial numbers Serial numbers
WA Document Detail	1	HEADER	ZXY_PDDN_SWA_DOC_GRID1 ZXY_PDDN_LWA_DOC_GRID1	Header Header data
WA Document Detail	2	ITEMS	ZXY_PDDN_SWA_DOC_GRID2 ZXY_PDDN_LWA_DOC_GRID2	Items Items
WA Document Detail	3	PO_ITEMS	ZXY_PDDN_SWA_DOC_GRID3 ZXY_PDDN_LWA_DOC_GRID3	PO items PO items
WA List Header	1	HEADER	ZXY_PDDN_SWA_LIST_GRID1 ZXY_PDDN_LWA_LIST_GRID1	Header Header

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
Workflow status	I	HEADER	ZXY_PDDN_SWA_STAT_GRIDI ZXY_PDDN_LWA_STAT_GRIDI	Header Header
Header data for archive log	I	HEADER	ZXY_PDDN_SARC_LOG_GRIDI ZXY_PDDN_LARC_LOG_GRIDI	Header Header

## Document statuses

PROCESS DIRECTOR ships with these system document statuses for Goods Receipts:

Status	Status Description
CA	Goods movement cancelled
DL	Deleted
ER	In error - no goods movement
NN	No goods movement posted
OK	Goods movement posted
XX	Rejected

## Text types

PROCESS DIRECTOR ships with this system text type for Goods Receipts:

Logical level	Text object	Description	ID	Description	Multiple
HEADER	/EBY/PD	PROCESS DIRECTOR texts	PDN	PROCESS DIRECTOR note	X

## User exits / BAdIs

### BAdI Posting (Goods Receipts)

This user exit / BAdI currently provides the interface methods [BEFORE\\_POST](#), [BEFORE\\_POST\\_BI](#), [AFTER\\_POST](#) and [AFTER\\_POST\\_BI](#) that you can implement to add your own customer logic before or after a Goods Receipt document is posted.

#### BEFORE\_POST method

#### Calling sequence

This method is called from the action `POST_DOC` Post document. It runs immediately *before* document posting occurs.

#### Method signature

Type	Parameter	Description
▶□	VALUE( IR_OBJECT )	Delivery notes object
▶□	IS_BO_HDR	Data fields of a PD base object
▶□	IS_DN_HDR	Delivery notes header data
▶□	IT_ITEMS	Delivery notes DN items
▶□	IT_PO_ITEMS	Generic PO items data
▶□	IT_SERIALS	Delivery notes serial numbers
▶▶	CS_HEADER	BAPI Communication Structure: Material Document Header Data
▶▶	CS_CODE	MMIM: New Key Assignment GM_CODE to Transaction of INV. Mgmt
▶▶	CT_ITEMS	BAPI Table Type: Create Material Document Item

Type	Parameter	Description
▶▶	CT_SERIALS	BAPI Communication Structure: Create Mat. Doc., Serial No.
▶▶	CB_XNAPR	Print via output control

## BEFORE\_POST\_BI method

### Method signature

Type	Parameter	Description
▶□	VALUE( IR_OBJECT )	Delivery notes object
▶□	IS_BO_HDR	Data fields of a PD base object
▶□	IS_DN_HDR	Delivery notes header data
▶□	IT_ITEMS	Delivery notes DN items
▶□	IT_PO_ITEMS	Generic PO items data
▶□	IT_SERIALS	Delivery notes serial numbers
▶▶	CT_BDCDATA	Batch Input Branch to Objects
▶▶	CC_TCODE	Transaction code

## AFTER\_POST method

### Calling sequence

This method is called from the action `POST_SUCCESS Posting succeeded`. It runs after the document status `Posted` has been set.

**Note:** If document posting fails, `AFTER_POST` is *not* called.

## Method signature

Type	Parameter	Description
▶□	VALUE ( IR_OBJECT )	Delivery notes object
▶□	IS_HEADER	BAPI Communication Structure: Material Document Header Data
▶□	IS_CODE	MMIM: New Key Assignment GM_CODE to Transaction of Inv. Mgmt
▶□	IT_ITEMS	BAPI Table Type: Create Material Document Item
▶□	IT_SERIALS	BAPI Communication Structure: Create Mat. Doc., Serial No.
▶□	IB_XNAPR	Print via output control
▶□	IT_PO_ITEMS	Generic PO items data
▶▶	CS_BO_HDR	Data fields of a PD base object
▶▶	CS_DN_HDR	Delivery notes header data
▶▶	CT_ITEMS	Delivery notes DN items
▶▶	CT_SERIALS	Delivery notes serial numbers

## AFTER\_POST\_BI method

### Method signature

Type	Parameter	Description
▶□	VALUE ( IR_OBJECT )	Delivery notes object

Type	Parameter	Description
▶□	IT_PO_ITEMS	Generic PO items data
▶□	IT_BDCDATA	Batch Input Branch to Objects
▶□	IC_TCODE	Transaction code
▶□	IT_MESSAGES	Collecting messages in the SAP system
▶□	IS_OPTIONS	Parameter string for runtime of CALL TRANSACTION USING ...
▶□	II_RC	Return code after CALL TRANSACTION
▶▶	CS_BO_HDR	Data fields of a PD base object
▶▶	CS_DN_HDR	Delivery notes header data
▶▶	CT_ITEMS	Delivery notes DN items
▶▶	CT_SERIALS	Delivery notes serial numbers

## Invoice Block and Cancellation (IB)

SAP Object type: /EBY/PDIB

Package: /EBY/PDIB

This chapter provides reference information for Invoice Block/Cancellation.

### Checks for Invoice Block/Cancellation

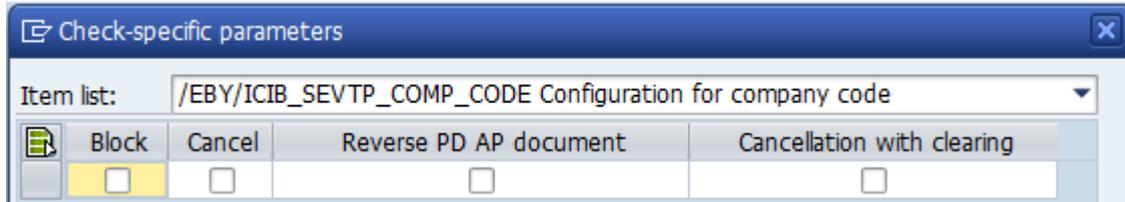
The following checks are available for Invoice Block/Cancellation in the PROCESS DIRECTOR standard. You can also use [general checks](#) that are available for all process types.

## Check allowed transaction type for company code

/EBY/CL\_ICIB\_EVT\_CHK\_COMP\_CODE

Checks whether specific actions are allowed for the company code given in the invoice.

Specify which actions should be possible for the company code in the initialization parameters.



Initialization parameter	Description
<b>Block</b>	Allows invoice blocking.
<b>Cancel</b>	Allows invoice cancelation.
<b>Reverse PD AP document</b>	Allows reversal of the PROCESS DIRECTOR Accounts Payable document.
<b>Cancelation with clearing</b>	Allows cancelation with clearing.

## Check for a valid payment block key

Checks whether an entered payment block can be used.

Specify which payment blocks are *not* allowed in the initialization parameters.



## Document model customization

The document model provides an interface structure for each logical level containing all fields that are available for that level. Using the customer includes (CIs) provided in this structure,

you can extend the document model to include customer-specific fields. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

- Use the include ending in `_DATA` if the custom field should be automatically persisted in the database.
- Use the include ending in `_DISP` if you only want to automatically calculate a field at runtime and display it.
- 🔴 **Important:** Never add customer-specific fields directly to the customer include, but use your own customer-specific structure instead. For example, first create a customer-specific structure corresponding to each used CI, include that structure in the CI and put all customer-specific fields in the customer-specific structure. **Do not use an APPEND structure.** Set the enhancement category of the customer-specific structure to **Can Be Enhanced, Character-Type or Numeric-Type** (menu **Extras**). Never use the enhancement category **Can Be Enhanced (Deep)**. See the SAP documentation for more information.

Logical level	Package	Interface structure	Customer Includes
Header data	/EBY/ICIB_MODEL	/EBY/ICIB_SHDR_IF	CI_EBY_ICIB_SHDR_DATA

## View model customization

This section provides detailed information about which settings to enter when creating a customer view model for Invoice Block/Cancelation. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

## By GUI element

- SAP document overview list:

Display Invoice block/cancellation #35 (1 of 28)

PROCESS DIRECTOR by ReadSoft

St...	Wf...	Att...	PD doc no	Doc. no.	CoCd	Invoice doc. no	Year	Posting Date	Reference	Amount	Cry	Vendor	Name 1	Pbk	Pbk	...	Posting Date
			35	19809	1000	5100000258	2012	23.08.2012		200.00	EUR	2500	Jürgen Rötter		R		
			34	19809	1000	5100000258	2012	23.08.2012		200.00	EUR	2500	Jürgen Rötter		R		
			33	18106	1000	1900000078	2012	13.04.2012		10,000,000...	EUR	1000	C.E.B. BERLIN		A		
			32	18106	1000	1900000078	2012	13.04.2012		10,000,000...	EUR	1000	C.E.B. BERLIN		A		
			31	18106	1000	1900000078	2012	13.04.2012		10,000,000...	EUR	1000	C.E.B. BERLIN		A		
			30	18106	1000	1900000078	2012	13.04.2012		10,000,000...	EUR	1000	C.E.B. BERLIN		A		
			29	17713	1000	1900000067	2012	12.03.2012	ASASAQQW	116.00	EUR	1060	ReadSoft Ebydos AG		A		
			28	17713	1000	1900000067	2012	12.03.2012	ASASAQQW	116.00	EUR	1060	ReadSoft Ebydos AG		A		
			27	17750	1000	1900000007					EUR	1060	ReadSoft Ebydos AG		A		
			26	17750	1000	1900000007					EUR	1060	ReadSoft Ebydos AG		A		
			25	18106	1000	1900000078	2012	13.04.2012		10,000,000...	EUR	1000	C.E.B. BERLIN		A		
			24	18106	1000	1900000078	2012	13.04.2012		10,000,000...	EUR	1000	C.E.B. BERLIN		A		
			23	18106	1000	1900000078	2012	13.04.2012		10,000,000...	EUR	1000	C.E.B. BERLIN		A		
			22	10282	1000	5100000015	2008	02.04.2008	1128023	380.32	EUR	3200	Blumenbecker Industrie		R		
			21	10282	1000	5100000015	2008	02.04.2008	1128023	380.32	EUR	3200	Blumenbecker Industrie		R		
			20	18106	1000	1900000078	2012	13.04.2012		10,000,000...	EUR	1000	C.E.B. BERLIN		+		
			19	18106	1000	1900000078	2012	13.04.2012		10,000,000...	EUR	1000	C.E.B. BERLIN		A		
			18	18106	1000	1900000078	2012	13.04.2012		10,000,000...	EUR	1000	C.E.B. BERLIN		A		
			17	17731	1000	5100000183	2011	15.03.2012	ASDASDASD	100.00	EUR	1000	C.E.B. BERLIN		A		
			16	0	1000	5100000190	2012	04.01.2012		100.00	GBP	1060	ReadSoft Ebydos AG		A		
			15	0	1000	5100000183	2011	13.12.2011		17.00	EUR	1060	ReadSoft Ebydos AG		A		
			12	10282	1000	5100000015	2008	02.04.2008	1128023	380.32	EUR	3200	Blumenbecker Industrie		R		
			11	0	1000	5100000017	2008	02.04.2008	42150	548.34	EUR	3600	Funcke Büroarkt		+		
			10	10291	1000	5100000017	2008	02.04.2008	42150	548.34	EUR	3600	Funcke Büroarkt		R		
			6	16510	1000	1900000033	2011	13.12.2011		22.00	EUR	1060	ReadSoft Ebydos AG		P		
			3	11516	1000	5100000107	2010	05.11.2010	QA2	58.00	EUR	1060	ReadSoft Ebydos AG		+		
			2	0													
			1	10120	1000	5100000008	2008	28.03.2008	DFASD	119.00	EUR	1060	ReadSoft Ebydos AG		A		

Header data grid:  
Component type: Blank, Grid No.: 1, Logical level: HEADER

- SAP document detail:

To maintain the SAP GUI detail screen, use SAP transaction /EBY/PDVI\_VDSTC.

- Web Application document overview list:

Worklist S47

Invoice Blocking/Unblocking (28)

Status	WF status	PD doc no	Document no.	Invoice doc. no	Fisc. Year	CoCode	Postg Date	Vendor	Reference	Amount	Currency	Print block
		35	19809	5100000258	2012	1000	08/23/2012	2500		200.00	EUR	
		34	19809	5100000258	2012	1000	08/23/2012	2500		200.00	EUR	R
		33	18106	1900000078	2012	1000	04/13/2012	1000		10,000,000.00	EUR	
		32	18106	1900000078	2012	1000	04/13/2012	1000		10,000,000.00	EUR	
		31	18106	1900000078	2012	1000	04/13/2012	1000		10,000,000.00	EUR	A
		30	18106	1900000078	2012	1000	04/13/2012	1000		10,000,000.00	EUR	
		29	17713	1900000067	2012	1000	03/12/2012	1060	ASASAQQW	116.00	EUR	
		28	17713	1900000067	2012	1000	03/12/2012	1060	ASASAQQW	116.00	EUR	A
		27	17713	1900000067	2012	1000	03/12/2012	1060	ASASAQQW	2,552,000.00	EUR	
		26	17713	1900000067	2012	1000	03/12/2012	1060	ASASAQQW	380.32	EUR	A
		25	18106	1900000078	2012	1000	04/13/2012	1000		10,000,000.00	EUR	
		24	18106	1900000078	2012	1000	04/13/2012	1000		10,000,000.00	EUR	A
		23	18106	1900000078	2012	1000	04/13/2012	1000		10,000,000.00	EUR	
		22	10282	5100000015	2008	1000	04/02/2008	3200	1128023	380.32	EUR	
		21	10282	5100000015	2008	1000	04/02/2008	3200	1128023	380.32	EUR	R
		20	18106	1900000078	2012	1000	04/13/2012	1000		10,000,000.00	EUR	+
		19	18106	1900000078	2012	1000	04/13/2012	1000		10,000,000.00	EUR	
		18	18106	1900000078	2012	1000	04/13/2012	1000		10,000,000.00	EUR	A
		17	17731	5100000238	2012	1000	03/15/2012	1000	ASDASDASD	100.00	EUR	A

Header data grid:  
Component type: WA List Header, Grid No.: 1, Logical level: HEADER

■ Web Application document detail:

**Header**

General	Block	Attachments (1)
Transaction type	0	
Document status	<span style="color: green;">●</span>	Workflow status
PD document number	34	
Document number	19809	
Invoice doc. number	5100000258	Fiscal Year 2012
Company Code	1000	
Posting Date	08/23/2012	
Reference		
Vendor	2500	Jürgen Ritter
Amount	200.00	Currency EUR

**Header data grid:**  
Component type: WA Document Detail, Grid No.: 1, Logical level: HEADER

■ Workflow status:

**Workflow selection**

Workflow	Approve and create PO (2200)
Status description	Running
Started by	Blake E...
Created on	08/22/2011
Time created	17:11:21
Workflow due date	08/30/2011

**Workflow status header grid:**  
Component type: Workflow status, Grid No.: 1, Logical level: HEADER

**Next**

Status	Step	Date	Time	Processor	Overdue
	Create PO			Karina Kunz	

**Current**

Status	Step	Date	Time	Processor	Overdue
	Approve PO	08/22/2011	17:11:42	Roger Tillman	08/30/2011

Continue

## By view model component type

GUI area	Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming suggestions)	Btn. label Grid title (naming suggestions)
SAP document list		1	HEADER	ZXY_PDIB_SGRID1 ZXY_PDIB_LGRID1	Header Header data
Web Application document detail header data	WA Document Detail	1	HEADER	ZXY_PDIB_SWA_DOC_GRID1 ZXY_PDIB_LWA_DOC_GRID1	Header Header data
Web Application document list	WA List Header	1	HEADER	ZXY_PDIB_SWA_LIST_GRID1 ZXY_PDIB_LWA_LIST_GRID1	Header Header data
Workflow status popup	Workflow status	1	HEADER	ZXY_PDIB_SWA_STAT_GRID1 ZXY_PDIB_LWC_STAT_GRID1	Header Header data
Archive log	Header data for archive log	1	HEADER	ZXY_PDIB_SARC_LOG_GRID1 ZXY_PDIB_LARC_LOG_GRID1	Header Header data

## Document statuses

PROCESS DIRECTOR ships with these system document statuses/substatuses for Invoice Block/Cancelation:

### Statuses

Status	Status Description
ER	In error or incomplete
NN	New or ready for processing
OK	Processed

The following system document substuses are available for invoice block/cancelation:

Substatus	Description
IC	Invoice cancelation
NB	Set new payment block

Substatus	Description
UB	Remove payment block

## Text types

PROCESS DIRECTOR ships with this system text type for Invoice Block/Cancelation:

Logical level	Text object	Description	ID	Description	Multiple
HEADER	/EBY/PD	PROCESS DIRECTOR texts	PDN	PROCESS DIRECTOR note	X

## Master Data Maintenance (MD)

The following standard master data types are currently available:

Master data type	SAP Object type	Package
Asset master	/EBY/PDMD_ASSET	/EBY/PDMD_ASSET
Cost center	/EBY/PDMD_COSTCTR	/EBY/PDMD_COSTCENTER
Customer master	/EBY/PDMD_CUSTMA	/EBY/PDMDCM
G/L account	/EBY/PDMD_GLACC	/EBY/PDMD_GLACCOUNT
Profit center	/EBY/PDMD_PRCTR	/EBY/PDMD_PROFITCENTER
Vendor master	/EBY/PDMD_VENDOMA	/EBY/PDMD_VENDORMASTER

This chapter provides reference information for these master data types.

## Preset classes for Master Data Maintenance

The following preset classes are available for Master Data Maintenance in the PROCESS DIRECTOR standard. You can also use [general preset classes](#) that are available for all process types.

## **Asset Master**

### **Preset business area from cost center**

/EBY/CL\_PDMDAS\_PRESET\_BUSAREA

Presets the business area from the entered cost center.

### **Preset vendor name**

/EBY/CL\_PDMDAS\_PRESET\_VENDNAME

Presets the vendor name.

## **Cost Centers**

### **TEMPLATE: Preset header**

/EBY/CL\_PDMDCC\_PRESET\_TEMPL

This preset serves as a template for your own coding to preset header data in a cost center maintenance request.

## **Customer Master**

### **Preset: IBAN number from bank account**

/EBY/CL\_PDMDCM\_PRESET\_BNK2IBAN

Presets the IBAN number from the entered bank account.

### **Preset: SWIFT number from bank account**

/EBY/CL\_PDMDCM\_PRESET\_BNK2SWFT

Presets the SWIFT number from the entered bank account.

### **Preset: Bank account from IBAN number**

/EBY/CL\_PDMDCM\_PRESET\_IBAN2BANK

Presets the bank account number from the entered IBAN number.

## **G/L Accounts**

### **Presets based on company code**

/EBY/CL\_PDMDGL\_CCODE\_PRESETS

Presets data based on the entered company code.

## Preset language from chart of account

/EBY/CL\_PDMDGL\_PRESET\_COA\_LANG

Presets the language from the entered Chart of Accounts.

## Vendor Master

### IBAN preset

/EBY/CL\_PDMDVM\_EVT\_PRESET\_IBAN

Presets the IBAN number based on the entered country and bank key. Add this preset class for the field /EBY/IBAN.

### Bank details preset: name, SWIFT code

/EBY/CL\_PDMDVM\_PRESET\_BANKDETS

Presets the bank name and SWIFT code based on the entered country and bank key. Add this preset class for the fields BANKA and SWIFT.

### Preset: Bank account from IBAN number

/EBY/CL\_PDMDVM\_PRESET\_IBAN2BNK

Presets the bank country, key, account number and control key based on the entered IBAN number. Add this preset class for the fields BANKS, BANKL, BANKN and BKONT.

## Checks for Master Data Maintenance

The following check events are available for master data in the PROCESS DIRECTOR standard. You can also use [general checks](#) that are available for all process types.

## Asset Master

### Simulate asset master creation/change/delete

/EBY/CL\_PDMDAS\_EVT\_CHK\_POST

Simulates the creation/change/deletion of the asset master data in SAP and returns all SAP messages generated during the simulation.

## Cost Centers

### Simulate cost center creation/change/delete

/EBY/PDMDCC\_EVT\_CHK\_POST

Simulates the creation/change/deletion of the cost center in SAP and returns all SAP messages generated during the simulation.

## Customer Master

### Address check

/EBY/CL\_PDMDCM\_EVT\_CHK\_ADDR

Checks the address data.

### Allowed alternative payer check

/EBY/CL\_PDMDCM\_EVT\_CHK\_AAP

Checks the allowed alternative payer.

### Bank existence check

/EBY/CL\_PDMDCM\_EVT\_CHK\_BANK

Checks if the combination of country and bank key entered in the customer master maintenance request exists in the SAP system and returns a message if this is the case. This prevents existing bank data from being overwritten.

### Check email address

/EBY/CL\_PDMDCM\_EVT\_CHK\_EMAIL

Checks if there is an email address in the customer master, and if yes, checks whether the email address is valid or not.

### Check IBAN validity

/EBY/CL\_PDMDCM\_EVT\_CHK\_IBAN

Uses the SAP function CHECK\_IBAN to check whether the IBAN number entered in the customer master maintenance request is valid and returns a message if this is not the case.

### Check withholding tax

/EBY/CL\_PDMDCM\_EVT\_CHK\_WHT

Checks the withholding tax.

## Sales area existence check

/EBY/CL\_PDMDCM\_EVT\_CHK\_SAREA

Checks if the sales area entered in the customer master maintenance request exists in the SAP system.

## G/L Accounts

### Check deletions flags - block is required

/EBY/CL\_PDMDGL\_EVT\_CHK\_DELETE

Checks whether deletion flags are activated for the document. This prevents the processor or approver from processing the document without looking at the **Block/Deletion** tab and missing deletion flags that have been set.

### Check duplicates for same language

/EBY/CL\_PDMDGL\_EVT\_CHK\_DPL\_LNG

Checks whether duplicate keywords and translations (G/L account descriptions) have been defined in PROCESS DIRECTOR for the same language.

### Check G/L account posting

/EBY/CL\_PDMDGL\_EVT\_CHK\_POST

Simulates the creation/change/deletion of the G/L account in SAP and returns all SAP messages generated during the simulation.

### Cost element existence check

/EBY/CL\_PDMDGL\_EVT\_CHK\_COSTELM

Checks whether the cost element entered in the G/L account maintenance request exists in the SAP system.

## Profit Centers

### Simulate profit center creation/change

/EBY/PDXY\_EVT\_CHK\_PRCRT

Simulates the creation/change/deletion of the profit center in SAP and returns all SAP messages generated during the simulation.

## Vendor Master

### Address check

/EBY/CL\_PDMDVM\_EVT\_CHK\_ADDR

Checks the address data.

### Bank existence check

/EBY/CL\_PDMDVM\_EVT\_CHK\_BANK

Checks if the combination of country and bank key entered in the vendor master maintenance request exists in the SAP system and returns a message if this is the case. This prevents existing bank data from being overwritten.

### Check email address

/EBY/CL\_PDMDVM\_EVT\_CHK\_EMAIL

Checks if there is an email address in the vendor master, and if yes, checks whether the email address is valid or not.

### Check IBAN validity

/EBY/CL\_PDMDVM\_EVT\_CHK\_IBAN

Uses the SAP function CHECK\_IBAN to check whether the IBAN number entered in the vendor master maintenance request is valid and returns a message if this is not the case.

### Check vendor master data for mandatory fields

/EBY/CL\_PDMDVM\_EVT\_CHK\_POST

Simulates vendor master data posting based on the standard SAP account group field status customizing.

### Check permitted alternative payee

/EBY/CL\_PDMDVM\_EVT\_CHK\_PAP

Checks the permitted alternative payee.

### Check withholding tax

/EBY/CL\_PDMDVM\_EVT\_CHK\_WHT

Checks the withholding tax.

## Document model customization

The document model provides an interface structure for each logical level containing all fields that are available for that level. Using the customer includes (CIs) provided in this structure, you can extend the document model to include customer-specific fields. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

- Use the include ending in `_DATA` if the custom field should be automatically persisted in the database.
  - Use the include ending in `_DISP` if you only want to automatically calculate a field at runtime and display it.
- 🔴 **Important:** Never add customer-specific fields directly to the customer include, but use your own customer-specific structure instead. For example, first create a customer-specific structure corresponding to each used CI, include that structure in the CI and put all customer-specific fields in the customer-specific structure. **Do not use an APPEND structure.** Set the enhancement category of the customer-specific structure to **Can Be Enhanced, Character-Type or Numeric-Type** (menu **Extras**). Never use the enhancement category **Can Be Enhanced (Deep)**. See the SAP documentation for more information.

Master data type	Logical level	Package	Interface structure	Customer Include
Cost center	Header data	/EBY/PDMD_COSTCENTER	/EBY/PDMDCC_SHDR_IF	CI_EBY_PDMDCC_SHDR_DATA
Cost center	Collection of cost centers	/EBY/PDMD_PROFITCENTER	/EBY/PDMDCC_SCCC_IF	CI_EBY_PDMDCC_SCCC_DATA
Profit center	Header data	/EBY/PDMD_PROFITCENTER	/EBY/PDMDPC_SHDR_IF	CI_EBY_PDMDPC_SHDR_DATA

Master data type	Logical level	Package	Interface structure	Customer Include
Vendor master	Header data	/EBY/PDMD_VENDORM ASTER	/EBY/PDMDVM_SHDR_ IF	CI_EBY_PDMDVM_SHDR_ DATA  CI_EBY_PDMDVM_SHDR_ DISP
Vendor master	Bank details	/EBY/PDMD_VENDORM ASTER	/EBY/PDMDVM_SBANK _DATA	CI_EBY_PDMDVM_SBANK _DATA
G/L account	Header data	/EBY/PDMD_GLACCOU NT	/EBY/PDMDGL_SHDR_ DISP	CI_EBY_PDMDGL_SHDR_ DISP
G/L account	Keywords	/EBY/PDMD_GLACCOU NT	/EBY/PDMDGL_SKYW_ DATA	CI_EBY_PDMDGL_SKYW_ DATA
G/L account	Translation	/EBY/PDMD_GLACCOU NT	/EBY/PDMDGL_STRL_ DATA	CI_EBY_PDMDGL_STRL_ DATA

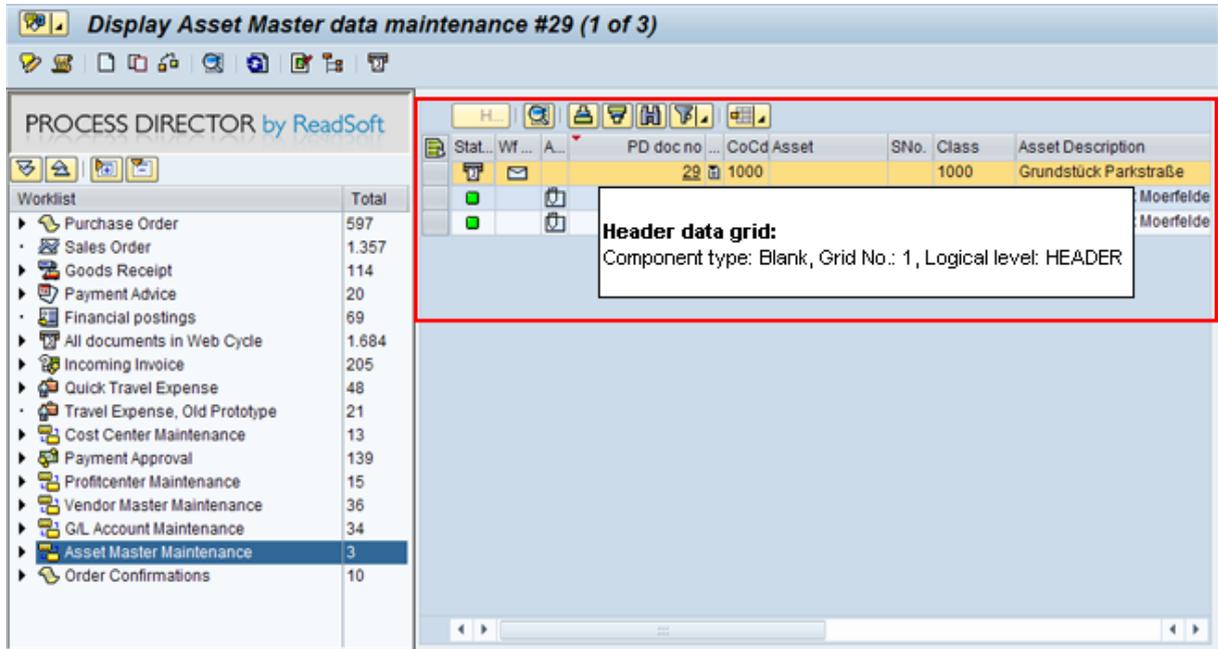
## View model customization

### Asset Master Maintenance

This section provides detailed information about which settings to enter when creating a customer view model for Asset Master Data Maintenance. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

## By GUI element

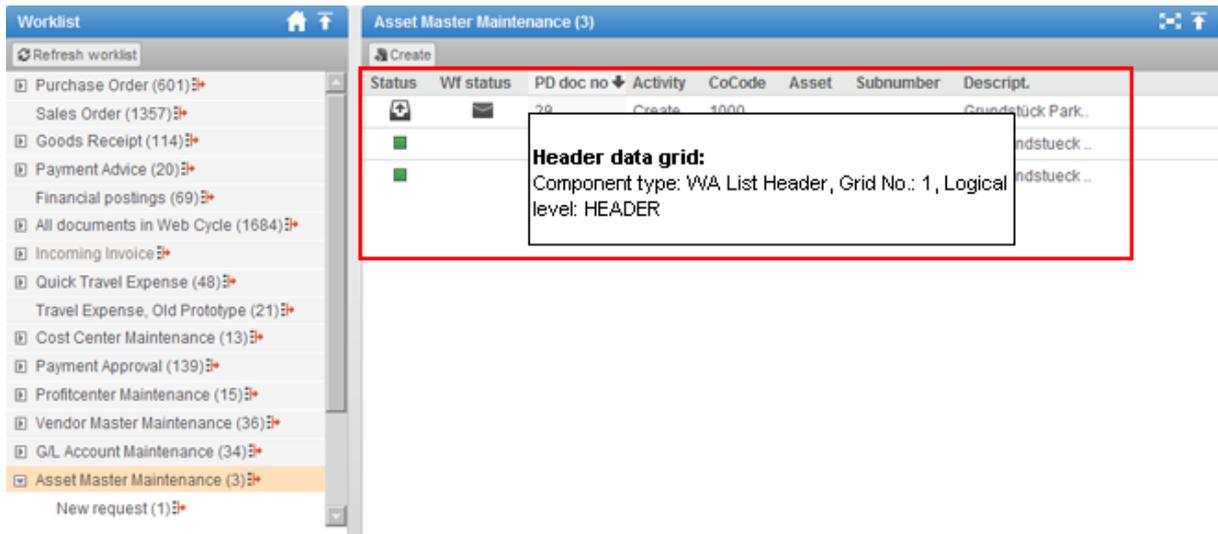
- SAP document overview list:



- SAP document detail:

To maintain the SAP GUI detail screen, use SAP transaction /EBY/PDVI\_VDSTC.

- Web Application document overview list:



- Web Application document detail:

**Header data grid:**  
Component type: WA Document Detail, Grid No.: 1,  
Logical level: HEADER

**Current messages:**  
No workflow step of THOMAS2 in Asset Master data maintenance 29

- Workflow status:

**Workflow status header grid:**  
Component type: Workflow status, Grid No.: 1,  
Logical level: HEADER

Status	Step	Date	Time	Processor	Overdue
	Create PO			Karina Kunz	
	Approve PO	08/22/2011	17:11:42	Roger Tillman	08/30/2011

Continue

## By view model component type

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
	I	HEADER	ZXY_PDMDAS_SGRID I ZXY_PDMDAS_LGRID I	Header Header data
WA Document Detail	I	HEADER	ZXY_PDMDAS_SWA_DOC_GRID I ZXY_PDMDAS_LWA_DOC_GRID I	Header Header data
WA List Header	I	HEADER	ZXY_PDMDAS_SWA_LIST_GRID I ZXY_PDMDAS_LWA_LIST_GRID I	Header Header data
Workflow status	I	HEADER	ZXY_PDMDAS_SWA_STAT_GRID I ZXY_PDMDAS_LWC_STAT_GRID I	Header Header data
Header data for archive log	I	HEADER	ZXY_PDMDAS_SARC_LOG_GRID I ZXY_PDMDAS_LARC_LOG_GRID I	Header Header data

## Cost Center Maintenance

This section provides detailed information about which settings to enter when creating a customer view model for Cost Center Maintenance. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

By GUI element

- SAP document overview list:

Status	Wf status	Att. ind.	PD doc no	Activity	Cost Center	CO Area	Valid From	Valid To
◇			13	U	NIPPEL	5000	15.09.2010	31.12.9999
■			12	U	2555	1000	30.11.2010	31.12.2012
■			11	C	2555	1000	30.11.2010	31.12.2012
■			10	U	PL_TEST_02	1000	01.02.2010	31.12.9999
■								31.12.9999
■								31.12.9999
■								31.12.9999
■								31.12.9999
◇			4	U	1000	2000	01.01.1994	31.12.9999
◇			3	U	1000	2000	01.01.1994	31.12.9999
◇			2	U	71130	F100	01.01.1999	31.12.9999
■			1	U	PL_TEST_02	1000	01.02.2010	31.12.9999

- SAP document detail:

To maintain the SAP GUI detail screen, use SAP transaction /EBY/PDVI\_VDSTC.

- Web Application document overview list:

Status	Wf status	PD doc no	Activity	Cost Ctr	CO Area	Valid From	to	Name
■		13	Update	NIPPEL	5000	09/15/10	12/31/99	Feierei
■		12	Update	2555	1000	11/30/10	12/31/12	HTH_Test1
■								31/12 HTH_Test1
■								31/99 Peter's test cost -
■	✓							31/99 Corporate Services
■	⚠							31/99 Test costcenter
■	✓	7	Update	1000	1000	01/01/94	12/31/99	Corporate Services
■	⚠	6	Create	1000	0001	10/04/10	12/31/99	
■	✓	5	Create	NIPPEL	5000	09/15/10	12/31/99	Feierei
■		4	Update	1000	2000	01/01/94	12/31/99	Corporate Services
■		3	Update	1000	2000	01/01/94	12/31/99	Corporate Services
■		2	Update	71130	F100	01/01/99	12/31/99	Accounting
■		1	Update	PL_TEST_02	1000	02/01/10	12/31/99	Peter's test cost

- Web Application document detail:

**Header data grid:**  
Component type: WA Document Detail, Grid No.: 1,  
Logical level: HEADER

**Valid From:** 11/30/10  
**Valid To:** 12/31/12

**Current messages:**  
For messages from previous steps use the "Audit trail" button

- Workflow status:

**Workflow status header grid:**  
Component type: Workflow status, Grid No.: 1,  
Logical level: HEADER

**Workflow selection:** Approve and create PO (2200)

**Status description:** Running  
**Started by:** Blake E...  
**Created on:** 08/22/2011  
**Time created:** 17:11:21  
**Workflow due date:** 08/30/2011

Status	Step	Date	Time	Processor	Overdue
	Create PO			Karina Kunz	

Status	Step	Date	Time	Processor	Overdue
	Approve PO	08/22/2011	17:11:42	Roger Tillman	08/30/2011

Continue

## By view model component type

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
	I	HEADER	ZXY_PDMDCC_SGRID I ZXY_PDMDCC_LGRID I	Header Header data
WA Document Detail	I	HEADER	ZXY_PDMDCC_SWA_DOC_GRID I ZXY_PDMDCC_LWA_DOC_GRID I	Header Header data
WA List Header	I	HEADER	ZXY_PDMDCC_SWA_LIST_GRID I ZXY_PDMDCC_LWA_LIST_GRID I	Header Header data
Workflow status	I	HEADER	ZXY_PDMDCC_SWA_STAT_GRID I ZXY_PDMDCC_LWC_STAT_GRID I	Header Header data
Header data for archive log	I	HEADER	ZXY_PDMDCC_SARC_LOG_GRID I ZXY_PDMDCC_LARC_LOG_GRID I	Header Header data

## G/L Account Maintenance

This section provides detailed information about which settings to enter when creating a customer view model for G/L Account Maintenance. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

### By GUI element

- SAP document overview list:

**Header data grid:**  
Component type: Blank, Grid No.: 1, Logical level: HEADER

**Keywords grid:**  
Component type: Blank, Grid No.: 2, Logical level: KEYWORD

**Translation grid:**  
Component type: Blank, Grid No.: 3, Logical level: TRANSLATION

- SAP document detail:

To maintain the SAP GUI detail screen, use SAP transaction /EBY/PDVI\_VDSTC.

- Web Application document overview list:

Status	WF status	PD doc no	Action	G/L Acct	CoCode	Chrt/Accts	Acct Group	B/S acct	Acct cur
■		186	Insert	7770	1000	INT	SAKO	<input checked="" type="checkbox"/>	EUR
■		185	Insert	7773	2000	INT	SAKO	<input checked="" type="checkbox"/>	BAD
■		184	Insert	7771	2000	INT	SAKO	<input checked="" type="checkbox"/>	BAD
■								<input checked="" type="checkbox"/>	BAD
■								<input checked="" type="checkbox"/>	EUR
■								<input checked="" type="checkbox"/>	EUR
■								<input checked="" type="checkbox"/>	GBP
■		179	Insert	7784	1000	INT	SAKO	<input checked="" type="checkbox"/>	EUR
■		178	Change	7771	1000	INT	SAKO	<input checked="" type="checkbox"/>	HUF
■		177	Change	7771	1000	INT	SAKO	<input checked="" type="checkbox"/>	HUF
■		175	Insert	7777	1000	INT	SAKO	<input type="checkbox"/>	EUR

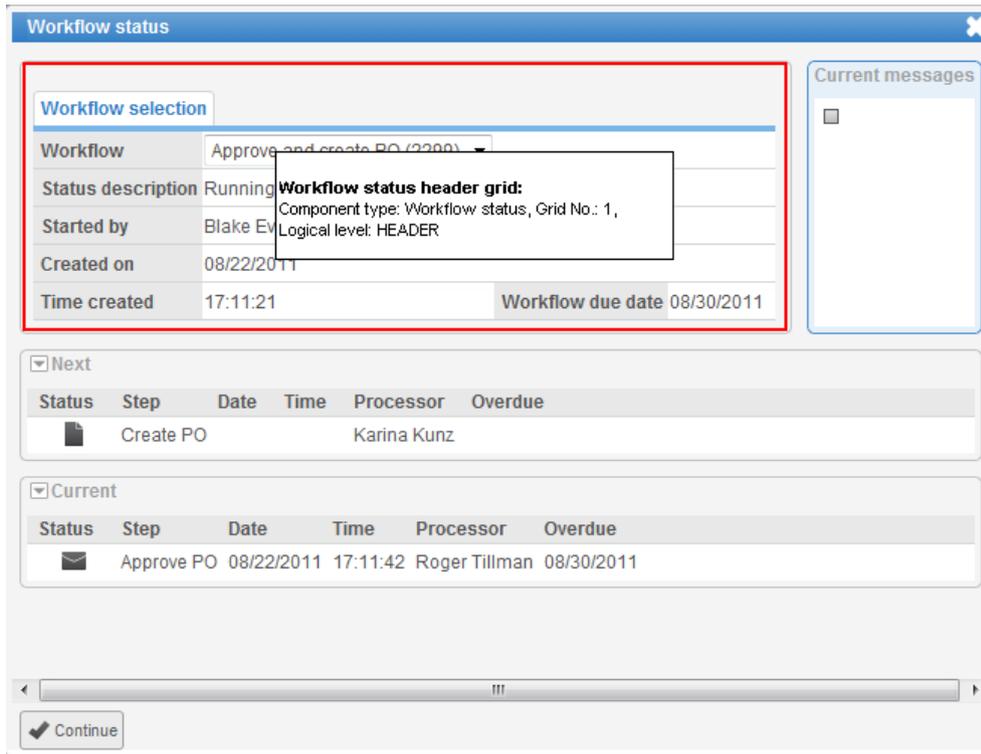
- Web Application document detail:

**Header data grid:**  
Component type: WVA Document Detail, Grid No.: 1, Logical level: HEADER

**Keywords grid:**  
Component type: WVA Document Detail, Grid No.: 2, Logical level: KEYWORD

**Translation grid:**  
Component type: WVA Document Detail, Grid No.: 3, Logical level: TRANSLATION

- Workflow status:



### By view model component type

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
	1	HEADER	ZXY_PDMDGL_SGRID1 ZXY_PDMDGL_LGRID1	Header Header data
	2	KEYWORD	ZXY_PDMDGL_SGRID2 ZXY_PDMDGL_LGRID2	Keywords Keywords
	3	TRANSLATION	ZXY_PDMDGL_SGRID3 ZXY_PDMDGL_LGRID3	Translation Translation
WA Document Detail	1	HEADER	ZXY_PDMDGL_SWA_DOC_GRID1 ZXY_PDMDGL_LWA_DOC_GRID1	Header Header data
WA Document Detail	2	KEYWORD	ZXY_PDMDGL_SWA_DOC_GRID2 ZXY_PDMDGL_LWA_DOC_GRID2	Keywords Keywords
WA Document Detail	3	TRANSLATION	ZXY_PDMDGL_SWA_DOC_GRID3 ZXY_PDMDGL_LWA_DOC_GRID3	Translation Translation

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
WA List Header	1	HEADER	ZXY_PDMDGL_SWA_LIST_GRIDI ZXY_PDMDGL_LWA_LIST_GRIDI	Header Header data
Workflow status	1	HEADER	ZXY_PDMDGL_SWA_STAT_GRIDI ZXY_PDMDGL_LWC_STAT_GRIDI	Header Header data
Header data for archive log	1	HEADER	ZXY_PDMDGL_SARC_LOG_GRIDI ZXY_PDMDGL_LARC_LOG_GRIDI	Header Header data

## Profit Center Maintenance

This section provides detailed information about which settings to enter when creating a customer view model for Profit Center Maintenance. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

### By GUI element

- SAP document overview list:

The screenshot displays the SAP GUI interface for 'Display Profitcenter Maintenance #15 (1 of 15)'. The main window contains a table with the following columns: Status, Wf status, Att. ind., PD doc no, Activity, Profit..., COAr, Valid From, Valid To, and Name. The table lists several entries, with the first row highlighted. A red box is drawn around the table area, and a tooltip points to the first row, displaying the text: 'Header data grid: Component type: Blank, Grid No.: 1, Logical level: HEADER'. The left sidebar shows a 'Worklist' tree with 'Profitcenter Maintenance' selected, showing a count of 15. The top of the screen shows the title bar and a toolbar with various icons.

- SAP document detail:

To maintain the SAP GUI detail screen, use SAP transaction /EBY/PDVI\_VDSTC.

- Web Application document overview list:

The screenshot shows the 'Profitcenter Maintenance (15)' list view. A red box highlights the table content, and a callout box provides details for the header data grid.

Status	WF status	PD doc no	Activity	CO Area	Profit Ctr
		15	Create	1000	TEST
		14	Create	1000	TEST
		13	Update	1000	TEST
		8	Update	1000	1017
		7	Create	1000	TESTI
		6	Create	1000	TEST
		5	Create	1000	TESTI

**Header data grid:**  
Component type: WA List Header, Grid No.: 1, Logical level: HEADER

▪ Web Application document detail:

The screenshot shows the 'Profitcenter Maintenance 15' document detail view. A red box highlights the form content, and a callout box provides details for the header data grid.

**Header data grid:**  
Component type: WA Document Detail, Grid No.: 1, Logical level: HEADER

Document status:

Workflow status: PROCESSED

Activity:

Profit Ctr:

Control:

Valid From: 12/01/10

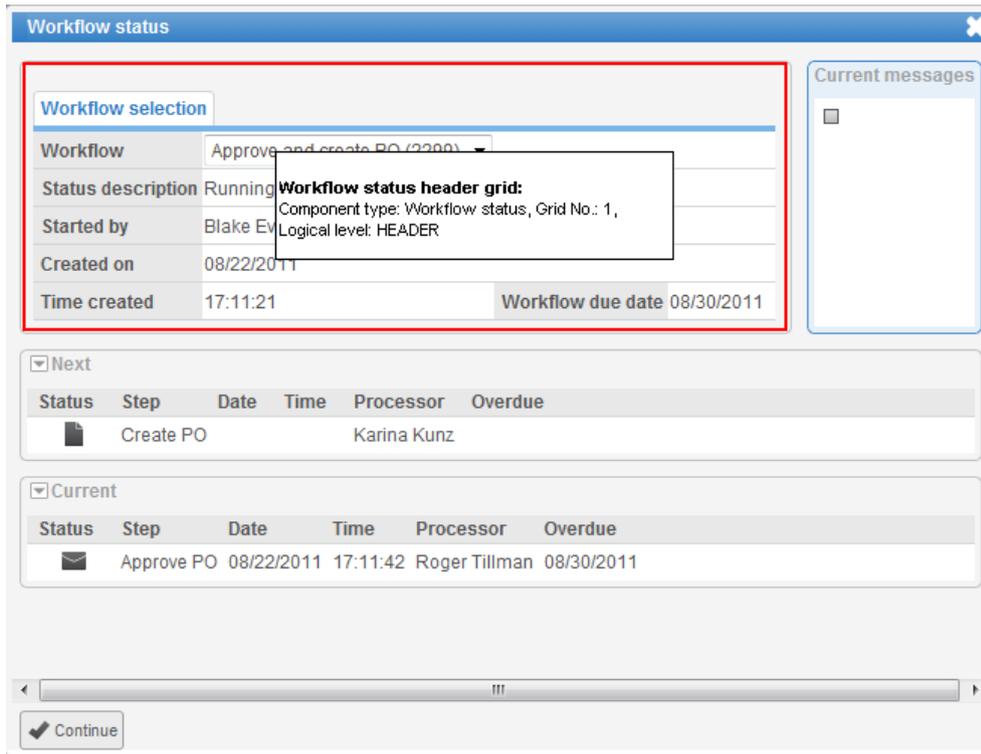
Valid To: 12/31/99

Name: Test Profitcenter

Long Text: Now with even long text

Current messages: For messages from previous steps use the "Audit trail" button

- Workflow status:



### By view model component type

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
	I	HEADER	ZXY_PDMDPC_SGRID I ZXY_PDMDPC_LGRID I	Header Header data
WA Document Detail	I	HEADER	ZXY_PDMDPC_SWA_DOC_GRID I ZXY_PDMDPC_LWA_DOC_GRID I	Header Header data
WA List Header	I	HEADER	ZXY_PDMDPC_SWA_LIST_GRID I ZXY_PDMDPC_LWA_LIST_GRID I	Header Header data
Workflow status	I	HEADER	ZXY_PDMDPC_SWA_STAT_GRID I ZXY_PDMDPC_LWC_STAT_GRID I	Header Header data
Header data for archive log	I	HEADER	ZXY_PDMDPC_SARC_LOG_GRID I ZXY_PDMDPC_LARC_LOG_GRID I	Header Header data

## Vendor Master Maintenance

This section provides detailed information about which settings to enter when creating a customer view model for Vendor Master Data Maintenance. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

### By GUI element

- SAP document overview list:

**Display Vendor master maintenance #39 (1 of 36)**

Bank

PROCESS DIRECTOR by ReadSoft

Worklist	Total
Purchase Order	597
Sales Order	1,357
Goods Receipt	114
Payment Advice	20
Financial postings	69
All documents in Web Cycle	1,684
Incoming Invoice	205
Quick Travel Expense	48
Travel Expense, Old Prototype	21
Cost Center Maintenance	13
Payment Approval	139
Profitcenter Maintenance	15
<b>Vendor Master Maintenance</b>	<b>36</b>
GL Account Maintenance	34
Asset Master Maintenance	3
Order Confirmations	10

Status	Sub state	Wf status	PD doc no	Activity	Vendor	CoCode	P Org	Group	Las
◇			39	C	45566	1000	1000		
■			38	C	KN_301			0001	KN
■			37	C	KN_300			0001	KN
■			36	U	KN_100			0001	KN
■								0001	KN
■								0001	JUU
■								0001	JUU
■								0001	RS
■								0001	RS
◇			31	U	4100			0001	RS
◇			30	U	4100	1000	1000	0001	RS
■			28	U	KUTAS	1000	1000	0001	kut
■			25	C	HU_TEST			0001	Val
■			24	C	PD_IS_FD			0001	PD

Ctry	Bank Key	Bank Account Number	CK	N
	123456	55566611		M

**Header data grid:**  
Component type: Blank, Grid No.: 1, Logical level: HEADER

**Bank data grid:**  
Component type: Blank, Grid No.: 2, Logical level: BANK

- SAP document detail:

To maintain the SAP GUI detail screen, use SAP transaction /EBY/PDVI\_VDSTC.

- Web Application document overview list:

The screenshot shows the 'Vendor Master Maintenance (36)' overview list. The left sidebar contains a 'Worklist' with various document types, and the main area displays a table of vendor records. A red box highlights the table, and a callout box provides details for the 'Header data grid'.

Status	Sub state	Wf status	PD doc no	Activity	Vendor	CoCode	Purch.Org.	Acct Group	Last na
			39	Create	45566	1000	1000		
	@3P@		38	Create	KN_301			0001	KN_30
	@3P@		37	Create	KN_300			0001	KN_30
	@3P@							0001	KN_10
	@3P@							0001	KN_10
								0001	JUUU
								0001	JUUU
			32	Update	4100			0001	RS
			31	Update	4100			0001	RS
			30	Update	4100	1000	1000	0001	RS
			28	Update	KUTAS	1000	1000	0001	kutas

**Header data grid:**  
Component type: WVA List Header, Grid No.: 1, Logical level: HEADER

- Web Application document detail:

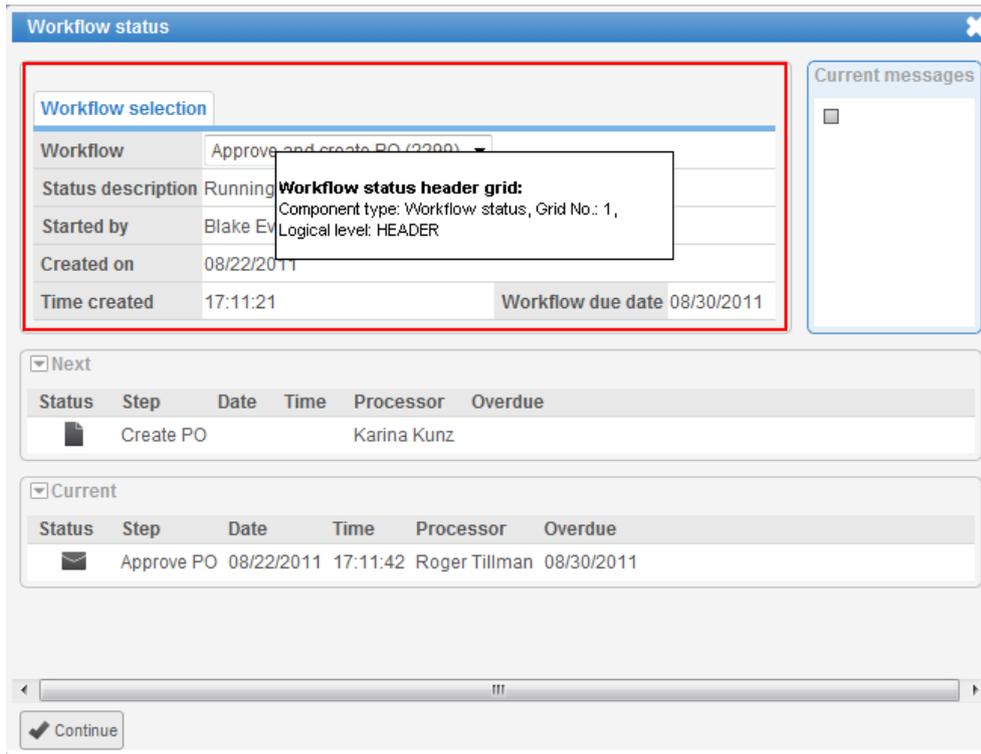
The screenshot shows the 'Vendor master maintenance 30' document detail view. The left sidebar contains a 'Worklist' with various document types, and the main area displays a form with tabs for 'General data', 'Address data', 'Block/Deletion', and 'Notes (0)'. A red box highlights the form, and two callout boxes provide details for the 'Header data grid' and 'Bank data grid'.

**Header data grid:**  
Component type: WVA Document Detail, Grid No.: 1, Logical level: HEADER

**Bank data grid:**  
Component type: WVA Document Detail, Grid No.: 2, Logical level: BANK

Bank ctry	Bank Key	Bank A
<input checked="" type="checkbox"/>	DE	123456
<input type="checkbox"/>		555666

- Workflow status:



### By view model component type

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
	1	HEADER	ZXY_PDMDVM_SGRID1 ZXY_PDMDVM_LGRID1	Header Header data
	2	BANK	ZXY_PDMDVM_SGRID2 ZXY_PDMDVM_LGRID2	Bank Bank
WA Document Detail	1	HEADER	ZXY_PDMDVM_SWA_DOC_GRID1 ZXY_PDMDVM_LWA_DOC_GRID1	Header Header data
WA Document Detail	2	BANK	ZXY_PDMDVM_SWA_DOC_GRID2 ZXY_PDMDVM_LWA_DOC_GRID2	Bank Bank
WA List Header	1	HEADER	ZXY_PDMDVM_SWA_LIST_GRID1 ZXY_PDMDVM_LWA_LIST_GRID1	Header Header data
Workflow status	1	HEADER	ZXY_PDMDVM_SWA_STAT_GRID1 ZXY_PDMDVM_LWC_STAT_GRID1	Header Header data

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
Header data for archive log	I	HEADER	ZXY_PDMDVM_SARC_LOG_GRIDI ZXY_PDMDVM_LARC_LOG_GRIDI	Header Header data

## Document statuses

PROCESS DIRECTOR ships with these system document statuses for all Master Data Maintenance types:

Status	Status Description
ER	In error or incompletd
NN	New or ready for processing
OK	Processed
WR	Postable with warning(s) (Customer and Vendor master data only)
XX	Rejected

The following system document substatuses are available for specific master data maintenance types:

### Vendor Master

Substatus	Description
LA	General data posted
LB	Accounting data posted
LF	Posting completed
LM	Purchasing data posted

## Customer Master

Substatus	Description
LA	General data posted
LB	Accounting data posted
LF	Posting completed
LS	Sales area data posted

## Asset Master

Substatus	Description
CE	Create - External number range
CI	Create - Internal number range
D	Delete
U	Update

## Text types

PROCESS DIRECTOR ships with this system text type for Master Data Maintenance:

Logical level	Text object	Description	ID	Description	Multiple
HEADER	/EBY/PD	PROCESS DIRECTOR texts	PDN	PROCESS DIRECTOR note	X

## User exits / BAdIs

### BAdI Posting (G/L Account Maintenance)

This user exit / BAdI currently provides the interface methods [BEFORE\\_POST](#) and [AFTER\\_POST](#) that you can implement to add your own customer logic before or after a G/L Account Maintenance document is posted.

#### BEFORE\_POST method

#### Calling sequence

This method is called from the action `POST_DOC` Post document. It runs immediately *before* document posting occurs.

#### Method signature

Type	Parameter	Description
▶□	VALUE ( IR_OBJECT )	Generic header
▶□	IS_BO_HDR	Data fields of a PD base object
▶□	IS_MDGL_HDR	G/L account header data
▶□	IT_MDGL_KYW	G/L account keywords data
▶□	IT_MDGL_TRL	G/L account translations data
▶▶	CS_ACCOUNT_COA	G/L account master record: Chart of Accounts
▶▶	CT_ACCOUNT_NAMES	G/L account master record: Descriptions (Table)
▶▶	CT_ACCOUNT_KEYWORDS	G/L account master record: Key Word (Table)
▶▶	CT_ACCOUNT_CCODES	G/L account master record: Company Code (Table)

## AFTER\_POST method

### Calling sequence

This method is called from the action `POST_SUCCESS` Posting succeeded. It runs after the document status `Posted` has been set.

**Note:** If document posting fails, `AFTER_POST` is *not* called.

### Method signature

Type	Parameter	Description
▶	(VALUE) IR_OBJECT	Generic header
▶	IS_ACCOUNT_COA	G/L account master record: Chart of Accounts
▶	IT_ACCOUNT_NAMES	G/L account master record: Descriptions (Table)
▶	IT_ACCOUNT_KEYWORDS	G/L account master record: Key Word (Table)
▶	IT_ACCOUNT_CCODES	G/L account master record: Company Code (Table)
▶▶	CS_BO_HDR	Data fields of a PD base object
▶▶	CS_MDGL_HDR	PROCESS DIRECTOR generated structure
▶▶	CT_MDGL_KYW	PROCESS DIRECTOR generated table type
▶▶	CT_MDGL_TRL	PROCESS DIRECTOR generated table type

## BAdI Posting (Vendor Master Data Maintenance)

This user exit / BAdI currently provides the interface methods [BEFORE\\_POST](#) and [AFTER\\_POST](#) that you can implement to add your own customer logic before or after a Vendor Master Data Maintenance document is posted.

### BEFORE\_POST method

#### Calling sequence

This method is called from the action `POST_DOC` Post document. It runs immediately *before* document posting occurs.

## Method signature

Type	Parameter	Description
▶□	VALUE ( IR_OBJECT )	Generic header
▶□	IS_BO_HDR	Data fields of a PD base object
▶□	IS_MDVM_HDR	PROCESS DIRECTOR generated structure
▶□	IT_MDVM_BANK	PROCESS DIRECTOR generated table type
▶▶	CT_IDOIC_DATA	Table type for EDIDD (IDoc Data Records)

## AFTER\_POST method

### Calling sequence

This method is called from the action `POST_SUCCESS` `Posting succeeded`. It runs after the document status `Posted` has been set.

**Note:** If document posting fails, `AFTER_POST` is *not* called.

### Method signature

Type	Parameter	Description
▶□	( VALUE ) IR_OBJECT	Generic header
▶□	IT_IDOC_DATA	
▶▶	CT_MDVM_BANK	PROCESS DIRECTOR generated table type
▶▶	CT_MDVM_HDR	PROCESS DIRECTOR generated structure
▶▶	CS_BO_HDR	Data fields of a PD base object

## Order Confirmations (OR)

SAP Object type: /EBY/PDOR

Package: /EBY/PDOR

This chapter provides reference information for Order Confirmations.

### Checks for Order Confirmations

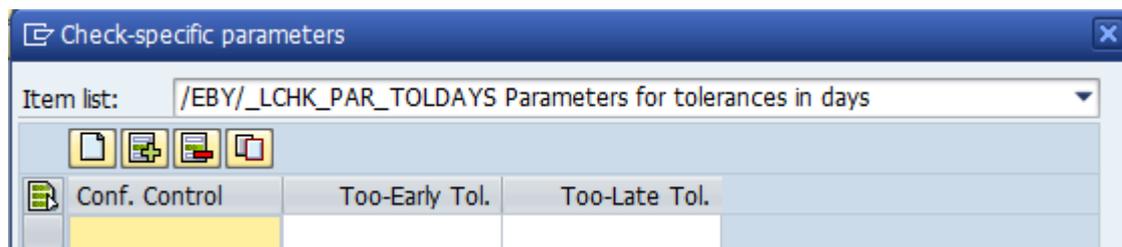
The following check events are available for order confirmations in the PROCESS DIRECTOR standard. You can also use [general checks](#) that are available for all process types.

#### Check confirmed delivery date

/EBY/CL\_PDOR\_EVT\_CHK\_DATE

Checks the delivery date given in the order confirmation against the PO.

In the initialization parameters, you can specify tolerances for specific confirmation control keys.



#### Check confirmed Incoterms

/EBY/CL\_PDOR\_EVT\_CHK\_INCO

Checks the Incoterms given in the order confirmation against the PO.

#### Check confirmed price

/EBY/CL\_PDOR\_EVT\_CHK\_PRICE

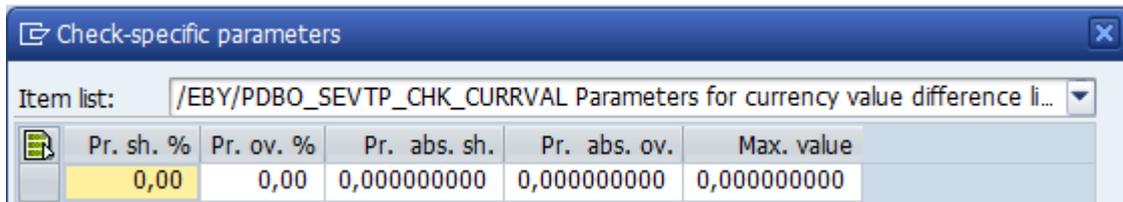
Checks the price given in the order confirmation against the PO.

In the initialization parameters, you can define the following rules:

- value shortfall in %
- value overrun in %

- absolute value shortfall
- absolute value overrun
- limit for standard line item value

If one of the defined rules is broken, a message is generated.



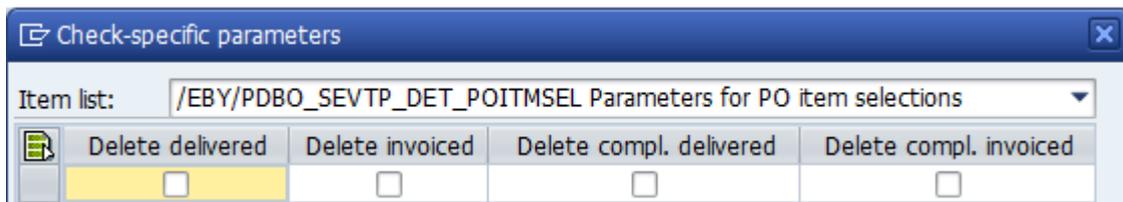
### Check confirmed quantity

/EBY/CL\_PDOR\_EVT\_CHK\_QTY

Checks the quantity given in the order confirmation against the PO.

Overdelivery and underdelivery tolerances are taken from the PO.

In the initialization parameters, you can exclude delivered and invoiced items.



Initialization parameter	Description
<b>Delete delivered</b>	Excludes items that have been partially delivered.
<b>Delete invoiced</b>	Excludes items that have been partially invoiced.
<b>Delete compl. delivered</b>	Excludes items that have been completely delivered.
<b>Delete compl. invoiced</b>	Excludes items that have been completely invoiced.

## Check whether the data can be processed

/EBY/CL\_PDOR\_EVT\_CHK\_PROC

Simulates the creation of the order confirmation in SAP and returns all SAP messages generated during the simulation.

## Determinations for Order Confirmations

These determinations are available for Order Confirmations in the PROCESS DIRECTOR standard:

### Purchase order line item assignment

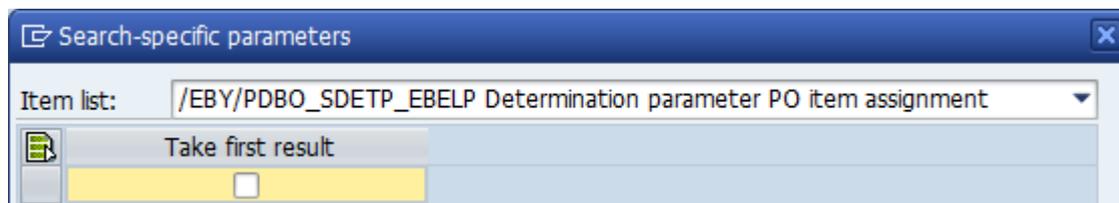
/EBY/CL\_PDOR\_DET\_EBELP\_INIT

Determines the purchase order line item number by comparing (depending on the defined searches):

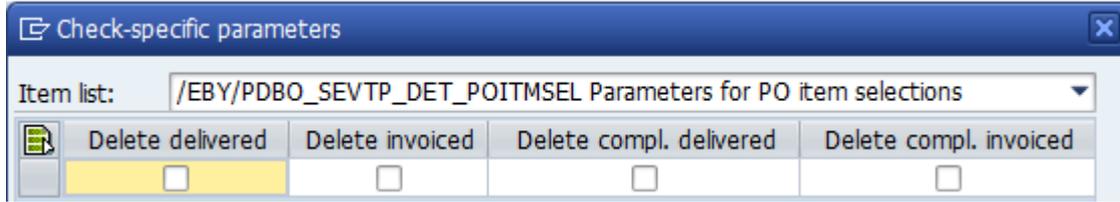
Search type	Search name
The vendor material number, line item number and quantity given on the order confirmation with the undelivered order quantity on the purchase order.	/EBY/CL_PDOR_DET_EBELP_IDNLF
The material number, line item number and quantity given on the order confirmation with the undelivered order quantity on the purchase order.	/EBY/CL_PDOR_DET_EBELP_MAT
The price and line item number given on the order confirmation with the price on the purchase order.	/EBY/CL_PDOR_DET_EBELP_PRICE
The quantity and line item number given on the order confirmation with the quantity on the purchase order.	/EBY/CL_PDOR_DET_EBELP_QTY

An exact match takes preference over a partial delivery.

In the search parameters, you can specify that the system should use the first match found.



You can also exclude delivered and invoiced items.



Initialization parameter	Description
<b>Delete delivered</b>	Excludes items that have been partially delivered.
<b>Delete invoiced</b>	Excludes items that have been partially invoiced.
<b>Delete compl. delivered</b>	Excludes items that have been completely delivered.
<b>Delete compl. invoiced</b>	Excludes items that have been completely invoiced.

### PO item proposal as determination

/EBY/CL\_PDOR\_DET\_ITEMPROP\_INIT

Adds line items from the purchase order to the order confirmation. Only line items that have confirmation control set in the **Confirmations** tab in the PO can be added. Uses the search **Item proposal from PO** (/EBY/CL\_PDOR\_DET\_EBELP\_ITEMPROP).

### Material determination

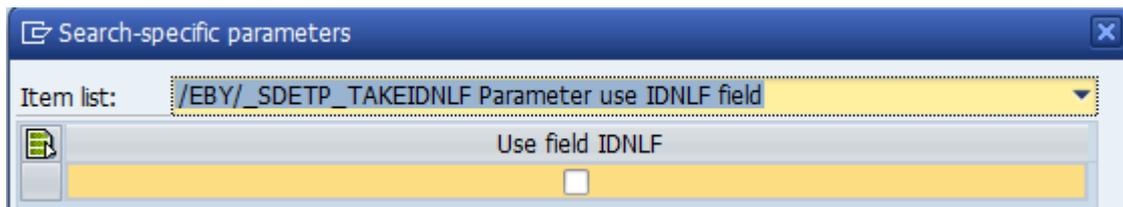
/EBY/CL\_PDOR\_DET\_MAT\_INIT

Determines the material number by checking whether the material number given on the order confirmation is (depending on the defined searches):

Search type	Search name
The material number from the material master data	/EBY/CL_PDOR_DET_MAT_MARA
A manufacturer part number maintained in the material master data	/EBY/CL_PDOR_DET_MAT_MANU

Search type	Search name
An EAN maintained in the material master data	/EBY/CL_PDOR_DET_MAT_EAN
A vendor material number maintained in the table EINA	/EBY/CL_PDOR_DET_MAT_EINA
The vendor material number given on the purchase order	/EBY/CL_PDOR_DET_MAT_VENDOR

In the initialization parameters, you can specify that the field IDNLF should be used:



### Purchase Order determination

/EBY/CL\_PDOR\_DET\_PO\_INIT

Verifies the purchase order number given on the order confirmation by searching the EKKO (purchasing document header) table for this number. Uses the search **Search PO in table EKKO** (/EBY/CL\_PDOR\_DET\_PO\_EKKO).

### Vendor determination

/EBY/CL\_PDOR\_DET\_VENDOR\_INIT

Verifies the vendor based on (depending on the defined searches):

Search type	Search name
The account number given on the order confirmation	/EBY/CL_PDOR_DET_VENDOR_ACC
The account number given on the purchase order	/EBY/CL_PDOR_DET_VENDOR_ACCPO
The vendor number given on the order confirmation	/EBY/CL_PDOR_DET_VENDOR_LFA1
The vendor number given on the purchase order	/EBY/CL_PDOR_DET_VENDOR_PO
The VAT number given on the order confirmation	/EBY/CL_PDOR_DET_VENDOR_STCEG

## Document model customization

The document model provides an interface structure for each logical level containing all fields that are available for that level. Using the customer includes (CIs) provided in this structure, you can extend the document model to include customer-specific fields. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

- Use the include ending in `_DATA` if the custom field should be automatically persisted in the database.
  - Use the include ending in `_DISP` if you only want to automatically calculate a field at runtime and display it.
- 🔴 **Important:** Never add customer-specific fields directly to the customer include, but use your own customer-specific structure instead. For example, first create a customer-specific structure corresponding to each used CI, include that structure in the CI and put all customer-specific fields in the customer-specific structure. **Do not use an APPEND structure.** Set the enhancement category of the customer-specific structure to **Can Be Enhanced, Character-Type or Numeric-Type** (menu **Extras**). Never use the enhancement category **Can Be Enhanced (Deep)**. See the SAP documentation for more information.

Logical level	Package	Interface structure	Customer Includes
Header data	/EBY/PDOR_MODEL	/EBY/PDOR_SHDR_IF	CI_EBY_PDOR_SHDR_DATA CI_EBY_PDOR_SHDR_DISP
Item data	/EBY/PDOR_MODEL	/EBY/PDOR_SITM_IF	CI_EBY_PDOR_SITM_DATA CI_EBY_PDOR_SITM_DISP
PO item data	/EBY/PDOR_MODEL	/EBY/PDOR_SEKPO_IF	CI_EBY_PDOR_SEKPO_DATA CI_EBY_PDOR_SEKPO_DISP

## View model customization

This section provides detailed information about which settings to enter when creating a customer view model for Order Confirmations. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

## By GUI element

- SAP document overview list:

**Header data grid:**  
Component type: Blank, Grid No.: 1, Logical level: HEADER

St...	Wf...	At...	PD doc no	Order acknowl.	Created on	Purch.Doc.	Crcy	Delivery da...	Vendor	Name 1	Acct w/ vndr
			Z9	3231941	03.08.2012	4500015523	GBP	15.08.2012	87241	John Doe Inc	
			Z8	3231941	03.08.2012	4500015523	GBP	15.08.2012	87241	John Doe Inc	
			Z7	3231941	03.08.2012	4500015523	GBP	10.08.2012	87241	John Doe Inc	
			Z6	3231941	03.08.2012	4500015523	GBP	10.08.2012	87241	John Doe Inc	
			Z5	3231941	03.08.2012	4500015523	GBP	10.08.2012	87241	John Doe Inc	
			Z4	3231941	03.08.2012	4500015523	GBP	10.08.2012	87241	John Doe Inc	
			Z3	3231941	03.08.2012	4500015523	GBP	10.08.2012	87241	John Doe Inc	
			Z2	3231941	03.08.2012	4500015523	GBP	10.08.2012	87241	John Doe Inc	
			Z1	3231941	03.08.2012	4500015523	GBP	10.08.2012	87241	John Doe Inc	
			Z0	3231941	03.08.2012	4500015523	GBP	10.08.2012	87241	John Doe Inc	
			69	3231941	03.08.2012	4500015523	GBP	10.08.2012	87241	John Doe Inc	
			68	3231941	08.03.2012	4500015523	GBP	08.10.2012	87241	John Doe Inc	
			67	3231942	08.03.2012	4500015523	GBP	10.08.2012	87241	John Doe Inc	
			66	3231941	08.03.2012	4500015523	GBP	08.10.2012	87241	John Doe Inc	
			65	3231941	08.03.2012	4500015523	GBP	08.10.2012	87241	John Doe Inc	
			64	3231941	08.03.2012	4500015523	GBP	08.10.2012	87241	John Doe Inc	

**Items grid:**  
Component type: Blank, Grid No.: 2, Logical level: ITEMS

Purch.Doc.	It...	Material	MPN	Quantity	OUUn	Deliv. date	Net price	Net va
4500015523	10		63302	3	PC	15.08.2012	26.11	78.

**PO Items grid:**  
Component type: Blank, Grid No.: 2, Logical level: ITEMS

Purch.Doc.	It...	Material	Short text	Vendor mat. no.	Quantity	OUUn	Delivery da...	Hst.
4500015523	10		Jacket	63302	3	PC	05.08.2012	
	20		Gloves	63305	60	PC	06.08.2012	

- SAP document detail:

To maintain the SAP GUI detail screen, use SAP transaction /EBY/PDVI\_VDSTC.

- Web Application document overview list:

**Header data grid:**  
Component type: WA List Header, Grid No.: 1, Logical level: HEADER

Status	WF status	PD doc no ↓	Acknowl.no	Created on	Purch.Doc.	Currency
<input type="checkbox"/>		2	AB38224	12/13/10	4500016167	EUR
<input type="checkbox"/>		1	AB875764	12/13/10	4500016166	EUR

▪ Web Application document detail:

The screenshot displays the 'Order Response 2' web application interface. On the left is a 'Worklist' sidebar with categories like Requisition (156), Goods Receipt (344), Financial Posting (65), Payment Advice (124), Customer Order (367), Incoming Invoice (61), Document Approval (12), Payment Approval (154), Order Confirmations (2), Cost Center Maintenance (14), Vendor Master Maintenance (2), GL Account Maintenance (1), Profitcenter Maintenance (1), Asset Master Maintenance (1), and All Workflows (150). The main area is titled 'Order Response 2' and contains three sections:

- Header:** A form with 'General Data' (Notes: 0, Attachments: 1), 'Document status', 'Workflow status', 'PROCESS DIRECTOR document number' (2), and 'Order acknowledgement' (AR20224). A callout box states: 'Header data grid: Component type: WA Document Detail, Grid No.: 1, Logical level: HEADER'.
- Items (4):** A table with columns: Item, Material, Vend. Mat., Quantity, Order Unit, Deliv. date, Net price, Net, Currency. Callout box: 'Items grid: Component type: WA Document Detail, Grid No.: 2, Logical level: ITEMS'.
- PO Items (4):** A table with columns: Purch.Doc., Item, Material, Short text, Vend. mat., Quantity, Order Unit, Deliv. date, Net price, Net, Currency, Ackn. reqd, Acknow.lno, Conf/ContrK, GR, Del.Com. Callout box: 'PO items grid: Component type: WA Document Detail, Grid No.: 3, Logical level: PO\_ITEMS'.

At the top right, there is a 'Current messages' section with a message: 'For messages from previous steps use the "Audit trail" button'.

- Workflow status:

**Workflow selection**

Workflow	Approve and create PO (2200)
Status description	Running
Started by	Blake Ev
Created on	08/22/2011
Time created	17:11:21
Workflow due date	08/30/2011

**Current messages**

**Next**

Status	Step	Date	Time	Processor	Overdue
	Create PO			Karina Kunz	

**Current**

Status	Step	Date	Time	Processor	Overdue
	Approve PO	08/22/2011	17:11:42	Roger Tillman	08/30/2011

Continue

### By view model component type

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
	1	HEADER	ZXY_PDOR_SGRID1 ZXY_PDOR_LGRID1	Header Header data
	2	ITEMS	ZXY_PDOR_SGRID2 ZXY_PDOR_LGRID2	Items Items
	3	PO_ITEMS	ZXY_PDOR_SGRID3 ZXY_PDOR_LGRID3	PO Items PO Items
WA Document Detail	1	HEADER	ZXY_PDOR_SWA_DOC_GRID1 ZXY_PDOR_LWA_DOC_GRID1	Header Header data

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
WA Document Detail	2	ITEMS	ZXY_PDOR_SWA_DOC_GRID2 ZXY_PDOR_LWA_DOC_GRID2	Items Items
WA Document Detail	3	PO_ITEMS	ZXY_PDOR_SWA_DOC_GRID3 ZXY_PDOR_LWA_DOC_GRID3	PO Items PO Items
WA List Header	1	HEADER	ZXY_PDOR_SWA_LIST_GRID1 ZXY_PDOR_LWA_LIST_GRID1	Header Header data
Workflow status	1	HEADER	ZXY_PDOR_SWA_STAT_GRID1 ZXY_PDOR_LWA_STAT_GRID1	Header Header data
Header data for archive log	1	HEADER	ZXY_PDOR_SARC_LOG_GRID1 ZXY_PDOR_LARC_LOG_GRID1	Header Header data

## Document statuses

PROCESS DIRECTOR ships with these system document statuses for Order Confirmations:

Status	Status Description
AP	Approved
ER	In error or incomplete
NN	New or ready for processing
OK	Processed
XX	Rejected

## Text types

PROCESS DIRECTOR ships with this system text type for Order Confirmations:

Logical level	Text object	Description	ID	Description	Multiple
HEADER	/EBY/PD	PROCESS DIRECTOR texts	PDN	PROCESS DIRECTOR note	X

## User exits / BAdIs

### BAdI Posting (Order Confirmations)

This user exit / BAdI currently provides the interface methods [BEFORE\\_POST](#) and [AFTER\\_POST](#) that you can implement to add your own customer logic before or after an Order Confirmation document is posted.

#### BEFORE\_POST method

#### Calling sequence

This method is called from the action `POST_DOC` Post document. It runs immediately *before* document posting occurs.

#### Method signature

Type	Parameter	Description
	IR_OBJECT	Order Response (OR) object
	IS_BO_HDR	Data fields of a PD base object
	IS_OR_HDR	Order Response (OR) header
	IT_ITEMS	Order Response (OR) items
	IT_PO_ITEMS	Generic PO items data
	CT_IDOIC_DATA	IDoc data

## AFTER\_POST method

### Calling sequence

This method is called from the action `POST_SUCCESS` Posting succeeded. It runs after the document status `Posted` has been set.

**Note:** If document posting fails, `AFTER_POST` is *not* called.

### Method signature

Type	Parameter	Description
▶	(VALUE) IR_OBJECT	Order Response (OR) object
▶	IT_IDOC_DATA	IDoc data
▶	IT_PO_ITEMS	Generic PO items data
▶▶	CS_BO_HDR	Data fields of a PD base object
▶▶	CS_OR_HDR	Order Response (OR) header
▶▶	CT_ITEMS	Order Response (OR) items

## Payment Advices (PA)

SAP Object type: /EBY/PDPA

Package: /EBY/PDPA

This chapter provides reference information for Payment Advices.

### Preset classes for Payment Advices

The following preset classes are available for Payment Advices in the PROCESS DIRECTOR standard. You can also use [general preset classes](#) that are available for all process types.

#### Preset customer name

/EBY/CL\_PDPA\_PRESET\_CUSTOMER

Presets the customer name from the customer number. Set this preset for the `NAME` field.

## Preset item data from BSID

/EBY/CL\_PDPA\_PRESET\_ITEM\_BSID

Presets the following item data from the table BSID:

- Accounting document line item number
- Fiscal year
- Alternative account
- Debit/credit indicator

The customer number in the header data is also populated, if it is initial.

The company code and accounting document number must be known.

Using this preset can be faster than using open item determinations.

## Preset payment amount from gross amount

/EBY/CL\_PDPA\_PRESET\_PAYMNT

At header level, presets the payment amount from the gross amount, or the gross amount from the payment amount.

## Updates items in the Electronic Bank Statement

/EBY/CL\_PDES\_PRESET\_ITEM

- ❗ Adds the Payment Advice number and customer number to line items in the Electronic Bank Statement.
- ❗ **Note:** Although this preset is a PDES class, it must be added to the Payment Advices presets configuration, not the Electronic Bank Statement preset configuration.

## Checks for Payment Advices

The following check events are available for Payment Advices in the PROCESS DIRECTOR standard. You can also use [general checks](#) that are available for all process types.

### Check company code

/EBY/CL\_PDPA\_EVT\_CHK\_COMP\_CODE

Checks whether the company code entered in the payment advice is maintained in SAP and is valid.

## Check if item matches a customer open item

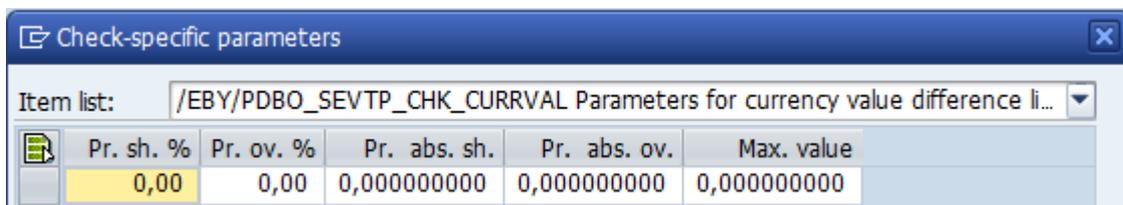
/EBY/CL\_PDPA\_EVT\_CHK\_OPEN\_ITEM

Checks whether this line item has been billed to a customer and whether the line item is still open (not paid).

In the initialization parameters, you can define the following rules:

- value shortfall in %
- value overrun in %
- absolute value shortfall
- absolute value overrun
- limit for standard line item value

If one of the defined rules is broken, a message is generated.



Pr. sh. %	Pr. ov. %	Pr. abs. sh.	Pr. abs. ov.	Max. value
0,00	0,00	0,000000000	0,000000000	0,000000000

## Check integrity of header amounts

/EBY/CL\_PDPA\_EVT\_CHK\_HDR\_AMNT

Checks whether the values in the total amount fields of the header are equal to the sum of the corresponding line item fields.

## Check integrity of item amounts

/EBY/CL\_PDPA\_EVT\_CHK\_ITM\_AMNT

Checks whether the values in the amount fields of the line items are correct (Payment amount should be equal to Gross amount - Discount amount - Deduction amount).

## Check the customer number of the payment advice

/EBY/CL\_PDPA\_EVT\_CHK\_CUSTOMER

Checks whether the customer number entered in the payment advice is maintained in SAP for the given company code and is valid.

## Determinations for Payment Advices

These determinations are available for Payment Advices in the PROCESS DIRECTOR standard.

### Payment Advice customer determination

/EBY/CL\_PDPA\_DET\_CUST\_INIT

Determines the customer number based on (depending on the defined searches):

Search type	Search name
KNAI (Customer Master General Data)	/EBY/CL_PDPA_DET_CUST_KNA1
KNBI (Customer Master Company Code Data)	/EBY/CL_PDPA_DET_CUST_KNB1
KNVV (Customer Master Sales Data)	/EBY/CL_PDPA_DET_CUST_KNVV
Amount. This search first tries to match the payment amount and gross amount. If no match is found, it then considers discounts, local currency, and statistical values to find a match.	/EBY/CL_PDPA_DET_CUST_AMOUNT
Accounting document number	/EBY/CL_PDPA_DET_CUST_BELNR
Document date	/EBY/CL_PDPA_DET_CUST_BLDAT
Posting date	/EBY/CL_PDPA_DET_CUST_BUDAT
Billing document number	/EBY/CL_PDPA_DET_CUST_VBELN
Reference document number	/EBY/CL_PDPA_DET_CUST_XBLNR
Reference document number from reference in item	/EBY/CL_PDPA_DET_CUST_XR_XBLNR

It searches in open items, cleared documents and parked documents to try and find the best match. Various [search parameters](#) are available.

## Payment Advice item mapper determination

/EBY/CL\_PDPA\_DET\_ITEMPROP\_INIT

Determines the document number, item number and fiscal year in the line items by searching for (depending on the defined searches):

Search type	Search name
Amount. This search first tries to match the payment amount and gross amount. If no match is found, it then considers discounts, local currency, and statistical values to find a match.	/EBY/CL_PDPA_DET_ITEMMAP_AMT
Document number	/EBY/CL_PDPA_DET_ITEMMAP_BELNR
Document date	/EBY/CL_PDPA_DET_ITEMMAP_BLDAT
Posting date	/EBY/CL_PDPA_DET_ITEMMAP_BUDAT
Item text	/EBY/CL_PDPA_DET_ITEMMAP_SGTXT
Billing document	/EBY/CL_PDPA_DET_ITEMMAP_VBELN
Reference number	/EBY/CL_PDPA_DET_ITEMMAP_XBLNR

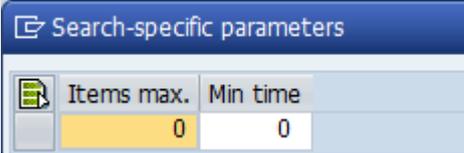
It searches in open items, cleared documents and parked documents to try and find the best match. Various [search parameters](#) are available.

## Payment Advice item proposal determination

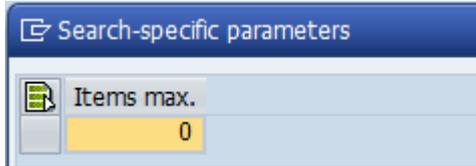
/EBY/CL\_PDPA\_DET\_ITEMPROP\_INIT

Retrieves open items for the company code and customer number entered in the document header and creates a corresponding line item for each open item. In the search parameters, you can specify the maximum number of items that should be retrieved.

Depending on the defined searches, the following items are added:

Search type	Search name
<p>Propose all items</p>	<p>/EBY/CL_PDPA_DET_ITEMPROP or /EBY/CL_PDPA_DET_ITEMPROP_Z00</p>
<p>Currently due items. This means items for which the invoice due date is later than the payment date, and the posting date lies before the payment date by a specified minimum number of days, which is defined in the search parameters (<b>Min time</b>).</p> 	<p>/EBY/CL_PDPA_DET_ITEMPROP_Z21</p>
<p>Overdue items</p>	<p>/EBY/CL_PDPA_DET_ITEMPROP_Z22</p>
<p>Unique combination of items. This means items are only added if there is a single, unique combination of open items that matches the payment amount.</p> <p>Example for unique combination:</p> <p>Assume the following customer open items are available:</p> <ul style="list-style-type: none"> <li>▪ Item 1 with the amount 100</li> <li>▪ Item 2 with the amount 200</li> <li>▪ Item 3 with the amount 150</li> <li>▪ Item 4 with the amount 250</li> </ul> <p>If the payment amount is 250, two different combinations of items match the payment amount:</p> <ul style="list-style-type: none"> <li>▪ Item 1 + Item 3 = 250</li> <li>▪ Item 4 = 250</li> </ul> <p>If the payment amount is 600, there is a unique combination, because only one combination of items matches the payment amount:</p> <p>Item 2 + Item 3 + Item 4 = 600 (no other combination amounts to 600)</p>	<p>/EBY/CL_PDPA_DET_ITEMPROP_Z231</p>

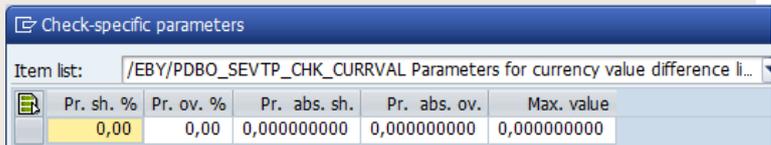
In the search parameters, you can specify the maximum number of items that should be retrieved:



## Search parameters

Which of these parameters are available depends on the selected search.

Parameter type	Description
/EBY/PDPA_LEVTP_BS_FLD_FILTER	<p><b>Condition for BSAD and BSEG filtering</b></p> <p>Enables you to specify conditions for the search. Only items that meet the specified criteria (or do not meet them, depending on the selected option) are included in the search.</p> <p>For example, particularly for cleared items, you may want to filter out items that are older than a specific date to exclude them from the search.</p>
/EBY/PDBO_SEVTP_CHK_CURRVAL	<p><b>Parameters for currency value difference limits</b></p> <p>Enables you to specify tolerances for currency values:</p> <ul style="list-style-type: none"> <li>▪ value shortfall in %</li> <li>▪ value overrun in %</li> <li>▪ absolute value shortfall</li> <li>▪ absolute value overrun</li> <li>▪ limit for standard line item value</li> </ul> <p>If one of the defined rules is broken, a message is generated.</p>



Parameter type	Description
/EBY/PDPA_LEVTP_ITEM_TO_BS_FLD	<p><b>Mapping of BSID, BSAD and BSEG fields to PDPA items</b></p> <p>Enables you to specify which fields in the SAP tables are compared with which PROCESS DIRECTOR Payment Advice fields, and in which order the fields are compared.</p>
/EBY/PDBO_SEVTP_FIELD_PROP	<p><b>Field properties</b></p> <p>Enables you to specify a minimum field length. If the field value is shorter than this length, it is not taken into account by the determination.</p> <p>If you check the <b>Wildcard</b> option, you can also specify a partial minimum length.</p>

## Document model customization

The document model provides an interface structure for each logical level containing all fields that are available for that level. Using the customer includes (CIs) provided in this structure, you can extend the document model to include customer-specific fields. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

- Use the include ending in `_DATA` if the custom field should be automatically persisted in the database.
- Use the include ending in `_DISP` if you only want to automatically calculate a field at runtime and display it.
- 🔴 **Important:** Never add customer-specific fields directly to the customer include, but use your own customer-specific structure instead. For example, first create a customer-specific structure corresponding to each used CI, include that structure in the CI and put all customer-specific fields in the customer-specific structure. **Do not use an APPEND structure.** Set the enhancement category of the customer-specific structure to **Can Be Enhanced, Character-Type or Numeric-Type** (menu **Extras**). Never use the enhancement category **Can Be Enhanced (Deep)**. See the SAP documentation for more information.

Logical level	Package	Interface structure	Customer Includes
Header data	/EBY/PDPA_MODEL	/EBY/PDPA_SHDR_IF	CI_EBY_PDPA_SHDR_DATA CI_EBY_PDPA_SHDR_DISP

Logical level	Package	Interface structure	Customer Includes
Item data	/EBY/PDPA_MODEL	/EBY/PDPA_SITM_IF	CI_EBY_PDPA_SITM_DATA CI_EBY_PDPA_SITM_DISP

## View model customization

This section provides detailed information about which settings to enter when creating a customer view model for Payment Advices. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

### By GUI element

- SAP document overview list:

The screenshot displays the 'Display Payment Advice #30 (1 of 17)' window. The main table contains the following data:

Stat.	VF	AB	PD do.	CoCd	Customer	Customer	Acct at cust.	Payment advice	Crty	Payment Amount	Gross amount	Cash discount	Payment	Pmnt
			20	1000	SYNTHES Corpor...				GBP	46.40	46.40	40.00	RA-6007432	06.00
			20	1000	Readsoft GmbH	PA2156			EUR	562.60	562.60	0.00	1212909	14.00
			20	1000	VWR International	10553			GBP	106.00	106.00	0.00	56-1234	03.00
			22	1000	SYNTHES Corpor...			040000000000000027	GBP	46.40	46.40	0.00	RA-6007432	06.00
			25	1000	VWR International	10553		040000000000000028	GBP	106.00	106.00	0.00	56-1234	03.00
			25	1000	SYNTHES Corpor...				GBP	46.40	46.40	40.00	RA-6007432	06.00
			11	1000	Readsoft GmbH	PA2156			EUR	562.60	562.60	0.00	1212909	14.00
			10	1000	VWR International	10553			GBP	106.00	106.00	0.00	56-1234	03.00
			10	1000	VWR International	10553		040000000000000024	GBP	106.00	106.00	0.00	56-1234	03.00
			10	1000	VWR International	10553		040000000000000023	GBP	106.00	106.00	0.00	56-1234	03.00
			10	1000	VWR International	10553		040000000000000022	GBP	106.00	106.00	0.00	56-1234	03.00
			10	1000	Readsoft GmbH	PA2156		040000000000000021	EUR	562.60	562.60	0.00	1212909	14.00

Below the main table, a summary table is visible:

Crty	Payment amount	Gross amount	Discount amnt	Document	Reference	Year
GBP	11.60	11.60	0.00	1800000009	12-0002	2007
GBP	23.20	23.20	0.00	1800000010	12-0003	2007
GBP	11.60	11.60	0.00	1800000011	12-0004	2007

- SAP document detail:

To maintain the SAP GUI detail screen, use SAP transaction /EBY/PDVI\_VDSTC.

- Web Application document overview list:

Status	WF status	PD doc no ↓	CoCode	Customer	Customer
■		125	1000	10553	WWR International
■					ReadSoft GmbH
■					WWR International
■		121	1000		
■		120	1000	2156	ReadSoft GmbH

**Header data grid:**  
Component type: WA List Header, Grid No.: 1,  
Logical level: HEADER

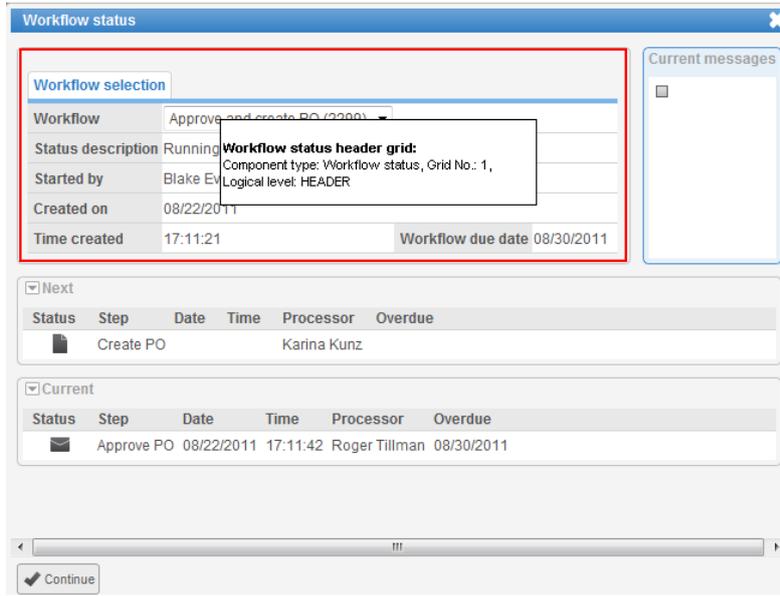
- Web Application document detail:

**Header data grid:**  
Component type: WA Document Detail, Grid No.: 1,  
Logical level: HEADER

Pymnt Amt	Gross	Discnt	Currency	DocumentNo ↑	Reference	Fisc. Year
0.00	31.80	0.00	GBP	1800000005	WWR-0001	
0.00	31.80	0.00	GBP	1800000006	WWR-0002	
0.00	42.40	0.00	GBP	1800000007	WWR-0003	

**Items grid:** Component type: WA Document Detail,  
Grid No.: 2, Logical level: ITEMS

▪ Workflow status:



**By view model component type**

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
	1	HEADER	ZXY_PDPA_SGRID1 ZXY_PDPA_LGRID1	Header Header data
	2	ITEMS	ZXY_PDPA_SGRID2 ZXY_PDPA_LGRID2	Items Items
WA Document Detail	1	HEADER	ZXY_PDPA_SWA_DOC_GRID1 ZXY_PDPA_LWA_DOC_GRID1	Remittance Advice Remittance Advice
WA Document Detail	2	ITEMS	ZXY_PDPA_SWA_DOC_GRID2 ZXY_PDPA_LWA_DOC_GRID2	Items Items
WA List Header	1	HEADER	ZXY_PDPA_SWA_LIST_GRID1 ZXY_PDPA_LWA_LIST_GRID1	Header Header data
Workflow status	1	HEADER	ZXY_PDPA_SWA_STAT_GRID1 ZXY_PDPA_LWA_STAT_GRID1	Header Header data
Header data for archive log	1	HEADER	ZXY_PDPA_SARC_LOG_GRID1 ZXY_PDPA_LARC_LOG_GRID1	Header Header data

## Document statuses

PROCESS DIRECTOR ships with these system document statuses for Payment Advices:

Status	Status Description
ER	New or in error
OK	Correct
PO	Posted
XX	Rejected

## User exits / BAdIs

### BAdI Posting (Payment Advices)

This user exit / BAdI currently provides the interface methods [BEFORE\\_POST](#) and [AFTER\\_POST](#) that you can implement to add your own customer logic before or after a Payment Advice document is posted.

#### BEFORE\_POST method

#### Calling sequence

This method is called from the action `POST_DOC Post document`. It runs immediately *before* document posting occurs.

#### Method signature

Type	Parameter	Description
▶□	VALUE ( IR_OBJECT )	Payment Advices object
▶□	IS_BO_HDR	Data fields of a PD base object
▶□	IS_PA_HDR	Payment Advices (PA) header data
▶□	IT_ITEMS	Payment Advices (PA) items

Type	Parameter	Description
▶□	IB_BACKGROUND	Background/online processing
▶□▶	CT_BDCDATA	Batch Input Branch to Objects
▶□▶	CC_TCODE	Transaction code
▶□▶	CS_OPTIONS	Parameter string for runtime of CALL TRANSACTION USING ...

## AFTER\_POST method

### Calling sequence

This method is called from the action `POST_SUCCESS Posting succeeded`. It runs after the document status `Posted` has been set.

**Note:** If document posting fails, `AFTER_POST` is *not* called.

### Method signature

Type	Parameter	Description
▶□	(VALUE) IR_OBJECT	Payment Advices object
▶□	IT_BDCDATA	Batch Input Branch to Objects
▶□	IC_TCODE	Transaction code
▶□	IT_MESSAGES	Collecting messages in the SAP system
▶□	IB_BACKGROUND	Background/online processing
▶□	IS_OPTIONS	Parameter string for runtime of CALL TRANSACTION USING ...

Type	Parameter	Description
▶□	II_RC	Return code after CALL TRANSACTION
▶▶	CS_BO_HDR	Data fields of a PD base object
▶▶	CS_PA_HDR	Payment Advices (PA) header data
▶▶	CT_ITEMS	Payment Advices (PA) items

## Programs

The following program is available for use with Payment Advices:

- [/EBY/PDPA\\_FINALIZE\\_DOC](#)

Checks if the open items of the payment advice document are already cleared and links the payment advice document to the clearing posting.

### [/EBY/PDPA\\_FINALIZE\\_DOC](#)

Use the /EBY/PDPA\_FINALIZE\_DOC program to link SAP clearing documents to PROCESS DIRECTOR payment advice documents.

The program checks if the open items of the payment advice document are already cleared and links the payment advice document to the clearing posting. Items that are not assigned are ignored.

## Payment Approvals (PR)

SAP Object type: /EBY/PDPR

Package: /EBY/PDPR

This chapter provides reference information for Payment Approvals.

### Checks for Payment Approvals

The following checks are available for Payment Approvals in the PROCESS DIRECTOR standard. You can also use [general checks](#) that are available for all process types.

## Check payment proposal was recreated

/EBY/CL\_PDPR\_EVT\_CHK\_CREA\_DATE

Checks whether the payment proposal was deleted and recreated in SAP with the same identification. If this is the case, a warning message is issued.

## Check payment block

/EBY/CL\_PDPR\_EVT\_CHK\_PMT\_BLOCK

Checks whether payment blocks have been set in the payment proposal document.

## Check payment proposal approved

/EBY/CL\_PDPR\_EVT\_CHK\_PMT\_PROPO

Checks whether workflow approval has been completed for the entire payment proposal.

## Check payment proposal approved for one PD document

/EBY/CL\_PDPR\_EVT\_CHK\_PMT\_PRP\_1

Checks whether workflow approval has been completed for a single PROCESS DIRECTOR document.

## Process parameters for Payment Approvals

The following process parameters are available for Payment Approvals. You can also use the general process parameters that are available for all process types.

Process	Description
<b>Archive the output of a print report</b> (Archiving)	<p>Enables users to create a report containing the line item proposals and archive it with the PROCESS DIRECTOR document. In the parameters, you specify:</p> <ul style="list-style-type: none"> <li>▪ the report to create, for example, RFZALI20 (Payment list)</li> <li>▪ the report variant to use</li> <li>▪ the archiving document type to create (this must have the document class PDF)</li> <li>▪ (optional) the format to use. This specifies, for example, the maximum number of lines and columns per page.</li> </ul>

Process	Description
<b>Parameters controlling document creation based on</b>  (Other)	Enables you to control which payment proposal data is retrieved from SAP. See <i>Configuring payment approval data retrieval</i> in the <i>PROCESS DIRECTOR SAP Configuration Guide</i> for more information.
<b>Retrieve new documents from SAP on Worklist double-click</b>  (Other)	Turn on to retrieve new SAP payment proposals when users double-click a Payment Approval Worklist node. Payment proposals are also retrieved when the program <a href="#">/EBY/PDPR_IMPORT</a> runs. Under certain circumstances this can lead to duplicate documents in PROCESS DIRECTOR. Turn this setting off if you are experiencing problems with duplicate documents.

## Document model customization

The document model provides an interface structure for each logical level containing all fields that are available for that level. Using the customer includes (CIs) provided in this structure, you can extend the document model to include customer-specific fields. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

- Use the include ending in `_DATA` if the custom field should be automatically persisted in the database.
  - Use the include ending in `_DISP` if you only want to automatically calculate a field at runtime and display it.
- 🔴 **Important:** Never add customer-specific fields directly to the customer include, but use your own customer-specific structure instead. For example, first create a customer-specific structure corresponding to each used CI, include that structure in the CI and put all customer-specific fields in the customer-specific structure. **Do not use an APPEND structure.** Set the enhancement category of the customer-specific structure to **Can Be Enhanced, Character-Type or Numeric-Type** (menu **Extras**). Never use the enhancement category **Can Be Enhanced (Deep)**. See the SAP documentation for more information.

Logical level	Package	Interface structure	Customer Includes
Header data	/EBY/PDPR_MODEL	/EBY/PDPR_SHDR_IF	CI_EBY_PDPR_SHDR_DATA  CI_EBY_PDPR_SHDR_DISP

Logical level	Package	Interface structure	Customer Includes
Item data	/EBY/PDPR_MODEL	/EBY/PDPR_SITM_IF	CI_EBY_PDPR_SITM_DATA CI_EBY_PDPR_SITM_DISP

## View model customization

This section provides detailed information about which settings to enter when creating a customer view model for Payment Approvals. See the *PROCESS DIRECTOR Configuration Guide* for more information.

### By GUI element

- SAP document overview list:

**Header data grid:**  
Component type: Blank, Grid No.: 1, Logical level: HEADER

Stat.	Wt...	PD do...	Run date	ID	CoCd	Crcy	Local curr	amnt	P
			144 08.10.2010	HTH05	1000	EUR	32.094,41-	U	
			143 09.09.2010	DMPA1	1000	EUR	5.581,12-	U	
			139 31.08.2010	HTH02	1000	EUR	4.960,33-	U	
			137	HTH03	1000	EUR	300.100,00-	U	
			134 20.08.2002	SRB01	3000	USD	197,00-	C	
			133 07.08.2002	REID4	4500	CAD	12.164,16-	C	
			132	REID3	4500	CAD	94.668,35-	C	
			126 31.12.2001	ZB300	R300	USD	6.014.092,38-	C	
			125	ZB100	R100	EUR	478.206,65-	S	
			124	Z3000	3000	USD	22.933,02-	C	
			123	Z1000	1000	EUR	480.098,17-	S	
			122	T1000	1000	EUR	187.387,97-	S	
			121 28.12.2001	BQ001	3000	USD	2.100,00-	C	
			120 08.10.2010	ZB000	R000	EUR	3.725,24	C	

**Items grid:** Component type: Blank, Grid No.: 2, Logical level: ITEMS

Inv.	Alt.	State	PBk	Vendor	Document No.	Amount	Crcy	PaY	Due date	Payment	Year	Co...
120					E110000001						2002	1000
120					E110000001						1000	1000
120					E110000001						1000	1000
120					E110000001						1000	1000
1200					1900001212	3.725,24	EUR	ZB00	08.10.2010	E110000001	1000	1000

- SAP document detail:

To maintain the SAP GUI detail screen, use SAP transaction /EBY/PDVI\_VDSTC.

▪ Web Application document overview list:

Worklist		Payment Approval (154)						
Refresh worklist		<b>Status</b>	<b>Wf status</b>	<b>PD doc no</b> ↓	<b>Date</b>	<b>ID</b>	<b>CoCd</b>	<b>Currency</b>
▶ Requisition (156)		\$		154	02/26/04	R0304	1000	EUR
▶ Goods Receipt (344)		\$		153	04/27/04	R0504	1000	EUR
▶ Financial Posting (65)		✓					000	EUR
▶ Payment Advice (124)		✓					000	USD
▶ Customer Order (367)		✓					000	EUR
Incoming Invoice		✓		149	02/27/03	R0303	1000	EUR
▶ Document Approval (12)		✓		148	08/26/03	R0903	1000	EUR
▶ Payment Approval (154)		✓		147	07/30/03	DG073	3000	USD
To approve (0)		✓		146	08/27/03	DGMAN	3000	USD
▶ My Workflow Inbox (0)								
▶ Payment Approval Cockpit (154)								

**Header data grid:**  
Component type: WA List Header, Grid No.: 1,  
Logical level: HEADER

▪ Web Application document detail:

Worklist		Payment Approval 136																																																																																							
Refresh worklist		← Back   Audit trail																																																																																							
<ul style="list-style-type: none"> <li>Requisition (156)</li> <li>Goods Receipt (344)</li> <li>Financial Posting (65)</li> <li>Payment Advice (124)</li> <li>Customer Order (367)</li> <li>Incoming Invoice</li> <li>Document Approval (12)</li> <li>Payment Approval (154)</li> <li>To approve (0)</li> <li>MyWorkflow Inbox (0)</li> <li>Payment Approval Cockpit (154)</li> <li>Order Confirmations (2)</li> <li>Cost Center Maintenance (14)</li> <li>Vendor Master Maintenance (2)</li> <li>GL Account Maintenance (1)</li> <li>Profitcenter Maintenance (1)</li> <li>Asset Master Maintenance (1)</li> <li>All Workflows (150)</li> <li>My Workflow Inbox (31)</li> </ul>		<b>Payment Approval - Header data</b> <b>General</b> Document status: ✓ Payment run Workflow status: PD document number: 136 <b>Header data grid:</b> Component type: WA Document Detail, Grid No.: 1, Logical level: HEADER Payment method: C											Current messages For messages from previous steps use the "Audit trail" button																																																																												
		<b>Items (4)</b> <table border="1"> <thead> <tr> <th>Att.</th> <th>State</th> <th>PKB</th> <th>Vendor</th> <th>DocumentNo</th> <th>Amount</th> <th>Currency</th> <th>Pmnt terms</th> <th>Due date</th> <th>Payment</th> <th>Fisc. Year</th> <th>CoCode</th> <th>Customer</th> <th>Account No</th> <th>Bank no</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td>Free for payment</td> <td>3906</td> <td>5100000148</td> <td>2,399.30</td> <td>USD</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td>Free for payment</td> <td>3906</td> <td>5100000197</td> <td>2,399.30</td> <td>USD</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td>Free for payment</td> <td>3906</td> <td>5100000218</td> <td>1,019.30</td> <td>USD</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td>Free for payment</td> <td>3906</td> <td>5100000243</td> <td>1,019.30</td> <td>USD</td> <td>ZB01</td> <td>01/02/03</td> <td>2000000034</td> <td>2002</td> <td>3000</td> <td></td> <td></td> </tr> </tbody> </table> <b>Items grid:</b> Component type: WA Document Detail, Grid No.: 2, Logical level: ITEMS													Att.	State	PKB	Vendor	DocumentNo	Amount	Currency	Pmnt terms	Due date	Payment	Fisc. Year	CoCode	Customer	Account No	Bank no				Free for payment	3906	5100000148	2,399.30	USD											Free for payment	3906	5100000197	2,399.30	USD											Free for payment	3906	5100000218	1,019.30	USD											Free for payment	3906	5100000243	1,019.30	USD	ZB01	01/02/03	2000000034	2002	3000		
Att.	State	PKB	Vendor	DocumentNo	Amount	Currency	Pmnt terms	Due date	Payment	Fisc. Year	CoCode	Customer	Account No	Bank no																																																																											
			Free for payment	3906	5100000148	2,399.30	USD																																																																																		
			Free for payment	3906	5100000197	2,399.30	USD																																																																																		
			Free for payment	3906	5100000218	1,019.30	USD																																																																																		
			Free for payment	3906	5100000243	1,019.30	USD	ZB01	01/02/03	2000000034	2002	3000																																																																													

- Workflow status:

**Workflow selection**

Workflow	Approve and create PO (2200)
Status description	Running
Started by	Blake Ev
Created on	08/22/2011
Time created	17:11:21
Workflow due date	08/30/2011

**Workflow status header grid:**  
Component type: Workflow status, Grid No.: 1,  
Logical level: HEADER

**Current messages**

**Next**

Status	Step	Date	Time	Processor	Overdue
	Create PO			Karina Kunz	

**Current**

Status	Step	Date	Time	Processor	Overdue
	Approve PO	08/22/2011	17:11:42	Roger Tillman	08/30/2011

Continue

### By view model component type

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
	1	HEADER	ZXY_PDPR_SGRID1 ZXY_PDPR_LGRID1	Header Header data
	2	ITEMS	ZXY_PDPR_SGRID2 ZXY_PDPR_LGRID2	Items Items
WA Document Detail	1	HEADER	ZXY_PDPR_SWA_DOC_GRID1 ZXY_PDPR_LWA_DOC_GRID1	Header Header data
WA Document Detail	2	ITEMS	ZXY_PDPR_SWA_DOC_GRID2 ZXY_PDPR_LWA_DOC_GRID2	Items Items

Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming proposals)	Btn. label Grid title (naming proposals)
WA List Header	I	HEADER	ZXY_PDPR_SWA_LIST_GRIDI ZXY_PDPR_LWA_LIST_GRIDI	Header Header data
Workflow status	I	HEADER	ZXY_PDPR_SWA_STAT_GRIDI ZXY_PDPR_LWA_STAT_GRIDI	Header Header data
Header data for archive log	I	HEADER	ZXY_PDPR_SARC_LOG_GRIDI ZXY_PDPR_LARC_LOG_GRIDI	Header Header data

## Document statuses

PROCESS DIRECTOR ships with these system document statuses Payment Approvals:

Status	Status Description
PP	Payment proposal
PR	Payment run

## Text types

PROCESS DIRECTOR ships with this system text type for Payment Approvals:

Logical level	Text object	Description	ID	Description	Multiple
HEADER	/EBY/PD	PROCESS DIRECTOR texts	PDN	PROCESS DIRECTOR note	X

## Programs

The following programs are available for use with Payment Approvals:

- [/EBY/PDPR\\_IMPORT](#)

Import payment proposals from SAP into PROCESS DIRECTOR.

▪ [/EBY/PDPR\\_INVOICE\\_UPDATER](#)

This program is only relevant if you have payment proposal documents that were created in a version of PROCESS DIRECTOR prior to 7.1 Service Pack 1. This report reads the payment proposal line items and checks whether a corresponding PROCESS DIRECTOR Accounts Payable invoice document is available. If there is, the connection between the line item and the invoice is saved in the database.

**[/EBY/PDPR\\_IMPORT](#)**

Use the /EBY/PDPR\_IMPORT program to import payment proposals from SAP into PROCESS DIRECTOR.

This program can be scheduled to run at regular intervals, for example, overnight. Proposals are still imported when the user clicks a payment approval Worklist node or the **Refresh** button, but only the proposals that have been created since the last program run are imported. This can considerably improve performance.

We recommend that you schedule this program if you have a large number of payment proposals, or payment proposals with many line items.

**Note:** If you are experiencing problems with duplicate PROCESS DIRECTOR documents created from SAP payment proposals, turn off the process parameter [Retrieve new documents from SAP on Worklist double-click](#).

**[/EBY/PDPR\\_INVOICE\\_UPDATER](#)**

This program is only relevant if you have payment proposal documents that were created in a version of PROCESS DIRECTOR prior to 7.1 Service Pack 1. This report reads the payment proposal line items and checks whether a corresponding PROCESS DIRECTOR Accounts Payable invoice document is available. If there is, the connection between the line item and the invoice is saved in the database. This is necessary because prior to PROCESS DIRECTOR 7.1 SP1, these connections were read whenever a document was displayed, which had a negative effect on performance. From PROCESS DIRECTOR 7.1 SP1, these connections are automatically saved to the database.

A link to the PROCESS DIRECTOR Accounts Payable document is displayed in the **Inv.** column in the payment approval line item:

Inv.	PD AP ...	Att.	Edit	St...	PBk	Vendor	DocumentNo
						1000	<a href="#">1900000023</a>
	14804					1000	<a href="#">5100000146</a>
	14806					1000	<a href="#">5100000147</a>

## Requisitions (PO)

SAP Object type: /EBY/PDPO

Package: /EBY/PDPO

This chapter provides reference information for Requisitions.

### Preset classes for Requisitions

The following preset classes are available for Requisitions in the PROCESS DIRECTOR standard. You can also use [general preset classes](#) that are available for all process types.

#### Preset accounting from previous item

/EBY/CL\_PDPO\_PRESET\_ACC

Copies the values from the previous accounting item to new accounting items.

#### Preset company code from purchasing org

/EBY/CL\_PDPO\_PRESET\_COCO\_PORG

Sets the company code from the purchasing organization (if there is a unique assignment in table T024E).

#### Preset material master data

/EBY/CL\_PDPO\_PRESET\_MATMAS

Sets the material group and the order unit values from the material master data (MARA table). If the order unit is not available in the master data, the base unit of measure is set instead.

#### Preset vendor data for header

/EBY/CL\_PDPO\_PRESET\_HDR\_VENDOR

Sets the **Currency**, **Incoterm 1**, **Incoterm 2** and **Payment terms** values from the vendor master data.

#### TEMPLATE: Preset PO header

/EBY/CL\_PDPO\_PRESET\_HEADER\_TMP

This preset class is provided as a template that you can use to configure your own custom presets for Requisitions header data.

- 🔴 **Note:** Do not add this preset class to the presets configuration. Instead, create your own preset class based on the template and add this custom preset class check to the configuration.

## Checks for Requisitions

The following check events are available for Requisitions in the PROCESS DIRECTOR standard. You can also use [general checks](#) that are available for all process types.

### Check Company Code

/EBY/CL\_PDPO\_EVT\_CHK\_COMP\_CODE

Checks whether the company code entered in the requisition is maintained in SAP and is valid.

### Check Vendors returned from OCI

/EBY/CL\_PDPO\_EVT\_CHK\_OCI\_VEND

Checks if more than one vendor is returned from the OCI catalog.

### Check SAP Purchase Order creation

/EBY/CL\_PDPO\_EVT\_CHK\_POST

Simulates the creation of the purchase order in SAP and returns all messages generated during the simulation.

### Check purchase organization

/EBY/CL\_PDPO\_EVT\_CHK\_PURCH\_ORG

Checks whether the company code entered in the requisition is maintained in SAP and is valid.

### Check item quantity and account quantity sum or percentage

/EBY/CL\_PDPO\_EVT\_CHK\_QUANTITY

Checks the following and returns error messages if any apply:

- The item quantity differs from the account assignment quantity.
- The item quantity differs from the delivery schedule quantity.
- The total account assignment percentage distribution is not equal to 100%.

### Check maintained schedule lines

/EBY/CL\_PDPO\_EVT\_CHK\_SCHEDULE

Checks if there are line items without a scheduled delivery date or with a delivery date in the past.

### Check vendor

/EBY/CL\_PDPO\_EVT\_CHK\_VENDOR

Checks whether the vendor entered in the requisition is maintained in SAP for the given company code and purchasing organization, and whether the vendor is valid and not blocked.

## Process parameters for Requisitions

The following process parameters are available for Requisitions. You can also use the general process parameters that are available for all process types.

Process	Description
<b>Define XFlags for PO BAPI posting</b> (Checks, Posting)	

Process	Description																										
<p><b>Posting parameters</b> (Posting)</p>	<p>Enables you to override the default settings for posting requisitions.</p> <table border="1" data-bbox="480 391 1362 1704"> <thead> <tr> <th data-bbox="485 391 887 459">Parameter</th> <th data-bbox="890 391 1362 459">Description</th> </tr> </thead> <tbody> <tr> <td data-bbox="485 463 887 583"><b>Hold Purchase Order if faulty</b></td> <td data-bbox="890 463 1362 583"></td> </tr> <tr> <td data-bbox="485 587 887 706"><b>Hold Purchase Order if not faulty</b></td> <td data-bbox="890 587 1362 706"></td> </tr> <tr> <td data-bbox="485 710 887 829"><b>No message determination</b></td> <td data-bbox="890 710 1362 829"></td> </tr> <tr> <td data-bbox="485 834 887 953"><b>No messages necessary</b></td> <td data-bbox="890 834 1362 953"></td> </tr> <tr> <td data-bbox="485 957 887 1076"><b>No authorization check</b></td> <td data-bbox="890 957 1362 1076"></td> </tr> <tr> <td data-bbox="485 1081 887 1200"><b>No price transfer from last document</b></td> <td data-bbox="890 1081 1362 1200"></td> </tr> <tr> <td data-bbox="485 1204 887 1323"><b>Set update task local</b></td> <td data-bbox="890 1204 1362 1323"></td> </tr> <tr> <td data-bbox="485 1327 887 1447"><b>Call BAPI PO</b></td> <td data-bbox="890 1327 1362 1447"></td> </tr> <tr> <td data-bbox="485 1451 887 1570"><b>Wait on commit</b></td> <td data-bbox="890 1451 1362 1570"></td> </tr> <tr> <td data-bbox="485 1574 887 1693"><b>Call BAPI RQ</b></td> <td data-bbox="890 1574 1362 1693"></td> </tr> <tr> <td data-bbox="485 1698 887 1817"><b>Automatic source determination</b></td> <td data-bbox="890 1698 1362 1817"></td> </tr> <tr> <td data-bbox="485 1821 887 1725"><b>Skip faulty items</b></td> <td data-bbox="890 1821 1362 1725"></td> </tr> </tbody> </table>	Parameter	Description	<b>Hold Purchase Order if faulty</b>		<b>Hold Purchase Order if not faulty</b>		<b>No message determination</b>		<b>No messages necessary</b>		<b>No authorization check</b>		<b>No price transfer from last document</b>		<b>Set update task local</b>		<b>Call BAPI PO</b>		<b>Wait on commit</b>		<b>Call BAPI RQ</b>		<b>Automatic source determination</b>		<b>Skip faulty items</b>	
Parameter	Description																										
<b>Hold Purchase Order if faulty</b>																											
<b>Hold Purchase Order if not faulty</b>																											
<b>No message determination</b>																											
<b>No messages necessary</b>																											
<b>No authorization check</b>																											
<b>No price transfer from last document</b>																											
<b>Set update task local</b>																											
<b>Call BAPI PO</b>																											
<b>Wait on commit</b>																											
<b>Call BAPI RQ</b>																											
<b>Automatic source determination</b>																											
<b>Skip faulty items</b>																											
<p><b>Requisition specific posting parameters</b> (Checks, Posting)</p>	<p>Determines whether a purchase order or a requisition will be created in SAP when a PROCESS DIRECTOR requisition document is posted.</p>																										

Process	Description
<b>Deactivation of popup to select contract</b>  (Other)	In the parameters, activate <b>Deactivation flag</b> to disable the popup that is displayed by default when a user creates a new requisition. This popup enables the user to create a new requisition based on an existing purchasing document or outline agreement. See also <i>Activating outline agreements</i> in the <i>PROCESS DIRECTOR SAP Configuration Guide</i> .

## Determinations for Requisitions

These determinations are available for Requisitions in the PROCESS DIRECTOR standard.

### Purchase Order delivery date determination

/EBY/CL\_PDPO\_DET\_SCHED\_INIT

Determines the delivery date based on a specific number of working days after the document date of the requisition. This calculated delivery date is entered for all line items of the requisition.

The number of working days can be entered in the parameters of the **Purchase order delivery date determination by working days** search (/EBY/CL\_PDPO\_DET\_SCHED\_WDAYS). If you also enter a factory calendar ID, the delivery date calculation takes this calendar into account.

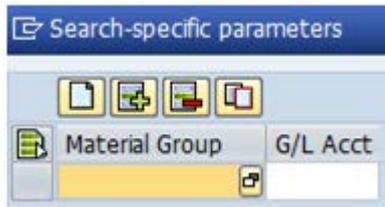
Search-specific parameters		
Days diff.	Factory Cal.ID	+/- sign
0		

### Purchase Order G/L account determination

/EBY/CL\_PDPO\_DET\_GLACCT\_INIT

Determines the G/L account based on the material group entered in the requisition line item.

Material groups and their corresponding G/L account numbers can be specified in the search parameters of the **Purchase order G/L account determination by mapping matgrp** search (/EBY/CL\_PDPO\_DET\_GLACCT\_MATGRP).

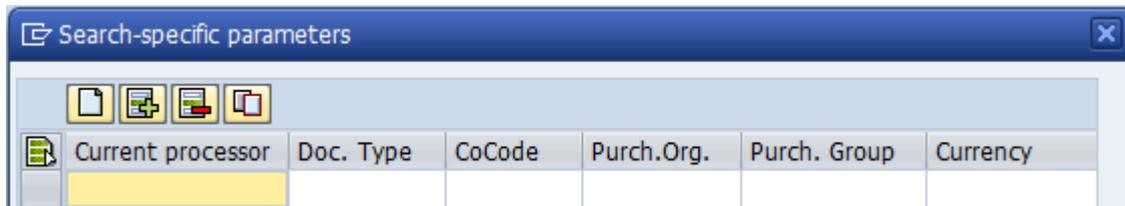


## Purchase Order org. data determination

/EBY/CL\_PDPO\_DET\_ORG\_INIT

Determines the organization data based on the current processor of the document.

Processors and their corresponding organization data can be entered in the parameters of the **Purchase order org. data determination by mapping table** search.



## Purchase Order vendor determination from OCI

/EBY/CL\_PDPO\_DET\_VENDOR\_INIT

Determines the vendor from the OCI catalog based on (depending on the defined searches):

Search type	Search name
The account number	/EBY/CL_PDPO_DET_VENDOR_ACC
The account number from the purchase order	/EBY/CL_PDPO_DET_VENDOR_ACCPO
A mapping table	/EBY/CL_PDPO_DET_VENDOR_MAP

## Document model customization

The document model provides an interface structure for each logical level containing all fields that are available for that level. Using the customer includes (CIs) provided in this structure, you can extend the document model to include customer-specific fields. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

- Use the include ending in `_DATA` if the custom field should be automatically persisted in the database.

- Use the include ending in `_DISP` if you only want to automatically calculate a field at runtime and display it.
- 🔴 **Important:** Never add customer-specific fields directly to the customer include, but use your own customer-specific structure instead. For example, first create a customer-specific structure corresponding to each used CI, include that structure in the CI and put all customer-specific fields in the customer-specific structure. **Do not use an APPEND structure.** Set the enhancement category of the customer-specific structure to **Can Be Enhanced, Character-Type or Numeric-Type** (menu **Extras**). Never use the enhancement category **Can Be Enhanced (Deep)**. See the SAP documentation for more information.

Logical level	Package	Interface structure	Customer Includes
Header data	/EBY/PDPO_MODEL	/EBY/PDPO_SHDR_IF	CI_EBY_PDPO_SHDR_DATA CI_EBY_PDPO_SHDR_DISP
Item data	/EBY/PDPO_MODEL	/EBY/PDPO_SITM_IF	CI_EBY_PDPO_SITM_DATA CI_EBY_PDPO_SITM_DISP
Account data	/EBY/PDPO_MODEL	/EBY/PDPO_SACC_IF	CI_EBY_PDPO_SACC_DATA CI_EBY_PDPO_SACC_DISP
Item conditions data	/EBY/PDPO_MODEL	/EBY/PDPO_SICO_IF	CI_EBY_PDPO_SICO_DATA CI_EBY_PDPO_SICO_DISP
Partner data	/EBY/PDPO_MODEL	/EBY/PDPO_SPAR_IF	CI_EBY_PDPO_SPAR_DATA CI_EBY_PDPO_SPAR_DISP
Schedule data	/EBY/PDPO_MODEL	/EBY/PDPO_SSCH_IF	CI_EBY_PDPO_SSCH_DATA CI_EBY_PDPO_SSCH_DISP
Header texts	/EBY/PDPO_MODEL	/EBY/PDPO_SHTX_IF	CI_EBY_PDPO_SHTX_DATA CI_EBY_PDPO_SHTX_DISP
Item texts	/EBY/PDPO_MODEL	/EBY/PDPO_SITX_IF	CI_EBY_PDPO_SITX_DATA CI_EBY_PDPO_SITX_DISP

## BAPI structure fields

Some document data models include BAPI structures. These structures can contain a large number of fields, many of which are not used. Therefore, most document data model BAPI structure fields are transient by default.

However, a customer may require that one or more of these fields be made persistent. PROCESS DIRECTOR offers an easy way to fulfill this requirement. In SAP transaction SE80, simply add the BAPI structure field(s) to the appropriate customer include.

Logical level	Package	Structure	Customer Include
Header data	/EBY/PDPO_PERSISTENCE	/EBY/PDPO_THDR	CI_EBY_PDPO_SHDR_BAPI
Item data	/EBY/PDPO_PERSISTENCE	/EBY/PDPO_TITM	CI_EBY_PDPO_SITM_BAPI
Account data	/EBY/PDPO_PERSISTENCE	/EBY/PDPO_TACC	CI_EBY_PDPO_SACC_BAPI
Item conditions data	/EBY/PDPO_PERSISTENCE	/EBY/PDPO_TICO	CI_EBY_PDPO_SICO_BAPI
Partner data	/EBY/PDPO_PERSISTENCE	/EBY/PDPO_TPAR	CI_EBY_PDPO_SPAR_BAPI
Schedule data	/EBY/PDPO_PERSISTENCE	/EBY/PDPO_TSCH	CI_EBY_PDPO_SSCH_BAPI
Header texts	/EBY/PDPO_PERSISTENCE	/EBY/PDPO_THTX	CI_EBY_PDPO_SHTX_BAPI
Item texts	/EBY/PDPO_PERSISTENCE	/EBY/PDPO_TITX	CI_EBY_PDPO_SITX_BAPI

## View model customization

This section provides detailed information about which settings to enter when creating a customer view model for Requisitions. See the *PROCESS DIRECTOR SAP Configuration Guide* for more information.

## By GUI element

- SAP document overview list:

The screenshot displays the SAP GUI interface for a Purchase Requisition. The main window is titled "Display Purchase Requisition #446 (7 of 195)". On the left, there is a "PROCESS DIRECTOR by ReadSoft" sidebar with a tree view of document types. The main area shows a table of requisition items. Several callout boxes with red borders and text describe specific grid components:

- Partners grid:** Component type: Additional grid 01, Grid No.: 5, Logical level: PARTNERS. This callout points to a small table within the main requisition table.
- Header data grid:** Component type: Blank, Grid No.: 1, Logical level: HEADER. This callout points to the top row of the main requisition table.
- Items grid:** Component type: Blank, Grid No.: 2, Logical level: ITEMS. This callout points to the main table of requisition items.
- Accounts grid:** Component type: Additional grid 01, Grid No.: 1, Logical level: ITEMS\_ACCOUNTS. This callout points to a sub-window titled "Accounts".
- Schedules grid:** Component type: Additional grid 01, Grid No.: 3, Logical level: ITEMS\_SCHEDULES. This callout points to a sub-window titled "Item Schedules".
- Conditions grid:** Component type: PO, Grid No.: 4, Logical level: ITEMS\_CONDITIONS. This callout points to a sub-window titled "Item Conditions".

- SAP document detail:

To maintain the SAP GUI detail screen, use SAP transaction /EBY/PDVI\_VDSTC.

- Web Application document overview list:

Status	Wf status	PD doc no	Purch.Doc.	Purch.req.	Vendor	Doc. date
		160			1060	12/07/10
	<input checked="" type="checkbox"/>				1060	12/05/10
						11/26/10
		156			1060	11/26/10
						11/24/10
	<input checked="" type="checkbox"/>	155	4500016161		1060	11/24/10

**Header data grid:**  
Component type: WVA List Header, Grid No.: 1,  
Logical level: HEADER

- Web Application document detail:

**Header data grid:**  
Component type: WA Document Detail, Grid No.: 1,  
Logical level: HEADER

**Items grid:** Component type: WA Document Detail,  
Grid No.: 2, Logical level: ITEMS

**Accounts grid:** Component type: WA Document Detail,  
Grid No.: 3, Logical level: ITEMS\_ACCOUNTS

- Workflow status:

**Workflow selection**

Workflow	Status description	Started by	Created on	Time created	Workflow due date
Approve and create PO (2200)	Running	Blake Evans	08/22/2011	17:11:21	08/30/2011

**Workflow status header grid:**  
Component type: Workflow status, Grid No.: 1,  
Logical level: HEADER

**Next**

Status	Step	Date	Time	Processor	Overdue
	Create PO			Karina Kunz	

**Current**

Status	Step	Date	Time	Processor	Overdue
	Approve PO	08/22/2011	17:11:42	Roger Tillman	08/30/2011

Continue

### By view model component type

GUI area	Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming suggestions)	Btn. label Grid title (naming suggestions)
SAP document list		1	HEADER	ZXY_PDPO_SGRID1 ZXY_PDPO_LGRID1	Header Header data
SAP Line items grid		2	ITEMS	ZXY_PDPO_SGRID2 ZXY_PDPO_LGRID2	Items Items
SAP Accounts grid	Additional grid 01	1	ITEM_ACCOUNTS	ZXY_PDPO_SGRID3 ZXY_PDPO_LGRID3	Accounts Accounts

GUI area	Comp. type	Grid No.	Logical level	Cust. display struct. Cust. table type name (naming suggestions)	Btn. label Grid title (naming suggestions)
SAP Schedules grid	Additional grid 01	3	ITEM_SCHEDULES	ZXY_PDPO_SGRID4 ZXY_PDPO_LGRID4	Schedules Item Schedules
SAP Conditions grid	Additional grid 01	4	ITEM_CONDITIONS	ZXY_PDPO_SGRID5 ZXY_PDPO_LGRID5	Conditions Item Conditions
SAP Partners grid	Additional grid 01	5	PARTNERS	ZXY_PDPO_SGRID7 ZXY_PDPO_LGRID7	Partners Partners
Web Application document detail header data	WA Document Detail	1	HEADER	ZXY_PDPO_SWA_DOC_GRID1 ZXY_PDPO_LWA_DOC_GRID1	Header Header data
Web Application document detail Line items grid	WA Document Detail	2	ITEMS	ZXY_PDPO_SWA_DOC_GRID2 ZXY_PDPO_LWA_DOC_GRID2	Items Items
Web Application Accounts grid	WA Document Detail	3	ITEM_ACCOUNTS	ZXY_PDPO_SWA_DOC_GRID3 ZXY_PDPO_LWA_DOC_GRID3	Accounts Accounts
Web Application document list	WA List Header	1	HEADER	ZXY_PDPO_SWA_LIST_GRID1 ZXY_PDPO_LWA_LIST_GRID1	Header Header data
Workflow status popup	Workflow status	1	HEADER	ZXY_PDPO_SWA_STAT_GRID1 XY_PDPO_LWC_STAT_GRID1	Header Header data
Archive log	Header data for archive log	1	HEADER	ZXY_PDPO_SARC_LOG_GRID1 ZXY_PDPO_LARC_LOG_GRID1	Header Header data

## Document statuses

PROCESS DIRECTOR ships with these system document statuses/substatuses for Requisitions:

### Statuses

Status	Status Description
ER	Not postable
NP	Postable
NW	New
OK	Posted
WR	Postable with warning(s)

### Substatuses

Substatus	Description
PM	Purchase Order Multiple
PO	Purchase Order
RM	Purchase Requisition Multiple
RQ	Purchase Requisition

## Text types

PROCESS DIRECTOR ships with these system text types for Requisitions:

Logical level	Text object	Description	ID	Description	Multiple
HEADER	/EBY/PD	PROCESS DIRECTOR texts	PDN	PROCESS DIRECTOR note	X
HEADER	EKKO	Purchasing doc. header texts	F01	Header text	
ITEMS	EBAN	Purchase requisition texts	B04	Material PO text	
ITEMS	EKPO	Purchasing doc. item texts	F03	Material PO text	

## User exits / BAdIs

PROCESS DIRECTOR provides the following user exits/BAdIs for use with Requisitions:

[Initialize values](#)

[Posting](#)

[Check](#)

### BAdI Initialize values (Requisitions)

This user exit / BAdI currently provides the interface method [INITIALIZE\\_VALUES](#) that you can implement to add your own customer rules for initializing field values for Requisitions. You can use it as an alternative or supplement to presets.

#### INITIALIZE\_VALUES method

The INITIALIZE\_VALUES method offers access to the Requisition document model logical level structures as CHANGING parameters. To initialize one or more fields, simply assign values to them in your coding, like this:

```
METHOD /eby/if_pdpo_badi_init_val~initialize_values .
```

```
cs_po_hdr-currency = 'GBP' .
ENDMETHOD.
```

## Calling sequence

In the PROCESS DIRECTOR standard, this user exit is called from the final event of the CREAT1 Create document action. It runs *after* presets are applied, but *before* the document is actually created on the database.

**Important:** Document creation will be prevented if an INITIALIZE\_VALUES method implementation issues an error message, unless the error message is filtered with a Remove message filter.

## Method signature

Since the user exit / BAdI is document-type-specific, the INITIALIZE\_VALUES method signature depends on the document type.

Type	Parameter	Description
▶□	IR_OBJECT	Requisitions object
▶▶	CS_BO_HDR	Data fields of a PD base object
▶▶	CS_PO_HDR	Structure containing requisition header data
▶▶	CT_ITEMS	Table containing requisition line item data
▶▶	CT_ITEM_ACCOUNTS	Table containing requisition accounts data
▶▶	CT_ITEM_SCHEDULES	Table containing requisition schedules data
▶▶	CT_ITEM_CONDITIONS	Table containing requisition conditions data
▶▶	CT_PARTNERS	Table containing partners

## BAdI Posting (Requisitions)

This user exit / BAdI currently provides the interface methods [BEFORE\\_POST](#), [BEFORE\\_POST\\_RQ](#), [AFTER\\_POST](#) and [AFTER\\_POST\\_RQ](#) that you can implement to add your own customer logic before or after a Requisition document is posted.

Both interface methods offer access to SAP PO BAPI structures as parameters:

- The `BEFORE_POST` interface method exposes the PO BAPI structures as `CHANGING` parameters pre-populated with the PO data to be posted to SAP.
- THE `AFTER_POST` interface method exposes the PO BAPI structures as `IMPORTING` parameters containing the PO data that has just been posted to SAP.

### BEFORE\_POST method

#### Calling sequence

This method is called from the action `POST_DOC` Post document. It runs immediately *before* document posting occurs.

#### Method signature

Type	Parameter	Description
▶	IR_OBJECT	Requisitions object
▶	IS_BO_HDR	Data fields of a PD base object
▶	IS_PO_HDR	Requisition header data
▶	IT_ITEMS	Requisition line item data
▶	IT_ITEM_ACCOUNTS	Requisition accounts data
▶	IT_ITEM_SCHEDULES	Requisition schedules data
▶	IT_ITEM_CONDITIONS	Requisition conditions data
▶	IT_PARTNERS	Requisition partner data

Type	Parameter	Description
▶▶	CS_HEADER	Requisition header data
▶▶	CS_HEADERX	Requisition header data (change toolbar)
▶▶	CS_ADDRVENDOR	Requisition header: address structure BAPIADDR1 for vendor
▶▶	CS_EXPHEADER	Requisition header data
▶▶	CT_ITEMS	Table type for BAPIMEPOITEM
▶▶	CT_ITEMSX	Table type for BAPIMEPOITEMX
▶▶	CT_ACCOUNTS	Table type for BAPIMEPOACCOUNT
▶▶	CT_ACCOUNTSX	Table type for BAPIMEPOACCOUNTX
▶▶	CT_ACCPROFITSEGMENTS	Table type for BAPIMEPOACCOUNTPROFITSEGMENT
▶▶	CT_SCHEDULES	Table Type for BAPIMEPOSCHEDULE
▶▶	CT_SCHEDULESX	Table Type for BAPIMEPOSCHEDULX
▶▶	CT_CONDHEADER	Table Type for BAPIMEPOCONDHEADER
▶▶	CT_CONDHEADERX	Table Type for BAPIMEPOCONDHEADERX
▶▶	CT_CONDITIONS	Table Type for BAPIMEPOCOND
▶▶	CT_CONDITIONSX	Table Type for BAPIMEPOCONDX
▶▶	CT_LIMITS	Table Type for BAPIESUHC
▶▶	CT_CONTRACTLIMITS	Table Type for BAPIESUCC

Type	Parameter	Description
▶▶	CT_SERVICES	Table Type for BAPIESLLC
▶▶	CT_SRVACCESSVALUES	Table Type for BAPIESKLC
▶▶	CT_PARTNERS	Table Type for BAPIEKKOP
▶▶	CT_ADDRDELIVERY	Table Type for BAPIMEPOADDRDELIVERY
▶▶	CT_HEADERTEXTS	Table Type for BAPIMEPOTEXTHEADER
▶▶	CT_ITEMTEXTS	Table Type for BAPIMEPOTEXT
▶▶	CT_SERVICESTEXT	Table Type for BAPIESLLTX
▶▶	CT_EXTENSIONIN	Table with BAPIPAREX as Row Type
▶▶	CT_EXTENSIONOUT	Table with BAPIPAREX as Row Type
▶▶	CT_RETURN	Return parameter table
▶▶	CT_ITEM_HIST_TOTALS	Table of structure BAPIEKBES
▶▶	CT_SELECTED_ITEMS	Select options for GUIDs
▶▶	CT_ITEM_MAPPING	Item numbers mapping

## BEFORE\_POST\_RQ method

### Method signature

Type	Parameter	Description
▶	IR_OBJECT	Requisitions object

Type	Parameter	Description
▶□	IS_BO_HDR	Data fields of a PD base object
▶□	IS_PO_HDR	Requisition header data
▶□	IT_ITEMS	Requisition line item data
▶□	IT_ITEM_ACCOUNTS	Requisition accounts data
▶□	IT_ITEM_SCHEDULES	Requisition schedules data
▶□	IT_ITEM_CONDITIONS	Requisition conditions data
▶□	IT_PARTNERS	Requisition partner data
▶▶	CC_VENDOR	Account number of the vendor
▶▶	CT_SERVICES	Table Type for BAPIESLLC
▶▶	CT_SERVICES_TEXTS	Table Type for BAPIESLLTX
▶▶	CT_SRV_ACCASS_VALUES	Table Type for BAPIESKLC
▶▶	CT_ACCOUNTS	Table Type for BAPIEBKN
▶▶	CT_ACCOUNTS_OLD	Table Type for BAPIEBKN
▶▶	CT_ADDRDELIVERY	Type BAPIMERQADDRDELIVERY
▶▶	CT_CONTRACT_LIMITS	Table Type for BAPIESUCC
▶▶	CT_EXTENSIONIN	Table Type for BAPIPAREX
▶▶	CT_ITEMS	Table Type for BAPIEBAN

Type	Parameter	Description
▶▶	CT_ITEMS_OLD	Table Type for BAPIEBANC
▶▶	CT_ITEM_MAPPING	Item numbers mapping
▶▶	CT_ITEM_TEXTS	Table Type for BAPIEBANTX
▶▶	CT_ITEM_TEXTS_OLD	Table Type for BAPIEBANTX
▶▶	CT_LIMITS	Table Type for BAPIESUHC
▶▶	CT_RETURN	Table of structure BAPIRETURN
▶▶	CT_SELECTED_ITEMS	Select options for GUIDs

## AFTER\_POST method

### Calling sequence

This method is called from the action `POST_SUCCESS` Posting succeeded. It runs after the document status `Posted` has been set.

**Note:** If document posting fails, `AFTER_POST` is *not* called.

### Method signature

Type	Parameter	Description
▶	IR_OBJECT	Requisitions object
▶	IS_HEADER	Requisition header data
▶	IS_HEADERX	Requisition header data (change toolbar)
▶	IS_ADDRVENDOR	PO Header: Address Structure BAPIADDR I for Vendor
▶	IS_EXPHEADER	Purchase Order Header Data

Type	Parameter	Description
▶	IT_ITEMS	Table Type for BAPIMEPOITEM
▶	IT_ITEMSX	Table Type for BAPIMEPOITEMX
▶	IT_ACCOUNTS	Table Type for BAPIMEPOACCOUNT
▶	IT_ACCOUNTSX	Table Type for BAPIMEPOACCOUNTX
▶	IT_ACCPROFITSEGMENTS	Table Type for BAPIMEPOACCOUNTPROFITSEGMENT
▶	IT_SCHEDULES	Table Type for BAPIMEPOSCHEDULE
▶	IT_SCHEDULESX	Table Type for BAPIMEPOSCHEDULX
▶	IT_CONDHEADER	Table Type for BAPIMEPOCONDHEADER
▶	IT_CONDHEADERX	Table Type for BAPIMEPOCONDHEADERX
▶	IT_CONDITIONS	Table Type for BAPIMEPOCOND
▶	IT_CONDITONSX	Table Type for BAPIMEPOCONDX
▶	IT_LIMITS	Table Type for BAPIESUHC
▶	IT_CONTRACTLIMITS	Table Type for BAPIESUCC
▶	IT_SERVICES	Table Type for BAPIESLLC
▶	IT_SRVACCESSVALUES	Table Type for BAPIESKLC
▶	IT_PARTNERS	Table Type for BAPIEKKOP
▶	IT_ADDRDELIVERY	Table Type for BAPIMEPOADDRDELIVERY

Type	Parameter	Description
▶□	IT_HEADERTEXTS	Table Type for BAPIMEPOTEXTHEADER
▶□	IT_ITEMTEXTS	Table Type for BAPIMEPOTEXT
▶□	IT_SERVICESTEXT	Table Type for BAPIESLLTX
▶□	IT_EXTENSIONIN	Table with BAPIPAREX as Row Type
▶□	IT_EXTENSIONOUT	Table with BAPIPAREX as Row Type
▶□	IT_RETURN	Return parameter table
▶□	IT_ITEM_HIST_TOTALS	Table of structure BAPIEKBES
▶□	IT_SELECTED_ITEMS	Select options for GUIDs
▶□	IT_ITEM_MAPPING	Item numbers mapping
▶▶	CS_BO_HDR	Data fields of a PD base object
▶▶	CS_PO_HDR	Requisition header data
▶▶	CT_ITEMS	Requisition line item data
▶▶	CT_ITEM_ACCOUNTS	Requisition accounts data
▶▶	CT_ITEM_SCHEDULES	Requisition schedules data
▶▶	CT_ITEM_CONDITIONS	Requisition conditions data
▶▶	CT_PARTNERS	Requisition partner data

## AFTER\_POST\_RQ method

### Method signature

Type	Parameter	Description
▶	IR_OBJECT	Requisitions object
▶	IC_VENDOR	Account number of the vendor
▶	IT_SERVICES	Table Type for BAPIESLLC
▶	IT_SERVICES_TEXTS	Table Type for BAPIESLLTX
▶	IT_SRV_ACCASS_VALUES	Table Type for BAPIESKLC
▶	IT_ACCOUNTS	Table Type for BAPIEBKN
▶	IT_ACCOUNTS_OLD	Table Type for BAPIEBKN
▶	IT_ADDRDELIVERY	Type BAPIMERQADDRDELIVERY
▶	IT_CONTRACT_LIMITS	Table Type for BAPIESUCC
▶	IT_EXTENSIONIN	Table Type for BAPIPAREX
▶	IT_ITEMS	Table Type for BAPIEBAN
▶	IT_ITEMS_OLD	Table Type for BAPIEBANC
▶	IT_ITEM_MAPPING	Item numbers mapping
▶	IT_ITEM_TEXTS	Table Type for BAPIEBANTX
▶	IT_ITEM_TEXTS_OLD	Table Type for BAPIEBANTX
▶	IT_LIMITS	Table Type for BAPIESUHC

Type	Parameter	Description
▶□	IT_RETURN	Table of structure BAPIRETURN
▶□	IT_SELECTED_ITEMS	Select options for GUIDs
▶▶	CS_BO_HDR	Data fields of a PD base object
▶▶	CS_PO_HDR	Requisition header data
▶▶	CT_ITEMS	Requisition line item data
▶▶	CT_ITEM_ACCOUNTS	Requisition accounts data
▶▶	CT_ITEM_SCHEDULES	Requisition schedules data
▶▶	CT_ITEM_CONDITIONS	Requisition conditions data
▶▶	CT_PARTNERS	Requisition partner data

## BAdI Check (Requisitions)

This user exit / BAdI currently provides the interface method [CHECK](#) that you can implement to add your own customer check routine for Requisitions. You can use it as an alternative or supplement to checks.

### CHECK

The [CHECK](#) method offers access to the Requisitions document model logical level structures as `IMPORTING` parameters.

Here is a sample implementation of the [CHECK](#) interface method that shows how to generate a message from a BAPI call. Note that you must export the proper message group as a parameter to the `ADD_BAPIMSG` method call:

```
METHOD /EBY/IF_PDPO_BADI_CHECK~CHECK.
```

```

DATA: ls_bapireturn TYPE bapireturn,
        lr_bapi TYPE REF TO /eby/cl_bapi.

CREATE OBJECT lr_bapi.
ls_bapireturn = lr_bapi->check_purch_org_exists(
is_po_hdr-purch_org ).
IF NOT ls_bapireturn IS INITIAL.
    CALL METHOD /eby/cl_pdbo_exc=>add_bapimsg
        EXPORTING
            ic_grp = /eby/if_pdbo_con_exc=>grp_check
            is_bapi = ls_bapireturn.
ENDIF.
ENDMETHOD.

```

### Method signature

Type	Parameter	Description
▶	IR_OBJECT	Requisitions object
▶	IS_BO_HDR	Data fields of a PD base object
▶	IS_PO_HDR	Requisition header data
▶	IT_ITEMS	Requisition line item data
▶	IT_ITEM_ACCOUNTS	Requisition accounts data
▶	IT_ITEM_SCHEDULES	Requisition schedules data
▶	IT_ITEM_CONDITIONS	Requisition conditions data
▶	IT_PARTNERS	Requisition partner data

## Calling sequence

The CHECK BAdI method is called from the actions CHECK Check document and CHECKM Check documents (multiple).

It is run *after* activated determination events, activated user exit / BAdI **Determination** implementations, and activated check events are executed, but immediately *before* the document status is set depending on the check result.

Like standard PROCESS DIRECTOR check events, a user exit / BAdI **Check** implementation communicates its results by issuing messages.

# Index

## A

accounts payable	
accrual report.....	89
document model customization .....	82
document statuses .....	87
overview .....	81
programs.....	88
registering documents.....	88
single document link.....	35
text types.....	88
view model customization .....	83
accrual report.....	89
actions	
BAdl Set Web Application actions .....	57
archiving	
change archive links .....	47
process parameters .....	21
asset acquisition	
checks.....	89
document model customization .....	90
document statuses .....	93
text types.....	94
view model customization .....	90
asset master	
checks.....	177
document statuses .....	197
presets.....	175
view model customization .....	182
asset retirement	
checks.....	94
document model customization .....	94
document statuses .....	98
presets.....	94
text types.....	99
view model customization .....	95
asset transfer	
checks.....	99
document model customization .....	100
document statuses .....	104

presets .....	99
text types .....	105
view model customization .....	100

attachments	
check for required attachments.....	18

## B

BAdlS	
available BAdlS .....	48
Check (Requisitions) .....	257
general BAdlS (platform)	
Determination results.....	55
Modify field profiles .....	53
Modify Worklist .....	51
Rejection reason.....	55
Screen PBO/PAI .....	49
Set actions (Web Application).....	57
Set messages (Web Application).....	56
Set Worklist node.....	54
User logon .....	58
Workflow email handling.....	60
Workflow handling.....	62
Workflow steps handling .....	65
Initialize values (PO).....	247
Modify Worklist .....	51
posting	
customer orders .....	127
financial postings.....	145
general ledger .....	199
goods receipts.....	166
order confirmations .....	212
payment advices.....	224
requisitions .....	249
vendor master.....	200
process-type specific BAdlS	
Check (PO).....	257
Initialize values (PO).....	247
Posting (DN) .....	166
Posting (FI) .....	145
Posting (MDGL).....	199

Posting (MDVM) .....	200	process parameters .....	23
Posting (OR) .....	212	split configuration screen .....	38
Posting (PA) .....	224	cost centers	
Posting (PO) .....	249	checks .....	177
Screen PBO/PAI .....	49	document model customization .....	181
Set Web Application actions .....	57	document statuses .....	197
Set Web Application messages .....	56	presets .....	175
Set Worklist node .....	54	view model customization .....	185
Workflow email handling .....	60	customer orders	
Workflow handling .....	62	checks .....	105
Workflow steps handling .....	65	determinations .....	112
<b>BAPI</b>		document model customization .....	120
structure fields		document statuses .....	126
customer orders .....	120	overview .....	105
financial postings .....	140	text types .....	127
goods receipts .....	160	view model customization .....	122
purchase orders .....	239		
		<b>D</b>	
<b>C</b>		determinations	
catalog		available determinations .....	20
check vendors from OCI catalog .....	235	BAAd determination results .....	55
determine vendors from OCI catalog .....	238	split configuration screen .....	38
checking		<b>E</b>	
against check tables .....	18	electronic bank statements	
automatic via REPETITOR .....	42	determinations .....	132
customer orders .....	105	document model customization .....	133
financial postings .....	138	document statuses .....	136
for attachments .....	18	presets .....	132
goods receipts .....	156	text types .....	137
mandatory fields .....	18	view model customization .....	134
master data maintenance requests .....	177	emails	
order confirmations .....	202	standard email texts .....	31
Payment Advices .....	214		
payment approvals .....	226	<b>F</b>	
requisitions .....	235	field statuses	
checks		BAAd Modify field profiles .....	53
available checks .....	18	fields	
BAAd Check (PO) .....	257	populating .....	28
for asset acquisitions .....	89	financial postings	
for asset retirement .....	94	checks .....	138
for asset transfer .....	99	document model customization .....	140
for customer orders .....	105	document statuses .....	144
for financial postings .....	138	overview .....	137
for goods receipts .....	156	presets .....	137
for invoice block/cancelation .....	169	process parameters .....	139
for master data maintenance requests .....	177	programs .....	150
for order confirmations .....	202		
for Payment Advices .....	214		
for payment approvals .....	226		
for requisitions .....	235		

text types.....	145
view model customization .....	141

## G

G/L accounts	
BAdI Posting .....	199
checks.....	177
document statuses .....	197
presets.....	175
view model customization .....	188
generic archiving	
document model customization .....	152
document statuses .....	155
text types.....	156
view model customization .....	152
goods receipts	
checks.....	156
determinations .....	157
document model customization .....	160
document statuses .....	165
overview .....	156
presets.....	156
process parameters .....	160
text types.....	165
view model customization .....	162

## H

HTML emails	
standard texts for.....	31

## I

invoice block/cancelation	
checks.....	169
document statuses.....	174
document model customization .....	170
text types.....	175
view model customization .....	171

## M

mapping	
available mapping conversions .....	7
master data	
available types of master data .....	175
document model customization .....	181
text tyües.....	198

## O

order confirmations	
checks .....	202
determinations.....	204
document model customization.....	207
document statuses .....	211
overview .....	202
text types .....	212
view model customization .....	207

## P

Payment Advices	
checks .....	214
determinations.....	216
document model customization.....	220
document statuses .....	224
link to clearing posting.....	226
overview .....	213
presets .....	213
programs .....	226
view model customization .....	221
payment approvals	
checks .....	226
document model customization.....	228
document statuses.....	232
overview .....	226
process parameters .....	227
programs .....	232
text types .....	232
view model customization .....	229
posting	
automatically via REPETITOR.....	42
process parameters .....	24
presets	
available preset IDs.....	16
BAdI Initialize values (Requisitions).....	247
for goods receipts.....	156
for master data maintenance requests.....	175
for Payment Advices .....	213
for requisitions.....	234
preset classes .....	17
process parameters	
configuring.....	21
for archiving.....	21
for checks.....	23
for financial postings.....	139
for goods receipts.....	160
for other processes .....	28
for payment approvals .....	227

for posting.....	24
for requisitions.....	236
profit centers	
view model customization .....	191
programs	
/EBY/ICIV_ACCRUAL_REPORT.....	89
/EBY/ICIV_DOC_SYNCHRONIZE.....	88
/EBY/PDBO_CHANGE_ARC_LINKS.....	47
/EBY/PDBO_CHECK_NESTEDSET_NEW .....	46
/EBY/PDBO_CHECK_WL_DOC_COUNT... 47	
/EBY/PDBO_CONFIG_DUMP.....	43
/EBY/PDBO_CONFIG_GEN_WORKLIST .....	46
/EBY/PDBO_CONFIG_WRKL_ENV_CONV 47	
/EBY/PDBO_GEN_FAILED_NSTDSETS .....	47
/EBY/PDBO_REPETITOR .....	42
/EBY/PDFI_GEN_RECURRING_DOCS.....	150
/EBY/PDFI_PROVISION_REMINDER .....	151
/EBY/PDPA_FINALIZE_DOC .....	226
/EBY/PDPR_IMPORT .....	233
/EBY/PDPR_INVOICE_UPDATER.....	233
/EBY/PDWC_DUE_DATE_CHECK .....	40

## Q

query classes	
available query classes .....	5

## R

REPETITOR.....	42
requisitions	
checks.....	235
determinations .....	238
document model customization .....	239
document statuses .....	246
overview .....	234
presets.....	234
process parameters .....	236

text types .....	247
user exits/BADs .....	247
Check .....	257
Initialize values .....	247
Posting.....	127
view model customization .....	241

## U

user profile parameters	
/EBY/PDBO/SPLITV .....	38

## V

vendor master	
BAdI Posting .....	200
checks .....	177
document model customization.....	181
document statuses .....	197
presets .....	175
view model customization .....	194

## W

Web Application	
BAdI Set actions .....	57
BAdI Set messages .....	56
BAdI User logon.....	58
workflow steps	
email texts.....	35
workflows	
control patterns .....	30
step types.....	30
Worklist	
programs .....	39