PP Production Planning and Control

SAP R/3 Enterprise

Release 4.70

Release Notes

Design: SAP Communications Media

19 PP Production Planning and Control

19.1 PP-MRP Material Requirements Planning

19.1.1 Business Add-Ins in MRP

Use

As of *SAP R/3 Enterprise 4.70 (SAP_APPL 470)* Business Add-Ins (BAdIs) are available for the following functions in material requirements planning (MRP):

MRP General

- Changing subcontracting logic: MD_SUBCONT_LOGIC
  Using this BAdI, you can deactivate the separate consideration of provision stocks and requirements in MRP and the stock/requirements list.
- Changing logic for importing MRP data: MD_CHANGE_MRP_DATA
  Using this BAdI, you can exclude from, add to, or change the characteristics for certain elements for MRP when the MRP elements are read from the database.
- Changing the requirements dates of stock transfers: MD_STOCK_TRANSFER
  Using this BAdI, you can change the requirements dates for requirements for procurement proposals in the issuing plant.

MRP Planning Valuation (PP-MRP-BD)

- Processing the MRP list further: MD_MRP_LIST
  Using this BAdI, you can process the MRP list data further during posting in the planning run.
- Changing the display of MRP elements: MD_DISPLAY_ELEMENT
  Using this BAdI, you can change how the system displays MRP elements in the valuation list and the planning run.
- Planning and preparing planning data: MD_PLANNING_RESULT
  Using this BAdI, you can process the statistical data in the planning run further. You can, for example, trigger a separate workflow, so that you are informed of dumps or cancellation MRP list via mail.

MRP Procurement Proposal (PP-MRP-PP)

- Changing the procurement proposal: MD_PURREQ_CHANGE
  Using this BAdI, you can change the data in purchase requisition before posting in the planning run or before posting during planned order conversion.
- Processing purchase requisitions further: MD_PURREQ_POST
  Using this BAdI, you can further process data from planned orders that are posted in the planning run or during planned order conversion.
- Changing planned orders: MD_PLDORD_CHANGE
  Using this BAdI, you can change the data from planned orders before posting in the planning run, or before posting during manual planned order changes.

- Processing planned orders further: MD_PLDORD_POST
  Using this BAdI, you can further process the data from planned orders, which are posted in the planning run or posted during manual planned order processing. You can, for example, log any changes made.
- Changing source assignment: MD_MODIFY_SOURCE
  Using this BAdI, you can change the assignment of sources in the planning run according to your own criteria.
- Determining production order: MD_MODIFY_PRODVERS
  Using this BAdI, you can determine production versions according to your own criteria, both during the planning run and during manual creation of production versions.
- Logging planned order postings due to time stamp problems: MD_PLDORD_TIME_STAMP
  Using this BAdI, you can log planned order postings that were rejected by the system due to time stamp problems.
- Scheduling planned orders: MD_PLDORD_SCHEDULING
  Using this BAdI, you can change the scheduling parameters for planned orders.

MRP Planning Execution (PP-MRP-PR)

- Changing the package size in a planning run parallel to MRP: MD_MRP_RUN_PARALLEL
  Using this BAdI, you can individually determine the size of material packages used for the parallel planning run.

Long-term planning/Simulation (PP-MP-LTP)

- Defining additional parameters in the planning scenario: MD_PLANNING_SCENARIO
  Using this BAdI, you can display additional data in the planning scenario and maintain your own planning scenario parameters.

Effects on Customizing

To activate a Business Add-In, you have to create an active implementation. To do this, choose *Tools* -> *ABAP Workbench* -> *Utilities* -> *Business Add-Ins* -> Implementation in the SAP menu.

You can find further information in the SAP Library under *Basis* -> *ABAP-Workbench* -> *Changing the SAP standard* -> *Business Add-Ins* -> *Implementing a Business Add-Ins*.

19.1.2 Usability Improvements in MRP

Use

As of *SAP R/3 Enterprise Core 4.70 (SAP_APPL 470)* the following new or changed functions are available:

- **Order tree in the evaluation list (changed)**
  If you work with the order tree in the stock/requirements list or MRP list, you can now display the corresponding component requirements on multiple levels for a selected order or requirement. This function is based on the order report. If problems occur in MRP, the system issues an error message. When navigating in the order tree, the corresponding stock/requirements list is displayed in part of the screen and the cursor is automatically placed on the corresponding component requirement. This

way you get a quick overview of the stock situation at all levels within the order structure. In earlier releases, there was no discernible relationship between an order and its dependent requirements. You can find more information here

o **Navigating to pegged requirements with a double mouse click (new)**
You can now navigate directly to the pegged requirement on the next highest level for certain objects in the stock/requirements list and the MRP list. The system displays the stock/requirements list or the MRP list of the superordinate level and places the cursor automatically on the right line of the pegged requirement. This means that it is now much easier to find pegged requirements on the different MRP levels than in earlier releases.
You can select this function via the detail display of the requirements element. This function is available for the following MRP elements:

- Dependent requirements
- Order reservations
- Stock transfer release orders
- Stock transfer reservations

o **Stock transfer from plant to MRP area with stock transfer purchase requisitions (new)**
Up to now, the procurement of materials assigned to a storage location MRP area, could be achieved alongside the normal external procurement via the special procurement type *stock transfer from plant to MRP area* delivered in the standard system. In this case, the planning run of requirements planning creates stock transfer reservations from the plant to the MRP area. As of *SAP R/3 Enterprise Core 4.70,* this process can also be established via stock transfer purchase requisitions. If you execute planning of materials provided for a subcontractor using an MRP area of the type "subcontractor", you can now move the materials provided for subcontractors from plant to MRP area via stock transfer purchase requisitions.
For the new function, you can now also enter a material dependent planned delivery time in the MRP area segment. When you create stock transfer purchase orders from the plant to the MRP area (manually or via materials planning), this time is taken into account in the stock transfer time. The planned delivery time of the MRP area segment is only used if the indicator Consider Planned Delivery Time is set.
You control whether stock transfer reservations or stock transfer purchase requisitions are to be created for the process described above by defining the special procurement type *stock transfer from plant to MRP area* in the Customizing for Material Requirements Planning.

o **Printing the period totals display of the valuation list (new)**
As of the current release, the print display in the period totals display of the stock/requirements list and the MRO list is WYSIWYG (what you see is what you get). Up until now, this was only possible in the individual line display. Now, a print function for cancellation MRP lists is also available.

o **Stock/requirements list for date problems (new)**
In earlier releases, an error message was issued when a stock/requirements list was called, for which there were MRP elements with invalid dates according to the factory calendar. In this case, the list was not displayed at all and the document that caused the problem, could not be determined. Nor did the system construct the valuation lists if no forecast was executed for materials with a forecasting requirement.
From now on, you can display the stock/requirements list even if the master data has not been fully maintained. For the date problem described above, you can now localize the relevant MRP element. This function is available for the areas of purchasing, consumption, and forecasting.

- **Number ranges for manual planned order processing (changed)**
  As of the current release, you can define number ranges in Customizing for simulative planned order types depending on the planned order type (Customizing transaction OMI3). In earlier releases, you could only define order specific number ranges for operative planned orders.
- **Default parameters for long-term planning (changed)**
  Until now, it was not possible to save the parameter with fixed planned orders user-specifically in the planning transactions of long-term planning. This is possible as of SAP R/3 Enterprise.
- **Indicator *Read order report with buffers* (new)**
  As of the current release, you can set the new indicator Buffered reading of order report in the stock/requirements list and in the MRP list. This means that when the order report and the order tree are constructed, the stock/requirements situation at the respective MRP levels is buffered in the memory. Subsequent calls of the order report or the order tree are therefore much quicker than before. You should set this indicator if you use the order report or the order tree for various orders and requirements.
  The new function is not valid for make-to-order or engineer-to-order production since only part of the MRP element is needed and read per material.
  You can set this indicator in the user settings for the stocks/requirements list and the MRP list on the tab page *General Settings*.

Effects on Customizing

- **Defining number ranges for manual planned order processing**
  To do this, go to the Customizing for *Production* and choose *Material Requirements Planning* -> *Number Ranges* -> Define Number Ranges for Manual Processing.
- **Stock transfer from plant to MRP area with stock transfer purchase requisitions (new)**
  To do this, go to the Customizing for *Production* and choose *Material Requirements Planning* -> *Master Data* -> Define Special Procurement Type.
  Create the special procurement type *Stock transfer from plant to MRP area* for the plant where the MRP area is. Maintain the following parameters for the special procurement type:
  - *Procurement type: F* for external procurement
  - *Special procurement: U* for stock transfer
  - Plant that belongs to the MRP area
  - Stock transfer*: 1* for purchase requisitions from plant to MRP area.

19.2 PP-SFC Production Orders

19.2.1 Business Add-Ins for Production Orders and Process Orders

Use

In shop floor control, the following Business Add-Ins (BAdI) for production and process orders are available as of *SAP R/3 Enterprise Core 4.70 (SAP_APPL 470)*:

- BAdI WORKORDER_INFOSYSTEM to fill additional (customer-defined) fields in the order information system (new).
  In addition, you can define authorization checks for variants in and changes to the field selection.
- BAdI WORKORDER_REWORK to adjust the yield during confirmation of rework with own tools (new)
  With the help of the new BAdI *WORKORDER_REWORK*, it is possible to implement customer-specific rework logic. It is available for one interface to transfer confirmed quantities outwards, and therefore activate the rework processing. For other interfaces, information can be transferred inwards from the rework processing - like the successfully reworked yield - and is therefore considered in the quantity logic of the order and confirmation.

The following BAdIs were changed:

- BAdI WORKORDER_UPDATE was extended by the *BEFORE_UPDATE* method. This method is implemented directly before calling the posting. The order data is provided in table form as is the existing customer exit PPCO0001. Modifications to this data can no longer be made, but subsequent processes can be derived and triggered.
- BAdI WORKORDER_CONFIRM was extended by the *BEFORE_UPDATE* BAdI method. This allows, aside from the standard check for whether the selected confirmation can be cancelled, the execution of additional, customer-specific checks.

Effects on Customizing

To activate a Business Add-In, you must create an active implementation. To do this, choose from the SAP menu *Tools* -> *ABAP Workbench* -> *Utilities* -> *Business Add-Ins -> Implementation.*

For more information on the procedure, see the SAP Library under *Basis Components* -> *ABAP Workbench* -> *Changing the SAP Standard* -> *Business Add-Ins* -> *Implementing Business Add-Ins*.

19.2.2 Preconfigure Batch Split for Production Order (new)

Use

As of *SAP R/3 Enterprise Core 4.70 (SAP_APPL 470)*, the *Always batch split* indicator is included in the production scheduling profile. Previously, batch split records for components in the production order were only created if the batch determination had found more than one batch for a component. In the process order, on the other hand, split records are created in batch determination, even if only one batch is found.

If this indicator is set in Customizing, then a batch split record is now created in the production order for a single found batch.
If the indicator is blank, then the previous system behavior is still the same, in other words, the batch number is still directly entered in the reservation record.

19.2.3 Changes to Production and Process Orders

Use

As of *SAP R/3 Enterprise Core 4.70 (SAP_APPL 470)*, the production orders and process orders function has been enhanced as following:

- Joint venture connection for process order (only for process orders).
  In the process order, on the detail screen *Assignments*, a *Joint Venture* button is displayed if the joint venture accounting is active. If the process order is assigned to a WBS element, the relevant joint venture data is derived from this. Choose the *Joint Venture* button to display or change the joint venture data.

- Document link from production version (only for production orders).
  Previously, the documents that were copied into the production order were linked with the material master of the assembly to be produced or included in the BOM related to the order. Now, documents can also be assigned to the production version. These documents are then copied into the production order.

- Capacity requirement reduction by confirmations (production and process orders).
  In the work center there is a new indicator *Control Capacity Reduction* for each capacity segment (set up, process, tear down). The indicator controls the capacity reduction of the segment with confirmations. It has the following features:

  - In proportion to quantity
    The quantity-proportional capacity reduction corresponds with the solution implemented in the releases up to SAP R/3 4.6. The capacity requirement is therefore considerably reduced if the relevant operation is finally confirmed, or with quantity-dependent forms, (at least) the complete quantity of the operations was partially confirmed.

  - Complete in partial confirmation
    With a complete reduction in partial confirmation, the status of the relevant operation is analyzed. If this is partially or finally confirmed, the relevant capacity requirement is completely reduced. A partial reduction is not possible.

  - Complete in final confirmation
    With a complete reduction in final confirmation, the status of the relevant operation is analyzed. If this is finally confirmed, the relevant capacity requirement is completely reduced. A partial reduction is not possible.

  - In proportion to activity
    If the capacity reduction takes place in proportion to the activity, the activity that is assigned in your record type group (see tab page *Cost Center Assignment* from work center) to the relevant capacity segment is determined. The planned capacity is then reduced by the confirmed activity quantity.

- Transfer of WM warehouse numbers from updated supply area (production and process orders).
  When a new supply area is created on the goods movement overview of the pick list or the confirmation, checks are carried out to determine whether this supply area exists and is maintained for the storage location. When you save the goods movement, the system determines the WM warehouse numbers using the current supply area.

- Enhancement of BAdI *WORKORDER_UPDATE* (production and process orders).

See release info Business Add-Ins for Production and Process Orders

- Enhancement of BAdI *WORKORDER_CONFIRM* (production and process orders).
  See release info Business Add-Ins for Production and Process Orders

19.2.4 Confirmation: Termination if Incorrect Actual Costs/Goods Movements

Use

As of *SAP R/3 Enterprise Core 4.70 (SAP_APPL 470)*, you can prevent a confirmation from being saved in production and process orders, if error records appeared during actual cost determination or goods movement postings.

Previous process:
When creating a confirmation for orders, processes such as the posting of actual costs or goods movements can be linked. If errors appear when posting, a log is issued if you set the corresponding setting in Customizing. In the case of incorrect goods movements, the user has the option of making corrections. If the error still exists, the confirmation itself, however, can be saved. In this case, postprocessing records are created that can be processed with transactions COFC (for actual costs) and COGI (for goods movements).

New:
For single entry, you can now prevent saving in the situation described above, by setting the corresponding indicator in the confirmation parameters. See also Termination if Incorrect Actual Costs and Termination if Incorrect Goods Movements.
The system then reacts by terminating the confirmation transaction. The indicators are intepreted both when creating as well as when cancelling the confirmation.

19.2.5 PP-SFC-IS Information System

19.2.5.1 DRB (Enhanced)

Use

From *SAP R/3 Enterprise Core 4.70 (SAP_APPL 470)* , the Document Relationship Browser (DRB) offers an extended selection of object types. Object types have been added to the DRB, and object types that were already connected in earlier releases have been completed.

New object types:

- Profit center document
- Costing-based profitability analysis
- Electronic account statement
- Production order
- Backflush document

The connection of these object types only affects the functions within the DRB. There are no separate access programs.

Enhanced object types:

- Accounting document
  From the document display (transaction FB03), you can branch to the DRB. You can use report RDRBFI00 to access the DRB and to find archived accounting documents via the archive information system (SAP AS).
- Controlling document
  From the line item reports (for example, transaction KSB1, KSB5, KOB1), you can branch to the DRB.
  The DRB functions can find Controlling documents in the archives of all archiving objects that archive Controlling line items. There are several field catalogs and info structures for this.

See also

For more information about the Document Relationship Browser, see the SAP note 492938.

19.3 PP-REM Repetitive Manufacturing

19.3.1 Business Add-Ins in Repetitive Manufacturing (REM)

Use

In Repetitive Manufacturing, Business Add-Ins (BAdI) are available as of *SAP R/3 Enterprise Core 4.70 (SAP_APPL 470)* for the following functions:

Repetitive Manufacturing - Planning

- Change Layout of Production List: BAdI RM_LIST_PRODUCTION
  You can use this BAdI to replace the default production lists created in SAP R/3 with your own-defined production lists.
- Hide Lines in Planning Table: BAdI RM_MF50_ROWS
  You can use this BAdI to hide individual line categories for certain planning segments in the planning table in Repetitive Manufacturing.

Repetitive Manufacturing - Backflush

- Enhance Repetitive Manufacturing Confirmation with HR Data: BAdI RM_HR_INTEGRATION
  You can use this BAdI to enhance confirmations in REM with HR-relevant data. With the connection to the HR-components, it is possible to create time tickets based on the confirmations.
- Improve Performance of Repetitive Manufacturing Confirmation: BAdI RM_PERFORMANCE_DREQ
  You can use this BAdI to improve the performance of confirmations in REM with a limited reading of dependent requirements.

Effects on Customizing

To activate a Business Add-In, you must create an active implementation. To do this, choose *Tools* -> *ABAP Workbench* -> *Utilities* -> *Business Add-Ins -> Implementation.*

For more information on the procedure see the SAP Library, under *Basis* -> *ABAP Workbench* -> *Changing the SAP Standard* -> *Business Add-Ins* -> *Implementing Business Add-Ins*.

19.3.2 Usability Improvements in Repetitive Manufacturing (REM)

Use

As of *SAP R/3 Enterprise 4.70 (SAP_APPL 470)*, the following new or changed functions are available:

- **Customizing parameter for synchronous posting during the collective backflush (new)**
  Block problems may occur during the collective backflush (transaction MF42N) if, for example, two or more items are backflushed to the same assembly. This block problem is caused because postings in the standard system are generally asynchronous postings. In other words, one backflush process can still be active, whilst the next one sets blocks.
  There is now a new indicator Synchronous Posting in Customizing for REM. If you set this indicator, you can ensure that the posting of one backflush item is finished before the processing of the next backflush item starts. The block problem is therefore avoided.
  Note that setting this indicator leads to longer runtimes in the collective backflush.

- Batch where-used list **in REM (new)**
  Before Release 4.70 there was no connection from the batch where-used list to the Repetitive Manufacturing backflush. Until now, there has therefore been no significant evaluation regarding the link of produced and entered batches for materials in repetitive manufacturing.

  The batch where-used list now also exists in repetitive manufacturing. This enables you to track the component batches, which flow into production, for each produced batch. You must specify the relevant product batch for each component posting.
  You cannot use this function if you are only working with cumulated postprocessing records.
  You can activate the batch where-used list for repetitive manufacturing in Customizing with the indicator Update batch where-used list.

- **Classification of output batch during backflush (new)**
  As of the current release you can also run a classification of the output batch if it is determined in the movement type Customizing that the classification is to take place online.
  In Customizing of the movement type, it is determined whether a classification of the batch is allowed, and in which form it will be executed. In the Repetitive Manufacturing backflush, all output batches to be classified online have until now not been considered, in other words, in this case, a classification was not possible. You can activate the new function in Customizing for REM with the indicator Batch Classification Active .

  If the backflush is started in the background (not online), there is generally no call for classifying the output batch. The same goes for reporting point quantities without goods receipt posting.

- **Non-document-related reversal for each BAPI (new)**
  As of *SAP R/3 Enterprise 4.70* , you can now also execute a non-document-related reversal for each BAPI. For the backflush in the warehouse scenario, the non-document-related reversal is possible for the following cases:
  - Final backflushes
  - Reporting point backflushes
  - Separate component scrap message
  - Separate component consumption message

  Furthermore, you can also execute non-document-related reversal for final backflushes in the individual customer scenario and production lot scenario.
  You can activate the new function in BAPI *RepManConfirmation1* from the input parameter BCKFLTYPE.
- **Variable file name for EXCEL-download in planning table (new)**
  As of *SAP R/3 Enterprise 4.70* , a dialog box appears during the EXCEL download in the planning table for repetitive manufacturing, in which you can enter a path and a name for the EXCEL file. In older releases, you could not change the file name.

Effects on Customizing

- **Determine synchronous posting during collective backflush**
  Under Customizing for *Production*, choose *Repetitive Manufacturing* -> *Backflushing* -> Maintain Global Settings for Confirmation and LIS. Under Collective Backflush, set the indicator Synchronous Posting.
- **Activate the batch where-used list in repetitive manufacturing**
  Under Customizing for *Production*, choose *Repetitive Manufacturing* -> *Control Data* -> Define Repetitive Manufacturing Profiles. Under Batch Where-Used List, set the indicator Update Batch Where-Used List.
- **Activate the classification of output batch during the online backflush**
  Under Customizing for *Production*, choose *Repetitive Manufacturing* -> *Backflushing* -> Maintain Global Settings for Confirmation and LIS. Under *Backflushing*, set the indicator *Batch Classification Active* to '*1*'.

19.4 PP-PI Production Planning for Process Industries

19.4.1 Changes to the IMG Structure for Production Planning - Process Industries (PP-PI)

Use

In *SAP R/3 Enterprise PLM Extension 1.10* (EA-APPL 110), the following changes were made to the

Implementation Guide (IMG) for *Production Planning - Process Industries (PP-PI)*:

New IMG Activities

Below *Process Management*, the structure node *Decentralized Process Management* was added along with the following activities:

- *Define RFC Destinations for Control Recipe Destinations*
- *Define and Set Up Control Recipe Destinations*
- *Transport Predefined Messages Categories to Central SAP R/3 System*
- *Transport Predefined Messages Categories to Decentralized SAP R/3 System*
- *Define and Set Up Message Destinations in Decentralized SAP R/3 System*
- *Set Up Process Message Categories in Decentralized SAP R/3 System*
- *Define Characteristics for Process Messages in Decentralized SAP R/3 System*

Below *Process Management* -> *Control Recipes/PI Sheets* -> *Control Recipe Destinations*:

- *Edit Filters for Process Instruction Generation*

Below *Process Management* -> *Process Messages* -> *Process Message Destinations*:

- *Define Alert Categories*

Below *Process Management* -> *Process Instructions:*

- *Transport Long Texts of Process Instruction Categories*

Below *Process Management*, the structure node *SAP ODA (OPC Data Access)* was added along with the following activities:

- *Define RFC Destination for SAP ODA*
- *Define Settings for SAP ODA*

Below *Process Management*, the structure node *Business Add-Ins in Process Management* was added along with the following activities:

- *Select Process Messages*
- *Manufacturing Data Access: Register Service*

Enhanced IMG Activities

Below *Process Management* -> *Control Recipes/PI Sheets* -> *Control Recipe Destinations*, a functional enhancement was made for the following activity:

- *Define and Set Up Control Recipe Destinations*

Restructured or Renamed Activities

Below *Process Management* -> *Control Recipes/PI Sheets* -> *Control Recipe Destinations*:

- The activity *Define and Set Up Control Recipe Destinations* was replaced by the structure node *Control Recipe Destinations*. The activity itself was moved below the structure node *Control Recipe Destinations.*

Below *Process Management* -> *Process Messages* -> *Message Destinations*:

- The activity *Define and Set Up Message Destinations* was replaced by the structure node *Message Destinations*. The activity itself was moved below the structure node *Message Destinations*.

19.4.2 PP-PI-MD Basic Data

19.4.2.1 Linking DMS Documents to Production Versions (New)

Use

As of *SAP R/3 Enterprise Core 4.70 (SAP_APPL 4.70)*, you can link documents that are stored in the SAP Document Management System (DMS) to production versions. This function is available in *Production Version Mass Processing* (transaction C223). In this way, you can call up the documents relevant for a production version directly from one transaction.

Creating Documents in DMS

To create a document in DMS, you choose *Logistics* -> *Central Functions* -> *Document Management* -> *Document* -> *Create* from the *SAP Easy Access* screen.

Prerequisites

To display a document for a production version, the following requirements must be met:

- The deletion flag must not be set for the document.
- The document version specified must be valid and released.
- Only one original application file with the application specified may be assigned to the document.
- The original application file must be active and stored in the secure storage area.

Linking Documents to Production Versions

1. To access the transaction for production version mass processing from the *SAP Easy Access* screen, you choose *Logistics* -> *Production - Process* -> *Master Data* -> *Production Versions*.
2. You select a production version and choose *Environment* -> *Document Assignment*.
3. In the dialog box, you enter a document that is stored in *DMS* and choose *Transfer*.
4. To display a document that is already linked, you select the document in the dialog box and choose *Display Originals*.

Note

Note that you **cannot** link DMS documents to production versions from the following transactions:

- Master Recipe
- Material Master Record

19.4.3 PP-PI-POR Process Order

19.4.3.1 Business Add-Ins for Production Orders and Process Orders

Use

In shop floor control, the following Business Add-Ins (BAdI) for production and process orders are available as of *SAP R/3 Enterprise Core 4.70 (SAP_APPL 470)*:

- BAdI WORKORDER_INFOSYSTEM to fill additional (customer-defined) fields in the order information system (new).
  In addition, you can define authorization checks for variants in and changes to the field selection.
- BAdI WORKORDER_REWORK to adjust the yield during confirmation of rework with own tools (new)
  With the help of the new BAdI *WORKORDER_REWORK*, it is possible to implement customer-specific rework logic. It is available for one interface to transfer confirmed quantities outwards, and therefore activate the rework processing. For other interfaces, information can be transferred inwards from the rework processing - like the successfully reworked yield - and is therefore considered in the quantity logic of the order and confirmation.

The following BAdIs were changed:

- BAdI WORKORDER_UPDATE was extended by the *BEFORE_UPDATE* method. This method is implemented directly before calling the posting. The order data is provided in table form as is the existing customer exit PPCO0001. Modifications to this data can no longer be made, but subsequent processes can be derived and triggered.
- BAdI WORKORDER_CONFIRM was extended by the *BEFORE_UPDATE* BAdI method. This allows, aside from the standard check for whether the selected confirmation can be cancelled, the execution of additional, customer-specific checks.

Effects on Customizing

To activate a Business Add-In, you must create an active implementation. To do this, choose from the SAP menu *Tools -> ABAP Workbench -> Utilities -> Business Add-Ins -> Implementation.*

For more information on the procedure, see the SAP Library under *Basis Components -> ABAP Workbench -> Changing the SAP Standard -> Business Add-Ins -> Implementing Business Add-Ins*.

19.4.3.2 Changes to Production and Process Orders

Use

As of *SAP R/3 Enterprise Core 4.70 (SAP_APPL 470)*, the production orders and process orders function has been enhanced as following:

- Joint venture connection for process order (only for process orders).

In the process order, on the detail screen *Assignments*, a *Joint Venture* button is displayed if the joint venture accounting is active. If the process order is assigned to a WBS element, the relevant joint venture data is derived from this. Choose the *Joint Venture* button to display or change the joint venture data.

- Document link from production version (only for production orders).
  Previously, the documents that were copied into the production order were linked with the material master of the assembly to be produced or included in the BOM related to the order. Now, documents can also be assigned to the production version. These documents are then copied into the production order.

- Capacity requirement reduction by confirmations (production and process orders).
  In the work center there is a new indicator *Control Capacity Reduction* for each capacity segment (set up, process, tear down). The indicator controls the capacity reduction of the segment with confirmations. It has the following features:

  - In proportion to quantity
    The quantity-proportional capacity reduction corresponds with the solution implemented in the releases up to SAP R/3 4.6. The capacity requirement is therefore considerably reduced if the relevant operation is finally confirmed, or with quantity-dependent forms, (at least) the complete quantity of the operations was partially confirmed.

  - Complete in partial confirmation
    With a complete reduction in partial confirmation, the status of the relevant operation is analyzed. If this is partially or finally confirmed, the relevant capacity requirement is completely reduced. A partial reduction is not possible.

  - Complete in final confirmation
    With a complete reduction in final confirmation, the status of the relevant operation is analyzed. If this is finally confirmed, the relevant capacity requirement is completely reduced. A partial reduction is not possible.

  - In proportion to activity
    If the capacity reduction takes place in proportion to the activity, the activity that is assigned in your record type group (see tab page *Cost Center Assignment* from work center) to the relevant capacity segment is determined. The planned capacity is then reduced by the confirmed activity quantity.

- Transfer of WM warehouse numbers from updated supply area (production and process orders).
  When a new supply area is created on the goods movement overview of the pick list or the confirmation, checks are carried out to determine whether this supply area exists and is maintained for the storage location. When you save the goods movement, the system determines the WM warehouse numbers using the current supply area.

- Enhancement of BAdI *WORKORDER_UPDATE* (production and process orders).
  See release info Business Add-Ins for Production and Process Orders

- Enhancement of BAdI *WORKORDER_CONFIRM* (production and process orders).
  See release info Business Add-Ins for Production and Process Orders

19.4.3.3 Confirmation: Termination if Incorrect Actual Costs/Goods Movements

Use

As of *SAP R/3 Enterprise Core 4.70 (SAP_APPL 470)*, you can prevent a confirmation from being saved in production and process orders, if error records appeared during actual cost determination or goods movement postings.

Previous process:
When creating a confirmation for orders, processes such as the posting of actual costs or goods movements can be linked. If errors appear when posting, a log is issued if you set the corresponding setting in Customizing. In the case of incorrect goods movements, the user has the option of making corrections. If the error still exists, the confirmation itself, however, can be saved. In this case, postprocessing records are created that can be processed with transactions COFC (for actual costs) and COGI (for goods movements).

New:
For single entry, you can now prevent saving in the situation described above, by setting the corresponding indicator in the confirmation parameters. See also Termination if Incorrect Actual Costs and Termination if Incorrect Goods Movements.
The system then reacts by terminating the confirmation transaction. The indicators are intepreted both when creating as well as when cancelling the confirmation.

19.4.4 PP-PI-PMA Process Management

19.4.4.1 Introduction to Decentralized Process Management

Use

As of *SAP R/3 Enterprise SCM Extension 1.10 (EA_APPL 110)*, you can use decoupled process management in a decentralized SAP R/3 System. This means that an additional SAP R/3 System that only contains data for production execution is installed at the plant level. By decoupling this system from the central, corporate SAP R/3 System, the availability of functions closely related to production can be guaranteed much easier. When the central SAP R/3 System is unavailable, for example, due to a release upgrade, production can still be executed.

Communication between the central and the decentralized SAP R/3 System takes place via the PI-PCS interface. You can link any number of decentralized SAP R/3 Systems to a central SAP R/3 System. Via the control recipe destination, you define in which decentralized system production is executed.

To transfer control recipes to a decentralized system, you must send them to a control recipe destination of *Type 2 - Download to external system, initiated by SAP process management*.

Scenario

In the central SAP R/3 System, you maintain the master data in the *Production Planning - Process Industries (PP-PI)* component as usual. You create process orders and release them. The control recipes created when the orders are released are sent to the decentralized SAP R/3 System via the PI-PCS interface. Here, the control recipes are either converted into a PI sheet or transferred further to an external process control system via the PI-PCS interface.

In the decentralized SAP R/3 System, you maintain the PI sheet and report actual data. The process messages resulting from this:

- Can be forwarded to the central system
In this way, stocks and the order status are updated. This guarantees data integration.
- Can be processed locally in the decentralized system
You can forward the data to external systems or store them for future analyses.

If the control recipes in the decentralized system are processed by an external process control system, this external system can report its actual data either to the decentralized system or directly to the central system.

Effects on Customizing

You make the following Customizing settings in the central and the decentralized system:

Settings in the Central SAP R/3 System

1. You create the RFC destination of the decentralized SAP R/3 System.
   Define RFC Destinations for Control Recipe Destinations
2. You define a control recipe destination of *Type 2 - Download to external system - initiated by SAP process management* that you want to use to send the control recipes intended for the decentralized system. You specify the RFC destination of the decentralized system as the destination address. This control recipe destination must have the same name as the destination in the decentralized system.
   Define and Set Up Control Recipe Destinations
3. To update the status of control recipes in the central and decentralized system, you need special process message categories and destinations.
   You copy the process message categories and destinations delivered by SAP to your plants for the central system.
   Transport Predefined Message Categories to Central SAP R/3 System

Settings in the Decentralized SAP R/3 System

1. You create the RFC destination of the central system in the decentralized system.
   Define RFC Destinations for Control Recipe Destinations
2. You create a control recipe destination with any destination type that you like. However, it must have the same name as the control recipe destination in the central system.
   Depending on how the control recipes are to be processed in the decentralized system, you must choose the corresponding destination type. If control recipes are to be processed by an external process control system, you must specify the RFC destination of the process control system as the destination address.
   Define and Set Up Control Recipe Destinations
3. To update the status of control recipes in the central and decentralized system, you need special process message categories and destinations.
   You copy the process message categories and destinations delivered by SAP to your plants for the decentralized system.
   Transport Predefined Message Categories to Decentralized SAP R/3 System
4. You create a new message destination to report the following message categories to the central SAP

R/3 System:

- Message category PI_CRST2
- All other message categories that you want to use for confirmations to the central SAP R/3 System (for example, PI_CONS, PI_PROD)
  Define and Set Up Message Destinations in Decentralized SAP R/3 System

5. You adjust the message categories that you want to use for confirmations to the central SAP R/3 System.

   a) You delete the assignment of message destination PI09 to message category PI_CRST.

   b) You assign the new message destination of *Type 2 - Destination is external function* that you have created in the decentralized SAP R/3 System to message category PI_CRST2.

   c) You decide which other message categories you want to send to the central SAP R/3 System (for example, PI_CONS, PI_PROD). You delete the message destinations of these message categories and assign the new message destination of *Type 2 - Destination is external function* to them.
   Note that the process message categories and the characteristics, destinations, and target fields they contain must be the same in the central and the decentralized system.
   Set Up Process Message Categories in Decentralized SAP R/3 System

6. You make additional settings in characteristics definition:
   Usually, all master data (for example, material masters) is maintained in the central SAP R/3 System. If you want to assign values to characteristics such as PPPI_MATERIAL in the decentralized system, it is helpful if all relevant material numbers can also be called up through value help there. However, this data is not available through the standard value help. Only the data maintained in the decentralized system is available.
   To access the master data from the central system in the decentralized system through value help, you must specify the RFC destination of the central system in which the master data is maintained for each characteristic in characteristics definition.
   Note that the relevant characteristic must also exist in the decentralized system and have the same technical properties (format, length, decimal places, and so on).
   To do so, choose the IMG activity Set Up Characteristics for Process Messages in Decentralized SAP R/3 System and choose *Proc.mgmt* on the *Basic Data* tab.

19.4.4.2 Data Exchange Between Applications in the SAP R/3 System (New)

Use

With *SAP R/3 Enterprise SCM Extension 1.10 (EA_APPL 110)*, the new internal interface *Manufacturing Data Access* is introduced. It enables exchanging manufacturing events and data with reference to manufacturing data points between different applications of the SAP R/3 System.

Functions

Any application can register at the interface as a *Service*. A server can be used as a client, a server, or both as a client and a server. Clients can request data from other services, while servers provide data for other services. Services that are used both as clients and servers can provide and request data.

Furthermore, *Manufacturing Data Access* provides additional functions that you can use to dynamically request data points or events as well as navigate to their maintenance environment.

Depending on whether the registered application serves as a client or as a server, you can use the following functions through *Manufacturing Data Access*:

A **client** can:

- Read data points from other applications
- Write data points to other applications
- Subscribe to events from other applications
  When an application subscribes to events from another application, these events are automatically reported to this application when they occur. In this way, you can, for example, subscribe to value changes.
  In addition, events may contain parameters that you can read while the events are processed.

A **server** provides:

- Data points that can be read or written by other applications
- Events to which other applications can subscribe
  Once events occur, they are published by the server.

In this way, you can:

- Synchronously read or write the values of one or more data points,
  this means on request (for example, by pressing a button)
- Read or write the values of one or more data points on an event-driven basis,
  this means that in running applications, data is automatically read or written - triggered by an event - and the user does not need to request it explicitly

Events can also be processed in different system logons running in parellel.
Example: User A maintains PI sheet 001. User B maintains PI sheet 002. Value changes in PI sheet 002 can be transferred to PI sheet 001 automatically although the two users are logged on at different terminals.

Services

The following **services** and their functions are available in the *Production Planning - Process Industries* environment at the moment. For more information on the individual services, see the corresponding release notes.

- MDA: Generic Services
- PSH: PI Sheet
- PMC: Manufacturing Cockpit
- ODA: OPC Data Access
- OAE: OPC Alarms/Events
- SYS: System Information

Data Acces and Addresses for Events and Data Points

For the data exchange between applications, addresses are used to address data points and events.

Addresses start with the name of the service. The other address parts are determined depending on the service.

You must define the data access to the services differently in the various client applications. For more information, see the release note on the relevant service.

Examples for Event Addresses:

- **ODA:0001.MYITEM.VCHD**
  Service ODA , plant 0001, OPC item MYITEM, value changed
- **OAE:0001.MYSUBSCRIPTION**
  Service OAE, plant 0001, OPC Alarms/Events subscription MYSUBSKRIPTION triggered
- **PMC:0001.MYCOCKPIT.VARCHGD**
  Service PMC, plant 0001, cockpit MYCOCKPIT, value changes to global variables occurred
- **PSH:100000000000004711.LOGIN**
  Service PSH, user logged on to maintain PI sheet **100000000000004711**
- **MDA:BROADCAST.DEMO**
  Service MDA, user-defined event **DEMO** triggered

Examples for Data Point Addresses:

- **ODA:0001.MYITEM.PV**
  Service ODA, plant 0001, OPC item MYITEM, current value
- **PMC:0001.MYCOCKPIT.MYVAR**
  Service PMC, plant 0001, cockpit MYCOCKPIT, value of variable MYVAR
- **PSH:100000000000004711.MYVAR**
  Service PSH, value of variable MYVAR in the PI sheet **100000000000004711**
- **SYS:TIME.UTC**
  Service SYS, current time (Universal Coordinated Time)

Registering New Applications as Services

You can use the **Business Add-In** (BAdI) SCM_CMX_DA_SRV to register any application of the SAP R/3 System as a service.

To do so, you go to Customizing for *Process Management* and choose *Business Add-Ins in Process Management* -> *Manufacturing Data Access: Register Service*.

For more information, see BAdI: Registering a Manufacturing Data Access Service.

Example

This section serves to illustrate the following functions:

- Data exchange between a PI sheet and a cockpit using Manufacturing Data Access
- Reading of OPC items from a cockpit using SAP ODA and Manufacturing Data Access
- Using of internal and external manufacturing events to carry out functions automatically in the PI sheet
- Using of global variables that are transferred as data points using Manufacturing Data Access

The PI sheet layout is defined in such a way that a cockpit is integrated in it in a frame. The cockpit serves to automatically read a value from the PI sheet and the value of an OPC item and add both values.

The PI sheet contains one input field. As soon as a value is entered in this field, it is automatically transferred to the cockpit. As soon as a variable changes in the cockpit, the system carries out the calculation automatically. The result of this calculation is automatically returned to the PI sheet and at the same time displayed in the cockpit. In this way, the reading of OPC items and the calculation function are independent from the PI sheet.

You define the following in the **master recipe or process order**:

- The PI sheet is to contain an input field to which variable VAR1 refers.
- When the variable VAR1 changes in the PI sheet, its value is to be transferred to the cockpit, to the global variable GINVAR. Using characteristic PPPI_EVENT, you specify that the value is to be transferred to the cockpit with the internal event *Parameter Changed*.
- When a variable changes in the cockpit MYCOCKPIT, the changed value is to be returned to the PI sheet, via the global variable GOUTVAR of the cockpit. This value is transferred to the PI sheet via the import parameter PMC:1100:MYCOCKPIT.GOUTVAR. For this, you specify the external event PMC:1100.MYCOCKPIT.VARCHGD so that the value can be transferred automatically.
- The imported value of global variable GOUTVAR from the cockpit is to be imported to the local variable VAR2 of the PI sheet and displayed there.

To do so, you define the following characteristics in a process instruction of type 0:

Characteristic	Characteristic Value
PPPI_INPUT_REQUEST	Value entry
PPPI_VARIABLE	VAR1
PPPI_REQUESTED_VALUE	PPPI_MATERIAL_QUANTITY
PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_BUTTON_TEXT	Send value to cockpit
PPPI_FUNCTION_DURING_DISPLAY	Permitted
PPPI_EVENT	Parameter Changed
PPPI_EXPORT_DATA	PMC:0001.MYCOCKPIT.GINVAR
PPPI_STRING_VARIABLE	VAR1
PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_BUTTON_TEXT	Read value from cockpit
PPPI_FUNCTION_DURING_DISPLAY	Permitted
PPPI_EVENT	PMC:0001.MYCOCKPIT.VARCHGD
PPPI_IMPORT_DATA	PMC:0001.MYCOCKPIT.GOUTVAR
PPPI_STRING_VARIABLE	VAR2
PPPI_OUTPUT_TEXT	Value calculated
PPPI_OUTPUT_VARIABLE	VAR2

In the **Cockpit** MYCOCKPIT, you define the following:

- You declare the global variable GINVAR and output the value of variable GINVAR.
- Using Manufacturing Data Access, the current value of the OPC Data Access item MYODA is to be read. You define the automatic reading operation by assigning the value ODA:0001.MYODA.VCHD to PPPI_EVENT. In addition, the OPC value is also to be read on request. For this, you specify PPPI_BUTTON_TEXT. The OPC value is to be displayed in the cockpit.
- When a variable changes in the cockpit, a calculation is to be carried out automatically. For this, you use the internal event *Parameter Changed*. The value of global variable GINVAR and the current value of the OPC item are to be added.
  Before the calculation, you define the global variable GOUTVAR so that the value can be output later.
- The result of the calculation (global variable GOUTVAR) is to be displayed in the cockpit.

To do so, you define the following characteristics in a process instruction of type 0:

Characteristic	Characteristic Value
PPPI_GLOBAL_VARIABLE	GINVAR
PPPI_MATERIAL_CONSUMED	0
PPPI_OUTPUT_TEXT	Value from PI sheet
PPPI_OUTPUT_VARIABLE	GINVAR
PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_BUTTON_TEXT	Read OPC item
PPPI_FUNCTION_DURING_DISPLAY	Permitted
PPPI_EVENT	ODA:0001.MYODA.VCHD
PPPI_IMPORT_DATA	ODA:0001.MYODA.PV
PPPI_FLOAT_VARIABLE	ODAVAR1
PPPI_OUTPUT_VARIABLE	ODAVAR1
PPPI_INPUT_REQUEST	Value from PI sheet + OPC value
PPPI_GLOBAL_VARIABLE	GOUTVAR
PPPI_CALCULATED_VALUE	PPPI_MATERIAL_CONSUMED
PPPI_EVENT	Parameter Changed
PPPI_CALCULATION_FORMULA	GINVAR + ODAVAR1
PPPI_OUTPUT_TEXT	Value calculated
PPPI_OUTPUT_VARIABLE	GOUTVAR

19.4.4.3 - Service MDA

Use

With *SAP R/3 Enterprise SCM Extension 1.10 (EA_APPL 110)*, the new internal interface *Manufacturing Data Access* is introduced. It enables the data exchange between different applications of the SAP R/3 System. Applications that are registered at this interface as services can exchange values from manufacturing data points and manufacturing events.

For more information, see the release note Data Exchange Between Applications in the SAP R/3 System that provides an overview of *Manufacturing Data Access*.

The MDA service is used as a server and contains the generic services BROADCAST and REDIRECT.

BROADCAST

You can use this service to define user-specific events. A generic data point is available for this to which you can write any value. The value change of the data point is published as an event. Other services can subscribe to this event.

Address for Event Propagation

You specify the user-specific event that you want to propagate in an address. The address is made up as follows:
MDA:BROADCAST.SEND.<*Category*>

You replace <*Category*> by any character string representing the user-specific value. Note that the entry is case-sensitive.

Address for Event Subscription

Other services can use the following address to subscribe to this event:
MDA:BROADCAST.<*Category*>

Address for Parameter Reading

The following information can be transferred as data points together with the event. The address is made up as follows:

- Category of the event
  MDA:BROADCAST.PARAM.CATEGORY
- The system logon from which the event was sent
  MDA:BROADCAST.PARAM.SESSION
- The terminal from which the event was sent
  MDA:BROADCAST.PARAM.TERMINAL
- The time stamp at which the event was sent
  MDA:BROADCAST.PARAM.TIMESTAMP
- The user who sent the event
  MDA:BROADCAST.PARAM.USER
- The value of the data point that is transferred with the event
  MDA:BROADCAST.PARAM.VALUE

Example for BROADCAST:

The fill level of tank #333 is to be recorded in a manufacturing cockpit. The up-to-date fill level is to be published as an event by pressing a button. In the address **MDA:BROADCAST.SEND.<*Category*>**, <*Category*> was replaced by **DPV** meaning *data point value*.
In the cockpit defintion, you define the following characteristics in the sequence specified:

PPPI_DATA_REQUEST_TYPE	Simple data request
PPPI_DATA_POINT_NAME	Tank #333
PPPI_OUTPUT_CHARACTERISTIC	PPPI_DATA_POINT_NAME

PPPI_INPUT_REQUEST	Record fill level
PPPI_VARIABLE	VAR_DPV
PPPI_REQUESTED_VALUE	PPPI_DATA_POINT_VALUE
PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_BUTTON_TEXT	Publish
PPPI_FUNCTION_DURING_DISPLAY	Permitted
PPPI_EXPORT_DATA	MDA:BROADCAST.SEND.DPV
PPPI_FLOAT_VARIABLE	VAR_DPV

In transaction *CO69 - Create Process Message Automatically*, a process message is always to be created automatically when the propagated event above is published. You create a variant in transaction CO69 and define the following:

- Option *Event-Driven Start* as the *Start Condition*
- Address **MDA:BROADCAST.DPV** as the event

At the same time, parameter *Value* of the event is to be read in the process message. To do so, select value assignment type *Data Point* for a message characteristic and assign address **MDA:BROADCAST.PARAM.VALUE.** to it.

REDIRECT

Using REDIRECT, you can address any number of data points dynamically and read or write the values belonging to them. Two generic data points are available for this. You use one of them to transfer the address dynamically and the other to transfer the value.

- **Address for the Data Point**
  The address selected dynamically is transferred to the generic data point **MDA:REDIRECT.ADDRESS.*<Index>***.
  <*Index*> can be replaced by any character string. Note that the entry is case-sensitive.
- **Address for the value of the data point**
  The value of the data point addressed dynamically can be read or written using the generic data point **MDA:REDIRECT.VALUE.*<Index>*** .
  You replace <*Index*> with the same value that you have already used in address MDA:REDIRECT.ADRESS.<Index>. Using the index that you can freely choose, you can create any number of pairs of dynamically selected addresses and values in one call. This means that you can address any number of data points dynamically in one call.

Example for REDIRECT:

The process operator is to read the fill levels of different tanks in manufacturing cockpit *MYCOCKPIT*. He or she can choose between three different tanks in a dropdown box. The tank he or she selects is written to the generic data point as the address. The current fill level of the tank selected is read by an OPC Data Access server and displayed in the cockpit.

To get a dropdown box with different tanks in the cockpit, characteristic Z_MY_ADDRESS was created, which contains three fixed values. Characteristic Z_MY_ADDRESS contains the following fixed values:

Characteristic Value	Description
ODA:0001.TANK1.PV	Tank 1
ODA:0001.TANK2.PV	Tank 2
ODA:0001.TANK3.PV	Tank 3

In cockpit *MYCOCKPIT*, you define the following characteristics in the sequence specified:

PPPI_INPUT_REQUEST	Select tank
PPPI_VARIABLE	ADDRESS
PPPI_REQUESTED_VALUE	Z_MY_ADDRESS
PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_BUTTON_TEXT	Read fill level
PPPI_FUNCTION_DURING_DISPLAY	Permitted
PPPI_EVENT	PARAMETER_CHANGED
PPPI_EXPORT_DATA	MDA:REDIRECT.ADDRESS.1
PPPI_STRING_VARIABLE	ADDRESS
PPPI_IMPORT_DATA	MDA:REDIRECT.VALUE.1
PPPI_STRING_VARIABLE	VALUE
PPPI_OUTPUT_TEXT	Fill level
PPPI_OUTPUT_VARIABLE	VALUE

The following applications can access events and data points of the MDA service:

- Automatic Process Message Creation
- PI Sheet
- Manufacturing Cockpit

19.4.4.4 - Services PSH and PMC

Use

With *SAP R/3 Enterprise SCM Extension 1.10 (EA_APPL 110)*, the new internal interface *Manufacturing Data Access* is introduced. It enables the data exchange between different applications of the SAP R/3 System. Applications that are registered at this interface as services can exchange values from manufacturing data points and manufacturing events.

For more information, see the release note Data Exchange Between Applications in the SAP R/3 System that provides an overview of *Manufacturing Data Access*.

PI sheets are registered as *Service PSH* and cockpits as *Service PMC*. Both services serve as clients and servers, which means that they can both provide data points and events, and request them.

The following client functions are possible:

- Reading and writing data points from the following services:
  - PSH: PI sheet
  - PMC: Manufacturing cockpit
  - ODA: OPC Data Access
  - SYS: System information
- Subscribing to an event from the following services:
  - MDA: Generic service
  - PSH: PI sheet
  - PMC: Manufacturing cockpit
  - ODA: OPC Data Access
  - OAE: OPC Alarms/Events

  You can use events to subscribe to value changes.

The PSH and PMC services offer the following server functions:

- **Events** to which other services can subscribe:
  Events of Service PSH
  - PI sheet was completed
    Address: **PSH:<No. of PI sheet>.COMPLTD**
    Example: **PSH:100000000000004711.COMPLTD**
  - PI sheet was created
    Address: **PSH:<No. of PI sheet>.CREATED**
  - Maintenance was started
    Address: **PSH:<No. of PI sheet>.LOGIN**
  - Maintenance was exited
    Address: **PSH:<No. of PI sheet>.LOGOFF**
  - Data was reported
    Address: **PSH:<No. of PI sheet>.REPORTD**
  - Data was saved
    Address: **PSH:<No. of PI sheet>.SAVED**
  - Global variables were changed
    Address: **PSH:<No. of PI sheet>.VARCHGD**

  Events of Service PMC
  - Manufacturing cockpit was started
    Address: **PMC:<Plant>.<Cockpit name>.LOGIN**
    Example: **PMC:0001.MYCOCKPIT.LOGIN**
  - Manufacturing cockpit was exited
    Address: **PMC:<Plant>.<Cockpit name>.LOGOFF**
  - Global variables were changed

Address: **PMC:<Plant>.<Cockpit name>.VARCHGD**

- Global variables that other services can access as **data points**
  - **Data Point of Service PSH**:
    Address: **PSH:<Number of PI sheet>.<global variable>**
    Example: **PSH:100000000000004711.MYVAR**
  - **Data Point of Service PMC**:
    Address: **PMC:<Plant>.<Cockpit name>.<Globale variable>**
    Example: **PMC:0001.MYCOCKPIT.MYVAR**

The following applications can access data points and events from the services PSH and PMC:

- Automatic Process Message Creation
- PI Sheet
- Manufacturing Cockpit

Data Access Definition

You can define the data access using *Manufacturing Data Access* in the master recipe, process order, and in the cockpit defintion. The new characteristics listed below were defined for this. The syntax for *Manufacturing Data Access* is similar to that of dynamic function calls. Here, import and export parameters are also transferred.

When defining *Manufacturing Data Access*, you define the characteristics below in the sequence specified in a process instruction of *Type 0: Universal process instruction category*. In one Manufacturing Data Access definition, you can read (PPPI_IMPORT_DATA) or write (PPPI_EXPORT_DATA) any number of data points. Within the Manufacturing Data Access definition, however, you must not add other characteristics.

Read Data Point

Characteristic	Characteristic Value
PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_IMPORT_DATA	Data point or global variable from which the value is to be transferred to the subsequent variable
Parameter characteristic	Variable names with the corresponding data type

Example:

You want to read the current value of data point MYODA in plant 0001 of service ODA and transfer it to the local variable MYVAR.

PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_IMPORT_DATA	ODA:0001.MYODA.PV
PPPI_STRING_VARIABLE	MYVAR

Write Data Point

Characteristic	Characteristic Value
PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_EXPORT_DATA	Data point or global variable of the application to which the value of the subsequent variable is to be transferred

Parameter characteristic	Variable names with the corresponding data type

Example:

You want to transfer the value of the local variable MYVAR to the global variable GINVAR and export it as a data point to service PMC. Here, the value of GINVAR is to be written to cockpit MYCOCKPIT in plant 1100.

PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_EXPORT_DATA	PMC:1100.MYCOCKPIT.GINVAR
PPPI_STRING_VARIABLE	MYVAR

Read Data Point Value on Event-Driven Basis

Characteristic	Characteristic Value
PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_EVENT	Internal or external event
PPPI_IMPORT_DATA	Data point or global variable of the application from which the value is to be transferred to the subsequent variable
Parameter characteristic	Variable names with the corresponding data type

Example:

The value change of data point MYODA in plant 0001 of service ODA is defined as an event. When the value of this data point changes, it is to be read and written to the local variable MYVAR.

PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_EVENT	ODA:0001.MYODA.VCHD
PPPI_IMPORT_DATA	ODA:0001.MYODA.PV
PPPI_STRING_VARIABLE	MYVAR

Write Data Point Value on Event-Driven Basis

Characteristic	Characteristic Value
PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_EVENT	Internal or external event
PPPI_EXPORT_DATA	Data point or global variable of the application to which the value of the subsequent variable is to be transferred
Parameter characteristic	Variable names with the corresponding data type

See also

For more information, see
Automatic Execution of Functions in PI Sheets and Manufacturing Cockpits
Using Global Variables in PI Sheets and Manufacturing Cockpits

19.4.4.5 - Services ODA and OAE

Use

With *SAP R/3 Enterprise SCM Extension 1.10 (EA_APPL 110)*, the new internal interface *Manufacturing Data Access* is introduced. It enables the data exchange between different applications of the SAP R/3 System. Applications that are registered at this interface as services can exchange values from manufacturing data points and manufacturing events.

For more information, see the release note Data Exchange Between Applications in the SAP R/3 System that provides an overview of *Manufacturing Data Access*.

The OPC Data Access and OPC Alarms/Events functions of SAP OPC Data Access (SAP ODA) were registered as *Service ODA* and *Service OAE*. Both services serve as servers only, which means that they provide:

- OPC items that can be read by other services (service ODA):
  - Current value
    Address: **ODA:<Plant>.<OPC item>.PV**
    Example: **ODA:0001.MYITEM.PV**
  - Date
    Address: **ODA:<Plant>.<OPC item>.DT**
  - Time
    Address: **ODA:<Plant>.<OPC item>.TM**
  - Quality of item value
    Address: **ODA:<Plant>.<OPC item>.QA**
  - Substatus of quality
    Address: **ODA:<Plant>.<OPC item>.QS**
  - Quality limit
    Address: **ODA:<Plant>.<OPC item>.QL**
  - Result text
    Address: **ODA:<Plant>.<OPC item>.RT**
- Value changes of OPC items to which other services can subscribe as events (service ODA)
  Address: **ODA:<Plant>.<OPC item>.VCHD**
  Example: **ODA:0001.MYITEM.VCHD**
- OPC events to which other services can subscribe (service OAE)
  Address: **OAE:<Plant>.<OPC subscription>**
  Example: **OAE:0001.MYSUBSCRIPTION**

The following applications can access events and data points from the services ODA and OAE:

- Automatic Process Message Creation
- PI Sheet
- Manufacturing Cockpit

For more information, see Introduction of SAP OPC Data Access (SAP ODA).

19.4.4.6 - Service SYS

Use

With *SAP R/3 Enterprise SCM Extension 1.10 (EA_APPL 110)*, the new internal interface *Manufacturing Data Access* is introduced. It enables the data exchange between different applications of the SAP R/3 System. Applications that are registered at this interface as services can exchange values from manufacturing data points and manufacturing events.

For more information, see the release note Data Exchange Between Applications in the SAP R/3 System that provides an overview of *Manufacturing Data Access*.

The service SYS serves as a server. It provides the following data points containing system information:

- Date according to local settings by the user
  SYS:DATE.LOCAL
- Date, Universal Coordinated Time (UTC)
  SYS:DATE.UTC
- Time according to local settings by the user
  SYS:TIME.LOCAL
- Time, Universal Coordinated Time (UTC)
  SYS:TIME.UTC
- Time stamp, long format
  SYS:TIMESTAMP.LONG
- Time stamp, short format
  SYS:TIMESTAMP.SHORT
- Logon language
  SYS:SESSION.LANGUAGE
- Name of terminal
  SYS:SESSION.TERMINAL
- User logged on
  SYS:SESSION.USER.NAME
- User parameters
  SYS:SESSION.USER.PARAM.<Parameter name>
- ABAP text symbol, language selected, or logon language
  SYS:TEXT.SYMBOL.<Program>.<ID>.<Language>
- ABAP message text in logon language
  SYS:TEXT.MESSAGE.<Message class>.<Message number>.<Variable texts>...
- Globally unique key, 16 characters, binary
  SYS:GUID.BIN16
- Globally unique key, 22 characters, text type
  SYS:GUID.CHAR22

- Globally unique key, 32 characters, text type
  SYS:GUID.CHAR32

The following applications can access data points of service SYS:

- Automatic Process Message Creation
- PI Sheet
- Manufacturing Cockpit

19.4.4.7 PP-PI-PMA-MSG Process Message

19.4.4.7.1 New Message Destination for Sending Alerts

Use

As of *SAP R/3 Enterprise SCM Extension 1.10 (EA_APPL 110)*, the new standard process message destination *Type 5: Alert Category* is available. You can use this message destination, for example, to send alerts to one or more recipients from PI sheets, external process control systems, or manually. The alerts can be sent to the recipients through SMS or e-mail. In this way, you can inform certain recipient groups about unexpected and maybe critical situations in the plant.

To use a process message destination of type 5 to send alerts, you proceed as follows:

1. You carry out the steps described in the *Effects on Customizing* section.
2. You create process messages with reference to the message destination that you have defined in Customizing. You can create process messages:
   - In the PI sheet
   - In transaction *CO69 - Create Process Messages Automatically*
   - Via the PI-PCS interface
3. You send the process messages.

Effects on Customizing

1. You define an alert category.
   - If required, you can define container elements that you can insert in the alert text as text variables.
   - You define a short text and a long text for the alert category.

   Define Alert Categories
2. You assign the alert recipients to the alert category.
3. You define a process message destination of type 5. You enter the alert category as the destination address.
   Define and Set Up Message Destinations
4. If you have defined container elements for the alert category, you must define destination-specific

target fields for them in the message destination. You enter the names of the container elements as the target field names.

5. You define a process message category to which you assign the message destination along with the destination-specific target fields.
   Define and Set Up Process Message Categories

19.4.4.7.2 New Transaction for Automatic Process Message Creation (New)

Use

As of *SAP R/3 Enterprise SCM Extension 1.10 (EA_APPL 110)*, the new transaction Create Process Message Automatically - CO69 is available.

You can use this transaction in particular to specify that process messages are automatically created at certain intervals or when an event occurs. In this way, you can make sure, for example, that process messages for a certain event are created even if the corresponding PI sheet is not being maintained.

Start Conditions

You create a variant for every process message category in the transaction. You can choose from the following start conditions:

- ***Start not Allowed***
  The variant must not be started.
- ***Manual Individual Start***
  The variant can be started manually on the overview screen. The process message is created only once.
- ***Periodic Start***
  With this condition, you enter the time intervals at which the variant is to be started automatically. The process messages are automatically created in the time intervals specified.
- ***Event-Driven Start***
  With this condition, you specify the manufacturing event that is to trigger message creation. Events are provided via the *Manufacturing Data Access* interface by the services registered there. Addresses are used to access events. They start with the name of the service providing the event.

Examples for Event Addresses:

- **ODA:0001.MYITEM.VCHD**
  Service ODA , plant 0001, OPC item MYITEM, value changed
- **OAE:0001.MYSUBSCRIPTION**
  Service OAE, plant 0001, OPC Alarms/Events subscription MYSUBSKRIPTION triggered
- **PMC:0001.MYCOCKPIT.VARCHGD**
  Service PMC, plant 0001, cockpit MYCOCKPIT, value changes to global variables occurred
- **PSH:100000000000004711.LOGIN**
  Service PSH, user logged on to maintain PI sheet **100000000000004711**
- **MDA:BROADCAST.DEMO**

Service MDA, user-defined event **DEMO** triggered
For more information, see Data Exchange Between Applications in the SAP R/3 System.

Characteristic Value Assignment

You can assign the following values to the characteristics of the message:

- ***No value***
- ***Any fixed value***
- ***Manufacturing data point***
  Manufacturing data points are provided via the *Manufacturing Data Access* interface by the services registered there. Addresses are used to access data points. They start with the name of the service providing data points. You can use value help to select these data points.

Examples for Data Point Addresses:

- **ODA:0001.MYITEM.PV**
  Service ODA, plant 0001, OPC item MYITEM, current value
- **PMC:0001.MYCOCKPIT.MYVAR**
  Service PMC, plant 0001, cockpit MYCOCKPIT, value of variable MYVAR
- **PSH:100000000000004711.MYVAR**
  Service PSH, value of variable MYVAR in the PI sheet **100000000000004711**
- **SYS:TIME.UTC**
  Service SYS, current time (Universal Coordinated Time)
  For more information, see Data Exchange Between Applications in the SAP R/3 System.

Logging

All variants defined in a plant are displayed on the overview screen. Here, you can start or stop the variants. The system logs all start activities and the success or error messages that may occur for the variants.

19.4.4.8 PP-PI-PMA-PMC Browser-Based PI Sheet / Cockpit

19.4.4.8.1 User Change in PI Sheets and Cockpits (New)

Use

Until now, only the person logged on to the SAP R/3 System could maintain the PI sheet or manufacturing cockpit at a certain time. If the processor changed, the current processor first had to exit the PI sheet or cockpit and log off from the SAP R/3 System before the new processor could log on the system and maintain the PI sheet or cockpit.

As of *SAP R/3 Enterprise SCM Extension 1.10 (EA_APPL 110)*, you can switch between an anonymous service user and an individual user while maintaining the PI sheet or cockpit. A service user is a user that is available to a larger, anonymous user community. The service user should only have display authorization for the PI sheet or cockpit. The header of the PI sheet or cockpit always shows the name of the user currently logged on to the system. To switch from the service user to the individual user, choose

Log On. If you choose *Log Off*, the system switches back to the service user.

Scenario

The scenario describes how a process operator shift logs on to the system using the service user and calls up the PI sheet or cockplit in display mode. If entries need to be made or functions need to be called in the PI sheet or cockpit, the corresponding process operator must log on using his or her individual user ID and password. The system then switches over to the change mode. The process operator can choose *Log Off* any time to log off again, for example, if he or she has completed processing. The system then automatically changes back to the service user and the PI sheet and cockpit cannot be changed anymore.

You can benefit from this function as follows:

- **Role-Specific Authorization Concept**
  You can, for example, create a role-specific authorization concept for PI sheet maintenance. A service user may, for example, only display the PI sheet, whereas a process operator may also change it. However, process steps may only be locked or unlocked by the shift manager.

- **Increased Security**
  Using such an authorization concept, you gain higher security since PI sheets can only be maintained with an individual user. If you have activated logging, the system logs who carried out which activities in the PI sheet. Thus you can, for example, track jumps to quality management (QM).
  For more information on logging in the PI sheet, see New Functions in the PI Sheet Context Menu.

- **Faster User Change**
  Since you do not need to log off from the SAP R/3 System when switching between the service user and the individual user, a user change is much faster.

Note the following for this function:

- You can only change users if the first logon to the SAP R/3 System was made with a service user. Otherwise, the function is not available.

- You cannot change between different languages.

- You cannot change between different clients.

19.4.4.8.2 Asynchronous Signature Process for Process Steps in the PI Sheet (Changed)

Use

Until now, you could define an asynchronous signature only at the end of a PI sheet. This last asynchronous signature also automatically completes the PI sheet.

As of *SAP R/3 Enterprise SCM Extension 1.10 (EA_APPL 110)*, you can define a signature strategy to be executed asynchronously for every process step. However, only the last asynchronous signature in the PI sheet triggers the *Complete PI Sheet* function. Using an asynchronous signature, actual data cannot be reported until one or more additional persons have confirmed the entries by executing their signatures.

Example

In the course of production, the process operator enters values and executes his or her signature to sign

these entries. However, the system does not yet create a process message since the entries must be checked and signed again by the shift manager. In the meantime, the process operator can maintain another section in the PI sheet. Only the input field that still needs to be signed is still locked for further entries until the shift manager has signed.
If the values entered by the process operator are not allowed, the shift manager can cancel the signature process. The input fields in the PI sheet become ready for input again. New values can be entered. The process messages belonging to the process step are not created until all required signatures of the signature strategy have been executed.

19.4.4.8.3 New Functions in the PI Sheet Context Menu (Enhanced)

Use

New Deactivate Process Step and Activate Process Step Functions

Until now, you could only activate or deactivate process instructions in the PI sheet. As of *SAP R/3 Enterprise SCM Extension 1.10 (EA_APPL 110)*, you can deactivate and activate entire process steps, this means none of the input fields and function keys belonging to a process step are ready for input or executable after deactivation.

Deactivated process steps are not taken into account when the PI sheet is checked. Neither are variables contained in the process step. In calculations, for example, values in deactivated process steps are not included in the calculation.

To execute this function, the user needs the authorization for *Activity 02 - Change* in the authorization object C_CRPI_BER.

In Customizing for Control Recipe Destinations, you can specify that this function is to be logged whenever it is executed. The log records functions both executed manually by the user and executed automatically by the system.
For more information, see the section *Logging*.

Lock and Unlock Functions

In this release, you can lock and unlock certain areas (for example, process steps, process instructions) in the PI sheet. Locked areas are no longer ready for input or executable. You cannot complete the PI sheet when these areas are locked. Deactivated areas, by contrast, are not taken into account when the PI sheet is completed.
These functions are available in the context menu of the PI sheet. They can be executed for the following areas:

- Process steps
- Process instructions
- Tables
- Table lines

To execute these functions, the user needs the authorization for *Activity 05 - Lock* in the authorization object C_CRPI_BER. For more information on the authorization concept for the PI sheet, see User Change in PI Sheets and Manufacturing Cockpits.

In Customizing for Control Recipe Destinations, you can specify that these functions are logged

whenever they are executed. The log records functions both executed manually by the user and executed automatically by the system.
For more information, see the section *Logging.*

Display Properties

As of this release, you can display the properties for every input field in the PI sheet in a dialog box. To display properties, you choose *Display Properties* in the context menu. Depending on the actions made at an input field, the maximum content of the *Properties* dialog box can be as follows:

- **Position**
  Specifies, for example, the phase, process instruction and characteristic of the input field.
- **Log**
  Contains all values entered, the date and time when the values were entered as well as the terminal number and user ID of the user logged on. Both rejected and accepted values are logged.
  For more information, see the section *Logging.*
- **Deviations**
  Lists all deviations that occurred during value entry. The value entered, the validation rule defined and a note about whether the value was rejected or accepted are displayed for every deviation.
- **Comments**
  Contains all comments, the date and time when the comments were entered, and the user ID of the user logged on for the input field.
- **Signatures**
  Lists all digital signatures executed in input fields. The name of the signatory, the date and time when the signature was executed, and a comment that the signatory may have entered are displayed for every signature.

Logging

In this release, user actions are logged in the PI sheet. When you choose *Display Properties* in the context menu, you can display the logged data in the *Properties* dialog box on the *Log* tab.
The system logs the following events:

- When a user enters data.
- When a user carries out the *Activate* or *Deactivate* function.
- When the system automatically carries out the *Activate* or *Deactivate* function.
- When the user carries out the *Lock* or *Unlock* function.
- When the systen automatically carries out the *Lock* or *Unlock* function.
- When the user completes a process step or a process instruction.

On the *Log* tab, a list with all entries that were logged is displayed. The following is recorded for each event:

- Date and time when the event was executed
- A text describing which event was executed
- The description of the terminal at which the function was carried out
- The user ID of the user logged on

- o If a signature was required to confirm the entry, the user ID of the signatory as well as the date and time when the signature was executed

Effects on Customizing

Logging

In Customizing, you can specify whether and which events are to be logged. In addition, you can determine for each event whether a signature is required to confirm the event to be logged. If the signature is canceled, the action just triggered is also canceled (for example, the process instruction will not be activated).

To do so, go to Customizing for *Production Planning - Process Industries* and choose *Process Management* -> *Control Recipes/PI Sheets* -> *Control Recipe Destinations* -> Define and Set Up Control Recipe Destinations.

19.4.4.8.4 Automatic Execution of Functions in PI Sheets and Manufacturing Cockpits (New)

Use

Until now, you could execute a number of functions, such as *Deactivate Process Instruction* and *Add Table Line*, manually in the PI sheet via the context menu. As of *SAP R/3 Enterprise SCM Extension 1.10 (EA_APPL 110)*, you can define conditions in the master recipe or process order to trigger the execution of these functions automatically.

Example:
You want to specify, for example, that the quantity entry of a certain material depends on the current pH value of the mixture in the vessel. This means that the material quantity may not be entered until the process operator has measured a pH value of less than 7 (< 7).

Using this function, you can define this condition in the process order or master recipe. When the PI sheet is opened, the input field for the material quantity entry would first be locked. The lock at the input field for material quantities would only be removed after a pH value of less than seven was entered in the previous input field, and the material quantity field would become ready for input.

Functions that can Be Executed Automatically

The following functions can be executed automatically by conditions:

- *Save Document*
- *Document Report Data*
- *Activate Process Step*
- *Deactivate Process Step*
- *Lock Process Step*
- *Unlock Process Step*
- *Complete Process Step*
- *Activate Process Instruction*
- *Deactivate Process Instruction*
- *Lock Process Instruction*
- *Unlock Process Instruction*
- *Complete Process Instruction*

- *Activate Table*
- *Deactivate Table*
- *Lock Table*
- *Unlock Table*
- *Complete Table*
- *Add Table Line*
- *Activate Table Line*
- *Deactivate Table Line*
- *Lock Table Line*
- *Unlock Table Line*
- *Complete Table Line*

New Characteristics

The following new characteristics have been introduced to be able to formulate these functions in the process order, master recipe, or cockpit definition:

Characteristic	***Meaning***
PPPI_COMMAND	Specifies the function that is to be executed, shown, or hidden when the condition is fulfilled
PPPI_ACTION	Specifies whether the function is to be shown or hidden in the context menu. This characteristic is optional. If you do not define this characteristic, the system uses the *Execute* option by default.
PPPI_FORMULA	Specifies the calculation formula in which the conditions are defined. When the conditions are fulfilled, the function is triggered.
PPPI_FUNCTION	Specifies the function module in which the conditions are defined. When the conditions are fulfilled, the function is triggered.
PPPI_EVENT	Specifies an internal event (for example *Process Instruction Was Locked*) or an external event (for example, OPC event). When this event occurs, the function is triggered.

Formulating Conditions

You can define conditions for the automatic execution of functions as follows:

- In a formula
- In a function module
- As a manufacturing event

In a Formula

In a formula, you can, for example, define that a certain area is deactivated when a value that was entered or calculated somewhere else in the PI sheet is above a certain limit.
In a second formula, you define that the same area is activated again when the value is below this limit. You use characteristic PPPI_FORMULA for this.

In a process instruction of type 0, you assign the following characteristics in the sequence specified:

Characteristic	**Characteristic Value**
PPPI_COMMAND when the condition is fulfilled	Function that is to be executed, shown, or hidden
PPPI_ACTION	Execute, show, or hide
PPPI_FORMULA	Formula

Example for the "Formula" Condition:

Process instruction 0020 is to be deactivated when the value entered in process instruction 0010 is greater than 100. At the same time, process instruction 0020 is to be activated again when the value is less or equal to 100.

Process Instruction 0010:

PPPI_INPUT_REQUEST	Enter quantity ABC:
PPPI_VARIABLE	VAR
PPPI_REQUESTED_VALUE	PPPI_MATERIAL_CONSUMED

Process Instruction 0020:

PPPI_INPUT_REQUEST	Enter quantity XYZ
PPPI_REQUESTED_VALUE	PPPI_MATERIAL_CONSUMED
PPPI_COMMAND	Deactivate Process Instruction
PPPI_ACTION	Execute
PPPI_FORMULA	VAR > 100
PPPI_COMMAND	Activate Process Instruction
PPPI_ACTION	Execute
PPPI_FORMULA	VAR =< 100

In a Function Module

You can write your own function module that contains one or more conditions. When the conditions are fulfilled, the corresponding function is triggered.
You use characteristic PPPI_FUNCTION for this. The syntax for characteristics definition is the same as that for dynamic function modules.

In a process instruction of type 0, you assign the following characteristics in the sequence specified:

Characteristic	***Characteristic Value***
PPPI_COMMAND when the condition is fulfilled	Function that is to be executed, shown, or hidden
PPPI_ACTION	Execute, show, or hide
PPPI_FUNCTION	Name of the function module
Parameter characteristic	Import or export parameter
Characteristic for variable/constant with the same data type as the parameter	Variable name

Example for the "Function Module" Condition:

Process instruction 0010: The pH value is to be entered for material CATALYST.

Process instruction 0020: Depending on the pH Value measured, the process operator is to add lye and record the quantity added. At the same time, function module CHECK_PHVAL is defined to check whether the pH value from process instruction 0010 is at least four. It also checks other factors such as the material quality by consulting data from material classification.
If the pH value is less than four (< 4) or other inspection criteria are not met, process instruction 0020 is deactivated and the process operator must not add lye.

Process Instruction 0010:

PPPI_DATA_REQUEST_TYPE	Simple Data Request
PPPI_VARIABLE	MY_MAT
PPPI_MATERIAL	KATALYSATOR
PPPI_OUTPUT_CHARACTERISTIC	PPPI_MATERIAL
PPPI_INPUT_REQUEST	Enter pH value
PPPI_VARIABLE	PHVALUE
PPPI_REQUESTED_VALUE	PH_VALUE

Process Instruction 0020:

PPPI_INPUT_REQUEST	Lye Quantity Added:
PPPI_VARIABLE	MY_MAT
PPPI_REQUESTED_VALUE	PPPI_MATERIAL_CONSUMED
PPPI_COMMAND	Deactivate Process Instruction
PPPI_ACTION	Execute
PPPI_FUNCTION	PHWERT_PRUEF
PPPI_EXPORT_PARAMETER	MATNR
PPPI_STRING_VARIABLE	MY_MAT
PPPI_EXPORT_PARAMETER	PHVALUE
PPPI_FLOAT_VARIABLE	MY_VALUE

As Manufacturing Events

You can specify an event that is to trigger the function automatically. You use characteristic PPPI_EVENT for this.

In the process instruction, you assign the following characteristics in the sequence specified:

Characteristic	Characteristic Value
PPPI_COMMAND when the condition is fulfilled	Function that is to be executed, shown, or hidden
PPPI_ACTION	Execute, show, or hide
PPPI_EVENT	Internal or external event

The system distinguishes between internal and external events. You can use the value help for characteristic PPPI_EVENT to select all events available.

- **Internal Events**
  Internal events are events that occur within the same PI sheet or cockpit in which the function is to be triggered.
  Examples: *Process Step Locked* or *User Logged On*

- **External Events**
  External events are events that occur in other applications. Applications that are registered as services at the internal interface *Manufacturing Data Access* can provide or receive events.
  At the moment, you can receive external events from the following applications in PI sheets and cockpits:

  - Other PI sheets (service PSH)
  - Other manufacturing cockpits (service PMC)
  - OPC Data Access servers (Service ODA)
  - OPC Alarms/Events servers (Service OAE)

  Addresses are used to access events. They start with the name of the service providing the event. You can use the value help for characteristic PPPI_EVENT to select these addresses.

Examples for Event Addresses:

- **ODA:0001.MYITEM.VCHD**
  Service ODA , plant 0001, OPC item MYITEM, value changed
- **OAE:0001.MYSUBSCRIPTION**
  Service OAE, plant 0001, OPC Alarms/Events subscription MYSUBSKRIPTION triggered
- **PMC:0001.MYCOCKPIT.VARCHGD**
  Service PMC, plant 0001, cockpit MYCOCKPIT, value changes to global variables occurred
- **PSH:100000000000004711.LOGIN**
  Service PSH, user logged on to maintain PI sheet **100000000000004711**
- **MDA:BROADCAST.DEMO**
  Service MDA, user-defined event **DEMO** triggered
  The data exchange between the different applications is realized internally by the *Manufacturing Data Access* interface.
  For more information, see
  Introduction of SAP OPC Data Access (SAP ODA)
  Data Exchange Between Applications of the SAP R/3 System

Example for the "Event" Condition:

In operation 0010, the quantity of material MAT003 charged into vessel 123 is to be entered and then confirmed by executing a signature.

In operation 0020, the quantity of material MAT005 charged into vessel 155 is to be entered and then confirmed by executing a signature. In addition, it is defined that the system is to lock the current process step as soon as the external event MY_SUBSCRIPTION of service OAE from plant 0001 occurs. This means that the section between the first and the second signature is locked since this is considered a process step.

Operation 0010:

PPPI_DATA_REQUEST_TYPE	Simple data request
PPPI_INPUT_REQUEST	Vessel 123:
PPPI_REQUESTED_VALUE	PPPI_MATERIAL_CONSUMED

PPPI_MATERIAL	MAT003
PPPI_OUTPUT_CHARACTERISTIC	PPPI_MATERIAL
PPPI_INPUT_REQUEST	Signature
PPPI_REQUESTED_VALUE	PPPI_SIGNATURE

Operation 0020:

PPPI_DATA_REQUEST_TYPE	Simple data request
PPPI_INPUT_REQUEST	Vessel 155:
PPPI_REQUESTED_VALUE	PPPI_MATERIAL_CONSUMED
PPPI_MATERIAL	MAT005
PPPI_OUTPUT_CHARACTERISTIC	PPPI_MATERIAL
PPPI_COMMAND	Lock Process Step
PPPI_ACTION	Execute
PPPI_EVENT	OAE:0001.MY_SUBSCRIPTION.EVT
PPPI_INPUT_REQUEST	Signature
PPPI_REQUESTED_VALUE	PPPI_SIGNATURE

19.4.4.8.5 Using Global Variables in PI Sheets and Manufacturing Cockpits (Enhanced)

Use

You can define variables in process management to transfer characteristic values from one process instruction to antother. The system distinguishes between simple and global variables. Until now, the following applied:

- **Simple variables** could only be transferred within one control recipe or one cockpit.
- **Global variables** could only be transferred between different control recipes of the same process order. Exchanging variable values between PI sheets and manufacturing cockpits as well as between cockpits was not possible either.

As of *SAP R/3 Enterprise SCM Extension 1.10 (EA_APPL 110)*, you can now use the new *Manufacturing Data Access* interface to also exchange global variables as follows:

- Between PI sheets of different process orders
- Between PI sheets and cockpits
- Between different cockpits

Global Variable Definition

Global variables are defined as before by assigning characteristic PPPI_GLOBAL_VARIABLE to the process instruction immediately before the message characteristic whose value you want to transfer.

Characteristic	**Value Assigned**
PPPI_GLOBAL_VARIABLE	Variable name
Message characteristic	Characteristics of the message category assigned

Using Global Variables

To use Manufacturing Data Access to work with global variables, you define the following in the process instruction:

Read Data

Characteristic	**Value Assigned**
PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_IMPORT_DATA	Data point or **global variable** of the application from which the value is to be transferred to the subsequent variable

Example:

PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_IMPORT_DATA	PMC:1100.MYCOCKPIT.**GINVAR**
PPPI_STRING_VARIABLE	VAR

Write Data

Characteristic	**Value Assigned**
PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_EXPORT_DATA	Data point or **global variable** of the application into which the value of the subsequent variable is to be transferred
Parameter characteristic	Variable names with the relevant data type

For more information, see Data Exchange Between Applications in the SAP R/3System.

Note that global variables that are transferred using Manufacturing Data Access may only be defined in *Process instructions of type 0: universal process instruction category*.

Example

This section serves to illustrate the following functions:

o Data exchange between a PI sheet and a cockpit using Manufacturing Data Access

o Reading of OPC items from a cockpit using SAP ODA and Manufacturing Data Access

o Using of internal and external manufacturing events to carry out functions automatically in the PI sheet

o Using of global variables that are transferred as data points using Manufacturing Data Access

The PI sheet layout is defined in such a way that a cockpit is integrated in it in a frame. The cockpit serves to automatically read a value from the PI sheet and the value of an OPC item and add both values.

The PI sheet contains one input field. As soon as a value is entered in this field, it is automatically transferred to the cockpit. As soon as a variable changes in the cockpit, the system carries out the calculation automatically. The result of this calculation is automatically returned to the PI sheet and at the same time displayed in the cockpit. In this way, the reading of OPC items and the calculation function are independent from the PI sheet.

You define the following in the **master recipe or process order**:

- The PI sheet is to contain an input field to which variable VAR1 refers.
- When the variable VAR1 changes in the PI sheet, its value is to be transferred to the cockpit, to the global variable GINVAR. Using characteristic PPPI_EVENT, you specify that the value is to be transferred to the cockpit with the internal event *Parameter Changed*.
- When a variable changes in the cockpit MYCOCKPIT, the changed value is to be returned to the PI sheet, via the global variable GOUTVAR of the cockpit. This value is transferred to the PI sheet via the import parameter PMC:1100:MYCOCKPIT.GOUTVAR. For this, you specify the external event PMC:1100.MYCOCKPIT.VARCHGD so that the value can be transferred automatically.
- The imported value of global variable GOUTVAR from the cockpit is to be imported to the local variable VAR2 of the PI sheet and displayed there.

To do so, you define the following characteristics in a process instruction of type 0:

Characteristic	Characteristic Value
PPPI_INPUT_REQUEST	Value entry
PPPI_VARIABLE	VAR1
PPPI_REQUESTED_VALUE	PPPI_MATERIAL_QUANTITY
PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_BUTTON_TEXT	Send value to cockpit
PPPI_FUNCTION_DURING_DISPLAY	Permitted
PPPI_EVENT	Parameter Changed
PPPI_EXPORT_DATA	PMC:0001.MYCOCKPIT.GINVAR
PPPI_STRING_VARIABLE	VAR1
PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_BUTTON_TEXT	Read value from cockpit
PPPI_FUNCTION_DURING_DISPLAY	Permitted
PPPI_EVENT	PMC:0001.MYCOCKPIT.VARCHGD
PPPI_IMPORT_DATA	PMC:0001.MYCOCKPIT.GOUTVAR
PPPI_STRING_VARIABLE	VAR2
PPPI_OUTPUT_TEXT	Value calculated
PPPI_OUTPUT_VARIABLE	VAR2

In the **Cockpit** MYCOCKPIT, you define the following:

- You declare the global variable GINVAR and output the value of variable GINVAR.
- Using Manufacturing Data Access, the current value of the OPC Data Access item MYODA is to be read. You define the automatic reading operation by assigning the value

ODA:0001.MYODA.VCHD to PPPI_EVENT. In addition, the OPC value is also to be read on request. For this, you specify PPPI_BUTTON_TEXT. The OPC value is to be displayed in the cockpit.

- When a variable changes in the cockpit, a calculation is to be carried out automatically. For this, you use the internal event *Parameter Changed*. The value of global variable GINVAR and the current value of the OPC item are to be added.
  Before the calculation, you define the global variable GOUTVAR so that the value can be output later.
- The result of the calculation (global variable GOUTVAR) is to be displayed in the cockpit.

To do so, you define the following characteristics in a process instruction of type 0:

Characteristic	Characteristic Value
PPPI_GLOBAL_VARIABLE	GINVAR
PPPI_MATERIAL_CONSUMED	0
PPPI_OUTPUT_TEXT	Value from PI sheet
PPPI_OUTPUT_VARIABLE	GINVAR
PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_BUTTON_TEXT	Read OPC item
PPPI_FUNCTION_DURING_DISPLAY	Permitted
PPPI_EVENT	ODA:0001.MYODA.VCHD
PPPI_IMPORT_DATA	ODA:0001.MYODA.PV
PPPI_FLOAT_VARIABLE	ODAVAR1
PPPI_OUTPUT_VARIABLE	ODAVAR1
PPPI_INPUT_REQUEST	Value from PI sheet + OPC value
PPPI_GLOBAL_VARIABLE	GOUTVAR
PPPI_CALCULATED_VALUE	PPPI_MATERIAL_CONSUMED
PPPI_EVENT	Parameter Changed
PPPI_CALCULATION_FORMULA	GINVAR + ODAVAR1
PPPI_OUTPUT_TEXT	Value calculated
PPPI_OUTPUT_VARIABLE	GOUTVAR

19.4.4.8.6 Automatic Execution of Calculations and Dynamic Function Calls (New)

Use

You can execute calculations and dynamic function calls in the PI sheet and manufacturing cockpit by choosing the relevant pushbutton. As of *SAP R/3 Enterprise SCM Extension 1.10 (EA_APPL 110)* , these functions can be executed automatically by an event.

You define a manufacturing event that is to trigger this function in the master recipe or process order. You use characteristic PPPI_EVENT for this.

Executing Calculations Automatically

In the process instruction, you assign the following characteristics in the sequence specified:

Characteristic	**Characteristic Value**
PPPI_CALCULATED_VALUE	Characteristic to be calculated
PPPI_EVENT	Internal or external event
PPPI_CALCULATION_FORMULA	Calculation formula

Example of an automatic calculation

The input field "Value" contains default value 25. Variable MYVAR refers to this input field. Value *Parameter Changed* was defined as the triggering event. This means that once the input value changes, the calculation function is executed automatically and the result of the calculation is updated.

Characteristic	**Characteristic Value**
PPPI_INPUT_REQUEST	Value
PPPI_DEFAULT_VALUE	25
PPPI_VARIABLE	MYVAR
PPPI_INPUT_REQUEST	Result
PPPI_VARIABLE	MYRESULT
PPPI_CALCULATED_VALUE	PPPI_MATERIAL_CONSUMED
PPPI_EVENT	Parameter Changed
PPPI_CALCULATION_FORMULA	MYVAR + 3

Executing Dynamic Function Calls Automatically

In the process instruction, you assign the following characteristics in the sequence specified:

Characteristic	**Characteristic Value**
PPPI_FUNCTION_NAME	Name of function module
PPPI_EVENT	Internal or external event
Parameter characteristic	Import or export parameter
Characteristic for variable/constant with the data type of the parameter	Variable name

Example of Function Executed Automatically

The input field "Date" contains the current date as the default value. Variable MYDATE refers to this input field. A function module has been defined that is to calculate the day from the date when the corresponding button is pressed. In addition, a triggering event with the value *Process Instruction Locked* was defined, which means that once the process instruction is locked, the system calculates the day automatically and displays it.

Characteristic	**Characteristic Value**
PPPI_INPUT_REQUEST	Date
PPPI_VARIABLE	MYDATE
PPPI_REQUESTED_VALUE	PPPI_EVENT_DATE

PPPI_FUNCTION_NAME	DATE_TO_DAY
PPPI_FUNCTION_DURING_DISPLAY	Permitted
PPPI_EVENT	Process Instruction Locked
PPPI_EXPORT_PARAMETER	DATE
PPPI_DATE_VARIABLE	MYDATE
PPPI_IMPORT_PARAMETER	WEEKDAY
PPPI_STRING_VARIABLE	MYWEEKDAY
PPPI_OUTPUT_TEXT	Day
PPPI_OUTPUT_VARIABLE	MYWEEKDAY

19.4.4.8.7 Determining Control Recipe Numbers (New)

Use

As of *SAP R/3 Enterprise SCM Extension 1.10 (EA_APPL 110)*, you can use the internal interface *Manufacturing Data Access* to exchange manufacturing data points and manufacturing events between the PI sheets of the same process order.

To access data points and events from PI sheets, you must specify the number of the corresponding PI sheet when you define the data access in the master recipe or process order. Since the corresponding control recipe has usually not been sent by this time, you do not yet know the number of the PI sheet.

You can use the following new characteristics to access data points and events from PI sheets whose numbers you do not know. The control recipe number is determined through the control recipe destination specified. You specify a text symbol in the event or data point address, which is then replaced by the number of the control recipe when the recipe has been created.

Characteristic	**Meaning**
PPPI_OTHER_DESTINATION	Destination name of the control recipe from which the data point or event is to come
PPPI_OTHER_CONTROL_RECIPE	Control recipe number for the destination specified. The value of this characteristic is assigned automatically.

In a process instruction of *Type 0: Universal process instruction category*, you define the following characteristics in the sequence specified:

Read Data Point

Characteristic	**Value Assigned**
PPPI_OTHER_DESTINATION	Destination address of the control recipe from which the data point or event is to come
PPPI_OTHER_CONTROL_RECIPE	Automatically
PPPI_DATA_ACCESS	Manufacturing Data Access
PPPI_IMPORT_DATA	Data point or global variable of the application from which the value is to be transferred to the subsequent variable
Parameter characteristic	Variable names with the corresponding data type

Example

In the PI sheet for control recipe destination N1, you want to read variable MYVAR from the PI sheet for control recipe destination N2. You assign N2 to PPPI_OTHER_DESTINATION. In the data point address, you define text symbol *&0020&* instead of the PI sheet number. When the control recipe for destination N2 is created, the system replaces the text symbol &0020& by the value that is assigned to characteristic number 0020.

Char.No.	Characteristic	Value Assigned
0010	PPPI_OTHER_DESTINATION	N2
0020	PPPI_OTHER_CONTROL_RECIPE	*Automatically*
0030	PPPI_DATA_ACCESS	Manufacturing Data Access
0040	PPPI_BUTTON_TEXT	Number of the PI sheet
0050	PPPI_FUNCTION_DURING_DISPLAY	Permitted
0060	PPPI_IMPORT_DATA	PSH:&0020&.MYVAR
0070	PPPI_FLOAT_VARIABLE	VAR
0080	PPPI_OUTPUT_VARIABLE	VAR

For more information on *Manufacturing Data Access* and how to define the data access, see

- Data Exchange Between Applications of the SAP R/3 System
- Service PSH

19.4.4.8.8 Intergrating DMS Documents to the PI Sheet (New)

Use

Until now, you could call up production-relevant documents that were stored in the *SAP Document Management Service (DMS)* in the PI sheet by means of a dynamic function call. The document displayed was not archived in the batch record.

As of S*AP R/3 Enterprise SCM Extension 1.10 (EA_APPL 110)*, you can use the following enhancements to this function:

- You can directly display the DMS document in the PI sheet in an embedded frame or as a hyperlink.
- The system can determine and display the valid, released document version automatically.
- The documents are archived in the batch record together with the PI sheet when they can be read in an embedded frame.

Creating Documents in DMS

To create a document in *DMS* from the *SAP Easy Access* screen, you choose *Logistics* -> *Central Functions* -> *Document Management* -> *Document* -> *Create*.

Prerequisites

To display a document in the PI sheet, the following requirements must be met:

- The deletion indicator must not be set for the document.

- The document version specified must be valid and released.
- Only one original application file with the specified workstation application may be assigned to the document.
- The original application file to be displayed must be active and stored in the secure storage area.

Linking Documents to PI Sheets

1. In the master recipe or process order, you assign a process instruction of type 0 with a long text characteristic, for example PPPI_INSTRUCTION.
2. In the long text characteristic, you define an HTML page in which you address the document via a URL. Use the paragraph format HTML for this.
   If the document is to be displayed within the PI sheet and archived together with it, use an embedded frame. However, you can also call up the document in a separate window by using a hyperlink.

Structure of the Document URL

The URL that you use to address a document in the PI sheet must follow the following conventions:
SAPR3-DMS-<*Document number*>-<*Document type*>-<*Document part*>-<*Document version*>-<*Workstation application of the original*>

Example

You want to link a document with the following data to the PI sheet:

Dokument number:	DOC1
Document type:	PI
Document part:	001
Version:	03
Workstation application:	WWW
Additional files:	None

To link the document to an embedded frame, you define, for example, the following long text in your process instruction: Note the <a href="SAPR3-DMS-DOC1-PI-001-03-WWW" scrolling=yes>master formula </a>.

19.4.4.9 PP-PI-PMA-OPC OPC Interface

19.4.4.9.1 Introduction of SAP OPC Data Access (SAP ODA) (New)

Use

As of *SAP R/3 Enterprise SCM Extension 1.10 (EA_APPL 110)*, the new SAP OPC Data Access (SAP ODA) component is available. It supports data exchange between SAP R/3 and process control systems from different vendors. You can use SAP ODA to receive data and events from the process control level in the SAP R/3 System automatically or on request, and send data back.

SAP ODA is based on the OPC (OLE for Process Control) industry standard that defines vendor-independent interfaces for process control engineering. OPC servers provide process data that can be requested by OPC clients. SAP R/3 is an OPC client that communicates with OPC servers. One of the components of SAP ODA is the SAP ODA Connector, which is installed on a PC outside of the SAP System.

No additional programming or middleware is required to use SAP ODA. Unlike the existing PI-PCS interface, no certification of the individual process control sytems by SAP is required with SAP ODA. Thus, the installation and configuration effort is negligible.

Possible OPC Server Functions in SAP R/3

At present, SAP ODA supports the following OPC server functions:

- OPC Data Access - this means:
    - Synchronous reading and writing of values from and to any OPC Data Access server
    - Subscription to value changes from OPC Data Access servers
- OPC Alarms/Events - this means:
  Subscription to events from OPC Alarms/Events servers and acknowledgement of events

The OPC functions supported by SAP ODA represent a general interface that is available for all SAP applications. At the moment, the interface is used by *Production Planning - Process Industries (PP-PI)*.

Integration with the PI Sheet

In the *Production Planning for Process Industries (PP-PI)* environment, OPC items and OPC events can be integrated with the PI sheet and manufacturing cockpit . From there, you can request actual data from automated process parts interactively and write values to process control systems. By reading and writing values automatically, errors that may occur during manual entry are reduced. You can, for example, read temperature values of a production vessel and output them in the PI sheet. You can report a defective production vessel as an OPC event in the PI sheet. The process operator who maintains the PI sheet and has read the alarm, can acknowledge it by executing his or her signature in the PI sheet.

For more information, see:

- Data Exchange Between Applications in the SAP R/3 System
- Automatic Process Message Creation
- Automatic Execution of Functions in PI Sheets and Cockpits

Effects on Customizing

To use the functions of SAP ODA successfully, you must make the following settings in Customizing of the SAP R/3 System:

1. You install the SAP ODA Connector on the target host.
   Note that SAP ODA only supports COM mechanisms but not DCOM mechanisms. Therefore, the OPC server and the SAP ODA Connector must be installed on the same host. However, you can install any number of SAP ODA Connectors on various hosts.
2. In the SAP R/3 System, you define the RFC destination of the SAP ODA Connector that you have installed on the target host.
   Define RFC Destination for SAP ODA
3. In the SAP R/3 System, you make the basic settings required for communication between the SAP ODA Connector and the SAP R/3 System.

Define Basic Settings for SAP ODA
Customizing activity Define Settings for SAP ODA

4. In the SAP R/3 System, you define the names for the OPC servers that the SAP ODA Connector is to access. At the same time, you specify which of the following OPC server functions you want to use, provided that the relevant OPC server supports them:

   - Synchronous Data Access
   - Data Access Subscription
   - Alarms/Events Subscription

   Define OPC Servers
   Customizing activity Define Settings for SAP ODA

5. In the SAP R/3 System, you define the names for the OPC items that the SAP ODA Connector is to access.
   You can use the item names that you define here to read and write values in the following applications:
   Automatic Process Message Creation
   PI Sheet
   Manufacturing Cockpit

   Define OPC Data Access Items
   Customizing activity Define Settings for SAP ODA

6. You define in the SAP R/3 System that you want to subscribe to alarms and events from a lower-level system (for example, process control system) via an OPC Alarms/Events server. This means that an event is automatically reported to the SAP R/3 System as soon as it occurs in the process control system.
   You can use the subscription names defined here to subscribe to events in the following applications:
   Automatic Process Message Creation
   PI Sheet
   Manufacturing Cockpit

   Define OPC Alarms/Events Subscriptions
   Customizing activity Define Settings for SAP ODA