

Schedule Manager (CA)



HELP.COACCCHEDMAN

Release 4.6C



Copyright

© Copyright 2001 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft[®], WINDOWS[®], NT[®], EXCEL[®], Word[®], PowerPoint[®] and SQL Server[®] are registered trademarks of Microsoft Corporation.

IBM[®], DB2[®], OS/2[®], DB2/6000[®], Parallel Sysplex[®], MVS/ESA[®], RS/6000[®], AIX[®], S/390[®], AS/400[®], OS/390[®], and OS/400[®] are registered trademarks of IBM Corporation.

ORACLE[®] is a registered trademark of ORACLE Corporation.

INFORMIX[®]-OnLine for SAP and Informix[®] Dynamic Server[™] are registered trademarks of Informix Software Incorporated.

UNIX[®], X/Open[®], OSF/1[®], and Motif[®] are registered trademarks of the Open Group.

HTML, DHTML, XML, XHTML are trademarks or registered trademarks of W3C[®], World Wide Web Consortium, Massachusetts Institute of Technology.

JAVA[®] is a registered trademark of Sun Microsystems, Inc.

JAVASCRIPT[®] is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

SAP, SAP Logo, R/2, RIVA, R/3, ABAP, SAP ArchiveLink, SAP Business Workflow, WebFlow, SAP EarlyWatch, BAPI, SAPPHIRE, Management Cockpit, mySAP.com Logo and mySAP.com are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other products mentioned are trademarks or registered trademarks of their respective companies.

Icons

Icon	Meaning
	Caution
	Example
	Note
	Recommendation
	Syntax

Inhalt

Schedule Manager (CA)	6
Variables in Schedule Manager	7
Defining Global Variables for the Task List and Flow Definition	8
Defining Selection Variables	9
Specifying Selection Variables in the Program Variants	10
Specifying Selection Variables for Flow Definitions With Parallel Branches	12
Individual Functions of the Schedule Manager	14
Using the Scheduler	15
Task List.....	17
Task Types.....	18
Creating a Task List.....	19
Inserting Tasks/Task Groups into the Task List	20
Inserting Your Own Programs as Tasks	22
Inserting Your Own Programs as Tasks.....	27
Storing Task Documentation.....	29
Scheduling Tasks in the Daily Overview.....	30
Scheduling a Complete Task List in the Daily Overview	31
Relative Start Time	32
Runtime Analysis for Jobs.....	33
Controlling Whether Jobs Are Started	34
Using the Monitor	35
Monitor - Monitoring Active and Completed Jobs	37
Monitor - Working with the Object List	39
Using the Flow Definition	42
Flow Step Types	43
Creating a Flow Definition	44
Specifying Selection Variables for Flow Definitions With Parallel Branches	48
Choosing Objects for Processing.....	50
Defining Processing Options, Output Options and Execution Types.....	51
Error Handling	52
Error: A Job Stops.....	53
Error: A Job Becomes.....	55
Error: An Executed Report Found an Error	57
Error: Job Scheduling Was Unsuccessful.....	59
Error: The Workflow Stops.....	60
Error: The Workflow Becomes	61
Multilevel Worklist	62
Multilevel Worklist: Process Flow.....	75
Monitor - Working with the Object List	77
Processing Worklists.....	80
Choosing Objects for Processing.....	90
Processing Status of Objects.....	91
Scenarios for the Schedule Manager	97
Starting Transactions/Reports Online, Scheduling/ Monitoring Jobs	98

Starting Transactions/Reports Online, Scheduling Jobs/Job Chains, Monitoring Jobs..... 99
Starting Transactions/Reports Online, Scheduling Jobs and Job Chains, Worklist 100

Schedule Manager (CA)

The extensive automation features of the Schedule Manager facilitate the definition, scheduling, execution, and review of tasks that are executed on a regular basis, such as period-end closing.

The Schedule Manager consists of four independent components:

- [Flow Definition \[Seite 42\]](#)
- [Scheduler \[Seite 15\]](#)
- [Monitor \[Seite 35\]](#)
- [Worklist \[Seite 62\]](#)

Variables in Schedule Manager

By defining variables in Schedule Manager, you can minimize the amount of work that is required to make the necessary value changes (for example, Period) for tasks in the task list.

For more information, see:

[Defining Global Variables for the Task List and Flow Definition](#)

[\[Seite 8\]Defining Selection Variables \[Seite 9\]](#)

[Specifying Selection Variables in the Program Variants \[Seite 10\]](#)

[Specifying Selection Variables for Flow Definitions With Parallel Branches](#)

[\[Seite 48\]](#)

Defining Global Variables for the Task List and Flow Definition

Defining Global Variables for the Task List and Flow Definition

Use

Normally the selection criteria for period-end closing do not change very often. Criteria that do change regularly are the closing period and the fiscal year. The period and fiscal year must be changed for each program or flow definition specified as a task in the task list.

To avoid having to change these values for every single program or every flow definition, you can define selection variables in the program variants.

Features

You can define selection variables for single programs that are included as tasks in the task plan, as well as for flow definitions.

You can specify global parameters for the task list, such as the company code or profit center.

In Schedule Manager, the same flow definitions (= workflows) can be run in parallel, such as for different plants. The definition of the global variables defined for the main workflow can also be used for the parallel branches.

Activities

You can display the currently defined selection variables. To do so, choose *Extras* → *Settings* → *Selection variables*. You can also define new selection variables. You can then choose these as selection variables in the variable attributes when creating variants for single programs.

For more information, see:

[Defining Selection Variables \[Seite 9\]](#)

[Specifying Selection Variables in the Program Variants \[Seite 10\]](#)

[Specifying Selection Variables for Flow Definitions With Parallel Branches \[Seite 48\]](#)

Defining Selection Variables

Use

Normally the selection criteria for period-end closing do not change very often. Criteria that do change regularly are the closing period and the fiscal year. The period and fiscal year must be changed for each program or flow definition specified as a task in the task list.

To avoid having to change these values for every single program or every flow definition, you can define selection variables.

Prerequisites

You must assign a TVARV variable for the plant to each program that is to be run once for all plants.

Procedure

1. Call up the transaction **STVARV** (*Display table TVARV: Selection variables*).
2. Choose *Change*.
You can create, change and delete new variables.
3. Define a parameter name for your variant.
4. Call up the transaction **SM34**.
5. Go to view cluster **VSMANTVARV**.
6. Specify the same parameter name for the variable as you did in transaction **STVARV**.



Changes to the global selection variables specified for the task list are transferred to all variables specified in tasks in the task list.

Result

You defined a global selection variable for Schedule Manager.

For more information, see:

[Defining Selection Variables in the Program Variants \[Seite 10\]](#)

[Specifying Selection Variables for Flow Definitions With Parallel Branches](#)

[\[Seite 48\]](#)

Specifying Selection Variables in the Program Variants

Specifying Selection Variables in the Program Variants

Use

Normally the selection criteria for period-end closing do not change very often. Criteria that do change regularly are the closing period and the fiscal year. The period and fiscal year must be changed for each program or flow definition specified as a task in the task list.

To avoid having to change these values for every single program or every flow definition, you can define selection variables in the program variants.

Prerequisites

You have defined the variables already. For more information, see [Defining Selection Variables \[Seite 9\]](#).

You are in [flow definition \[Extern\]](#) in Schedule Manager (*Extras* → *Flow definition* → *Edit flow definition*).

1. Choose a processing step in the navigation area.
On the right of the screen, you see *Flow definition: Task details*.
2. Enter a new variant for this processing step and define a name for this new variant in the *Variant* field .
3. Choose *Change variant*.
The *Variant Maintenance* screen appears. *Program* <program name>, *variant* <variant name>.

Procedure

4. Choose *Attributes*.
5. Make an entry in the *Description* field.
6. Now you want to specify variables for the period and fiscal year that you can then change once centrally for all programs and flow definitions that use those variables. To do this, under the heading *Selection screen objects*, select the column S in the lines *Period* and *Fiscal year*.
7. Choose *Selection variables*.
8. To choose selection criteria for the variables P_FROM (period) and P_GJAHR (fiscal year), use the input help.



You previously entered these variables in table TVARV.

9. Save your entries.
The *Variant Maintenance* screen appears. *Program* <program name>, *variant* <variant name>.

Result

You can no longer make entries in the *Period* and *Fiscal year* fields. These fields are now always filled through the current entries in table TVARV.

Specifying Selection Variables in the Program Variants

Specifying Selection Variables for Flow Definitions With Parallel Branches

Specifying Selection Variables for Flow Definitions With Parallel Branches

Use

Normally the selection criteria for period-end closing do not change very often. Criteria that do change regularly are the closing period and the fiscal year. The period and fiscal year must be changed for each program or flow definition specified as a task in the task list.

To avoid having to change these values for every single program or every flow definition, you can define selection variables in the program variants.



You have five programs that must run with the same valuation for nine plants. That means that every program must run with a plant-specific variant for each plant. These variants differ only in their *Plant* specification.

As described below, you define global variables for *fiscal year*, *period* and *plant*. Create variants for the five programs and define the global variants that you created previously. Finally define a flow definition (A) for the five programs. Define a flow definition (B) with nine parallel branches (for the plants). Include the flow definition A in each of these branches. A dialog box appears that displays the three global variables (*fiscal year*, *period*, *plant*). Enter a value for the variable *Plant* and flag the entry as to be saved. The dialog box reappears when you plan flow definition B in the day view. Enter values for each fiscal year and period. Do not make an entry for the variable *Plant*. If you make an entry here, the system does not accept it. You defined the plant when creating the flow definition.

Prerequisites

You have defined the variables already. For more information, see [Defining Selection Variables \[Seite 9\]](#).

You are in [flow definition \[Extern\]](#) in Schedule Manager (*Extras* → *Flow definition* → *Edit flow definition*).

4. Choose a processing step in the navigation area.
On the right of the screen, you see *Flow definition: Task details*.
5. Enter a new variant for this processing step and define a name for this new variant in the *Variant* field.
6. Choose *Change variant*.
The *Maintain Variant: Program <program name>, Variant <variant name>* screen appears.

Procedure

9. Choose *Attributes*.
10. Make an entry in the *Description* field.
11. Now you want to specify variables for the period and fiscal year that you can then change once centrally for all programs and flow definitions that use those variables. To do this, under the heading *Selection screen objects*, select the column S in the lines *Period* and *Fiscal year*.

Specifying Selection Variables for Flow Definitions With Parallel Branches

12. Choose *Selection variables*.
13. To choose selection criteria for the variables P_FROM (period) and P_GJAHR (fiscal year), use the input help.



You previously entered these variables in table TVARV.

10. Save your entries.
The *Maintain Variant: Program <program name>, Variant <variant name>* screen appears.

Result

The system uses these entries for the following purposes:

- To feed these variables to a program created directly as a task in the workflow.
- To feed these variables to the parallel branches.



The system mixes the variables of the parallel branches with those of the main workflow.

If the parameter values of the main workflow are different from those of the subworkflows, the system uses the parameter values that were specified when the subworkflows were created.

Individual Functions of the Schedule Manager

Individual Functions of the Schedule Manager

Use

A number of periodic tasks are executed on a regular basis (daily, weekly, or monthly) in the SAP System. An example of such a task is period-end closing. This requires the processing of a large number of individual objects at certain times. This process is supported by the individual components of the Schedule Manager.

Features

Flow definition

In a flow definition, you can link [tasks \[Extern\]](#) to each other if they are related or if you wish to use a worklist in them. You can therefore schedule a flow definition as a task in the scheduler.

See also [Using the Flow Definition \[Seite 42\]](#) in the SAP Library.

Scheduler

In the scheduler, you can schedule tasks in a structure tree. You can use *drag-and-drop* in a daily overview to enable the system to execute the tasks at a certain time.

See also [Using the Scheduler \[Seite 15\]](#) in the SAP Library.

Monitor

The monitor gives you an overview of the scheduled tasks during and after processing. You can correct faulty objects in a worklist.

See also [Using the Monitor \[Seite 35\]](#) in the SAP Library.

Worklist

Objects that are to be processed in a processing step sequence are managed in the worklist.

The worklist monitor presents information such as which objects were processed without errors and which objects could not be processed. You can display information on the cause of errors, and thus control the way in which the object is processed further.

The worklist ensures that when a processing step sequence is processed again, the system only processes the objects which had errors or which you manually instructed the system to reprocess. Define the processing step sequence in the flow definition.

See also [Multilevel Worklist \[Seite 62\]](#) in the SAP Library.

Using the Scheduler

Use

In the scheduler, you can execute and monitor complex business flows, for example, period-end closing. You can define task lists if you have the corresponding authorization.

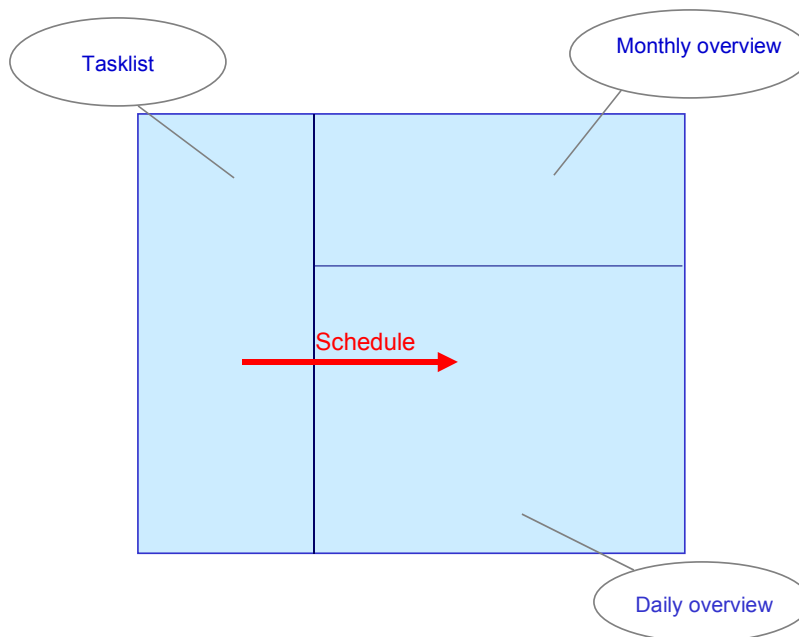
Integration

- You can group tasks that are to be executed in the background in a particular order, (and which are controlled by the workflow) into a [flow definition \[Seite 40\]](#).
- You monitor flows and jobs during and after processing in the [monitor \[Seite 33\]](#)
- Use the multilevel [worklist \[Seite 60\]](#) to improve performance and reduce error processing.

Prerequisites

Before using the scheduler, you need to create a task list in which you can later insert the tasks to be scheduled.

Features



The scheduler is divided up into three areas:

- **Task list**

Tasks structured into task groups in chronological order, which are executed periodically, possibly by more than one user to complete a certain process.

Individual Functions of the Schedule Manager

- **Monthly overview**

An overview of the current and previous month.

You can select a day from the monthly overview, which the system then displays in the daily overview in detail.

- **Daily overview**

Overview of the tasks to be done during the day.

Tasks created in the task list are scheduled in the daily overview. The system displays executed tasks in the daily overview with the time that they started.

Activities

Create a task list.

Schedule tasks in the daily overview and if required, have the system execute them.

Task List

Definition

A structured group of tasks, which are executed periodically, and possibly by more than one user to complete a certain process, such as, period-end closing.

Use

The task list enables you to schedule your process (split into tasks) in the daily overview ([Scheduling in the Daily Overview \[Seite 30\]](#)).

Structure

The task list is displayed as a structure tree, into which you can insert different [task types \[Seite 18\]](#).

Integration

The [Schedule Manager \[Extern\]](#) consists of the daily overview and the monthly overview.

Task Types

Task Types

You can use the following types of task in the task plan:

- Job chains executed in the background (= flow definition)
- Individual jobs executed in the background (= program with variant)
- Programs or transactions executed online.
- Notes as placeholder to describe a task that you do not process in the SAP System (such as "inform Mr. X")



It is only programs with variants that can be scheduled in the daily view and the system executes them automatically. The system **cannot** start programs and transactions. You can start these manually from the task list. To do so, use the right mouse button to choose *Execute*.

See also:

[Inserting Your Own Programs into the Task List \[Seite 27\]](#)

Creating a Task List

Use

The [task list \[Seite 17\]](#) forms the basis of the [scheduler \[Seite 15\]](#). This is where you arrange tasks or task groups into a chronological structure that represents an entire process (such as period-end closing).

Prerequisites

You have divided the process to be displayed into its components (programs, transactions, and jobs).

Procedure

1. Choose *Task list* → *Create*.
2. Enter a name for the new task list.
3. Choose *Insert*.
A detail screen appears.
4. Enter a description for the new task list.
5. Specify the display format for the daily and monthly overviews.
6. Save your entries.

Result

You have created a task list in which you can now insert tasks.

See also:

[Creating New Tasks/Task Groups in the Task List \[Seite 20\]](#)



To create and process a copy of the SAP task list in your own namespace, choose *Task list* → *Save*.

Inserting Tasks/Task Groups into the Task List

Inserting Tasks/Task Groups into the Task List

Use

To complete a process in Schedule Manager, you need to split the process into its parts (programs, transactions, jobs). You can then insert, delete and reorganize the tasks or task groups (tasks grouped according to content or time-based criteria) chronologically in the task list.

Prerequisites

You [create a task list \[Seite 19\]](#).

Procedure

Inserting Tasks/Task Groups in the Task List

1. Choose *Change task list*.
2. Select the highest node under which the new task/task group should appear.
3. Use a right mouse click to choose *Insert task*.
You can enter a different description for the task/task group and also choose the task type.
You can specify the person responsible for the task.
4. Choose *Insert*.
5. Save the changes.
The task types are displayed by various symbols in the structure tree of the task list.

Using Existing Tasks as Templates for New Tasks

1. Use a right mouse click to select the task that you want to use as the template for the new task.
2. Choose *Copy*.
3. Position your cursor on the node under which the new task is to be assigned.
4. Use a right mouse click to choose *Insert*.
5. Save your changes to the task list.

Changing or Deleting Tasks

1. Choose *Change task list*.
2. Use a right mouse click to select the task to be checked/deleted.
3. Choose *Change* or *Delete*.
4. Save the changes.

Reorganizing Tasks in the Task List

1. You can use *drag-and-drop* to move a task to another position in the task list.
2. Save the changes.

Inserting Tasks/Task Groups into the Task List**Result**

You can now process the task plan by scheduling the tasks in the daily overview.



Note that you can enter a [relative start time \[Seite 32\]](#) when creating a task. You require this start time, if you want to [schedule a complete task list \[Seite 31\]](#). To prevent tasks from being started at weekends or public holidays, you can restrict start dates to calendar days and weekdays.

Inserting Your Own Programs as Tasks

Inserting Your Own Programs as Tasks

Use

You can include your own transaction directly in the task plan without having to make changes to the transaction.

If you run your own program periodically, you can extend the program so that it can be used in flow definitions (Workflows) and you can see the processing status in the monitor.

For the program to use the worklist, the programming guidelines for the job monitor and the flow definition must be met. You must carry out further program extensions.

Procedure

Connecting Your Own Program to the Monitor

The program **CUSTOMER_REPORT** is available in the system as standard. This shows you how to call two function modules that enable the easiest connection to the monitor (**without** a workflow connection).

Incorporate the two Includes **SCHEDMAN_INIT** and **SCHEDMAN_CLOSE** in your program. The program is connected to the Job Monitor.

Utilizing Your Program for Flow Definition

If you also want to utilize your program in flow definitions, you must incorporate the Include **RKASMAWF** in the selection screen.

At the beginning of processing, install the following coding **instead of** the Include **SCHEDMAN_INIT** mentioned above:

```
data: gs_key      like schedman_key.

data: ls_detail like schedman_detail_user.

data: ls_appl    like schedman_customer.

data: ld_dummy(20) value 'ABCDEFGHijkl'.

constants: customer_appl like smmain-application value 'CUSTOMER'

      ls_detail-application = customer_appl.

      ls_detail-repid       = sy-repid.

*   ls_detail-testflag     = true.      "Decide if testrun or not
```

Inserting Your Own Programs as Tasks

```
ls_appl-customer_field = ld_dummy.
```

```
CALL FUNCTION 'KPEP_MONI_INIT_RECORD'
```

```
EXPORTING
```

```
LS_DETAIL = ls_detail
```

```
* LS_WITEM =
```

```
LS_APPL = ls_appl
```

```
* LD_WORKLIST_FLAG = ' '
```

```
IMPORTING
```

```
LS_KEY = gs_key.
```

```
* TABLES
```

```
* LT_SELKRIT =
```

```
* LT_PARAM =
```

In the structure **SCHEDMAN_CUSTOMER**, you can define your own fields, which you fill with values at runtime.

At the end of program processing, install the following coding **instead of** the Include **SCHEDMAN_CLOSE**:

```
data: ld_aplstat like smmain-aplstat.
```

```
data: LS_SCMA_EVENT LIKE SCMA_EVENT.
```

```
constants: c_status_ok value '0'.
```

```
constants: c_status_undefined value '2'.
```

```
constants: c_status_error value '4'.
```

Inserting Your Own Programs as Tasks

```
constants: c_status_aborted          value 'A'.

include schedman_events.

*.decide the status you want to send

    ld_aplstat = c_status_ok.

* ld_aplstat = c_status_undefined.

* ld_aplstat = c_status_error.

* ld_aplstat = c_status_aborted.

*.If the report ended with error -> stop whole workflow. Otherwise

*.start the next job

    if ld_aplstat = '4' or ld_aplstat = 'A'.

        ls_scma_event-wf_event = cs_wf_events-error.

    else.

        ls_scma_event-wf_event = cs_wf_events-finished.

    endif.

*.the variables wf_witem and wf_okey

*.are from include RKASMAWF and are filled AUTOMATICALLY

*.fill them into strcuture ls_scma_event

    ls_scma_event-WF_WITEM = wf_witem.

    ls_scma_event-WF_OKEY  = wf_okey.
```

Inserting Your Own Programs as Tasks

```
CALL FUNCTION 'KPEP_MONI_CLOSE_RECORD'

EXPORTING

    LS_KEY          = gs_key.
*   LS_MESSAGE     =
*   LD_OBJECTS     =
*   LS_EXT         =
*   LS_RL          =

    LS_SCMA_EVENT = ls_scma_event

*   TABLES
*   LT_SPOOL      =

CHANGING

    LD_APLSTAT = ld_aplstat

EXCEPTIONS

    NO_ID_GIVEN = 1

    OTHERS      = 2.
```



For programs that you want to include in flow definitions, note the following:

- If it is a cross-application program, you need to copy the program that was already stored with an application, and then store the new program in the **SCMAPROG** table with the new application.

Enter the name of your program as a customer program in Schedule Manager. For more information, see [Inserting Your Own Programs as Tasks \[Seite 27\]](#).

You can include your own program in a flow definition using these settings, and see the results in the monitor.

Inserting Your Own Programs as Tasks

Inserting Your Own Programs as Tasks

Prerequisites

To add your own programs to the task list and schedule them, you must store them in the system and make sure the Schedule Manager knows how to access them.

Procedure

1. To do so, choose *Extras* → *Settings* → *Customer Programs*.
This brings you to the table *Registration of Customer Schedule Manager Programs*.
2. Choose *New entries*.
3. Enter the ABAP program name or search for it using the input help.
4. Use the input help to select an application.
You can enter either a **CUSTOMER** application or a standard application. If you use a standard application, you must replace the constant value **CUSTOMER_APPL** in the coding for **SCHEDMAN_INIT** with your chosen value.



```
constants: customer_appl like smmain-application value '*****'.
```

5. You can set the following indicators:

AVo (worklist)

Determines that the program

- Receives the objects to be processed in the worklist of the Schedule Manager. This means that no scope of selection must be defined for the report itself. The scope of selection only has to be defined once in the flow definition.
- Processes these objects within the worklist, and that a processing status is set for each processing step and object during processing.
- Issues messages on the object within the worklist.

To ensure that these requirements are met, certain programming guidelines must be followed when the customer program is written.

Sel (Selection)

A program for which this indicator is set is a selection program for the worklist (or for a flow definition with worklist) for a customer application. Programming guidelines must be followed for such programs as well.

Rep (Reports)

This indicator is set for programs that generate reports that are based on worklists. Programming guidelines must be followed for such programs as well.

6. Save your entries.

Inserting Your Own Programs as Tasks



Essentially any program can be scheduled in Schedule Manager.

However, to enable information on the program to be output in the job monitor, the program must meet certain programming guidelines.

Additional programming guidelines must be met to enable the program to be scheduled with the workflow (flow definition).

For the program to use the worklist, the programming guidelines for the job monitor and the flow definition must be met. Further programming guidelines must also be noted.

For more information, see [Inserting Your Own Programs as Tasks \[Seite 22\]](#).

Storing Task Documentation

Use

You can attach Microsoft Office documents to the tasks where they are required for processing. You can also store straightforward long texts to document a task.

Procedure

Creating Microsoft Office Documentation

1. Choose *Change task list*.
2. Use the right mouse button to select the task to be documented.
3. Choose *Office document*.
The required Microsoft Office document type appears. You can now enter your text.
4. Save the document and return to the scheduler by choosing *File* → *Close* → *Back to the Schedule Manager*.

Creating a Long Text

1. Choose *Change task list*.
2. Use the right mouse button to select the task to be documented.
3. Choose *Long text*.
A window appears in which you can enter a note.
4. Choose *Continue*.
The system returns to the scheduler.



You can only store **one** long text and **one** Microsoft Office document **at the same time** for a given task.

All users assigned to this task list have access to these documents.

Sending Long Texts

1. Use the right mouse button to select the task for which you have entered a note.
2. Choose *Send note*.
3. Enter the *recipient* and a *recipient type*.
You can select the priority of the transmission and other attributes.
You can view the note again and create attachments.

Result

You have stored a document for a task, which is now permanently available to you for reference.

Scheduling Tasks in the Daily Overview

Scheduling Tasks in the Daily Overview

Use

To enable the system to start tasks at certain times, you need to schedule the tasks from the task list in the daily overview.

Prerequisites

You have inserted tasks into a task list.

Procedure

1. Choose *Insert task*.
2. Use *drag-and-drop* to schedule a task from the task list in the daily overview, by dropping the task onto an appropriate time.



Transactions and programs can only be started **directly** by you. You **cannot** schedule these [task types \[Seite 18\]](#) in the daily overview.

3. To start a transaction or program from the task list, select this task.
4. Use the right mouse button to choose *Execute*.

Result

The system executes the tasks at the times you specified.

Scheduling a Complete Task List in the Daily Overview

Use

You can schedule a complete task list. The advantage of doing so is that you can use the same task list each month for scheduling purposes simply by changing the start date and time.

Prerequisites

Before you can schedule the complete task list, you must specify a [relative start time \[Seite 32\]](#) for each task in the task list. This schedules the sequence of tasks independently of the concrete start date of the task list.

Scheduling a Complete Task List in the Daily Overview

1. Select the highest node of the task list with the right mouse button.
2. Choose *Schedule*.
3. Specify when the task list should be run.
You can choose between:
 - Starting on the current day
 - Starting on any other day

Scheduling the Complete Task List

Before actually scheduling the task list, you can run a simulation.

1. Select the highest node of the task list with the right mouse button.
2. Choose *Simulation of scheduling*.
3. Specify when the task list should be run.
The system outputs a list showing the starting dates of the tasks in the task list.

Deleting the Scheduling of a Complete Task List

1. Select the scheduled task list in the daily overview.
2. Choose *Delete*.
The system outputs a table showing the statuses of the tasks in the task list.



You can decide when the jobs in the task list should be started. For more information, refer to [Controlling Whether Jobs Are Started \[Seite 34\]](#).

Relative Start Time

Relative Start Time

Use

You can schedule complete task lists.

To do this, it is necessary to specify a *relative start time* for each task.

Features

The relative start time contains two time parameters:

- The number of calendar days or workdays (called the *offset*), such as 2 calendar days or 1 workday.
- The actual starting time, such as 12:15 pm.

Using these parameters, you can schedule and reschedule the individual tasks independently of the actual run date of the task list itself.

Example 1

You have created the task **foreign currency valuation** with the parameters **offset in working days: 1** and **start time: 12:15 pm**. If the task list is scheduled to start on a Friday and a factory calendar is being used, the task will be started the following Monday at 12:15 pm (weekends are not working days). If a Gregorian calendar is being used (in this case all days are working days), the task would be started on Saturday at 12:15 pm. If the same task list were scheduled to start on a Monday, in both cases the task would be started on the following Tuesday at 12:15 pm.

If the task list contains several tasks with relative start times, the actual start times are always calculated on the basis of the scheduled date and the specified calendar.

Example 2

The task **assessment** has an offset of 1 calendar day, and the task **settlement** has an offset of 2 calendar days. The schedule date of the task list is March 11. The task **assessment** is started on March 12 (March 11 + 1 day), and the task **settlement** is started on March 13 (March 11 + 2 days).



The offset for the start time is always based on the **concrete** schedule date. The system then calculates the concrete start date from this offset using the schedule date and the specified calendar.

Runtime Analysis for Jobs

Use

Runtime analysis shows you the average runtime of jobs or flow definitions. It also estimates the runtime of the next job to be executed.

Activities

1. Select a job with the right mouse button.
2. Choose **runtime analysis**.
The system shows you the following information:
 - The average runtime
 - The number of runs executed
 - The runtime of the last and next-to-last runs
 - The estimated runtime of the next run
3. You can also enter your own estimation of the runtime.



If any runtime information is available for tasks in the task list, it is displayed in the last three columns of the task list.

Controlling Whether Jobs Are Started

Controlling Whether Jobs Are Started

Use

If you schedule a job in the daily overview but do not release it, it is **not automatically started** by the system when the start time is reached.

This function is particularly useful for [scheduling a complete task list \[Seite 31\]](#).

Features

You can release a job:

- When inserting it into the task list
- After scheduling the job in the daily overview

Activities

You are in the mode *task list - scheduling*.

1. You select a job with the right mouse button.
2. You choose *Schedule*.
On the subsequent screen, you specify under *Release task* whether the job is executed automatically when the start time is reached.

Using the Monitor

Use

The monitor shows the information on an active or completed job that was scheduled in the scheduler.

To improve performance and facilitate error rectification, use the multilevel [worklist \[Seite 62\]](#).

Integration

The monitor is part of the [Schedule Manager \[Extern\]](#).

Other components are:

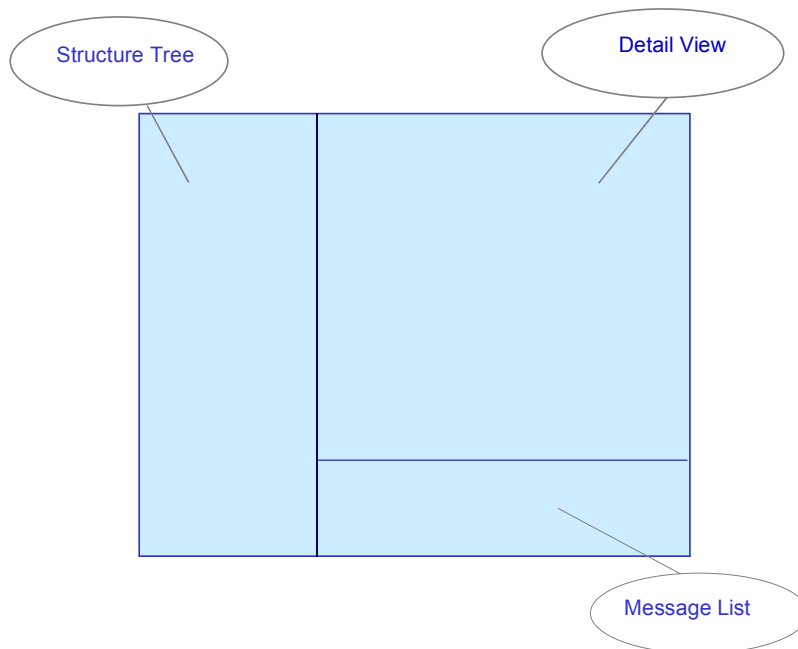
- **Flow definition**

You can group tasks that are to be executed in the background in a particular order (and which are controlled by the workflow) into a [flow definition \[Seite 42\]](#).

- **Scheduler**

You can create individual tasks for business transactions (which for example, make up period-end closing) in the [scheduler \[Seite 15\]](#) task list. By scheduling them in the daily overview, you enable the system to execute them.

Structure



For detailed information on the functions of the monitor, refer to:

- [Monitor - Monitoring Active and Completed Jobs \[Seite 37\]](#)

Using the Monitor

- [Monitor - Working with the Object List \[Seite 77\]](#)

Monitor - Monitoring Active and Completed Jobs

Definition

The monitor shows the information on an active or completed job that was scheduled in the scheduler.

Prerequisites

To enable the system to start jobs or job chains, you need to schedule them in the daily overview of the scheduler.

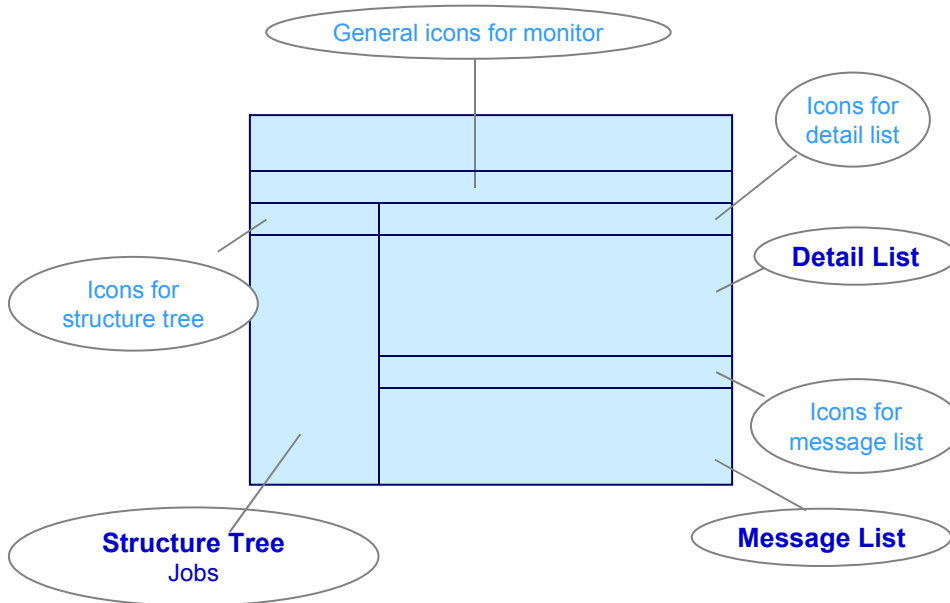
Features

To determine the current status of jobs that are still running, you can repeat the database selection.

If you only wish to see jobs with a certain status, you can hide the other statuses.

The monitor is divided into three areas:

Structure



- **Structure tree**

The structure tree displays the workflows (expanded into substeps) and jobs which ran at particular times, in chronological order.

You receive information on:

Monitor - Monitoring Active and Completed Jobs

- The job status
- The job runtime
- The update status

Use the right mouse button to see the different functions available for this job, such as:

- Start transaction
- Restart report
- Goto object monitor in the generalized worklist, and so on.

For more detailed information on a job, double-click on the required entry in the structure tree. The detail view appears.

- **Detail view of a job**

By switching the tab titles you can select different detailed information (details, parameters, additional information) on a job.

There are four pushbuttons underneath the tab page. These are only active if corresponding data for the selected job is available:

- Spool list
Display of batch spool list(s)
- Job log
Display of the job log belonging to the batch job
- Extract
Online display of results lists that were saved
- Basic list
Online display of a short list that was saved. This list contains the most important information on a job.

- **Message list**

If messages occurred for a job and these were saved, then the system displays them in the message area.

To go to a message long text, double-click on the corresponding message.

Activities

From a scheduled job, you call up the monitor from the daily overview of the scheduler.

You call up the required details for a selected job.

See also:

[Monitor - Working with the Object List \[Seite 77\]](#)

Monitor - Working with the Object List

Definition

The monitor of the *Schedule Manager* is a tool for processing multilevel worklists. It contains information on active or completed jobs that were planned in the scheduler.

Use

This section provides information on using the object list in the monitor of the Schedule Manager.

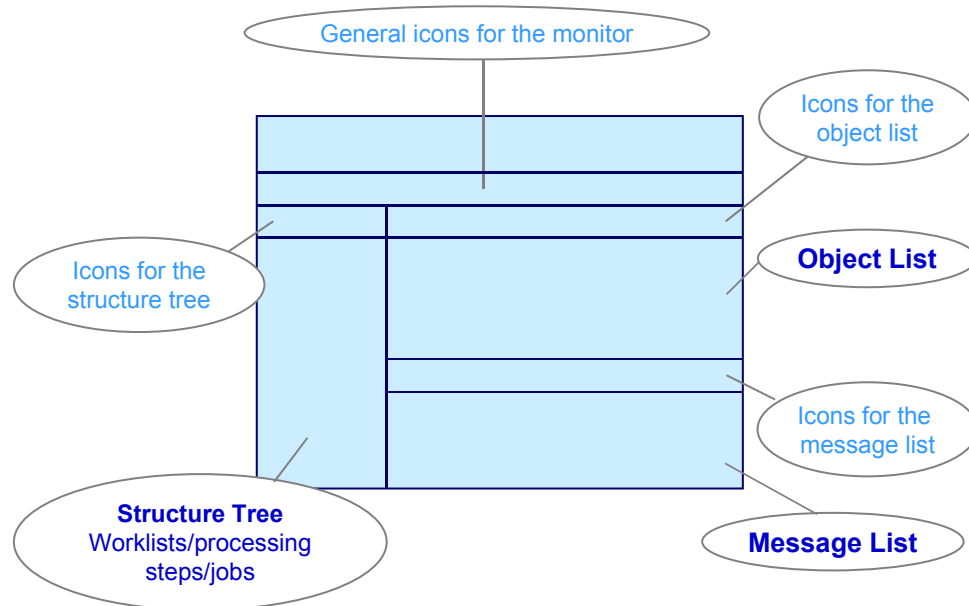
For more information on jobs in the monitor, see [Using the Monitor \[Seite 35\]](#).

The monitor performs the following functions:

- Displays the available worklists and their processing steps
- Keeps track of the processing status of each object (such as product cost collectors, WBS elements, internal orders, and production orders) and each processing step
- Assists you in analyzing the causes of errors for the objects and processing steps
- Displays the processing status of objects (particularly for the objects with errors)
- Shows the hierarchical relationships between the objects
- Compares object hierarchies with each other
- Controls whether objects enter the next processing step
- Sends objects to the responsible person for further processing

You can set the processing status so that objects of negligible value, for example, are not reprocessed in a processing step even if errors are issued for the objects in that step. In this case, it is not necessary to remove the cause of the error.

Monitor - Working with the Object List

Structure

The monitor is divided into the following screen areas:

Structure Tree with Worklists/Processing Steps/Jobs

The structure tree containing the worklists, processing steps and jobs is displayed on the left side of the monitor.

In the system, the worklists are updated separately for each scope of selection and processing step sequence. The processing step sequence is specified in the flow definition. You can view the worklists of the application component in which you are working, including their functions and objects.

The system displays the following information on each worklist:

- Number of objects in the worklist
- Processing steps
- Number of objects processed in each processing step
- Number of objects with errors in each processing step

Object List

The object list is located in the top right area of the screen. The object list can display one or more processing steps of a worklist. You select the processing steps in a worklist in the structure tree *worklists/processing steps/jobs*.

The object list includes the following information:

Monitor - Working with the Object List

- The objects for the selected processing steps. You can control which objects are displayed for each processing step by means of a filter that takes the individual processing statuses into account. In application toolbar *icons for object list*, choose the icon *Objects: Set status filter*
The system uses the last user-specific default setting of the dialog box. Note the default settings of this dialog window in the SAP standard system. The last user-specific setting of the dialog box is the default.
- The processing step performed for the object
- The processing status of the object for each processing step
- A check box you can select to indicate that you have processed the object in the worklist monitor
- The person responsible for the object (if any)

Message List

The message list is located in the lower right-hand portion of the screen. The message list displays messages for certain objects. You can use these messages to analyze the causes of error for each object and processing step.

See also:

[Schedule Manager: Multilevel Worklist \[Seite 62\]](#)

[Multilevel Worklist: Process Flow \[Seite 75\]](#)

For detailed information on using the monitor of the Schedule Manager, see the following section:

[Processing Status of Objects and Processing Steps \[Seite 91\]](#)

[Processing Worklists \[Seite 80\]](#)

Using the Flow Definition

Using the Flow Definition

Use

A flow definition consists of individual flow steps. These steps include scheduling programs with variants in the job control of the SAP System, and interaction with users by email.

The flow definition is a graphical summary of several steps. A step in the flow definition corresponds to a task in the task plan, except that the individual step does not appear directly in the task plan, rather it is displayed in the flow definition, which is included in the task plan.

Integration

The Schedule Manager provides a multilevel [worklist \[Seite 62\]](#). The multilevel worklist improves performance and facilitates error finding. To use the Schedule Manager worklist, create a flow definition and schedule it in the scheduler. For further information about this worklist, see [Choosing Objects for Processing \[Seite 90\]](#).

Prerequisites

Workflow profiles enable you to adapt the user interface of the flow definition according to your requirements. If required, the project team can create and provide this type of profile. To make individual processing available as flow steps, you need a program which provides all the processing parameters and displays an ergonomic user interface.



If you are using the workflow builder function for the first time in the SAP System, go into Customizing and choose *Basis* → *Business Management* → *SAP Business Workflow* → *Maintain Standard Settings for the SAP Business Workflow* and then *Automatic Customizing*.

For more information, see [Creating a Flow Definition \[Seite 44\]](#).

For more information on error handling in flow definitions, see: [Error Handling \[Seite 52\]](#).

Features

You can define individual flow definitions with as many flow steps as you like, or you can link flow definitions together within an "upper" flow definition. You must assign the "upper" flow definition to an application (or application component) that is on a higher level than the applications assigned to the lower-level flow definitions.

Flow Step Types

In the workflow builder, you can schedule four types of flow step:

Program with Variants

Enter a program and a variant.

User Decision

For a user decision, the system sends a message to a user. The system generates a text that creates a message header in the user's inbox. The system stops processing the individual tasks in the [flow definition \[Seite 42\]](#) until the user confirms the message. The text should therefore contain the information required to make the decision, such as which task the system just executed and which data needs checking.

Fork

You can define as many tasks as you wish in each of the parallel branches. All of the parallel branches join at the end, although the task that follows the join is only processed when all of the tasks in each branch are completed.

You cannot subsequently change the number of parallel branches. However, you can delete a branch by deleting all of the tasks in the branch. You **cannot** add another branch.

Flow Definition

The flow definition consists of several tasks that the system executes in the specified order, once you schedule the flow definition in the daily overview of the scheduler. You can create further "sub-" flow definitions within a flow definition.

Creating a Flow Definition

Creating a Flow Definition

Use

You use a [flow definition \[Seite 42\]](#) in the [Schedule Manager \[Extern\]](#). You can insert a flow definition in the task list of the scheduler, then schedule it into the daily overview and run it.

Procedure

1. Choose *Extras* → *Process flow definition* in the Scheduler

Enter a description for the flow definition

2. Choose *Create*.

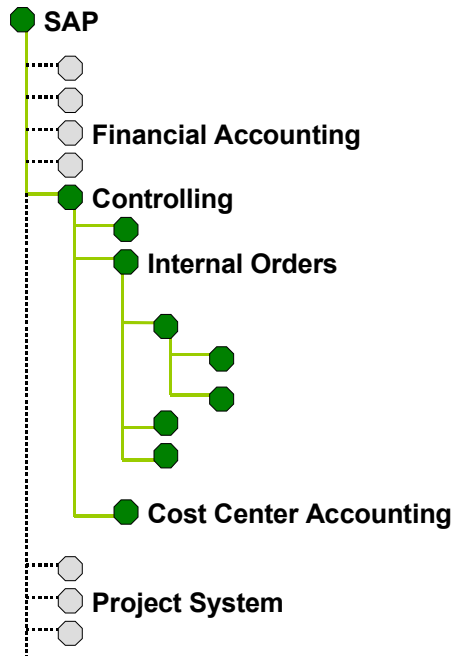


When creating without a template, the system generates a flow definition example (with flow steps) for the specified application. Do **not** delete the steps *Check objects in worklist* and *Renew worklist processing*, because these steps enable the recursive call of worklist processing. If the step *Renew worklist processing* is deleted, you **cannot** include it in the flow definition.

3. In the screen that follows, enter a description for your flow definition and then assign it to an application component.



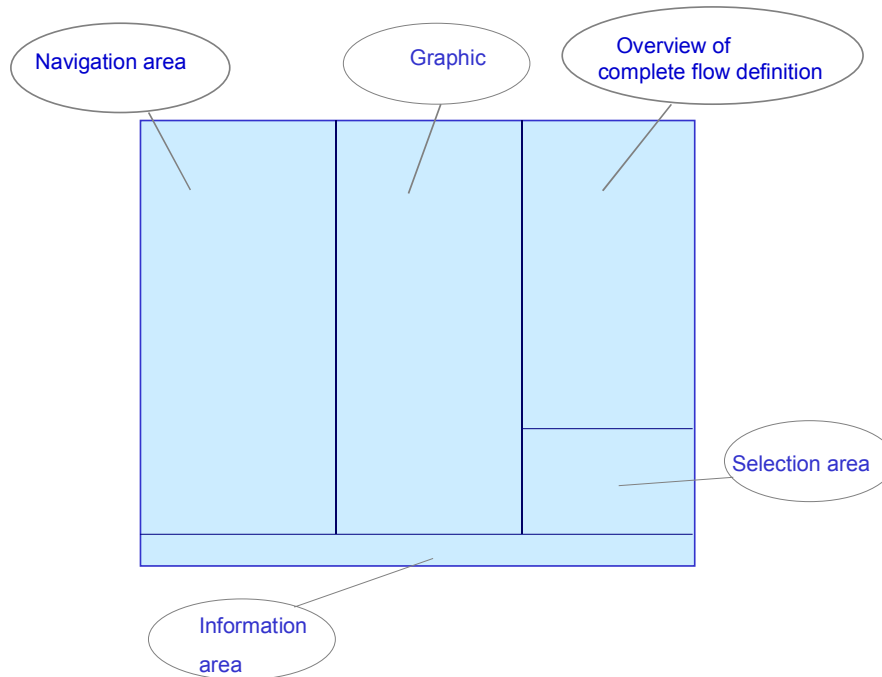
Note that all flow steps in a flow definition (which is also assigned to an application) must also be assigned to this application or to the application on the next level up or down.



If you have chosen the application *Controlling* for the flow definition, you can only create flow steps in this flow definition if the application that you define for the steps is **above** or **below** *Controlling* (SAP) in the hierarchy (for example, *Internal Orders*, *Cost Center Accounting*...). In this flow definition for example, you cannot insert a report in which the application *Financial Accounting* is defined.

4. If required, indicate that you are working with a [worklist \[Seite 62\]](#). To work with a multi-level worklist, you must have chosen an application with a worklist and the function *With worklist* during creation.
5. Enter a development class in the dialog box *Create object catalog entry*, then save your entry.
The workflow builder appears. Each node (step, event or operator) in the workflow builder has a menu that you can call up using the right mouse button. This immediately shows you the operations that are possible for a node, and cancels out the need for long menu structure or pushbutton paths.

Creating a Flow Definition



6. In the navigation area, select the node *Steps*.
7. Choose *Create step*.
8. Choose the [flow step type \[Seite 43\]](#).
9. Specify the required information for the flow step.



You can choose programs only using the input help. Programs are displayed in the input help only if you have registered them in the tables **SCMAPROGRAMS** (for SAP programs) or **SCMAPROG_CUST** (for your own programs).
For more information about working with your own programs, see [Inserting Own Programs as Tasks \[Seite 22\]](#).

10. Choose *Cancel* to return to the Workflow Builder.
11. You see the task in the structure tree of the task area. The task is integrated in the graphic of the flow definition. Move the task to the desired position in the flow definition.
12. Save your entries.

For more information on the Workflow Builder, see the SAP Library under *BC-Basis Components* → *Business Management (BC-BMT)* → *SAP Business Workflow Navigation(BC-BMT-WFM)* → [Workflow Builder \[Extern\]](#).



If you are using the workflow builder function for the first time in the SAP System, go to the Implementation Guide (IMG) and choose Basis → Business Management → SAP Business Workflow → Maintain Standard Settings for the SAP Business Workflow and then Automatic Customizing.

You can also use the SAP standard flow definitions. Variant names for these flow definitions begin with SAP& These variants only contain selection variables named SAP-SCMA Use the STVARV transaction or Schedule Manager, Extras → Settings → Selection variables to adapt the variable values according to your requirements. For further information on selection variables, see [Defining Global Variables for Task Lists/Flow Definitions \[Seite 8\]](#) and Defining Selection Variables for Flow Definitions With Parallel Branches.

Specifying Selection Variables for Flow Definitions With Parallel Branches

Use

Normally the selection criteria for period-end closing do not change very often. Criteria that do change regularly are the closing period and the fiscal year. The period and fiscal year must be changed for each program or flow definition specified as a task in the task list.

To avoid having to change these values for every single program or every flow definition, you can define selection variables in the program variants.



You have five programs that must run with the same valuation for nine plants. That means that every program must run with a plant-specific variant for each plant. These variants differ only in their *Plant* specification.

As described below, you define global variables for *fiscal year*, *period* and *plant*. Create variants for the five programs and define the global variants that you created previously. Finally define a flow definition (A) for the five programs. Define a flow definition (B) with nine parallel branches (for the plants). Include the flow definition A in each of these branches. A dialog box appears that displays the three global variables (*fiscal year*, *period*, *plant*). Enter a value for the variable *Plant* and flag the entry as to be saved. The dialog box reappears when you plan flow definition B in the day view. Enter values for each fiscal year and period. Do not make an entry for the variable *Plant*. If you make an entry here, the system does not accept it. You defined the plant when creating the flow definition.

Prerequisites

You have defined the variables already. For more information, see [Defining Selection Variables \[Seite 9\]](#).

You are in [flow definition \[Extern\]](#) in Schedule Manager (*Extras* → *Flow definition* → *Edit flow definition*).

7. Choose a processing step in the navigation area.
On the right of the screen, you see *Flow definition: Task details*.
8. Enter a new variant for this processing step and define a name for this new variant in the *Variant* field.
9. Choose *Change variant*.
The *Maintain Variant: Program <program name>, Variant <variant name>* screen appears.

Procedure

14. Choose *Attributes*.
15. Make an entry in the *Description* field.
16. Now you want to specify variables for the period and fiscal year that you can then change once centrally for all programs and flow definitions that use those variables. To do this, under the heading *Selection screen objects*, select the column S in the lines *Period* and *Fiscal year*.

Specifying Selection Variables for Flow Definitions With Parallel Branches

17. Choose *Selection variables*.
18. To choose selection criteria for the variables P_FROM (period) and P_GJAHR (fiscal year), use the input help.



You previously entered these variables in table TVARV.

11. Save your entries.
The *Maintain Variant: Program <program name>, Variant <variant name>* screen appears.

Result

The system uses these entries for the following purposes:

- To feed these variables to a program created directly as a task in the workflow.
- To feed these variables to the parallel branches.



The system mixes the variables of the parallel branches with those of the main workflow.

If the parameter values of the main workflow are different from those of the subworkflows, the system uses the parameter values that were specified when the subworkflows were created.

Choosing Objects for Processing

Choosing Objects for Processing

Use

An advantage of this [worklist \[Extern\]](#) is that the objects for processing only have to be selected once per flow definition.

Prerequisites

You are in [flow definition \[Extern\]](#) in Schedule Manager (*Extras* → *Flow definition* → *Edit flow definition*).

Procedure

10. Choose a processing step in the navigation area.
On the right of the screen, you see *Flow definition: Task details*.
11. Enter a new variant for this processing step and define a name for this new variant in the *Variant* field .
12. Choose *Create variant*.
The *Variant Maintenance* screen appears. Program <program name>, variant <variant name>.
Here you can determine the scope of selection of the program variants for the flow definition.
13. To be able to choose more extensive selection criteria, first complete the required entry fields, for example *Period* <006>, *Fiscal year* <2000>.
14. You can now define further selection parameters using the various pushbuttons that are offered in dialog boxes.
15. Choose *Attributes*.
The *ABAP: Save Attributes of Variant* <Variant name> screen appears.

Defining Processing Options, Output Options and Execution Types

You are in [flow definition \[Extern\]](#) in Schedule Manager (*Extras* → *Flow definition* → *Edit flow definition*).

16. Choose a processing step in the navigation area.
On the right of the screen, you see *Flow definition: Task details*.
17. Choose *Change variant*.
The screen *Variant Maintenance* appears. *Program* <program name>, *variant* <variant name>.
18. In the corresponding group frames, select the desired processing option, output option and execution type. For example, you can define whether parallel processing is allowed.
19. Perform variant maintenance for all programs that are linked to the flow definition.

For further information see [Defining Selection Variables in the Program Variants \[Seite 10\]](#) and [Selection Variables for Flow Definitions with Parallel Branches \[Seite 48\]](#).

Error Handling

Error Handling

Job error handling involves the following:

1. For checking purposes:
 - Periodically reading the job status
 - Checking the maximum runtime
2. When errors occur:
 - User decision "Continue the flow with the next task"
 - User decision "Reschedule the job"



The user decision "Reschedule job" is **not** available for worklists.

Error: A Job Stops

Cause

The job stops, if for example, the system is turned off.

Error Handling

The dialog for an error in the job administration is handled in its own sub-workflow, which consists of a user decision. This message is sent as a top priority mail to the person whose name is entered in the detail screen for maintenance of jobs in the flow definition, in the *When error, mail to* section (see also: [Creating a Flow Definition \[Seite 44\]](#)). That person receives the mail that an error has occurred in *job XY* in their mail inbox.

There are two options in the user decision.

Continue the flow with the next task

You choose this option if:

- You corrected the error and rescheduled the job manually in a separate session, and are waiting for the job to be executed.
- You executed the transaction online.
- You decided that the error is not relevant.



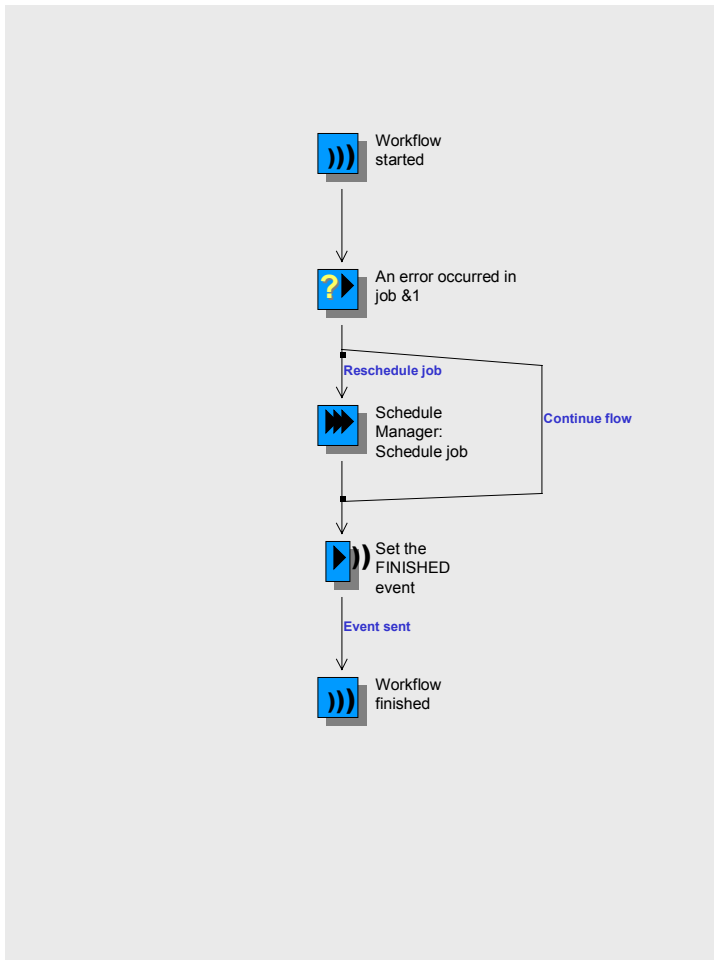
You can use this option without any problems, for jobs that have worklists. However, you need the worklist ID to be able to execute the job outside of the workflow.

Reschedule the job

You choose this option if you corrected the error and would like to use the [Schedule Manager \[Extern\]](#) to control the rescheduling of the job.

Error: A Job Stops

Graphical Representation of the Flow



Error: A Job Becomes "Stuck"

Symptom

The program has a runtime that is far longer than expected.

Error Handling

The dialog for a processing error in the program is handled in its own sub-workflow, which consists of a user decision. This message is sent as a top priority mail to the person whose name is entered in the detail screen for maintenance of jobs in the flow definition, in the *When error, mail to* section (see also: [Creating a Flow Definition \[Seite 44\]](#)). That person receives the mail that an error has occurred in *program X, variant Y* in their mail inbox.

There are two options in the user decision.

Continue the flow with the next task

You choose this option if:

- You corrected the error and rescheduled the job manually in a separate session, and are waiting for the job to be executed.
- You executed the transaction online.
- You decided that the error is not relevant.



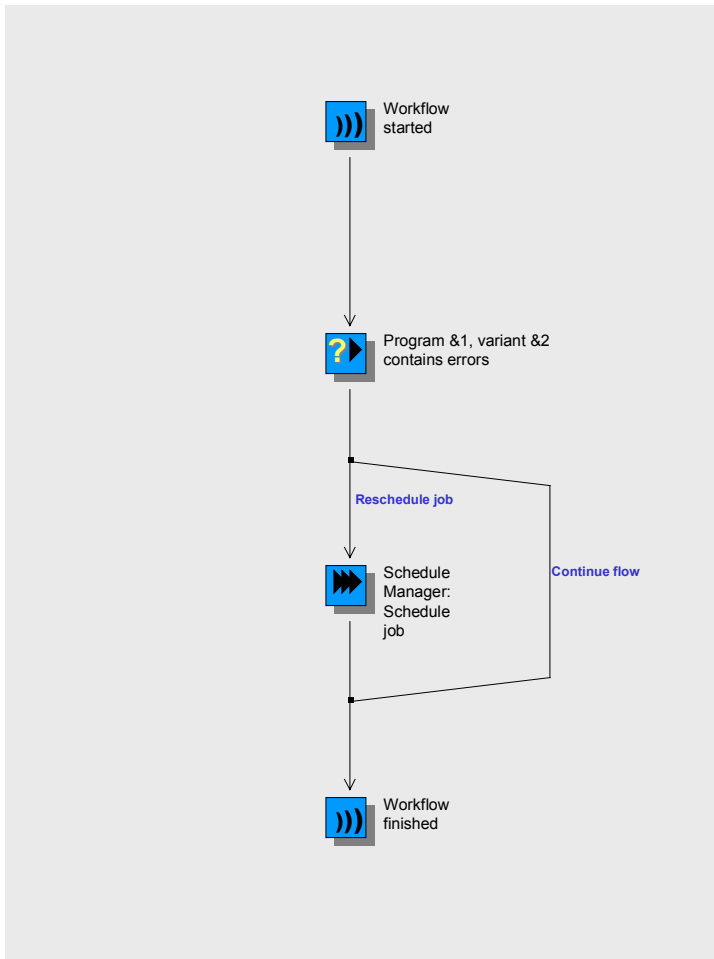
You can use this option for jobs that have worklists. However, you need the worklist ID to be able to execute the job outside of the workflow.

Reschedule the job

You choose this option if you corrected the error and would like to use the [Schedule Manager \[Extern\]](#) to control the rescheduling of the job.

Graphical Representation of the Flow

Error: A Job Becomes "Stuck"



Error: An Executed Report Found an Error

Symptom

Due to an error in the content, the REPORTERROR workflow event was triggered.

Error Handling

The workflow executes an error dialog. This contains a user decision. The user decision is sent as a top priority mail to the person whose name is entered in the detail screen for maintenance of jobs in the flow definition, in the *When error, send to* section (See also: [Creating a Flow Definition \[Seite 44\]](#)). That person receives the mail that an error has occurred in *program X, variant Y* in their mail inbox.

There are two options in the user decision.

Continue the flow with the next task

You choose this option if:

- You corrected the error and rescheduled the job manually in a separate session, and are waiting for the job to be executed.
- You executed the transaction online.
- You decided that the error is not relevant.

Reschedule the job

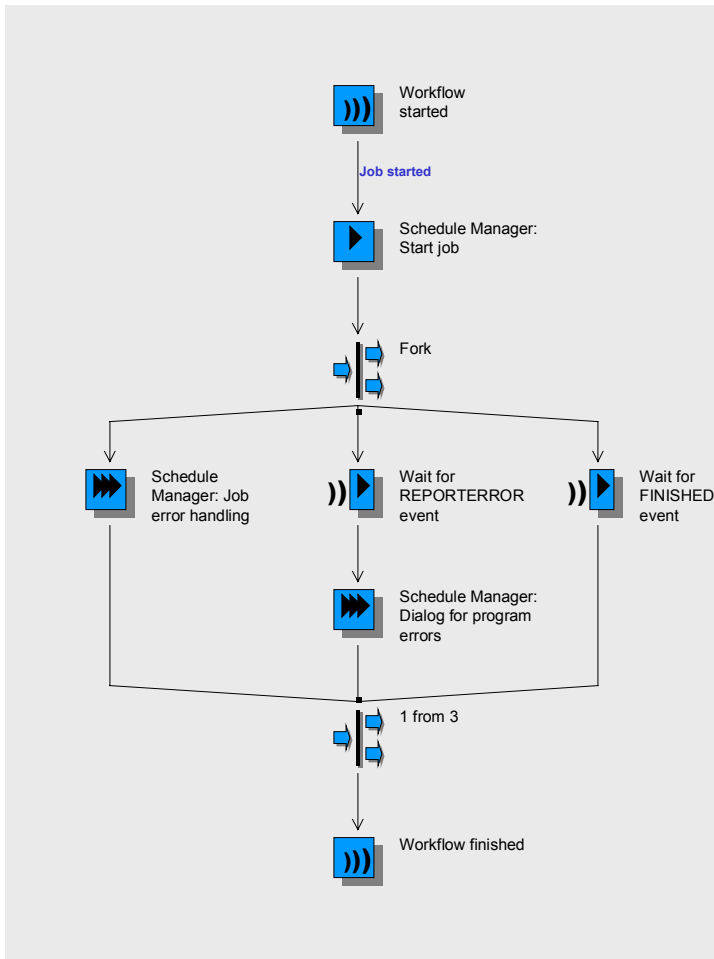
You choose this option if you corrected the error and would like to use the [Schedule Manager \[Extern\]](#) to control the rescheduling of the job.



No user decision is called up for jobs that have worklists. This is because the error handling for worklists is triggered by rerunning the task steps in the workflow. Therefore, programs written for worklists always trigger the FINISHED event.

Graphical Representation of the Flow

Error: An Executed Report Found an Error



Error: Job Scheduling Was Unsuccessful

Cause

Scheduling a job can be unsuccessful if the technical name of the report or variant was changed after the flow definition was created. This means that the technical names in the flow definition no longer exist.

Error Handling

The workflow recognizes the error and issues an error message displaying the faulty technical name of the report or variant.

The message is linked to a user decision. This message is sent as a top priority mail to the person whose name is entered in the detail screen for maintenance of jobs in the flow definition, in the *When error, mail to* section (see also: [Creating a Flow Definition \[Seite 44\]](#)). That person receives the message in their mail inbox.

There are two options in the user decision.

Continue the flow with the next task

You choose this option if:

- You corrected the error and rescheduled the job manually in a separate session, and are waiting for the job to be executed.
- You executed the transaction online.
- You decided that the error is not relevant.

Reschedule the job

You choose this option if you corrected the error and would like to use the [Schedule Manager \[Extern\]](#) to control the rescheduling of the job.

Error: The Workflow Stops

Error: The Workflow Stops

Cause

The workflow stops due to an error in the workflow runtime system.

Symptom

The job just scheduled is completed since the job control functions **separately** from the workflow. **However**, the following steps in the flow definition are **no** longer executed.

Error Handling

It is **not** possible to restart at the point where the workflow stopped. Therefore, you need to reschedule the whole flow definition.

Error: The Workflow Becomes "Stuck"

Cause

During job scheduling using the workflow, if an error occurs in the program section of the report and the report is being run at the AT SELECTION-SCREEN OUTPUT event, then the job is not scheduled.

Error Handling

The system issues an error message to notify you that scheduling is not complete.

The message is linked to a user decision. This message is sent as a top priority mail to the person whose name is entered in the detail screen for maintenance of jobs in the flow definition, in the *When error, mail to* section (see also: [Creating a Flow Definition \[Seite 44\]](#)). That person receives the message in their mail inbox.

There are two options in the user decision.

Continue the flow with the next task

You choose this option if:

- You corrected the error and rescheduled the job manually in a separate session, and are waiting for the job to be executed.
- You executed the transaction online.
- You decided that the error is not relevant.

Reschedule the job

You choose this option if you corrected the error and would like to use the [Schedule Manager \[Extern\]](#) to control the rescheduling of the job.

Multilevel Worklist

Multilevel Worklist

Use

The worklist of the Schedule Manager is a **multilevel worklist**. This worklist is particularly useful for the period-end closing activities.

Why Is the Worklist Multilevel?

In previous releases, the period-end closing process in the R/3 system consisted of a series of batch jobs. The sequence of the processing steps was established by the order in which the jobs were called. The objects were selected separately for each job. Through the selection criteria entered, it was possible to specify a unified scope of selection. This scope of selection had to be respecified for each processing step (that is, for each individual function of period-end closing).

When an object was processed, errors that occurred in previous processing steps were not taken into account. For this reason, it was necessary to check the objects that had errors once a job was completed. Any errors had to be corrected and then the job restarted for the entire scope of selection. In some areas (such as the period-end close in *Product Cost by Period*), it was already possible to create a single-level worklist for individual processing steps. With this single-level worklist, the objects with errors could be called up for each processing step, and the causes of the errors determined. The processing step could then be performed again for the object after the error was corrected. This worklist did not prevent objects with errors from being processed in the subsequent processing step (that is, in the subsequent job).

Advantages of the Multilevel Worklist

The worklist of the Schedule Manager is a multilevel worklist. This means that the worklist is generated for a sequence of processing steps rather than for just one processing step. The worklist therefore enables efficient execution of processing step sequences. Processes such as period-end closing can be performed much more efficiently with a multilevel worklist.

The multilevel worklist has the following advantages:

- The processing step sequences (such as in period-end closing) can be performed **faster** than before.
 - Manual processing after completion of each job is no longer necessary. Manual processing is only necessary after executing a sequence of processing steps that consists of multiple jobs (for example, complete closing of an application component).
 - Furthermore, if errors were issued for objects in the single-level worklist, it was often necessary to repeat the processing steps for the entire scope of selection (and not just for the objects with errors). With the multilevel worklist, the processing steps are repeated only for the objects that have errors.
- CPU time is reduced because objects are selected only once for each processing step sequence, instead of for each individual processing step. Objects are selected before the first processing step is executed. The multilevel worklist provides performance benefits particularly with complex structures in which dependencies between objects must be taken into account (such as complex project structures).

As a rule, jobs are planned and monitored by members of the EDP team. In many cases, these employees are not responsible for the correcting the errors shown in the error logs. With the multilevel worklist, you can directly inform the employees responsible for correcting the errors.

Multilevel Worklist

This notification takes place by means of a mail message that is sent automatically through the [workflow \[Extern\]](#).

Integration

The multilevel worklist is part of the *Schedule Manager* and is always used with the other functions of the Schedule Manager (see *Prerequisites*).

The following applications, functions, and objects are currently supported by the multilevel worklist:

Cost Object Controlling: Manufacturing Orders and Product Cost Collectors

Process Flow	Period-end closing for manufacturing orders and product cost collectors [Extern]
Scope of Selection for Processing Objects	<p>Closing encompasses production orders, CO production orders (production orders without a quantity structure), process orders; product cost collectors and QM orders. With co-products, some of the period-end closing work is performed at the level of the items of the manufacturing orders.</p> <p>A prerequisite is that the following requirements are met for these objects:</p> <ul style="list-style-type: none"> • The objects are not assigned to a cost object hierarchy, or it is specified in the cost object category that the individual orders of a material are processed outside the cost object hierarchy (see <i>Product Cost by Period</i>). • Account assignment can be made directly on the objects. This means that with regard to the selection of manufacturing orders, account assignment is made on the manufacturing orders themselves and not on a product cost collector. • The objects do not have status DLFL (deletion flag). <p>Product cost collectors are objects of the <i>Product Cost by Period</i> subcomponent.</p> <p>Manufacturing orders (including manufacturing orders without a quantity structure) are objects of the <i>Product Cost by Order</i> subcomponent.</p>
Processing Step	Objects
Template allocation	Order header (including product cost collectors)
Revaluation at actual prices [Extern]	Order header (including product cost collectors)
Actual overhead [Extern]	Order header (including product cost collectors)

Multilevel Worklist

Preliminary Settlement for Co-Products, Rework	<ul style="list-style-type: none"> • Preliminary settlement for co-products: Order header of manufacturing orders as processing objects; order items as receivers • Preliminary settlement of rework: Order header of manufacturing orders as processing objects, order header of manufacturing orders or product cost collectors as receivers; but not: Settlement of rework on product cost collectors or manufacturing orders assigned to a cost object hierarchy (see below) • Preliminary settlement of collective orders (old processing method without automatic goods movement) Header of manufacturing orders as processing objects, header of manufacturing orders as receivers
WIP calculation [Extern]	Production and process orders or, in joint production, their items, as well as CO production orders and product cost collectors
Variance calculation	Production and process orders or, in joint production, their items, as well as CO production orders and product cost collectors
Settlement [Extern]	Production and process orders or, in joint production, their items, as well as CO production orders and product cost collectors

Cost Object Controlling: Cost Object ID (Cost Object Nodes in a Cost Object Hierarchy and General Cost Objects)

Process Flow	Period-end closing for cost object ID
---------------------	---------------------------------------

Multilevel Worklist

Scope of Selection for Processing Objects	<p>Period-end closing includes:</p> <ul style="list-style-type: none"> • Cost object nodes of cost object hierarchies and the single objects assigned to the cost object hierarchy These can be the following: Product cost collectors, manufacturing orders, production orders without quantity structure, and (if applicable) order items of manufacturing orders (with joint production) for which the following conditions apply: <ul style="list-style-type: none"> – Account assignment can be made directly on the objects. This means that account assignment for manufacturing orders is made on the manufacturing order itself and not on a product cost collector. – The objects do not have the status DLFL (deletion flag). <p>Cost object hierarchies are part of the <i>Product Cost by Period</i> component.</p> • General cost objects <p>General cost objects are objects of the <i>Costs for Intangible Goods and Services</i> component.</p>
Processing Step	Objects
Template allocation	Cost object nodes of cost object hierarchies or the single objects assigned to the cost object hierarchy (product cost collectors, manufacturing orders or production orders without quantity structure); general cost objects
Revaluation at actual prices	Cost object nodes of cost object hierarchies or the single objects assigned to the cost object hierarchy; general cost objects
Actual cost distribution	Cost object nodes of cost object hierarchies ; the single objects assigned to the lowest cost object nodes are the final receivers
Actual overhead	Depending on the Customizing settings, cost object nodes of cost object hierarchies or the single objects assigned to a cost object hierarchy; general cost objects
Preliminary Settlement for Co-Products, Rework	<p>Only for orders assigned to the cost object hierarchy:</p> <ul style="list-style-type: none"> • Preliminary settlement for co-products: Order header of manufacturing orders as processing objects, order items as receivers • Preliminary settlement of rework: Order header of manufacturing orders as processing objects, order header of manufacturing orders or product cost collectors that are assigned to the cost object hierarchy as receivers
WIP calculation	The single objects assigned to a cost object hierarchy ; but not: Order items in joint production (manufacture of co-products), CO production orders

Multilevel Worklist

Variance calculation	Depending on Customizing settings, cost object nodes of cost object hierarchies or the single objects assigned to the cost object hierarchy
Settlement	Depending on Customizing settings, the top nodes of a cost object hierarchy or all nodes of the cost object hierarchy; if applicable, all orders assigned to the cost object hierarchy, and in joint production the items of the manufacturing orders; All general cost objects

Multilevel Worklist

Project System

Process Flow	Period-end closing for Project System
Scope of Selection for Processing Objects	WBS elements, networks, and orders
Processing Step	Objects
Generation of settlement rule	WBS elements
Template allocation	WBS elements, networks, and orders
Actual overhead	WBS elements, networks, and orders
Revaluation at actual prices	WBS elements, networks, and orders
Cost forecast	Networks
Interest calculation	WBS elements, networks, and orders
Project earned value	WBS elements, networks, and orders
Results analysis	WBS elements and orders
Incoming orders	WBS elements
Settlement	WBS elements, networks, and orders
Reporting	WBS elements, networks, and orders

Internal Orders

Processing Step	Period-end closing for internal orders
Scope of Selection for Processing Objects	Internal orders, maintenance orders
Processing Step	Objects
Template allocation	Internal orders, maintenance orders
Revaluation at actual prices	Internal orders, maintenance orders
Actual overhead	Internal orders, maintenance orders
Interest calculation	Internal orders, maintenance orders
Results analysis	Internal orders, maintenance orders
Settlement	Internal orders, maintenance orders

Sales Orders

Processing Step	Period-end closing for sales orders
Scope of Selection for Processing Objects	Sales order items that carry costs and revenues
Processing Step	Objects
Template allocation	Sales order items that carry costs and revenues
Revaluation at actual prices	Sales order items that carry costs and revenues
Actual overhead	Sales order items that carry costs and revenues
Results analysis	Sales order items that carry costs and revenues
Settlement	Sales order items that carry costs and revenues

Prerequisites

You are working with the *Schedule Manager* and are using all of its functions.

A prerequisite for the use of the multilevel worklist is that a constant quantity of objects (or, in subsequent executions, their subset) is processed in a predefined sequence of processing steps.

The selection set of the objects is determined through the application you select (for example, Cost Object Controlling: Manufacturing orders and product cost collectors) (see above), as well as through any additional entries that you may make when creating report variants (see below).

The sequence of processing steps is specified in the flow definition.

This means that you proceed as follows:

1. Go into the [Scheduler \[Seite 15\]](#) of the *Schedule Manager*.
2. From the scheduler, create a [flow definition \[Seite 42\]](#). In the flow definition, specify the processing step sequence (for example, a sequence of all single functions of period-end closing for product cost collectors). The creation of the flow definition is realized through the *Workflow Builder*.

You access the flow definition with the menu options *Extras* → *Process the flow definition*. When you create the flow definition, you should:

Multilevel Worklist

- Specify which application the flow definition is intended for (such as *Cost Object Controlling: Manufacturing Orders and Product Cost Collectors*)
 - Make sure you create a flow definition with a worklist
 - The *Workflow Builder* appears. Depending on the application you selected, you receive a SAP template that you can modify to meet your requirements.
3. Create a task for each processing step in the flow definition. Such tasks can be, for example, reports for the single functions of period-end closing or user decisions.

To facilitate maintenance of the flow definition, the system offers you a default template. This consists of a task placeholder at the beginning of the flow as well as a task placeholder in the feedback loop. You define the placeholder at the beginning of the flow before the feedback loop as a report task for selecting the worklist. You define the single functions of period-end closing as tasks in the feedback loop.

To define the selection or a **single function** (such as overhead calculation) as a task, select the *Program* indicator when you create the task. Then, select the report from a proposal list that you want to include in the processing sequence (such as *Overhead: Worklist of Manufacturing Orders*)

Then create a **report variant** for the execution of the report. Maintain the parameters for the variant. Here you can specify various parameters such as whether a detail list is output and whether processing takes place simultaneously on multiple servers.

You can further filter the scope of selection of the object to be processed by using a selection profile.

In addition, enter the period and fiscal year.



You can use the same flow definition each time if you enter selection variables instead of fixed values for the period and fiscal year when you maintain the report variants. In this case, the system calculates the period and fiscal year of the single functions dynamically from the current values for these variables when it executes the flow definition. This is only possible if you use selection variables (called TVARV variables) for the period and fiscal year parameters. When you create the report variant, you specify that you are working with selection variables. You maintain the selection variables by entering the transaction STVARV, or by starting the transaction through *Extras* → *Settings* → *Selection Variables* in the menu of the Schedule Manager and changing the variables there to the period and fiscal year to be processed. This enables you to use the same job variant every month. You only need to update the TVARV variable before executing the job variant.

You specify a **user decision** if you want the previous processing results to be checked after one or more processing steps. After executing the previous processing steps, the system automatically sends a mail message to the person responsible for checking the results (usually the cost accountant). When you create the flow definition, you specify which user receives this mail.

When you use the multilevel worklist, the system always specifies a user decision and a feedback loop for the reentry into postprocessing as the last step of the flow definition.

You can also insert user interactions at other points in the process definition if you want to check the objects processed up to a particular step before continuing with processing.

If you approve (release) the check of the objects with errors, you make it possible for the objects with errors to automatically be reprocessed later.

4. Enter the flow definition as a task in the task list of the *Scheduler*.
5. Start the task (the flow definition) in the *Scheduler*.
6. You monitor the flows and jobs during and after processing in the [monitor \[Seite 35\]](#) of the *Schedule Manager*.

Features

Basic Functions of the Multilevel Worklist

The multilevel worklist is generated for an entire sequence of processing steps.

The scope of selection is determined once and is valid for all processing steps. The worklist encompasses the objects of the scope of selection for which processing in the present processing step sequence is both possible and necessary. The scope of selection, therefore, equals the maximum scope of the worklist. Certain restrictions can be specified for individual processing steps in the scope of selection. These restrictions are usually determined through selection profiles that are specified when report variants are created.

The processing steps are performed in an order strictly defined in the flow control of the scheduler. A processing status is maintained for each object and processing step. The processing status indicates whether further processing of the object is allowed.

Each processing step only contains the objects for which (based on the processing status of the previous processing step) processing in this step is allowed.

In each processing step, dependencies between objects are interpreted according to the application. For this reason, it can be necessary to include other objects when processing an object in a step. The system accounts for such object dependencies automatically. You do not have to make any additional settings.



You want to perform results analysis for a WBS element (nonvaluated project stock). Production orders whose actual costs should be included in results analysis are assigned to the WBS element.

There are individual worklists for each processing step sequence (that is, each flow definition). Typical processing step sequences are the period-end closing sequences for the individual application components. Individual worklists are created for each application (for example, for internal orders and projects).

The multilevel worklist fulfills the following basic requirements:

- The selected objects are processed as far as possible.
- The selected objects are only processed when processing is both necessary and possible.
- Object dependencies are taken into account.

Object dependencies can be encountered, for example:

- In engineer-to-order environments
- When you are using cost object hierarchies

Multilevel Worklist



Suppose you are performing results analysis for WBS elements in an engineer-to-order environment. The system first determines which WBS elements are relevant on the basis of the selection criteria. These WBS elements are called *primary objects* because they are the original objects to be processed.

Results analysis also includes values that are posted to the production orders assigned to the WBS elements. These production orders are selected on the basis of their dependency on the WBS element. They are called secondary objects.

The determination of secondary objects depends not only on the type of worklist (for example, cost object hierarchies, projects), but also on the present business function and on the processed objects.



Suppose you are performing period-end closing for a cost object hierarchy.

You calculate the actual overhead at the level of the cost object nodes. In this case, there are no dependent objects because every cost object node is included in overhead calculation. Relationships to other cost object nodes do not play a role.

You also want to distribute the actual costs assigned to the cost object nodes to the lowest cost object nodes of the assigned product cost collector. The following situation arises during distribution:

As soon as a cost object node in a cost object hierarchy is a primary object, all other nodes are secondary objects (as long as they are not also primary objects).

During the execution of the flow definition, the individual processing steps receive a list of the objects to be processed. Each processing step sends the processing status of each object and a list of the displayed messages back to the worklist.



The scope of selection can contain all objects to be processed in a particular flow. However, in some situations you may want to perform the same flow more than once in parallel with different scopes of selection. This manual parallel processing can serve to reduce the overall run time.



Suppose you want to perform period-end closing for the application component *Product Cost by Order* in *Cost Object Controlling*. You can process all production orders and process orders in a plant, or all plants in a controlling area.

You create multiple scopes of selection in which you select by plant and order type. This means that, for example, one scope of selection includes all production orders in a plant, while another scope of selection includes all process orders in that same plant.

The previous processing step in the sequence must be fully completed before the next step can be started.

After the entire processing step sequence has been executed, the user forces a manual check.

The user-defined flow of individual processing steps (specified in the flow definition), the check of the objects with errors, as well as the release of this check (for renewed processing of objects) should be repeated until all objects in all processing steps have the status *OK*. Once this is achieved, processing is completed.

Processing of Objects in the Worklist

You process the worklist in the *monitor*. The monitor shows a list of the faulty objects and the messages issued for the objects. This information is needed for analyzing and correcting the errors.

In the monitor, you can specify how objects are to be processed the next time the processing step sequence is executed. For example, you can specify the following:

- That objects marked as faulty in a given processing step are excluded the next time that processing step is executed, and instead enter the subsequent processing step.
- That objects processed without errors are nevertheless reprocessed if they are faulty in a business sense (for example, as a result of incorrect Customizing settings)



If an object was processed without error in an update run for project interest calculation, new interest calculation can only be triggered if the previous interest calculation is first reversed. If no reversal is carried out, the object is not included in the recalculation of interest even though its processing status would normally allow this.

You control this through the processing status for the object and processing step.



Note the following:

If an object has been changed since it was processed in the processing step sequence defined by the flow definition (for example, additional costs have been assigned to the object), this change is not taken into account if the object has already been processed without error. In this case, you should change the processing status for the first processing step of the processing step sequence to force reprocessing.

See also:

See the following sections for additional information on the monitor and how to use it:

[Schedule Manager: Monitor - Working with the Object List \[Seite 77\]](#)

[Processing Worklists \[Seite 80\]](#)

[Processing Status \[Seite 91\]](#)

Triggering Reprocessing of Objects

Automatic Reprocessing

Once all objects have been processed and you have corrected the errors or specified that the processing step for which errors were issued should be skipped, the processing step sequence can be repeated in order to reprocess the objects that had been faulty in the previous run. You initiate reprocessing from the mail.

Multilevel Worklist

The system then processes the objects in the selection scope that had been processed with errors in the first run of the processing sequence and those that you instructed the system to reprocess (processing forced manually). For each object, processing starts with the processing step that had errors or for which processing was forced manually. The only processing steps repeated for an object are those for which errors occurred in the previous run, those that have not yet been performed, and those for which reprocessing was forced.

Both in the first run and in the repeated run, the only objects that are processed in each processing step are those that were successfully processed in the previous step and that have not yet been successfully processed in the current step. This limits the number of objects to be reprocessed in each step to those for which errors appeared in that step or in the preceding steps of the first run. Dependencies between objects are also taken into account. That is, depending on the object to be processed and the processing step, it may be necessary to reprocess additional objects even though they were already processed successfully.

Administrative Data Reorganization of the Multilevel Worklist

The administrative data of multilevel worklists encompasses the scope of selection, the step information (flow step), the processing status for the object, and the error messages for the object. This administrative data is deleted together with the workflow data of the *Schedule Manager*. It is not possible to archive the worklists.

Activities

Once all processing steps have been executed, the system informs you by mail that the results of the processing steps are ready for review. After you have checked the results and corrected any errors, the system asks you whether you would like to repeat reprocessing.

To make the relevant checks, access the monitor of the Schedule Manager.

You can access the monitor in the following ways:

- From the mail
- Directly from the menu of the application component

See also:

[Multilevel Worklist: Process Flow \[Seite 75\]](#)

Multilevel Worklist: Process Flow

Prerequisites

You have created a [flow definition \[Seite 42\]](#). In the flow definition, you define the sequence of processing steps that you want to perform. A processing step corresponds to a single function. Examples of individual processing steps are WIP calculation, variance calculation, and settlement. A processing step sequence is defined by linking these single functions to each other in a flow definition.

You have established the flow definition in the [scheduler \[Seite 15\]](#).

Process Flow

1. The system starts the sequence of processing steps in accordance with the planning specified in the scheduler. The objects to be processed are selected on the basis of the following logic:
 - Objects that were processed successfully in the previous flow step but have not yet been processed successfully in the current flow step are transferred from the selection to be processed. These objects are called *primary objects*.
 - The system also reads the dependent objects for the current flow step for all selected objects. These dependent objects are called *secondary objects*. If a primary object has one or more secondary objects that were not successfully processed in the previous flow step, neither the primary object nor its associated secondary objects can be processed in the current flow step.
2. The system executes the processing steps.
3. When the system has executed all processing steps, you receive a mail to inform you that the planned sequence of processing steps has been completed. This mail is generated using the workflow link.
4. You can reprocess the faulty objects directly from the mail.
5. The user can view the processed objects with their processing status in the monitor of the Schedule Manager.
6. The user processes the objects with errors.
7. The user retriggers the processing step sequence from the mail in his office inbox.
8. The system now processes the objects in the selection that were processed with errors in the first run of the processing sequence, or whose processing was forced manually.

For each object, processing starts with the step that was not processed correctly.

The subsequent processing steps are then started automatically in the defined sequence.



If an object that was processed without errors in the first run of the processing step sequence is changed before the second run of the processing step sequence, these changes do not result in an automatic reprocessing of the object. If you want the object to be reprocessed, manually set the corresponding processing status.

Multilevel Worklist: Process Flow

The production order 1000000 is included with all other production orders of plant 1000 in period-end closing of *Product Cost by Order*. Work in process for the order is calculated as EUR 1,000 and settled to *Financial Accounting (FI)*. All processing steps (for example, *WIP calculation* and *settlement*) of the processing step sequence were also performed without out errors.

However, not all production orders were processed without errors. In this way, for example, the production order 1000005 receives the processing status *error* for the processing step *WIP calculation* and the production order 1000010 receives the processing status *error* for the processing step *Settlement*.

After work in process has been settled for all production orders, including production order 1000000, additional costs amounting to EUR 200 are posted to production order 1000000.

Remove the errors for the production orders 1000005 and 1000010 and reinitiate the processing step sequence. The production orders 1000005 and 1000010 are reprocessed. Production order 1000000 is not processed as it was processed without errors in the first run.

If you want to force a reprocessing of production order 1000000, you can manually set the corresponding processing status.

See also:

[Schedule Manager: Worklist \[Seite 62\]](#)

[Schedule Manager: Monitor - Working with the Object List \[Seite 77\]](#)

[Processing Status of Objects and Processing Steps \[Seite 91\]](#)

[Processing Worklists \[Seite 80\]](#)

Monitor - Working with the Object List

Definition

The monitor of the *Schedule Manager* is a tool for processing multilevel worklists. It contains information on active or completed jobs that were planned in the scheduler.

Use

This section provides information on using the object list in the monitor of the Schedule Manager.

For more information on jobs in the monitor, see [Using the Monitor \[Seite 35\]](#).

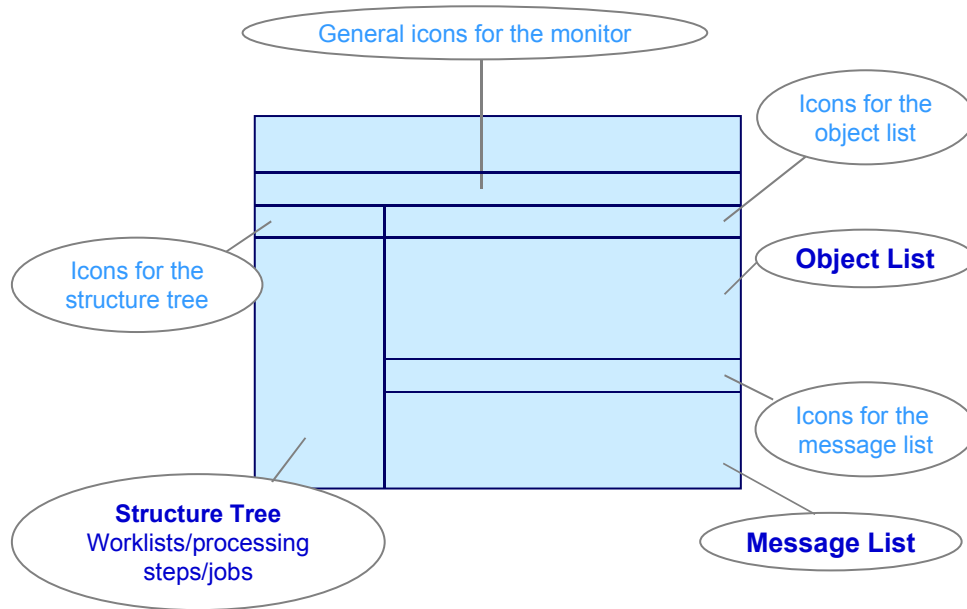
The monitor performs the following functions:

- Displays the available worklists and their processing steps
- Keeps track of the processing status of each object (such as product cost collectors, WBS elements, internal orders, and production orders) and each processing step
- Assists you in analyzing the causes of errors for the objects and processing steps
- Displays the processing status of objects (particularly for the objects with errors)
- Shows the hierarchical relationships between the objects
- Compares object hierarchies with each other
- Controls whether objects enter the next processing step
- Sends objects to the responsible person for further processing

You can set the processing status so that objects of negligible value, for example, are not reprocessed in a processing step even if errors are issued for the objects in that step. In this case, it is not necessary to remove the cause of the error.

Monitor - Working with the Object List

Structure



The monitor is divided into the following screen areas:

Structure Tree with Worklists/Processing Steps/Jobs

The structure tree containing the worklists, processing steps and jobs is displayed on the left side of the monitor.

In the system, the worklists are updated separately for each scope of selection and processing step sequence. The processing step sequence is specified in the flow definition. You can view the worklists of the application component in which you are working, including their functions and objects.

The system displays the following information on each worklist:

- Number of objects in the worklist
- Processing steps
- Number of objects processed in each processing step
- Number of objects with errors in each processing step

Object List

The object list is located in the top right area of the screen. The object list can display one or more processing steps of a worklist. You select the processing steps in a worklist in the structure tree *worklists/processing steps/jobs*.

The object list includes the following information:

Monitor - Working with the Object List

- The objects for the selected processing steps. You can control which objects are displayed for each processing step by means of a filter that takes the individual processing statuses into account. In application toolbar *icons for object list*, choose the icon *Objects: Set status filter*
The system uses the last user-specific default setting of the dialog box. Note the default settings of this dialog window in the SAP standard system. The last user-specific setting of the dialog box is the default.
- The processing step performed for the object
- The processing status of the object for each processing step
- A check box you can select to indicate that you have processed the object in the worklist monitor
- The person responsible for the object (if any)

Message List

The message list is located in the lower right-hand portion of the screen. The message list displays messages for certain objects. You can use these messages to analyze the causes of error for each object and processing step.

See also:

[Schedule Manager: Multilevel Worklist \[Seite 62\]](#)

[Multilevel Worklist: Process Flow \[Seite 75\]](#)

For detailed information on using the monitor of the Schedule Manager, see the following section:

[Processing Status of Objects and Processing Steps \[Seite 91\]](#)

[Processing Worklists \[Seite 80\]](#)

Processing Worklists

Processing Worklists

Use

You process worklists in the monitor of the Schedule Manager.

Navigating to the Monitor

Procedure

1. You can access the monitor in the following ways:
 - a. In the Schedule Manager, there are three ways of accessing the monitor from the screen *Schedule Tasks for Task List*: By selecting a calendar day, selecting the desired tasks in the daily overview, and then choosing *Monitor daily overview* in the application toolbar; by choosing the *Monitor* icon above the daily overview; or by double-clicking on a task in the daily overview.
 - b. From the menu of the application component. For example, to access the worklists of the *Product Cost by Period* component, go to the menu of *Product Cost by Period* and choose *Period-End Closing* → *Schedule Manager* → *Worklist Monitor*. Choose the following: *Product Cost Collector* or *Cost Object Hierarchy*. The screen *Select Worklist* appears.
In the selection screen, use the possible entries function for the *Application* field to select the application component for which you want to call up the worklist (such as *Cost Object Controlling: Manufacturing Orders and Product Cost Collectors*). Usually the appropriate application will already be shown as a default. Enter the additional selection criteria such as the period and fiscal year, and choose *Enter*. You can further restrict the selection with the icon *Further selection criteria*.

Result

The screen *Schedule Manager: Monitor* appears.

Working with the Monitor of the Schedule Manager

Using the Monitor

The monitor enables you to analyze the objects processed in a worklist and control how the objects are processed further. The focus is on the objects for which errors were issued.

Objects for which errors were issued in a processing step do **not** enter the next processing step.

You can analyze and correct the error, or you can instruct the system to skip processing (that is, not to process the objects in the corresponding step). Skipping processing can be appropriate for objects that have errors but whose value is negligible.

You instruct the system to skip processing by setting the processing status to *skip processing*. The system then processes the object in the next processing step as if it had been processed without errors in the previous step.

If the object is part of a hierarchy, this can also prevent the object from stopping the entire hierarchy from being processed.

The functions *hierarchy display* and *hierarchy comparison* provide support when you search for errors.

Processing Worklists



When actual overhead rates were calculated in the period-end closing process, one production order was not processed due to faulty Customizing settings. The production order is assigned to a WBS element. When the cost accountant reviewed the objects in the monitor, he decided that the missing overhead is insignificant and can be disregarded, since the effort involved in finding and correcting the error would be out of proportion to the amount in question. The faulty production order should therefore be included in results analysis for the WBS element. To prevent the entire project to which the production order is assigned from being excluded from processing, the accountant sets the processing status of the production order for the step *overhead* to *Skip processing*.

On the other hand, it may sometimes be necessary to process an object again even though it has been processed without error. You do this by setting the status *Repeat processing*. This can be appropriate in cases where incorrect Customizing settings have resulted in improper values. The processing steps only recognize errors that are indicated as errors in the monitor. However, an object can be faulty in a business sense without necessarily generating an error message.

Such errors can only be found by checking the results by hand.

In the flow definition, you can specify that a manual check (usually by the cost accountant) should take place after certain processing steps.

To force renewed processing of the object after correcting the error (in our example, this would involve selecting another results analysis method in Customizing), you change the processing status to *Repeat processing*.



Suppose you want to calculate results analysis data for a WBS element. In Customizing, you selected a results analysis method that causes unrealized profits to flow into inventory valuation. However, you don't want unrealized profits to be inventoried.

It may also be necessary to force an object to be processed due to the fact that postings which debited an object after period-end closing made the period-end closing data obsolete.



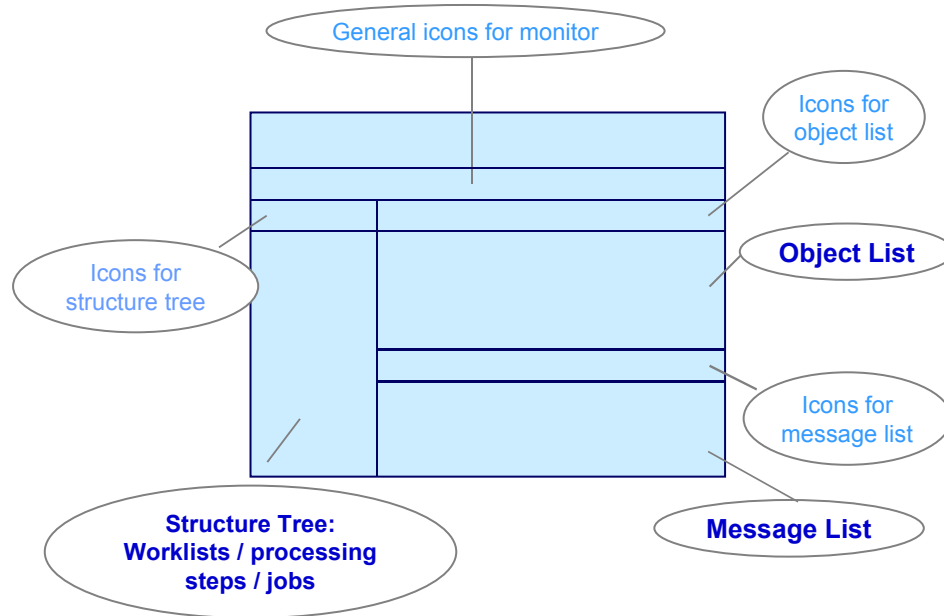
Suppose you have performed period-end closing for a production order. All processing steps of the production order were performed without error. The production order is then debited with additional costs that must be accounted for in period-end closing. You must therefore force the system to repeat the processing steps for the production order. In this case, you only need to force a repeat of the first processing step of the sequence. The production order is automatically included in all subsequent processing steps.

The monitor of the Schedule Manager is divided into three screen areas:

- Structure tree with worklists/processing steps/jobs
- Object list
- Message list (you have the choice between object messages, step messages, and messages of the originating objects)

Processing Worklists

Structure of the Monitor



(see also: [Schedule Manager: Monitor - Working with the Object List \[Seite 77\]](#))

Different functions are available in each screen area. These functions are described in the Procedure section below.

Procedure

Displaying Objects for One or More Processing Steps

You want to view the processed objects for one or more processing steps.



In the object list, you can choose to view all processed objects or only the objects with errors, for example, for one or more processing steps. Since the worklist's primary function is to assist you in analyzing the objects with errors, you should limit yourself to viewing only the objects that have errors. If you choose to view all processed objects, this will reduce system performance.

With the default settings of the standard system, objects with the following processing statuses are displayed:

- Flagged for postprocessing
- Processing forced manually
- Error
- Followup error

- Hierarchy error
- Minor error

For detailed information on the processing statuses updated for the objects, see the following section: [Processing Status for Objects \[Seite 91\]](#).

To view the objects, proceed as follows in the structure tree *worklists/processing steps/jobs*:

Double-click on a **processing step** in the *structure tree with worklists/processing steps/jobs* to generate the *object list* for that processing step. If there are any messages that refer to the processing step as a whole (rather than to specific objects), they are displayed in the *message list*.

Double-click on a **worklist** in the *structure tree with worklists/processing steps/jobs* to generate the *object list* for all processing steps. If there are any messages for the worklist, they are shown in the *message list*.

You can also select the processing steps with the context menu. Proceed as follows:

a) To display all processing steps for the objects in a worklist:

Select a worklist with the right mouse button and choose *Object list - All steps* in the context menu. The object list for the worklist is generated.



The object list can display up to ten processing steps. If there are more than ten steps, the dialog box *Select Processing Steps* appears, in which you can select a maximum of ten steps.

a) To display individual processing steps for the objects in a worklist:

If you don't want to see all processing steps for a worklist, select a worklist with the right mouse button and choose *Object list - Select steps* in the context menu. The dialog box *Select Processing Steps* appears, in which you can select a maximum of ten steps.



When you process worklists for manufacturing orders, the *object list* has a column for the material number. The material shown in this column is usually the material that is manufactured with the order shown in that row.


In joint production, the column shows the following:

- For the order header (i.e., for the object "ORD order"):
 - If a process material is being used, the process material is shown.
 - Otherwise the leading co-product is shown. The leading co-product is the co-product shown in the first order item.
- For the order item (i.e., for the object "OIT order item"), the material produced with that order item is shown.

Displaying the Processing Status for Each Object and Processing Step

The *object list* shows the processing status for each object and processing step. A column is displayed in the *object list* for each processing step selected in the structure tree

Processing Worklists

worklists/processing steps/jobs. This column shows the processing status of each object. The processing statuses are represented as icons. If you want to know which processing status is assigned to which icon, then click on the corresponding *Legend* icon above the *object list*. The screen *Processing Status: Icon Legend* appears. The  icon in the dialog box provides explanations of the processing statuses.

For detailed information on processing statuses, including examples, see the following section: [Processing Status for Objects \[Seite 91\]](#)

Filtering Objects

You can also display objects of other processing statuses through a status filter. To do so, click on the filter icon (*Objects: Set status filter*) above the object list. Select the relevant processing status and press *Enter*. All objects that have the corresponding processing status are displayed in the object list.

You can also filter objects according to criteria other than the processing status (function of the ABAP List Viewer).

Deciding Whether to Process Objects Further

You now have a list of objects. These are normally the objects with errors. You must decide whether to continue processing these objects. You have the following options:

- You correct the errors.
- You instruct the system to continue processing the objects even though you have not corrected the errors. This is appropriate for objects whose value is insignificant.

Reports on the objects are available to help you reach a decision.

Executing Reports

Reports can help you determine the following:

- Whether the value of the object is negligible

In this case, you instruct the system not to process the object. Otherwise, you correct the error.
- Whether the values calculated for an object are acceptable in a business sense. You may also want to do this for objects that were processed without error (processing status *OK*).

If this is not the case, then once you have corrected the problem (the error may have been caused by faulty Customizing settings), you can force *further processing*. You can also assign further reports with the function *report assignment* (if supported).

To view a report on an object, position the cursor on the object in the *object list* and click on the *Execute report* icon above the *object list*. Then enter the required parameters and select the report you want to see (only for projects; production orders do not allow reports to be selected). On the basis of the data in the report, you can now decide how the object is to be handled.

To analyze the error for the objects with the processing status *Error*, look at the messages for the object and processing step.

Messages for One or More Objects

To see the messages that were issued for one or more objects:

1. Select the objects.

Processing Worklists

In the *object list*, select an object by positioning the cursor on the object, holding down the left mouse button, and pressing the Ctrl key on the keyboard. Continue to select further objects in the same way.

2. View the messages.

To see the messages for the selected objects, click the *Display object messages* icon above the *object list*. You can also display the messages for **one** line of the object list by double-clicking on the line.

In the pull-down menu of the *Display object messages* icon above the object list, you can choose between the following types of messages:

- *Object messages*
- *Step messages*
- *Messages of originating objects*

The object messages appear in the *message list*.

3. View the long text of the messages.

To analyze the error, you will normally need to look at the long text of the message. To see the long text, double-click on the message in the *message list*.

Step-Specific Messages (Messages Without Reference to a Specific Object)

To see the messages for a particular processing step, go to the object list and choose the drop-down menu of the *Display object messages* icon, and then choose *Display step messages*. The step-specific messages are displayed in the message list.

Messages of Originating Objects

You should look at the *messages of the originating objects* particularly when you want to analyze objects with the processing status *Followup error*.

Skipping Processing Manually

You have decided to have the system skip a processing step for one or more objects.

- In the *object list*, select the step statuses that you want to change. Click the icon *Set processing status*.

If you select the status *Skip processing*, the system changes the processing status to *Skipped manually*. The object is not included in this processing step during the next execution of the processing step sequence. The object enters the subsequent processing step as if it had been processed without errors.

Forcing Processing Manually

There are two ways to force the system to reprocess objects:

Position your cursor in the *object list* and press the right mouse button. In the context menu, choose *Repeat processing*. The system changes the processing status to *Postprocessing forced manually*. The next time the processing step sequence is performed, the object is included in that processing step and in all subsequent processing steps.

You can also force the system to process objects by selecting the objects and clicking the icon *Set processing status* (icon in the object list) and choosing *Repeat processing*. You can select multiple objects at the same time with the following methods:

Processing Worklists

- To select an entire column, click on the column header.
- To select individual columns, lines, or cells, hold down the *Ctrl* key and click.

The system changes the processing status to *Postprocessing forced manually*. The next time the processing step sequence is performed, the object is included in that processing step and in all subsequent processing steps.

Editing Master Data

From the *object list*, you can access master data maintenance for the object. To do so, position your cursor on the relevant object and click on the *Edit master data* icon above the object list. The system starts a new session. You are now in the change mode of the relevant object. You can close the session by going back (green arrow).



The error message *Maintain the settlement rule of the sender* is issued for a production order for the processing step *Settlement*. To maintain the settlement rule, call up the production order.

Display as Hierarchy

In the hierarchy view, the objects are displayed according to their hierarchical relationship to each other.

Select one or more objects in the object list and choose *Display as hierarchy*. The screen *Object list - hierarchy view* appears. In the hierarchy view, the object or objects you branched from are selected.

Expand the object hierarchy. This provides a quick overview of the relationships between the faulty objects. You can process objects to which a particularly large number of faulty objects are assigned.

To go back to the object list from the hierarchy, choose *Back to object list*.



In the *object list hierarchy view* screen, you can only change the processing status of individual objects. Mass changes to the processing status of the objects is only possible in the *object list*.

In addition, a requirement for the *display hierarchy* function is that the objects have hierarchy information in the worklist. This is the case in the *Project System* and for cost object hierarchies.

Hierarchy Comparison

Objects that have hierarchy information (such as projects and cost object hierarchies) can be compared with each other using the *hierarchy comparison* function. This allows you to more efficiently find errors that are due to differences in the values of two periods. Comparing the data of two periods is particularly useful because the items and values of the objects in the hierarchy can change between periods.

Processing Worklists

In the project system, it is also possible to compare two versions of an original structure. The changes that were made to the version are shown in a separate window.

Before you start the comparison, you specify which two data structures you want to compare. You define one of the two structures as the original structure and the other as the version. During the comparison, the two structures are represented in the form of a hierarchical structure tree in a separate screen area. You can compare individual objects, subhierarchies, or the two complete structures. The results of the comparison are grouped into different categories:

- New objects (which objects were added to the version?)
- Changed objects (which object attributes were changed, and how?)
- Nonexistent objects (which objects do not exist or were deleted?)
- Hierarchy position of changed objects (which objects changed their position in the hierarchy?)

For detailed information on the hierarchy comparison, see the documentation under:

Logistics → Project System → Collaborative Engineering and Project Management → Monitoring → Comparing Two Folders

The Schedule Manager only supports the comparison function; synchronization is not possible.

Sending Objects

Select one or more objects in the object list and choose the *Send* icon above the object list. To receive a log of the transmission, choose *Send log*. If you choose *Send to agent*, the person responsible for the object receives a mail in his or her inbox (*Office → Workplace → Inbox*) with information on the required change. From this mail, you can access the screen *Schedule Manager: Monitor* which displays only the sent objects. Note that the filter function in the icons of the object list (*Objects: Set Status Filter*) is not available when you access the monitor through your office inbox.



To be able to use the function for sending objects, the following settings must be made in each client:

1. Choose *Tools → Business Workflow → Development → Definition Tools → Tasks/Task Groups → Change*.
2. Select task type *Workflow template* and enter task **20001060** (SCMA_WL_DISP). The screen *Workflow Template: Change* appears.
3. Select the tab *Triggering events*. Activate the event linkage by clicking the icon *Event linkage active* in the first column. The light symbol changes to green.
4. Save the data.

If no agent can be determined for the object, the dialog box *Entry of Responsible Agent* appears, in which you can specify the user responsible for further processing.

The object list contains the field *Agent ID* next to the existing column headers. This field shows the agent responsible for the object. The agent can be an R/3 user or a node in the organization plan.

Processing Worklists

On the basis of the organizational data of the objects (such as the plant, company code, and so on), you can specify criteria for each object type that are used to determine the agent.

You specify the agents responsible for the objects under:

Schedule Manager → Extras → Settings for worklist → Maintain responsibilities for objects.

For detailed information on maintaining responsibilities, refer to the SAP Library under:

Human Resources → Personnel Management → Organizational Management → Integration with SAP Business Workflow → Role Resolution → Role Definition → Define Roles Using Responsibilities

[Define Roles Using Responsibilities \[Extern\]](#)

Flagging Checked Objects

You can flag objects to indicate that you have checked them.

To do this, either select the check box **Object checked** directly, or select one or more objects and click the icon *Change "checked" indicators* and then in the pull-down menu:

- *Select indicators*: selects the check boxes for the selected objects
- *Deselect indicators*: deselects the check boxes for the selected objects
- *Toggle checked <-> not checked*: *Toggles the status for the selected objects*

The indicators are only visual and have no effect on processing.

Displaying Other Worklists

To display other worklists, click the *Other worklist* icon at the top of the screen. Then enter the required data and choose *Enter*. This function is only available when you access the Schedule Manager through the application components. When you access the Schedule Manager through a mail in your Office inbox, this function is **not** available.

Triggering Reprocessing of the Worklist

Prerequisites

Before you trigger reprocessing of the worklist, you must have done the following:

- Corrected the errors of the faulty objects, or instructed the system to skip processing for those objects
- Forced reprocessing of objects that are faulty in a business sense

Procedure

To trigger reprocessing, access the mail that was sent to you through the workflow. Start reprocessing from this mail.

Result

The following objects are processed:

- Objects that had errors in the previous execution of the processing step sequence and whose errors have been corrected

These objects are processed starting with the processing step in which the error occurred.

Processing Worklists

- Objects that had errors in the previous execution of the processing step sequence and for which the processing status *Skip processing* was set
These objects are processed starting with the processing step following the processing step for which the status *Skip processing* was set.
- Objects that received the status *Postprocessing forced manually*
These objects are processed starting with the processing step for which the status *Postprocessing forced manually* was set.

See also:

Many of the icons displayed above the individual screen areas support the standard functions of the ABAP List Viewer (ALV). These icons are not explicitly explained here. The same applies to the standard ALV functions in the context menu (right mouse button). For information on the ALV, refer to the SAP Library under *Cross-Application Components (CA) -> General Application Functions -> ABAP List Viewer*, or under: [Functions of the ABAP List Viewer \[Extern\]](#)

Choosing Objects for Processing

Choosing Objects for Processing

Use

An advantage of this [worklist \[Extern\]](#) is that the objects for processing only have to be selected once per flow definition.

Prerequisites

You are in [flow definition \[Extern\]](#) in Schedule Manager (*Extras* → *Flow definition* → *Edit flow definition*).

Procedure

20. Choose a processing step in the navigation area.
On the right of the screen, you see *Flow definition: Task details*.
21. Enter a new variant for this processing step and define a name for this new variant in the *Variant* field .
22. Choose *Create variant*.
The *Variant Maintenance* screen appears. Program <program name>, variant <variant name>.
Here you can determine the scope of selection of the program variants for the flow definition.
23. To be able to choose more extensive selection criteria, first complete the required entry fields, for example *Period* <006>, *Fiscal year* <2000>.
24. You can now define further selection parameters using the various pushbuttons that are offered in dialog boxes.
25. Choose *Attributes*.
The *ABAP: Save Attributes of Variant* <Variant name> screen appears.

Processing Status of Objects

Use

The processing status of an object tells you whether processing was successful in a specific processing step.

For every object listed in the [worklist of the Schedule Manager \[Seite 62\]](#), a processing status is indicated for each processing step. The system sets the processing status during processing. This status can be changed manually in the monitor.

In addition to information on the success of processing, the processing status shows whether further processing of an object is allowed. This ensures that an object which was not processed in a processing step due to errors is not passed on to the next step.

Integration

The processing status is displayed in the monitor. The status is not updated to the processed object. For more information on the monitor, see the following section: [Schedule Manager: Monitor - Working with the Object List \[Seite 77\]](#)



The examples described below are based on the use of the multilevel worklist in *Cost Object Controlling* and the *Project System*.

Features

Through the processing status, the worklist of the Schedule Manager controls whether an object is processed in the following processing step. The system automatically sets the following processing statuses:

Automatic Processing Statuses

- Not processed
The object has not been processed yet.
- OK
The object was processed without errors.
- Minor error
A warning or error occurred while the object was being processed. However, this error does not prevent further processing in a following step.
- Flagged for postprocessing
This status is set in the following case:
The object must be processed in a processing step because, for example, processing was forced. In the subsequent processing step, the object has one of the following processing statuses:
 - OK
 - *Not relevant*

Processing Status of Objects

- *Minor error*
- *Filtered*

Since the object normally would not be processed when it has this processing status, the system sets the processing status of the subsequent processing step to *Flagged for postprocessing*. This ensures that an object processed in one step is always included in the subsequent processing steps. The processing status is reset to *Flagged for postprocessing* regardless of whether the object was changed in the previous processing step.

- **Error**

An error occurred while the object was being processed that prevents further processing. You can either correct the error or instruct the system to include the object in the subsequent processing step despite the error (processing status *Processing skipped manually*).

- **Followup error**

An object receives the processing status *Followup error* for a processing step if it belongs to a group of objects that are processed together and this group contains one or more objects that were processed with errors (processing status *Error*), or were not processed, in the **previous** processing step.

The object with the processing status *Followup error* is not processed in the processing step for which it received that error. This ensures that the entire group of objects can only be processed when none of the objects contains an error that hinders the processing of an object in the group.

See the end of this text for an example of the processing statuses *Followup error* and *Hierarchy error*.

- **Hierarchy error**

An object receives the processing status *Hierarchy error* for a processing step if it belongs to a group of objects that are to be processed together and this group contains one or more objects that were processed with errors in the **current** processing step (processing status *Error*).

The object with the processing status *Hierarchy error* is not processed in the processing step for which it received that error. This ensures that the entire group of objects can only be processed when none of the objects contains an error that affects processing.

See the end of this text for an example of the processing statuses *Followup error* and *Hierarchy error*.

- **Filtered**

The user has excluded the object from processing by defining an application-specific filter.

- **Not relevant**

- The object is not relevant for processing in this processing step due to its attributes derived from Customizing or master data.
- The object is not relevant due to a status that has been set for it.

Processing Status of Objects



Example 1:

According to the selection parameters, all production orders of a plant are selected for variance calculation in period-end closing of *Product Cost by Order*. No variance key has been specified for some of the production orders. These production orders are not included in variance calculation. The production orders receive the processing status *Not relevant*.

Example 2:

According to the selection parameters, all production orders of a plant are selected for overhead calculation in period-end closing of *Product Cost by Order*. A system status or user status is set for some production orders, which does not allow processing. Examples of this are the system statuses *open*, *closed*, *deletion flag* and *locked*.



In some cases, a given status may make an object relevant or irrelevant only in certain processing steps.



Example:

A production order has a settlement rule with the settlement type FUL (full settlement). If neither the status DLV (delivered) nor the status TECO (technically completed) is set for the object, it is *not relevant* for *variance calculation* and receives the corresponding processing status. The order is relevant for the other processing steps of period-end closing such as template allocation, overhead calculation, calculation of work in process and settlement, regardless of whether one or none of the two statuses is set.

As a rule, with the processing status *not relevant*, no messages are issued for the object and processing step. To see the messages that caused the processing status to be set to *not relevant*, you can have the system process the object individually. Possibly, you can also force the issuance of messages through user-defined error management by putting the message type at the top.



For WIP calculation and results analysis, you can force the system to issue messages despite the status *not relevant*. You can guarantee the issuance of messages by setting the indicator *Log information messages*. If the indicator is set, all information messages are issued including those from processing within and outside of the multilevel worklist, even if the status *not relevant* is set for an object and processing step.

Manual Processing Statuses

You can set the following processing steps manually:

- Processing skipped manually

Processing Status of Objects

In the object list, the user manually sets the processing status of an object and processing step to *Skip processing*. The system changes the processing status to *Processing skipped manually*.

The object with this processing status is not included in the relevant processing step, and enters the next processing step.

- Processing forced manually

In the object monitor, the user manually sets the processing status of an object and processing step to *Repeat processing*. The system changes the processing status to *Processing forced manually*. The object is processed in the relevant processing step again even if the processing step had been executed without error.



If an object was processed without error in an update run for project interest calculation, new interest calculation can only be triggered if the previous interest calculation is first reversed. If no reversal is carried out, the object is not included in the recalculation of interest, despite its processing status.

Example

Example of the processing statuses *Follow-up error* and *Hierarchy error*:

During the calculation of actual overhead rates in period-end closing, a production order assigned to a WBS element was not processed due to deficient Customizing settings. The production order receives the processing status *Error* for the processing step *Actual overhead*.

In addition to this production order, other production orders are also assigned to the WBS element. All other production orders assigned to the WBS element receive the processing status *Hierarchy error* for the processing step *Actual overhead*. This status indicates that an object of an object group (all production orders assigned to a WBS element) contained errors in the processing step.

The processing step following the processing step *Actual overhead* is *Results analysis*. During results analysis for the WBS element, the data is included that was updated to the production orders assigned to the WBS element. Because one of the production orders assigned to the WBS element was marked as faulty and all other production orders received the corresponding processing status *Hierarchy error* in the processing step *Actual overhead*, the processing step *Results analysis* can not be carried out successfully for the WBS element.

The WBS element receives the processing status *Followup error* for the processing step *Results analysis*.

If you decide that the deficit resulting from the missing overhead is so low that it can be ignored, you can avoid time-consuming error analysis by specifying a processing status that prevents the production order from being included in the calculation of overhead. All production orders, including the one for which no overhead was calculated, are included in results analysis.

If the value of the production order is significant, you must remove the cause of error.

Additional Information

Note the following:

- There is no one-to-one assignment of message types to processing statuses in the worklist of the Schedule Manager. For example, not every message of message type "E" (error)

Processing Status of Objects

results in the processing status *Error* for an object in the Schedule Manager in that processing step.



The system issues the message CK 214 as an error message.

If this message is based on target cost version 0 in variance calculation, the processing status *Error* is set in the worklist of the Schedule Manager.

If this message is based on a target cost version in variance calculation that is not 0, the processing status *Minor error* is set in the worklist of the Schedule Manager.

- Some messages are issued with different message types based on the object to be processed. Different processing statuses are set in the Schedule Manager.



In variance calculation, if message KV 158 is based on a single production order, the system issues this message as an error message. The processing status *Error* is set in the worklist of the Schedule Manager.

In variance calculation, if this message is based on a product cost collector to which multiple production orders are assigned, the system issues a warning message. The processing status *OK* is set in the worklist of the Schedule Manager.

Note the following:

If you are using the multilevel worklist of the *Schedule Manager*, the objects with the processing status *Error* for a processing step are not processed in the subsequent step. In many cases, the processing status *Error* is set because a message was issued with the message type "E" (error). When calculating variances and scrap variances, you can use *user-defined error management* to influence the message type for a large number of messages. In the *worklist of the Schedule Manager*, this can result in a different processing status being updated to the processing step. This enables you to ensure that objects are processed in the subsequent processing step, for example.



A message was issued as an error message in variance calculation. The processing status *Error* is issued in the monitor.

The error is not serious, so you decide that the object should be processed in the next processing step. You therefore use user-defined error management to specify a message type of lower severity (such as "W" for a warning message). This ensures that the object does **not** receive the processing status *Error* for the processing step in the worklist due to this message. The object then enters the subsequent processing step.

No Processing Status Output

It may occasionally happen that no processing statuses are issued. See the following for an example of this: [Separation of Value Calculation from Goods Movements: Effects on Cost Object Controlling \[Extern\]](#).

See also:

For information on the following topics and many others, see the section [Processing Worklists \[Seite 80\]](#):

Processing Status of Objects

- When processing should be skipped or forced manually
- How to decide whether to do so
- How you can change the processing status

For detailed information on user-defined error management, see the following section:

[User-Defined Error Management in Cost Object Controlling \[Extern\]](#)

Scenarios for the Schedule Manager

Purpose

You can use the elements of the Schedule Manager in different scenarios in various combinations.

Prerequisites

You are working on a complex process, which consists of many different business transactions (tasks).

Process Flow

You decide which functions ([scheduler \[Seite 15\]](#), [monitor \[Seite 35\]](#), [flow definition \[Seite 42\]](#), [multilevel worklist \[Seite 62\]](#)) you require to execute and monitor your process.

You can choose between three scenarios:

- [Starting Transactions/Reports Online, Scheduling Jobs \[Seite 98\]](#)
- [Starting Transactions/Report Online, Scheduling Jobs and Job Chains \[Seite 99\]](#)
- [Starting Transactions/Reports Online, Scheduling Jobs and Job Chains, Worklist \[Seite 100\]](#)

Starting Transactions/Reports Online, Scheduling/ Monitoring Jobs

Use

Not only can you start transactions and reports online, and schedule jobs in the task list with this scenario, but you can also monitor the scheduled jobs in the [monitor \[Seite 35\]](#) while they are being executed.

Integration

You need the [scheduler \[Seite 15\]](#) and monitor from the Schedule Manager for this scenario.

Prerequisites

The application contains reports for your processing, which you can schedule as jobs.

Features

This scenario contains the following work steps:

- [Creating a Task List \[Seite 19\]](#)
- [Inserting a Task/Task Group in the Task List \[Seite 20\]](#)
- [Scheduling Tasks in the Daily Overview \[Seite 30\]](#)
- [Monitoring Jobs in The Monitor \[Seite 35\]](#)

Activities

In this scenario, you choose a task list in the scheduler or create a new task list.

You insert tasks in the daily overview. You can monitor the jobs in the monitor by selecting a task with the right mouse button and choosing *Monitor*.

Starting Transactions/Reports Online, Scheduling Jobs/Job Chains, Monitoring Jobs

Use

Not only can you start transactions and reports online, and schedule jobs/job chains from the flow definition in the task list with this scenario, but you can also monitor the scheduled jobs in the [monitor \[Seite 35\]](#) while they are being executed.

Integration

For this scenario, you need the following elements: the [scheduler \[Seite 15\]](#), the [monitor of the Schedule Manager \[Seite 35\]](#), and the [flow definition \[Seite 42\]](#).

Prerequisites

The application contains reports for your processing, which you can schedule as jobs.

Features

This scenario contains the following work steps:

- [Creating a Task List \[Seite 19\]](#)
- [Inserting a Task/Task Group in the Task List \[Seite 20\]](#)
- [Scheduling Tasks in the Daily Overview \[Seite 30\]](#)
- [Monitoring Jobs in The Monitor \[Seite 35\]](#)
- [Creating a Flow Definition \[Seite 42\]](#)

Activities

In this scenario, you choose a task list in the scheduler or create a new task list.

You insert tasks in the daily overview. You can monitor the jobs in the monitor by selecting a task with the right mouse button and choosing *Monitor*.

You can also insert job chains (that you specified in a flow definition) into the task list and schedule them in the daily overview. You can also use the monitor to check processing of the job chains.

Starting Transactions/Reports Online, Scheduling Jobs and Job Chains, Worklist

Use

Not only can you start transactions and reports online, and schedule jobs in the task list with this scenario, but you can also monitor the scheduled jobs in the [monitor \[Seite 35\]](#) while they are being executed. You can also use worklists and check them in the [object monitor \[Seite 77\]](#) in the generalized worklist.

Integration

For this scenario, you need the [scheduler \[Seite 15\]](#), the monitor of the Schedule Manager, the [flow definition \[Seite 42\]](#) and the [multilevel worklist \[Seite 62\]](#).

Prerequisites

The application contains reports for your processing, which you can schedule as jobs.

You are using worklists.

Features

This scenario contains the following work steps:

- [Creating a Task List \[Seite 19\]](#)
- [Inserting a Task/Task Group in the Task List \[Seite 20\]](#)
- [Scheduling Tasks in the Daily Overview \[Seite 30\]](#)
- [Monitoring Jobs in The Monitor \[Seite 35\]](#)
- [Creating a Flow Definition \[Seite 42\]](#)
- Working with the [Worklist Monitor \[Seite 77\]](#)

Activities

In this scenario, choose a task list from the scheduler, or create a new one. Only use task lists that have the task type *flow definition*.

You insert tasks in the daily overview. You can monitor the jobs in the monitor by selecting a task with the right mouse button and choosing *Monitor*.

In addition to the scheduler, monitor and flow definition, you can also work with a generalized worklist. Objects that are to be processed in a processing step sequence are managed in the worklist. Define the processing step sequence in the flow definition.

The worklist monitor shows you, for example, which objects were processed without errors, and which objects could not be processed. You can display information on the cause of errors, and thus control the way in which the object is processed further.

The generalized worklist ensures that when processing step sequences are repeated, the system only reprocesses the objects that previously contained errors and those for which you manually forced reprocessing.

Starting Transactions/Reports Online, Scheduling Jobs and Job Chains, Worklist