

**JP1 Version 12**

**JP1/Client Process Automation Configuration and  
Administration Guide**

**3021-3-D37-30(E)**

# Notices

## ■ Relevant program products

For details on supported operating system versions and service packs, patches, and other prerequisites for the target product, see the *Release Notes*.

*JP1/Client Process Automation (For Windows 10, Windows 8.1, Windows 8, Windows 7):*

P-2A12-3BCL JP1/Client Process Automation version 12-50

*JP1/Client Process Automation Option for Automatic Job Management System 3 (For Windows Server 2019, Windows Server 2016, Windows Server 2012, Windows Server 2012 R2):*

P-2A12-3CCL JP1/Client Process Automation Option for Automatic Job Management System 3 version 12-50

## ■ Trademarks

HITACHI and JP1 are either trademarks or registered trademarks of Hitachi, Ltd. in Japan and other countries.

Excel is either a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries.

Microsoft is either a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries.

Windows is either a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries.

Windows Server is either a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries.

Other company and product names mentioned in this document may be the trademarks of their respective owners.

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>).

Portions of this software were originally based on the following:

- software copyright (c) 1999, IBM Corporation., <http://www.ibm.com>.

## ■ Microsoft product name abbreviations

This manual uses the following abbreviations for Microsoft product names.

Abbreviation	Full name or meaning
Excel	Microsoft(R) Excel
	Microsoft(R) Office Excel
Windows 7	Microsoft(R) Windows(R) 7 Enterprise
	Microsoft(R) Windows(R) 7 Professional
	Microsoft(R) Windows(R) 7 Ultimate
Windows 8	Windows(R) 8 Enterprise
	Windows(R) 8 Pro
Windows 8.1	Windows(R) 8.1 Enterprise
	Windows(R) 8.1 Pro
Windows 10	Windows(R) 10 Enterprise

Abbreviation		Full name or meaning
		Windows(R) 10 Pro
Windows Server 2012	Windows Server 2012	Microsoft(R) Windows Server(R) 2012 Datacenter
		Microsoft(R) Windows Server(R) 2012 Standard
	Windows Server 2012 R2	Microsoft(R) Windows Server(R) 2012 R2 Datacenter
		Microsoft(R) Windows Server(R) 2012 R2 Standard
Windows Server 2016		Microsoft(R) Windows Server(R) 2016 Datacenter
		Microsoft(R) Windows Server(R) 2016 Standard
Windows Server 2019		Microsoft(R) Windows Server(R) 2019 Datacenter
		Microsoft(R) Windows Server(R) 2019 Standard

*Windows* is sometimes used generically, referring to Windows Server 2019, Windows Server 2016, Windows Server 2012, Windows 10, Windows 8.1, Windows 8, and Windows 7.

## ■ Restrictions

Information in this document is subject to change without notice and does not represent a commitment on the part of Hitachi. The software described in this manual is furnished according to a license agreement with Hitachi. The license agreement contains all of the terms and conditions governing your use of the software and documentation, including all warranty rights, limitations of liability, and disclaimers of warranty.

Material contained in this document may describe Hitachi products not available or features not available in your country.

No part of this material may be reproduced in any form or by any means without permission in writing from the publisher.

## ■ Issued

Mar. 2021: 3021-3-D37-30(E)

## ■ Copyright

Copyright (C) 2019, 2021, Hitachi, Ltd.

Copyright (C) 2019, 2021, Hitachi Solutions, Ltd.

## Summary of amendments

The following table lists changes in this manual (3021-3-D37-30(E)) and product changes related to this manual.

Changes	Location
The setup procedure for the AJS3 - View host was modified in the section for setting up the AJS3 linkage function.	3.3.2(3)
The execution user setting function was added, which enables a virtual user name to be specified as an execution user in the item definition.	4.4, 4.5.1, 5.3.1, 5.3.2, 5.5.1, 7.2.1
The <code>cpajobprint</code> command to output job or item definition information was added.	4.5.1, 9.
The following environment variables were added to the table of the command execution environment for action items: <ul style="list-style-type: none"> <li>CPAJDS</li> <li>CPAJCS</li> </ul> A description of how to use environment variables when an item is executed was also added.	5.3.1(3)table5-1, Appendixes B.2
The virtual user name was added in the subsections of <i>List of settings</i> and <i>Settings</i> for the following items. The description of the return value 126 was also modified. <ul style="list-style-type: none"> <li>Action items <ul style="list-style-type: none"> <li>Command execution</li> <li>Command execution by list</li> <li>Command execution with recovery option</li> </ul> </li> </ul>	8.2.1, 8.2.2, 8.2.3
The virtual user name was added in the subsections of <i>List of settings</i> and <i>Settings</i> for the following items. The description of the return value 0 was also modified. <ul style="list-style-type: none"> <li>Event monitor items <ul style="list-style-type: none"> <li>Monitor file</li> <li>Monitor any event</li> </ul> </li> </ul> The description of the return value 22 for the following item was also modified: <ul style="list-style-type: none"> <li>Event monitor item <ul style="list-style-type: none"> <li>Monitor any event</li> </ul> </li> </ul>	8.3.1, 8.3.2
The virtual user name was added in the subsections of <i>List of settings</i> and <i>Settings</i> for the following items. <ul style="list-style-type: none"> <li>Execution condition item <ul style="list-style-type: none"> <li>Monitor file</li> <li>Monitor any event</li> </ul> </li> </ul>	8.4.1, 8.4.2
The maximum number of concurrent connections from CPA Option for AJS3 to CPA was changed.	8.4.3(5)
The function as well as the description of argument <code>-a virtual-user-name</code> of the following command were modified: <code>cpasetumap</code> command	9.



Changes	Location
The <i>Command execution as virtual user</i> column was added to the table for the command execution environment.	<i>Appendixes A.3</i>
The number of custom jobs for CPA linkage that can be concurrently executed from AJS3 can now be expanded.	<i>Appendixes B.3, Appendixes E.2tableE-2</i>
The following process was added cpamonflchk process	<i>Appendixes F</i>
Changed how the product name is expressed in this manual.	<i>Appendixes H.2</i>

In addition to the above changes, minor editorial corrections were made.

# Preface

This manual describes Client Process Automation (simply called *CPA* hereafter) and Client Process Automation Option for Automatic Job Management System 3 (simply called *CPA Option for AJS3* hereafter).

## ■ Intended readers

This manual is intended for:

- Those who design the operation of a system that uses CPA to run client applications automatically
- Those who design the operation of or operate a system that uses CPA Option for AJS3 and Automatic Job Management System 3 (simply called *AJS3* hereafter) to run client applications automatically

## ■ Organization of this manual

This manual is organized into the following parts.

### PART 1: Overview

Provides an overview of CPA.

### PART 2: Installation and Setup

Describes how to install CPA and CPA Option for AJS3 and set up configurations for them.

### PART 3: Operations

Describes the operations that can be performed by operators and the functions necessary for operations of CPA.

### PART 4: Planning

Describes the operations that can be performed by designers and the functions necessary for planning.

### PART 5: Reference

Describes CPA windows and items as well as the commands provided by CPA and CPA Option for AJS3.

## ■ Conventions: Fonts and symbols

The following table explains the text formatting conventions used in this manual:

Text formatting	Convention
<b>Bold</b>	Bold characters indicate text in a window, other than the window title. Such text includes menus, menu options, buttons, radio box options, or explanatory labels. For example: <ul style="list-style-type: none"><li>• From the <b>File</b> menu, choose <b>Open</b>.</li><li>• Click the <b>Cancel</b> button.</li><li>• In the <b>Enter name</b> entry box, type your name.</li></ul>
<i>Italic</i>	Italic characters indicate a placeholder for some actual text to be provided by the user or system. For example: <ul style="list-style-type: none"><li>• Write the command as follows: <code>copy source-file target-file</code></li><li>• The following message appears: A file was not found. (file = <i>file-name</i>)</li></ul>

Text formatting	Convention
	<p>Italic characters are also used for emphasis. For example:</p> <ul style="list-style-type: none"> <li>Do <i>not</i> delete the configuration file.</li> </ul>
Monospace	<p>Monospace characters indicate text that the user enters without change, or text (such as messages) output by the system. For example:</p> <ul style="list-style-type: none"> <li>At the prompt, enter <code>dir</code>.</li> <li>Use the <code>send</code> command to send mail.</li> <li>The following message is displayed: <code>The password is incorrect.</code></li> </ul>

The following table explains the symbols used in this manual:

Symbol	Convention
	In syntax explanations, a vertical bar separates multiple items, and has the meaning of OR. For example: A   B   C means A, or B, or C.
{ }	In syntax explanations, curly brackets indicate that only one of the enclosed items is to be selected. For example: { A   B   C } means only one of A, or B, or C.
[ ]	In syntax explanations, square brackets indicate that the enclosed item or items are optional. For example: [ A ] means that you can specify A or nothing. [ B   C ] means that you can specify B, or C, or nothing.
. . .	In coding, an ellipsis ( . . . ) indicates that one or more lines of coding have been omitted. In syntax explanations, an ellipsis indicates that the immediately preceding item can be repeated as many times as necessary. For example: A, B, B, . . . means that, after you specify A, B, you can specify B as many times as necessary.

## ■ Conventions: Version numbers

The version numbers of Hitachi program products are usually written as two sets of two digits each, separated by a hyphen. For example:

- Version 1.00 (or 1.0) is written as 01-00.
- Version 2.05 is written as 02-05.
- Version 2.50 (or 2.5) is written as 02-50.
- Version 12.25 is written as 12-25.

The version number might be shown on the spine of a manual as *Ver. 2.00*, but the same version number would be written in the program as *02-00*.

# Contents

Notices	2
Summary of amendments	4
Preface	6

## Part 1: Overview

<b>1</b>	<b>Overview of CPA</b>	<b>13</b>
1.1	Features of CPA	14
1.1.1	Understanding jobs	14
1.1.2	Actions, event monitors, and action flows	14
1.1.3	Execution date/time and execution conditions	15
1.1.4	State management for jobs and actions	17
1.1.5	Custom items	19
1.2	Job components	20
1.3	Job execution unit and job execution ID	21

## Part 2: Installation and Setup

<b>2</b>	<b>Installation and Configuration of CPA</b>	<b>22</b>
2.1	Requirements	23
2.2	Installation	24
2.2.1	Installing a new instance	24
2.2.2	Overwriting or upgrading an existing instance	25
2.2.3	Performing silent installation	26
2.2.4	Notes on installation	26
2.3	Configurations	28
2.3.1	Configurations for the Send mail item	28
2.3.2	Configurations for the Monitoring mail reception item	29
2.3.3	Configurations for the AJS3 linkage function	30
2.3.4	Using the Windows log-on function	32
2.3.5	Using the Windows event monitor function	33
2.3.6	Creating a custom item	33
2.3.7	Suppressing the definition function in the operating environment	34
2.4	Uninstallation	35
2.5	Port number	36
2.6	Language setting	37

## **3 Installation and Configuration of CPA Option for AJS3 38**

- 3.1 Requirements 39
- 3.2 Installation 40
  - 3.2.1 Installing a new instance 40
  - 3.2.2 Overwriting or upgrading an existing instance 41
  - 3.2.3 Notes on installation 41
- 3.3 Configurations 42
  - 3.3.1 Configurations of the message communication service 42
  - 3.3.2 Setting up components for the AJS3 linkage function 42
  - 3.3.3 Clustering 44
- 3.4 Uninstallation 47
- 3.5 Port number 48

## **Part 3: Operations**

### **4 Operation of CPA 49**

- 4.1 Operating jobs 50
  - 4.1.1 Defining an item 50
  - 4.1.2 Registering a job 51
  - 4.1.3 Checking the state of a job 52
  - 4.1.4 Job operation function 52
- 4.2 Operating jobs that use automatic logon 53
  - 4.2.1 Windows log-on function 53
  - 4.2.2 Notes on Windows log-on function 53
- 4.3 Using the AJS3 linkage function 54
  - 4.3.1 List of CPA Option for AJS3 functions 54
  - 4.3.2 AJS3 linkage function 54
  - 4.3.3 System configuration 54
  - 4.3.4 Executing jobs 55
  - 4.3.5 Procedure to use the function 56
- 4.4 Job operation with the execution user specified 60
  - 4.4.1 Execution user setting function 60
- 4.5 Migrating definitions 62
  - 4.5.1 Saving and recovering definition files 62
  - 4.5.2 Notes on saving and recovering definitions 63
- 4.6 Migrating the CPA system 64
  - 4.6.1 Data to be saved 64
  - 4.6.2 Saving data 64
  - 4.6.3 Recovering data 65
  - 4.6.4 Notes on saving and recovering data 65

## Part 4: System Design

### 5 Business Operations Planning with CPA 66

- 5.1 Building an item creation environment 67
- 5.2 Common settings for items 68
- 5.3 Creating action items 69
  - 5.3.1 Basic settings for action items 69
  - 5.3.2 Option setting for an action item 71
  - 5.3.3 Setting an input item in an action item 71
- 5.4 Creating an action flow item 72
  - 5.4.1 Action flow execution function 72
  - 5.4.2 Item execution condition 77
  - 5.4.3 Examples of how to create an action flow 78
- 5.5 Event monitor functions 80
  - 5.5.1 List of event monitor functions 80
  - 5.5.2 Inherited information 81
- 5.6 Creating schedule items 83
  - 5.6.1 Schedule functions 83
- 5.7 Creating an input item 87
  - 5.7.1 Text box input item 87
  - 5.7.2 List box input item 88
  - 5.7.3 Check box input item 89
  - 5.7.4 Enabling or disabling an input item 91
  - 5.7.5 Inserting an option letter before the value 93
  - 5.7.6 Making an input item a mandatory field 94

## Part 5: Reference

### 6 Job Design View 96

- 6.1 List of windows 97
  - 6.1.1 Home window 98
- 6.2 Job operation windows 99
  - 6.2.1 Job exec. on time dialog box 100
  - 6.2.2 Job exec. on event dialog box 102
  - 6.2.3 Checking the status dialog box 103
- 6.3 Window used to define items 106
  - 6.3.1 Action flow dialog box 107
  - 6.3.2 Calendar dialog box 108

### 7 Item Design View 111

- 7.1 List of windows 112
  - 7.1.1 Home window 112

- 7.2 Item definition windows 114
- 7.2.1 Action dialog box 115
- 7.2.2 Action flow dialog box 126
- 7.2.3 Calendar dialog box 129
- 7.2.4 Exec. date dialog box 132
- 7.2.5 Exec. time dialog box 141

## **8 Standard items 143**

- 8.1 List of standard CPA items 144
- 8.2 Action items 146
  - 8.2.1 Command execution 146
  - 8.2.2 Command execution by list 147
  - 8.2.3 Command execution with recovery option 149
  - 8.2.4 Command execution in window 151
  - 8.2.5 Send mail 153
  - 8.2.6 Windows log on 155
- 8.3 Event monitor items 159
  - 8.3.1 Monitor file (event monitor item) 159
  - 8.3.2 Monitor any event (event monitor item) 161
  - 8.3.3 Monitoring mail reception (event monitor item) 164
  - 8.3.4 Monitor Windows event (event monitor item) 166
- 8.4 Execution condition items 169
  - 8.4.1 Monitor file event (execution condition item) 169
  - 8.4.2 Monitor any event (execution condition item) 170
  - 8.4.3 Monitor the linkage from JP1AJS3 (execution condition item) 174
  - 8.4.4 Event monitoring for receiving mail (execution condition item) 174
  - 8.4.5 Monitor Windows event (execution condition item) 176

## **Part 6: Reference**

### **9 Commands 178**

- Command notation conventions 179
- List of commands 180
  - cpajobdesign 181
  - cpajobentry 182
  - cpajobdelete 184
  - cpajobstop 186
  - cpajobrun 188
  - cpareport 190
  - cpajobprint 197
  - cpasswd 200
  - cpasetprof 202
  - cpasetfirewall 204

- cpaalsetup 205
- cpasetumap 207
- cpausrenvsetup 210
- cpaitemsetup 212
- cpaecho 214
- cpasleep 215

## Appendixes 216

- A Troubleshooting 217
  - A.1 Steps of troubleshooting 217
  - A.2 Collecting data 217
  - A.3 Different behaviors in different command execution environments 218
- B Tips to Operate Jobs 222
  - B.1 Using inherited information and profile items 222
  - B.2 How to use environment variables when an item is executed 222
  - B.3 Expanding the number of custom jobs for CPA linkage that are executed simultaneously on AJS3 225
- C Usage Estimation 226
  - C.1 Limit values 226
  - C.2 Conversion rate of characters into bytes 227
  - C.3 Memory requirements 227
  - C.4 Disk space requirements 227
- D Files and Folders 228
  - D.1 Files and folders in CPA 228
  - D.2 Files and folders in CPA Option for AJS3 229
- E Definition files 231
  - E.1 System settings definition file 231
  - E.2 Message communication service definition file 232
- F Processes 233
- G Version Changes 234
  - G.1 Changes in version 12-50 234
  - G.2 Changes in version 12-10 235
  - G.3 Changes in version 12-01 237
- H Reference Material for This Manual 239
  - H.1 Related publications 239
  - H.2 Conventions: Abbreviations for product names 239
  - H.3 Conventions: Acronyms 239
  - H.4 Conventions: KB, MB, GB, and TB 240

## Index 241



# 1

## Overview of CPA

This chapter describes the overview of CPA.

# 1.1 Features of CPA

Among the many business operations carried out by computers, some are performed regularly such as daily backups or month-end closing, while some are performed irregularly by events such as receiving orders.

Moreover, computer systems can be even more complicated with cloud services or data centers, where services and data are stored in virtual locations, as well as when different computers relate to one another to perform a set of operations. This, as a result, can impose a burden on services and system users who access the data.

CPA helps you to automate such recurring processing and routine processing. It also helps you to automate and reduce the operations that users perform on systems by making such processes interact with each other.

## 1.1.1 Understanding jobs

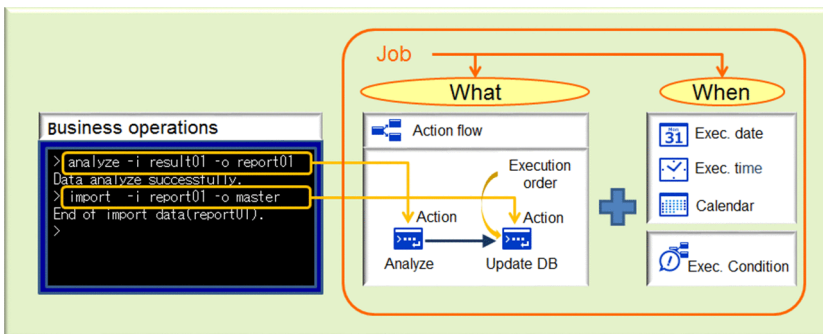
To automate business operations using CPA, you need to create jobs.

A job refers to processing to be executed on computers with one operation. A job includes multiple commands (What) and triggers for execution (When).

A job is an execution unit on computers. You can start or stop executing each job.

To execute processing on computers various information is required, such as the names of the commands to be executed, parameters specified for the commands, the order in which commands are executed, and the time when commands are executed. We call each job configuration component of such information an *item*.

Figure 1–1: Understanding jobs



For example, the figure above shows a job to execute two commands sequentially. In CPA, you use an action item to define a command to execute and parameters to specify for the command. You also use an action flow item to define the order in which commands will be executed. Additionally, you can add the execution date and execution time items to define when to execute the job.

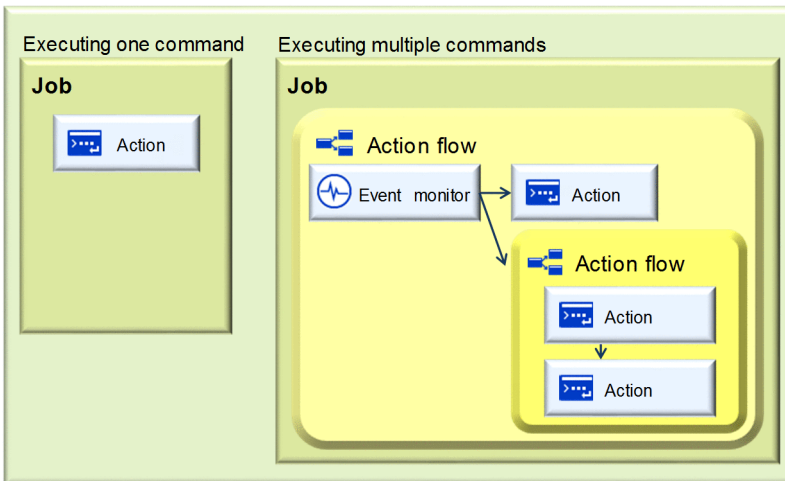
An item can be used by multiple jobs. For example, if you create an item that copies files, you can use it for another business operation where copying files is required. This makes it easier to create jobs. In addition, you can put your operation know-how to work.

## 1.1.2 Actions, event monitors, and action flows

Items responsible for what the job performs include actions, event monitors, and action flows. When you execute a single command, you call an action from a job. In contrast, when you execute multiple commands in combination with

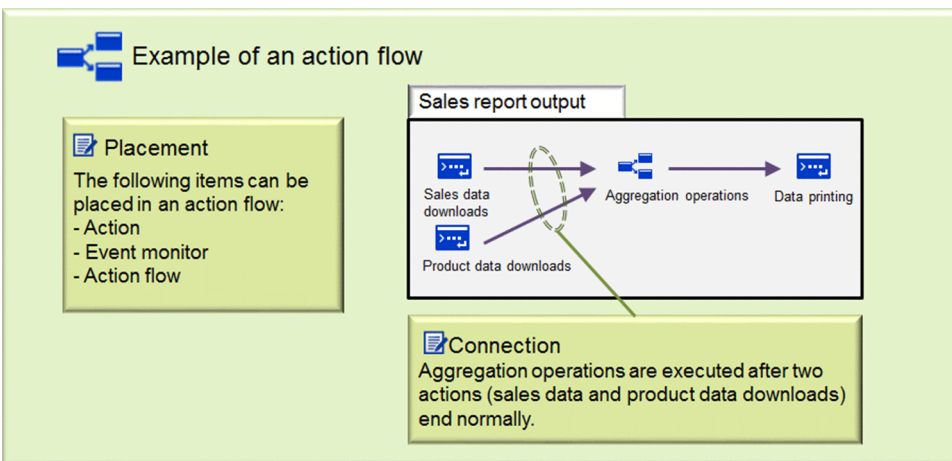
event monitoring, you call an action flow from a job. To monitor events, you specify an event monitor in an action flow. An action flow combines routine operations into one item. It can call actions, event monitors, and action flows.

Figure 1–2: Items executed in jobs



You can specify the order in which items placed are executed by connecting them with each other using relation lines (arrows). An item can be set to run after multiple items finish, while multiple items can be set to run after one item finishes.

Figure 1–3: Example of an action flow

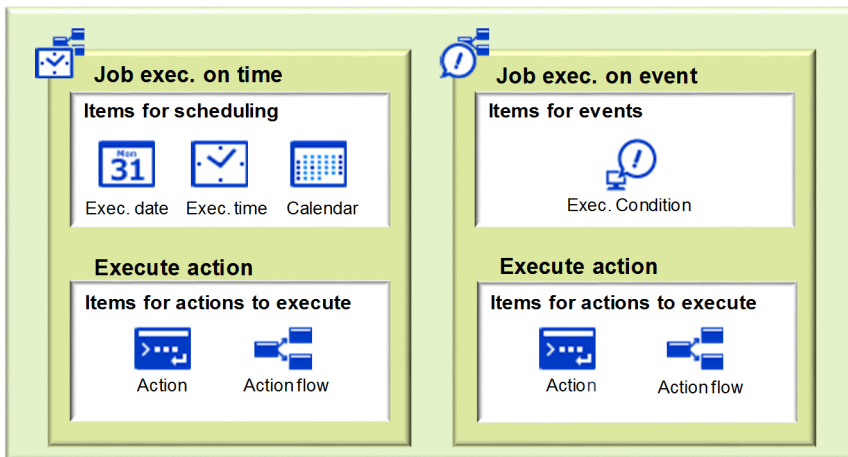


### 1.1.3 Execution date/time and execution conditions

There are two ways to execute jobs. First, you can specify the date and time (for example, at 9:00 every day) to execute jobs. Second, you can specify triggers of events (for example, after a file is created) by which jobs are executed. Depending on this difference in when executions take place, jobs are classified into the following two types. Each of them has a different combination of items.

- *Job executed on time:* A job runs on a scheduled date and time.
- *Job executed on event:* Job execution is triggered by an event.

Figure 1–4: Job types

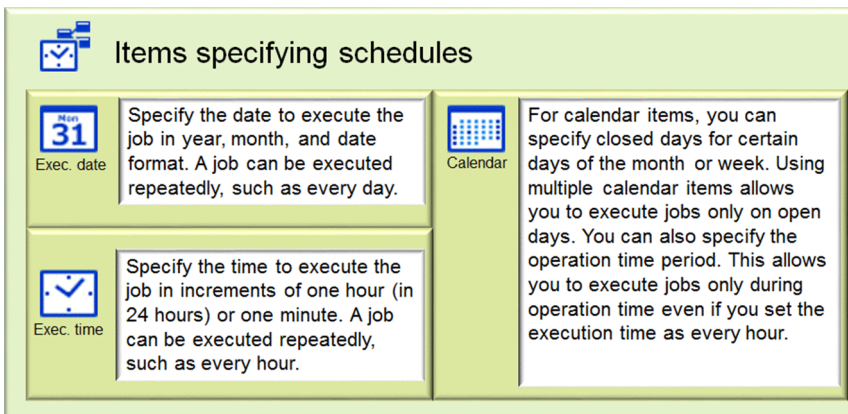


## (1) Job executed on time

The jobs of this type are set to run on a given schedule.

You can set a job execution schedule with a combination of the execution date, execution time, and calendar items. For example, you can use items such as the *execute everyday* item or the *execute in end of every month* item. Once you register the job with such items, the job is repeatedly executed on automatically calculated dates and times. You can create multiple calendar items depending on your business operations.

Figure 1–5: Items specifying schedules



## (2) Job executed on event

For this type of job, you specify events to trigger job execution.

You can specify events to trigger execution as execution condition items.

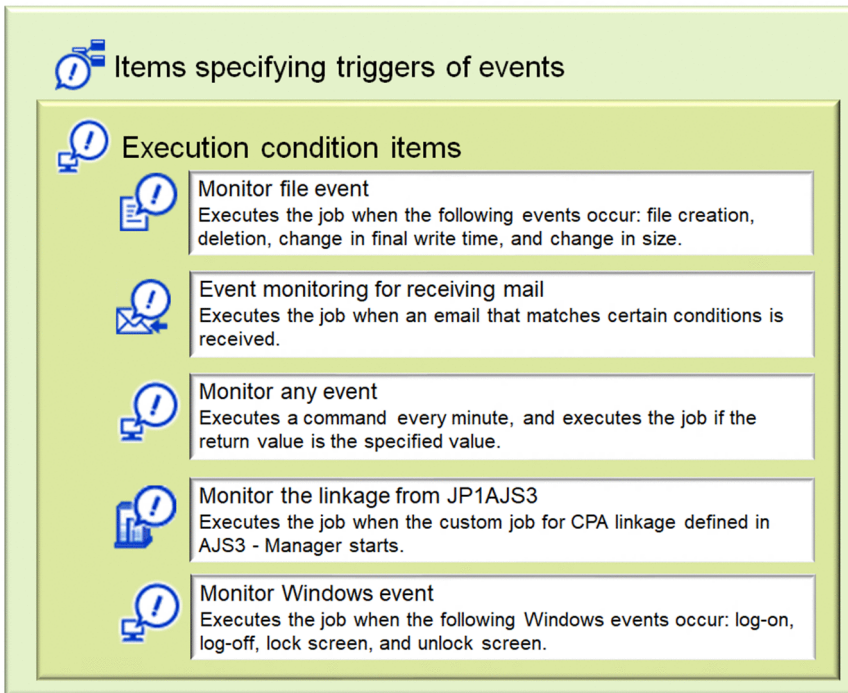
The following five execution condition items are available:

- Monitor file event
- Monitor any event
- Monitor the linkage from JP1AJS3<sup>#</sup>
- Event monitoring for receiving mail
- Monitor Windows event

#

Settings are required to use the item. For details on how to set up, see [2.3.3 Configurations for the AJS3 linkage function](#), [3.3.1 Configurations of the message communication service](#), and [3.3.2 Setting up components for the AJS3 linkage function](#).

Figure 1–6: Items specifying triggers of events



## 1.1.4 State management for jobs and actions

The following table describes states of jobs and actions.

Table 1–1: List of states

No.	Type	State	Description	Note
1	Job	Wait for start time (WAIT)	Waiting to start, such as waiting for the execution timing (the scheduled execution time) or waiting until the service starts	Applicable only for the job executed on time
2		Monitoring for event (MONITORING)	Waiting for event occurrence (execution condition)	Applicable only for the job executed on event
3		Overflowing events (MONITORING)	A status where no event can be accepted after the event queue overflows (Specifically, an event is detected when there are 16 jobs with the status of Wait for start.)	Applicable only for the job executed on event The resumption of monitoring cannot be initiated until all the events in the queue that are either waiting to start or currently running end.
4		Wait for start (QUEUEING)	Waiting to start	Applicable only for the job executed on event
5		Executing (RUN)	Started after all the start conditions are met, and not ended yet	--

No.	Type	State	Description	Note
6		Ended normally (ENDN)	All of action flows, actions, and event monitors normally ended	--
7		Ended with warning (ENDW)	One of the actions ended with a warning, and all the other action flows, actions, and event monitors ended.	--
8		Terminating event monitoring (ENDE)	Event monitoring is stopped by stopping the service or executing <code>cpajobstop -e</code> during a state of <i>monitoring for event</i> or <i>events in overflow</i> .	Applicable only for the job executed on event
9		Ended abnormally (ENDE)	One of action flows, actions, or event monitors ended abnormally, which caused all the other action flows, actions, and event monitors to end.	--
10		Ended abnormally (Cancel) (ENDE)	The job was ended (suspended) after <code>cpajobstop -c</code> was executed to terminate all the actions that were running.	--
11		Ended abnormally (Cancel) (ENDK)	The job was killed. The job is killed if it is forced to terminate by a command or the service stops during execution.	--
12	Action, event monitor, or action flow	Wait for start (WAIT)	Waiting to start, such as waiting for an execution timing (scheduled execution time or event occurrence), and waiting until the service starts or starting conditions (preceding conditions) are met.	--
13		Executing (RUN)	Not ended yet after all starting conditions (preceding conditions) are met.	--
14		Ended normally (ENDN)	An action, event monitor, or action flow ended normally. For actions, the return value of the execution command is determined; a return value of 0 means a normal end.	--
15		Ended normally (unexec) (ENDNS)	The conditions for executing an item, defined in the action flow, were not met and the item ended unexecuted.	--
16		Ended with warning (ENDW)	An action or action flow ended with a warning. This is not normal, but the action or action flow can be continued. When <b>Warning end if the return code is other than 0</b> is specified for <b>Operations when terminates</b> of a custom action item, and a value other than 0 is returned by the command, the action in question ends with a warning.	--
17		Ended abnormally (ENDE)	An action, event monitor, or action flow ended abnormally. In the case of an action, this state is determined based on the value returned by the execution command. Specifically, when <b>Abnormal end if the return code is other than 0</b> is specified for <b>Operations when terminates</b> of either a standard item or a custom item, and a value other than 0 is returned by the command, the action in question ends abnormally.	--
18		Ended abnormally (Cancel) (ENDK)	The job was killed. The job is killed if it is forced to terminate by a command or the service stops during execution.	--

No.	Type	State	Description	Note
19		Condition not satisfied (UNMATCHED)	An action flow defines branching with ending with a warning or ending normally, and one unit was not executed due to the branching.	--

Legend:

-- : Not applicable

## 1.1.5 Custom items

In addition to the standard items provided by CPA, you can create components that are necessary for the execution of business operations.

Such components are referred to as *custom items*. The created custom items can be combined with CPA action flows or job definitions to execute business operations.

Through componentization, which makes individual processes comprising a certain business operation available to other business operations as well and allows for the use of variable parameters as the essential parts of each process, you can operate your system in a highly efficient and maintainable manner. You can also operate the processes by flexibly defining action flows and schedules to suit your business operation. For example, different subsequent actions can be taken depending on the results of actions, or jobs can be executed at any intervals or on particular days of the week or dates.

To set up an environment for creating custom items, see [2.3.6 Creating a custom item](#). To design custom items, see [5. Business Operations Planning with CPA](#).

## 1.2 Job components

A job represents a unit where a set of actions and action flow are executed depending on schedules or execution conditions. You can start operations by registering jobs. The following table describes job components.

Table 1–2: Types of job components

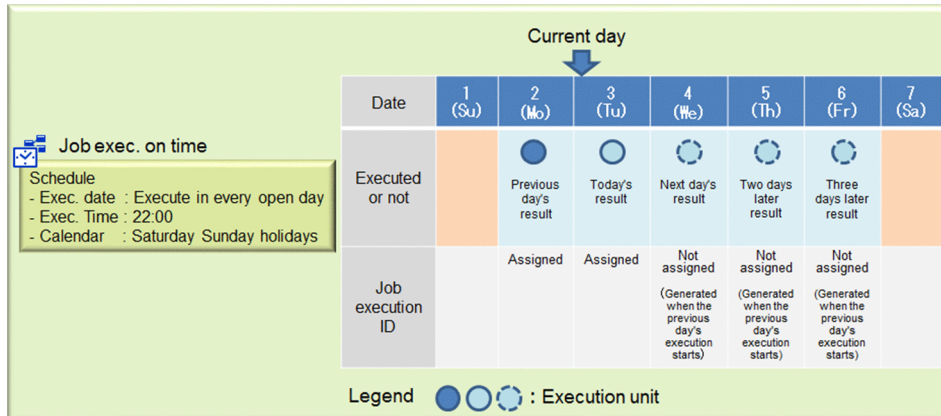
No.	Type	Description
1	Job	<p>A job consists of a group of configuration components which link business operations that you want to perform (What) to triggers by which such operations are performed (When).</p> <p>Depending on when executions take place, jobs are classified into the following two types: jobs executed on time, and jobs executed on event.</p> <p>You can schedule job executions by registering jobs each with a unique name (called a <i>job name</i>) in the job execution service.</p> <p>Specify the job name within 45 bytes. However, if multibyte characters are used, specify it within 15 characters.</p>
2	Item	<p>Job components are collectively referred to as items. There are seven types of items shown as Nos. 3 to 9 in this table.</p> <p>An item can be used by multiple jobs. Editing an item changes the execution settings of all the jobs executed on time that are registered and use the item. For a job executed on event, you need to register it again. Furthermore, users who design business operations (designers), such as individuals from the information systems division, can create custom items, which allow them to design highly sophisticated business operations. Users executing business operations (operators) can use the created custom items.</p>
3	Action	<p>An action defines a command executed in a job or a minimum execution unit on CPA, such as send mail. You can specify the file name of the command to be executed, a parameter to be specified for the command, or other settings. Different types of items are available depending on how actions are executed.</p>
4	Event monitor	<p>An event monitor defines the conditions for detecting event occurrence. Different types of items are available depending on what is being monitored (file monitor, mail reception, and so on).</p>
5	Action flow	<p>An action flow consolidates actions, event monitors, and action flows, and specifies the order in which items are executed.</p> <p>An action flow can be created and edited as an item, and it can be called from other action flows.</p>
6	Execution date	<p>An execution date determines the date when the job executed on time runs. You can specify a repeat pattern such as every day or every Monday.</p>
7	Execution time	<p>An execution time determines the time when the job executed on time runs. You can specify a repeat pattern such as every hour.</p>
8	Calendar	<p>A calendar refers to a date type used to determine job execution dates. There are two types of calendars: open day and closed day. You can define schedules with which jobs are executed only on weekdays by setting weekends and national holidays as closed days.</p> <p>You can also define operation time periods for calendars. No schedule is calculated for hours outside the operation time period.</p> <p>A calendar can be created and edited as an item. Different calendars can be registered for each job.</p>
9	Execution condition	<p>An execution condition defines conditions with which job execution is triggered by events. Execution conditions are used with event monitors. Once you register a job, CPA keeps monitoring event occurrence for the job.</p>
10	Unit	<p>A unit is a general term used to describe actions, event monitors, and action flows that are defined in an action flow. Ensure that every unit has a unit name that is unique in an action flow to enable the unit to be identified uniquely.</p>



## 1.3 Job execution unit and job execution ID

For each job, an ID (called a *job execution ID*) is assigned depending on its schedule and the results. A job execution ID is assigned to each execution unit, such as today's execution unit or the next day's execution unit. Each unit is determined by schedules or results of execution conditions.

Figure 1–7: Job execution unit and job execution ID



A job execution ID is assigned when an execution schedule is generated. For a job executed on time, a job execution ID is assigned when the job is registered and when the next execution schedule is calculated after the job is started. For a job executed on event, a job execution ID is assigned when a job is registered, or when a schedule to wait until the next execution condition is satisfied after an execution condition is met, is generated.

The format of a job execution ID is composed of 36 alphanumeric characters (in hexadecimal) and hyphens (-) as shown below.

36 characters

Example of a job execution ID : e66bf2f5-b920-4faf-b1f0-a56418af4682

Once a job execution ID is assigned, it does not change unless you re-register the job or change the schedule. Each job execution ID is unique. There are no duplicates of job execution IDs of other jobs or those of executions in the past.

You can use the `cpareport` command to see job execution IDs. For details, see *cpareport* in 9. *Commands*.

# 2

## Installation and Configuration of CPA

This chapter describes how to install CPA and set up configurations for it.

## 2.1 Requirements

---

- CPA must be installed on an NTFS file system.
- A host on which you want to install CPA must have a host name with a length of 128 bytes or less.

## 2.2 Installation

---

This section describes how to install CPA.

### 2.2.1 Installing a new instance

To install a new instance of CPA:

1. As a member of the Administrators group, log in to the host on which you want to install CPA.
2. Insert the CPA media.  
The Hitachi integrated installer window appears.
3. Follow the instructions from the Hitachi integrated installer and specify information as necessary to install the software.

Specify the following information during installation:

- User name<sup>#1</sup>  
Enter the name of the CPA user. The length must not exceed 50 bytes.
- Organization<sup>#1</sup>  
Enter the name of your company or department. The length must not exceed 80 bytes.
- Installation folder<sup>#2</sup>  
Specify the absolute path to the folder in which you want to install CPA. The length, including a backslash (\) that is automatically added to the end of the path, must not exceed 80 bytes. The absolute path must point to a local directory.  
If you do not specify any value, the following value is assumed:  
`system-drive\Program Files (x86)\HITACHI\JP1CPA#3`
- Data Folder<sup>#2</sup>  
Specify the absolute path to the folder in which you want to store CPA data. The length, including a backslash (\) that is automatically added to the end of the path, must not exceed 80 bytes. The absolute path must point to a local directory.  
If you do not specify any value, the following value is assumed:  
`system-drive\ProgramData\HITACHI\jpl\jpl_default\JP1CPA`  
#1: Do not include the following characters in the user name or organization name:
  - Symbols except for the following:  
Space, left parenthesis ( ( ), right parenthesis ( ) ), plus sign (+), hyphen (-), and underscore ( \_ )#2: Ensure that the installation folder and data folder do not include the following characters or paths:
  - Symbols except for the following:  
Space, left parenthesis ( ( ), right parenthesis ( ) ), plus sign (+), hyphen (-), and underscore ( \_ )Note that you cannot include a space at the beginning or end of the path, before or after a backslash (\), or in succession with another space.
  - Multibyte characters
  - Paths to non-fixed drives or UNC paths#3: If you are using a 32-bit operating system, replace `Program Files (x86)` with `Program Files`.

4. When the installation is completed, click the **Finish** button.

The following table lists default values for installation settings.

Table 2–1: Default values for installation settings

No.	Setting	Default value
1	Installation folder	<ul style="list-style-type: none"> <li>In a 32-bit operating system: %ProgramFiles%\Hitachi\JP1CPA</li> <li>In a 64-bit operating system: %ProgramFiles(x86)\Hitachi\JP1CPA</li> </ul>
2	Data folder	%ProgramData%\Hitachi\jpl\jpl_default\JP1CPA
3	Service name of the job execution service	JP1_CPA
4	Service display name of the job execution service	JP1/CPA
5	Startup type of the job execution service	Automatic (AUTO)
6	Log-on account for the job execution service	Local system account ( <b>Allow service to interact with desktop</b> is cleared.) Although changing the account allows you to work around the limitations imposed on the local system account during the execution of jobs, doing so prevents you from executing the <b>Command execution in window</b> and <b>Windows log on</b> items.
7	Service name of the message communication service	JP1_CPA_Message
8	Service display name of the message communication service	JP1/CPA Message
9	Startup type of the message communication service	Automatic (AUTO)
10	Log-on account for the message communication service	Local system account ( <b>Allow service to interact with desktop</b> is cleared.)
11	Name of the program folder in the <b>Start</b> menu	JP1_Client Process Automation
12	Name of a shortcut in the <b>Start</b> menu to open the Job Design View	Register job
13	Name of a shortcut in the <b>Start</b> menu to configure the firewall settings	Enable firewall settings
14	Name of a shortcut in the <b>Start</b> menu to revert the firewall settings	Disable firewall settings

## 2.2.2 Overwriting or upgrading an existing instance

To upgrade an existing instance of CPA:

1. As a member of the Administrators group, log in to the host on which you want to install CPA.
2. Exit all programs.  
Stop the running services and close the Job Design View, if any.
3. Insert the CPA media and start the installation.  
Follow the instructions from the Hitachi integrated installer to upgrade or overwrite the existing instance.

### Important

- If an unexpected error occurs while you overwrite or upgrade an existing instance, the data of the instance might be lost and no longer be restored. Make a backup copy of the data folder before you can start overwriting or upgrading.
- If you overwrite or upgrade the existing instance, the properties for the job execution service and message communication service are reset to the default values. If you have customized the properties for these services, customize them again.

For details on the service properties, see [Table 2-1 Default values for installation settings](#).

## 2.2.3 Performing silent installation

You can perform the silent installation of CPA. To perform the silent installation:

1. As a member of the Administrators group, log in to the host on which you want to install CPA.
2. Open the Command Prompt windows as an administrator.
3. Execute the following command:

```
CPA-media-drive\_OWNEXE\HPPSINST.BAT /CPA-media-drive\_PPDIR\product-model  
-name\DISK1 /"install-path"" [/A ["DATADIR=\"data-path\""] [USERNAME=\"user  
-name\"] [COMPANYNAME=\"organization\"]]
```

Specify the following for the product model name.

- For CPA: PLA123BC
- For CPA Option for AJS3: PLA123CC

### Important

- If you execute installation after copying the DVD-R Media to HDD, copy all the contents of the media onto a directory path that does not contain spaces. Also, make sure that the copied files are identical with the original ones at binary level.
- To confirm whether the installation completed successfully or not, check the return value of the execution result (0 means success).

## 2.2.4 Notes on installation

- Do not specify the location directly under a drive (ex. "C:\") as the installation destination of this product.
- For the installation destination folder and the data folder, do not specify a space at the beginning or end of folder names of the folders at the middle of the path.
- If the installation destination folder or the data folder contains spaces, installation and the installed product will not work properly if a file or folder which has the same path from the beginning to the spaces exists.

In this case, change the installation destination or the name of the file or the folder of the existing path.

(Example) When you install CPA to "C:\Program Files (x86)\HITACHI\JP1CPA", a file or a folder "C:\Program" exists.

- Before installing or uninstalling, stop other running applications, resident programs, and services as much as possible.
- Administrator authorities are required to execute installation, setup, and uninstallation operations.
- If a user other than Administrator executes these operations in an environment where UAC (User Account Control) is effective, promote to Administrator.  
In an environment where UAC is ineffective, execute these operations as a user who belongs to the Administrators group.
- While installing, uninstalling, or applying patches, do not set the startup type of the following services "invalid". If you set them "invalid", installation, uninstallation, or bug-fixing patch application may fail.
  - "Application Experience" service
  - "Windows Modules Installer" service
- This product sets the following path in the PATH environment variable at the time of installation.
  - Install path\bin  
(Example: C:\Program Files (x86)\Hitachi\JP1CPA\bin)

If the length of the PATH environment variable (including the path being set during installation) exceeds the limit value of Windows, the value set in the PATH environment variable does not become effective. Delete unnecessary PATH environment variables to keep the length of the setting value within the limit.  
For details on how to the Windows limit length, see the OS manual.
- When you want to keep the setting files and logs directly under the installation destination folder or the data folder, please backup the necessary files in advance because these are deleted by uninstallation.
- For Notes on WOW 64. The default installation destination is "System Drive\Program Files (x86)\Hitachi\JP1 CPA".  
If you change the destination to a path under "System Drive\Program Files\", (for example, to "System Drive\Program Files\Hitachi\JP1CPA"), this path will be redirected to a path that is not intended by the product or customer, due to the effect of WOW64 redirection.
- Installation on a network drive. Do not install this product while placing it on a disk which is mounted via a network file system such as NFS.

For other notes on installation, see the *Release Notes*.

## 2.3 Configurations

---

This section describes how to set up configurations for CPA.

### 2.3.1 Configurations for the Send mail item

When the Send mail item is available, a mail profile is automatically created upon installation.

However, you have to edit the mail profile settings manually.

#### File location

- `data-path\conf\mailprof.ini.model` (model file)
- `data-path\conf\mailprof.ini` (mail profile)

Note that the files are stored in the following locations by default:

- `%ProgramData%\HITACHI\jpl\jpl_default\JP1CPA\conf\mailprof.ini.model` (model file)
- `%ProgramData%\HITACHI\jpl\jpl_default\JP1CPA\conf\mailprof.ini` (mail profile)

#### Setting items in the mail profile

Include the following setting items in the mail profile:

- `DefaultFrom=sender-email-address`  
Specify only one email address that will be used as the sender of outgoing emails.  
When an email is sent, the `From` header of the email is set to this value.  
This setting item is required.
- `DefaultReplyTo=reply-to-email-address`  
Specify email addresses that will be used as the reply-to address of outgoing emails.  
When an email is sent, the `Reply-To` header of the email is set to this value.  
To specify multiple email addresses, list the addresses by separating each address with a comma (,). The specified email address must not exceed 256 bytes. When multiple email addresses are specified with commas (,) separating them, the specified addresses including the comma separators must not exceed 256 bytes. This setting item is required.
- `SmtPServer=SMTP-server-name`  
Specify the host name or IP address of the SMTP server to send emails.  
For the IP address, it must be specified in IPv4 or IPv6 format. For the SMTP server name, the length must not exceed 256 bytes. This setting item is required.
- `SmtPPort=SMTP-port-number`  
Specify the port number of the communication port on the SMTP server.  
If you omit this setting item, TCP 25 is used. You can use a number from 1 to 65535 for the port number.
- `SmtPAuthType={ 0 | 1 | 2 }`  
Specify the authentication method for outgoing emails.  
Specify 1 for the POP before SMTP authentication, or 2 for SMTP-AUTH. Omit this setting item or specify 0 when no authentication is required.
- `Pop3Server=POP3-server-name`



Specify the host name or IP address of the POP3 server for the POP before SMTP authentication.

For the IP address, it must be specified in IPv4 or IPv6 format. You can omit this setting item if you do not plan to use the POP before SMTP authentication. For the POP3 server name, the length must not exceed 256 bytes.

- `Pop3Port=POP3-port-number`

Specify the port number of the communication port on the POP3 server for the POP before SMTP authentication. If you omit this setting item, TCP 110 is used. You can use a number from 1 to 65535 for the port number.

- `Pop3User=mail-account`

Specify the authentication account name for the POP before SMTP authentication.

You can omit this setting item if you do not plan to use the POP before SMTP authentication. The mail account name must not exceed 255 characters in length.

- `Pop3Password=POP3-password`

Use the `cpasswd` command to specify the value of this setting item.

For details on the `cpasswd` command, see [cpasswd](#) in [9. Commands](#). You can omit this setting item if you do not plan to use the POP before SMTP authentication.

- `SmtAuthUser=mail-account`

Specify the authentication account name for SMTP-AUTH.

You can omit this setting item if you do not plan to use SMTP-AUTH. The mail account name must not exceed 255 characters in length.

- `SmtAuthPassword=SMTP-password`

Use the `cpasswd` command to specify the value of this setting item.

For details on the `cpasswd` command, see [cpasswd](#) in [9. Commands](#). You can omit this setting item if you do not plan to use SMTP-AUTH.

- `SmtAuthUseLOGIN={ 0 | 1 }`

Specify 1 to use LOGIN for SMTP-AUTH. Omit this setting item or specify 0 when you do not plan to use SMTP-AUTH.

- `SmtAuthUsePLAIN={ 0 | 1 }`

Specify 1 to use PLAIN for SMTP-AUTH. Omit this setting item or specify 0 when you do not plan to use SMTP-AUTH.

## 2.3.2 Configurations for the Monitoring mail reception item

If you want to use the Monitoring mail reception item, you need to configure the host running CPA to have a profile of mail reception for the Monitoring mail reception item to receive emails.

Create a new profile of mail reception by manually copying and editing the model file.

### File location

- ```
- data-path\conf\recvmailprof.ini.model (model file)
- data-path\conf\recvmailprof.ini (profile of mail reception)
```

Note that the files are stored in the following locations by default:

- ```
- %ProgramData%\HITACHI\jp1\jp1_default\JP1CPA\conf\recvmailprof.ini.model
(model file)
```

```
- %ProgramData%\HITACHI\jpl\jpl_default\JP1CPA\conf\rcvmailprof.ini (profile of mail reception)
```

## Setting items in the profile of mail reception

Include the following setting items in the profile of mail reception:

- `Protocol={ POP3 | IMAP4 }`  
Specify the protocol for the incoming mail server.  
If you omit this setting item, the value POP3 is assumed.
- `ServerName=incoming-mail-server-name-or-IP-address`  
Specify the name or IP address of the incoming mail server.  
For the IP address, it must be specified in IPv4 or IPv6 format. Note that the name or IP address of the incoming mail server must not exceed 256 bytes in length.  
This setting item is required.
- `Port=incoming-mail-server-port-number`  
Specify the port number of the communication port on the incoming mail server.  
If you omit this setting item, TCP 110 is used for the protocol POP3 or TCP 143 for IMAP4. You can use a number from 1 to 65535 for the port number.
- `UserName=user-name-of-OS`  
Specify the user name of OS that will be used to log in to the incoming mail server.  
The user name of OS must not exceed 255 alphanumeric characters in length. This setting item is required.
- `Password=password`  
Specify the password of the user of OS that will be used to log in to the incoming mail server.  
Use the `cpapasswd` command to specify the value of this setting item. For details on the `cpapasswd` command, see [cpapasswd](#) in [9. Commands](#).

## 2.3.3 Configurations for the AJS3 linkage function

The message communication service is required for using the AJS3 linkage function.

To use the AJS3 linkage function, you need to configure and start the message communication service on both the host running CPA and the host running CPA Option for AJS3.

For details on how to configure the message communication service on CPA Option for AJS3, see [3.3.1 Configurations of the message communication service](#).

### (1) Changing the port number of the message communication service

Installing CPA causes the `services` file to have a definition on the port number to be used by the message communication service. The port number must correspond with the port number that is defined for `cpamsgsrv` on the connected CPA Option for AJS3 host. For details on the port number, see [2.5 Port number](#).

To change the default port number, edit the `services` file.

To change the default service name for the port number, edit the message communication service definition file. For details on the message communication service definition file (`iesconf.xml`), see [E.2 Message communication service definition file](#).

Definition to be included in the `services` file

```
cpamsgsrv 22260/tcp # JP1 Client Process Automation
```

### Important

- The message communication service fails to start with the error `KNAO0950-E` when the port number is not defined in the `services` file.
- Ensure that the CPA Option for AJS3 host is also configured to have the `services` file defining the same port number.

## (2) Starting the service

You do not need to start the service manually because the installer ensures that the service starts automatically. However, you need to start the service manually if you change the startup type from `Automatic` to a different one after the installation process. To start the service:

1. Open **Control Panel** and select **Services**, or select **Administrative Tools** and then **Services**.
2. Select the JP1/CPA Message service.
3. Click the **Start** button.

## (3) Configuring communication with CPA Option for AJS3

### (a) Host name resolution

To communicate with CPA Option for AJS3, ensure that CPA hosts can resolve the name of the host running CPA Option for AJS3. A way to do this is to use the `hosts` file to resolve the host name to the IP address.

### (b) Configuring the firewall and other protections to allow traffic

To communicate with CPA Option for AJS3, ensure that the CPA hosts and the devices on network routes to CPA Option for AJS3 are configured with security protections to allow inbound and outbound traffic on the specified port number. The port number is defined as described in [2.3.3 \(1\) Changing the port number of the message communication service](#).

For Windows Firewall, you can use the **Enable firewall settings** command provided in the **Start** menu or the `cpasetfirewall` command to configure the settings.

Note that the CPA uninstaller does not revert the settings. If you no longer use CPA, you need to revert the settings by using the **Disable firewall settings** command provided in the **Start** menu or the `cpasetfirewall` command before uninstalling CPA.

Remember that the **Enable firewall settings** and **Disable firewall settings** commands require administrator privileges. You must run the commands as an administrator.

### Important

Ensure that the time zone is synchronized between the host on which CPA is installed and the one on which CPA Option for AJS3 is installed.

## 2.3.4 Using the Windows log-on function

To use the Windows log-on function, you have to set it up.

The Windows log-on function is available only with Windows 10. Do not set up this function with an OS other than Windows 10. If you set up this function with an OS other than Windows 10, unset up the function.

For details on the Windows log-on function, see [4.2.1 Windows log-on function](#).

CPA virtual users to be linked with information regarding the users logging on to Windows (Windows user names and passwords) must be registered in the user mapping file (`usrprof.xml`). You must therefore register virtual users when setting up the Windows log-on function.

### (1) Setting up the Windows log-on function

The procedure for setting up the Windows log-on function is provided below.

After setting up the function, restart the machine.

1. As a member of the Administrators group, log in to the host on which to set up the Windows log-on function.
2. Start the command prompt by selecting **Run as administrator**.
3. Execute the following command:

```
cpaalsetup
```

For details on the setup command (`cpaalsetup`), see [cpaalsetup](#) in [9. Commands](#).

4. Restart the machine.

#### Important

- If, in addition to CPA, an application in which the customized Windows log-on function is implemented runs in your environment, confirm that the application works fine even when the Windows log-on function is used. Even when the log-on action of the Windows log-on function implemented on CPA ends successfully, logging on to Windows while not logging on to the other application running in your environment can cause a problem.

Examples of applications that customize the Windows logon function

- JP1/HIBUN Data Encryption
  - HIBUN login window
  - Extended authentication
- JP1/HIBUN Device Control
  - Screen Lock function by area search
- When you no longer need the Windows log-on function, (when you set up the function with an OS other than Windows 10), or when a problem is discovered during the verification of Windows log-on behavior, you must use the `cpaalsetup` command to unset up the Windows log-on function. For details on the `cpaalsetup` command, see [cpaalsetup](#) in [9. Commands](#).
- If the `cpaalsetup` command is executed by a user who does not have administrator privileges, the command ends abnormally with a return value of 52. Always execute this command with administrator privileges.

- If the `cpaalsetup` command is executed while a file managed by CPA is opened in either a security software program or a text editor or when a log-on window is open, the command ends abnormally with a return value of either 45 or 111.

When you execute this command, ensure that no file managed by CPA is open and that the log-on window is not displayed.

- If you execute the `cpaalsetup` command when the setting specified for the `PATH` environment variable exceeds 1,958 bytes in length, the command might end abnormally with a return value of 119. When you execute this command, ensure that there is sufficient empty space in the length reserved for the `PATH` environment variable setting.
- Do not execute the `cpaalsetup` command while a Remote Desktop connection is established.

## (2) Setting virtual users

To register both virtual users and the user information to be linked to the virtual users in the user mapping file (`usrprof.xml`), you have to execute the `cpasetumap` command. For details on the `cpasetumap` command, see *cpasetumap* in [9. Commands](#).

### 2.3.5 Using the Windows event monitor function

The Windows event monitor function uses a support system event (System Event Notification Service)<sup>#</sup> as a means to detect Window events. For this reason, the COM+ Event System service, the COM+ System Application service, and System Event Notification Service service must be up and running when you use the Windows event monitor function. These services automatically start, and therefore you do not need to specify an additional setting.

Remember that the Windows event monitor function does not work when the above services are stopped. If you are using the Windows event monitor function, do not stop these services.

#

For details about the specifications of the support system event (the timing of event detection, etc.), contact Microsoft Corporation.

### 2.3.6 Creating a custom item

In order to be able to use the Item Design View, you have to set up a custom item creation environment. You cannot use the Item Design View without setting up this environment.

For details on the Item Design View, see [7. Item Design View](#).

The procedure for setting up the Item Design View is provided below.

1. As a member of the Administrators group, log in to the host on which to set up the Windows log-on function.
2. Start the command prompt by selecting **Run as administrator**.
3. Execute the following command:

```
cpaitemsetup
```

For details on the setup command (`cpaitemsetup`), see [cpaitemsetup](#) in *9. Commands*.

### 2.3.7 Suppressing the definition function in the operating environment

By taking the following steps, you can set up an environment in which the use of the definition function provided in the Job Design View is restricted so that jobs are managed by using items alone.

1. Log in to the host on which you want to suppress the definition function.
2. Execute the following command:

```
cpausrenvsetup
```

Note that, to be able to use the definition function provided in the Job Design View again, you have to unset up this environment. For details on the setup command and unsetup command (`cpausrenvsetup`), see [cpausrenvsetup](#) in *9. Commands*.

## 2.4 Uninstallation

---

To uninstall CPA:

1. As a member of the Administrators group, log in to the host on which you want to uninstall CPA.
2. Exit all programs.  
Stop the running services and close the Job Design View, if any.
3. If the Windows log-on function has been set up, unset up the function as follows:
  1. Start the command prompt by selecting **Run as administrator**.
  2. Execute the following command:

```
cpaalsetup -u
```

For details on the unsetup command (`cpaalsetup -u`), see *cpaalsetup* in [9. Commands](#).

4. If you have set up a custom item creation environment, take the following steps to unset up the environment:
  1. Start the command prompt by selecting Run as administrator.
  2. Execute the following command:

```
cpaitemsetup -u
```

For details on the unsetup command (`cpaitemsetup -u`), see *cpaitemsetup* in [9. Commands](#).

5. In Windows, open **Control Panel** and select **Programs and Features**.  
A dialog box appears prompting you to select the program you want to uninstall.
6. Select the program to uninstall and then click the **Uninstall** button.
7. When a dialog box appears prompting you to confirm the uninstallation, click the **Yes** button.  
A dialog box appears indicating that the uninstallation is in progress. The uninstallation is completed when the dialog box disappears. If you do not want to uninstall the program, click the **No** button. To suspend the uninstallation, click the **Cancel** button in the dialog box indicating that the uninstallation is in progress.
8. If necessary, manually delete the installation folder and data folder of CPA.

### Important

- If you start the uninstaller while a file is opened, the file or folder might remain undeleted. You need to delete such a file or folder manually.
- If you created files by using **Save job file** or **Save as file** in the Job Design View or Item Design View, you need to delete the files manually, if necessary.
- Files under the installation folder and the data folder (such as settings files and log files) will be deleted<sup>#</sup> during the uninstallation procedure. If you need them, you have to back them up before the uninstallation.  
<sup>#</sup>: User-created folders and files under the installation folder and the data folder, however, will not be deleted.

## 2.5 Port number

---

The following table describes the port number used by CPA and directions through the firewall.

Table 2–2: List of port numbers to be used

Service name	Port number	Used for	Direction through the firewall
cpamsgsrv	22260 <sup>#</sup>	Message communication service	--
		AJS3 linkage function	CPA Option for AJS3 (Custom jobs for CPA linkage) →CPA CPA →CPA Option for AJS3

Legend:

-- : Not applicable

#

- By default, the installer adds a definition to the `services` file so that `cpamsgsrv` can use port 22260. The port number must correspond with the port number that is defined for `cpamsgsrv` on the connected CPA Option for AJS3 host. If the `services` file already has a definition for port 22260, the installer adds another definition so that the service can use the smallest available port number greater than 22260, for example 22261. The installer might accidentally fail to add a definition to the `services` file. Before you can use the AJS3 linkage function, you need to ensure that a port number is defined and it corresponds with the port number that is defined on the connected CPA Option for AJS3 host.
- You can change this port number to any number to suit your environment. However, the number must not conflict with those for other programs.

In addition, the OS dynamically assigns some port numbers, which must not be used by the user. Different OSs assign different numbers depending on their types and versions, and the port numbers may conflict with some default values of the product. If this happens, change the values to avoid the conflict.

For details on the port numbers dynamically assigned by your OS, see the manual of the OS.



## 2.6 Language setting

---

CPA supports Japanese and English. Configure your operating system to use the same language as CPA.

However, you can change the display language for the view to Japanese, English, or Simplified Chinese by editing the language setting in the Job Design View or Item Design View. Once you change this language setting, the set language setting is always applied to the Job Design View or Item Design View even when you change the language for the OS.

With the above in mind, set the same language as the OS language setting for the Job Design View or Item Design View.

### Migration procedure to be used when you change the OS language setting

If you change the OS language setting, information such as logs output afterwards will be in the newly set language. For this reason, you have to follow the following migration procedure when changing the OS language setting:

1. Save files in the view, and execute the `cpasetprof` command to back up job definitions, item definitions, and profile information.
2. After uninstalling CPA in accordance with [2.4 Uninstallation](#), change the OS language setting, and then install CPA again by following the steps described in [2.2 Installation](#).
3. In a newly set language environment, use the definitions you backed up in step 1 to re-create job definitions, item definitions, and profile information, and then register them again.

If you are using the same language type with different character encoding, use the character encoding conversion function provided by the OS to change the character encoding of backed-up definitions.

### Important

- When you start the Job Design View or Item Design View, the OS language setting is applied to the displayed window. For this reason, we recommend that the OS language setting be used in the Job Design View or Item Design View.  
Note that if the OS does not have the language you try to select in the language setting of the Job Design View or Item Design View, the system does not work properly.
- When you share a job definition file across multiple language setting environments or when you manage jobs from AJS3 by using CPA Option for AJS3, all definitions must be in English (ASCII code).
- When Japanese is set as the OS language, logs are output in Japanese. If a language other than Japanese is set as the OS language, logs are output in English.
- When you start the window used to specify detailed definitions for the custom jobs provided by CPA Option for AJS3, the OS language setting is applied to the displayed window. If, however, a language other than Japanese or Simplified Chinese is set as the OS language, information on the window is displayed in English.

# 3

## Installation and Configuration of CPA Option for AJS3

This chapter describes how to install CPA Option for AJS3 and set up configurations for it.

For details on linking CPA Option for AJS3 with AJS3, see [4.3 Using the AJS3 linkage function](#).

## 3.1 Requirements

---

- CPA Option for AJS3 must be installed on an NTFS file system.
- A host on which you want to install CPA Option for AJS3 must have a host name with a length of 128 bytes or less.
- Before you install CPA Option for AJS3 on a host, ensure that AJS3 is set up on the host.

## 3.2 Installation

---

This section describes how to install CPA Option for AJS3.

### 3.2.1 Installing a new instance

To install a new instance of CPA Option for AJS3:

1. As a member of the Administrators group, log in to the host on which you want to install CPA Option for AJS3.
2. Insert the CPA Option for AJS3 media.  
The Hitachi integrated installer window appears.

3. Follow the instructions from the Hitachi integrated installer and specify information as necessary to install the software.

- User name<sup>#1</sup>

Enter the Windows user name which uses CPA Option for AJS3. The length must not exceed 50 bytes.

- Organization<sup>#1</sup>

Enter the name of your company or department. The length must not exceed 80 bytes.

- Installation folder<sup>#2</sup>

Specify the absolute path to the folder in which you want to install CPA Option for AJS3. The length, including a backslash (\) that is automatically added to the end of the path, must not exceed 80 bytes. The absolute path must point to a local directory.

If you do not specify any value, the following value is assumed:

*system-drive*\Program Files (x86)\HITACHI\JP1CPA<sup>#3</sup>

- Data Folder<sup>#2</sup>

Specify the absolute path to the folder in which you want to store CPA Option for AJS3 data. The length, including a backslash (\) that is automatically added to the end of the path, must not exceed 80 bytes. The absolute path must point to a local directory.

If you do not specify any value, the following value is assumed:

*system-drive*\ProgramData\HITACHI\jpl\jpl\_default\JP1CPA

#1: Do not include the following characters in the user name or organization name:

- Symbols except for the following:

Space, left parenthesis ( ( ), right parenthesis ( ) ), plus sign (+), hyphen (-), and underscore ( \_ )

#2: Ensure that the installation folder and data folder do not include the following characters or paths:

- Symbols except for the following:

Space, left parenthesis ( ( ), right parenthesis ( ) ), plus sign (+), hyphen (-), and underscore ( \_ )

Note that you cannot include a space at the beginning or end of the path, before or after a backslash (\), or in succession with another space.

- Multibyte characters

- Paths to non-fixed drives or UNC paths

#3: If you are using a 32-bit operating system, replace Program Files (x86) with Program Files.

4. When the installation is completed, click the **Finish** button.

The following table lists default values for installation settings.

Table 3–1: Default values for installation settings

No.	Setting	Default value
1	Installation folder	<ul style="list-style-type: none"> <li>In a 32-bit operating system: %ProgramFiles%\Hitachi\JP1CPA</li> <li>In a 64-bit operating system: %ProgramFiles(x86)%\Hitachi\JP1CPA</li> </ul>
2	Data folder	%ProgramData%\Hitachi\jpl\jpl_default\JP1CPA
3	Service name of the job execution service	JP1_CPA
4	Service display name of the job execution service	JP1/CPA
5	Startup type of the job execution service	Automatic (AUTO)
6	Service name of the message communication service	JP1_CPA_Message
7	Service display name of the message communication service	JP1/CPA Message
8	Startup type of the message communication service	Automatic (AUTO)
9	Name of the program folder in the <b>Start</b> menu	JP1_Client Process Automation
10	Name of a shortcut in the <b>Start</b> menu to configure the firewall settings	Enable firewall settings
11	Name of a shortcut in the <b>Start</b> menu to revert the firewall settings	Disable firewall settings

### 3.2.2 Overwriting or upgrading an existing instance

To upgrade an existing instance of CPA Option for AJS3:

- As a member of the Administrators group, log in to the host on which you want to install CPA Option for AJS3.
- Exit all programs.  
Stop the running services and close the Job Design View, if any.
- Insert the CPA Option for AJS3 media and start the installation.  
Follow the instructions from the Hitachi integrated installer to upgrade or overwrite the existing instance.

#### Important

If an unexpected error occurs while you overwrite or upgrade an existing instance, the data of the instance might be lost and no longer be restored. Make a backup copy of the data folder before you can start overwriting or upgrading.

### 3.2.3 Notes on installation

For details on the notes on installation, see [2.2.4 Notes on installation](#).

## 3.3 Configurations

---

This section describes how to set up configurations for CPA Option for AJS3.

### 3.3.1 Configurations of the message communication service

The message communication service is required for using the AJS3 linkage function. You need to configure and start the message communication service on both the host running CPA and the host running CPA Option for AJS3.

#### (1) Changing the port number of the message communication service

Installing CPA Option for AJS3 causes the `services` file to have a definition on the port number to be used by the message communication service. The port number must correspond with the port number that is defined for `cpamsgrv` on the connected CPA host. For details on the port number, see [3.5 Port number](#).

To change the default port number, edit the `services` file.

To change the default service name for the port number, edit the message communication service definition file (`iesconf.xml`). For details on the message communication service definition file, see [E.2 Message communication service definition file](#).

Definition to be included in the `services` file

```
cpamsgrv 22260/tcp # JP1 Client Process Automation
```

#### Important

- The message communication service fails to start with the error KNA00950-E when the port number is not defined in the `services` file.
- Ensure that every CPA host that will use the AJS3 linkage function is also configured to have the `services` file defining the same port number.

#### (2) Starting the service

You do not need to start the service manually because the installer ensures that the service starts automatically. However, you need to start the service manually if you change the startup type from `Automatic` to a different one after the installation process. To start the service:

1. Open **Control Panel** and select **Services**, or select **Administrative Tools** and then **Services**.
2. Select the JP1/CPA Message service.
3. Click the **Start** button.

### 3.3.2 Setting up components for the AJS3 linkage function

To use the AJS3 linkage function, you need to set up hosts where CPA or CPA Option for AJS3 is installed and then start the job execution service and the message communication service on each host.

## (1) Setting up a CPA host

1. Install CPA.

For details on how to install it, see [2.2 Installation](#).

2. Start the JP1/CPA service and the JP1/CPA Message service.

## (2) Setting up AJS3 - Manager and Agent hosts

1. Set up AJS3 - Manager and Agent.

For details on how to set up AJS3 - Manager and Agent, see the manual *JP1/Automatic Job Management System 3 Configuration Guide*.

2. Install CPA Option for AJS3.

For details on how to install it, see [3.2 Installation](#). If you want jobs to be managed and executed on the AJS3 - Manager host, you install CPA Option for AJS3 on the AJS3 - Manager host. If you want jobs to be executed on a separate host from the AJS3 - Manager host, you install CPA Option for AJS3 on the AJS3 - Agent host.

3. Start the JP1/CPA service and the JP1/CPA Message service.

## (3) Setting up a AJS3 - View host

1. Copy the following files, which are provided with CPA Option for AJS3, to the AJS3 - View host:

- The defining program for the CPA linkage custom job and the resource file storage folder

[File location]

- *data-path*\addin\forajs\view\cpacustomdef.exe

- *data-path*\addin\forajs\view\ja-JP\  
[Destination]

Any location on the AJS3 - View host

Ensure that the *cpacustomdef.exe* file and the *ja-JP* folder exist at the same level.

Some security products or other protections limit locations where executable files can run. In such a case, you need to select a location so that the executable file can be allowed to run.

- Icon for the CPA linkage custom job<sup>#</sup>

[File location]

*data-path*\addin\forajs\view\CUSTOM\_PC\_CPALINK.gif

[Destination]

*AJS3-View-install-path*\image\custom

<sup>#</sup>: In AJS3 - View version 12-50 or later, you do not have to copy it as it is a standard custom job.

2. Use AJS3 - View to register the CPA linkage custom job.

To register the CPA linkage custom job, you complete the registration information with values described in the table below.

For details on how to register a custom job, see the manual *JP1 Version 12 JP1/Automatic Job Management System 3 Linkage Guide*.

Table 3–2: Values you specify to register a custom job

Item	Value
Defining program	The file path name of the custom job definition program for CPA linkage, which you copied in step 1
Execution program	<i>CPA-Option-for-AJS3-install-path</i> \bin\cpalink.exe (fixed)
Version	1200 (fixed)
Class name	CPALINK (fixed)
Job type	PC

## (4) Configuring communication with CPA

### (a) Host name resolution

To communicate with CPA, ensure that the server running CPA Option for AJS3 can resolve the names of the connected hosts. A way to do this is to use the `hosts` file to resolve the host names to the IP addresses.

### (b) Configuring the firewall and other protections to allow traffic

To communicate with CPA, ensure that the CPA Option for AJS3 host and the devices on network routes to CPA are configured with security protections to allow inbound and outbound traffic on the specified port number. The port number is defined as described in *3.3.1(1) Changing the port number of the message communication service*.

For Windows Firewall, you can use the **Enable firewall settings** command provided in the **Start** menu or the `cpasetfirewall` command to configure the settings.

Note that the CPA Option for AJS3 uninstaller does not revert the settings. If you no longer use CPA Option for AJS3, you need to revert the settings by using the **Disable firewall settings** command provided in the **Start** menu or the `cpasetfirewall` command before uninstalling CPA Option for AJS3.

Remember that the **Enable firewall settings** and **Disable firewall settings** commands require administrator privileges. You must run the commands as an administrator.

To enable communication with CPA, configure Windows Firewall on both hosts to allow appropriate traffic.

#### Important

Ensure that the time zone is synchronized between the host on which CPA is installed and the one on which CPA Option for AJS3 is installed.

### (c) Using the AJS3 linkage function in a cluster

Ensure that hosts running CPA can communicate with both the IP address of the physical host running CPA Option for AJS3 and its logical IP address.

## 3.3.3 Clustering

This section describes how to set up a cluster.

This product supports an active-standby cluster configuration.



## (1) Setting up the active node

Install the product with the data path to a mounted shared disk.

1. Mount a shared disk. (Example path: v:\SHDISK\cpa)
2. Set the data path to the folder you mounted in step 1, and then install the product.
3. From Windows Administrative Tools, open **Services**. For both the job execution service and the message communication service, open the Properties window and set **Startup Type** to **Manual**.
4. From Windows Administrative Tools, open **Services**. Stop the job execution service and the message communication service.

## (2) Setting up the standby node

Install the product with the data path to a work folder, rather than a mounted shared disk. After the product is installed, edit the product information file to change the path to point to the shared disk.

1. Create a work data folder. (Example path: c:\temp\cpadata)
2. Set the data path to the folder you created in step 1, and then install the product.
3. Edit the product information file (ga.ini).

File location: *install-path*\ga.ini

(Default location: C:\Program Files (x86)\Hitachi\JP1CPA\ga.ini)

<Default value>

```
# INSTALL DATE : 2019/xx/xx xx:xx:xx.xx
# COMPUTERNAME : xxxxxxxxxxxxxx
# USERNAME      : xxxxxxxxxxxx
[ProductInfo]
InstallDir="C:\Program Files (x86)\Hitachi\JP1CPA"
DataDir="c:\temp\cpadata"
Version="120000"
SchemaVersion="1"
GAServiceName="JP1/CPA"
StartupType="AUTO"
IESCSERVICEName="JP1/CPA Message"
IESCStartupType="AUTO"
# END
```

<Changed value>

```
# INSTALL DATE : 2019/xx/xx xx:xx:xx.xx
# COMPUTERNAME : xxxxxxxxxxxxxx
# USERNAME      : xxxxxxxxxxxx
[ProductInfo]
InstallDir="C:\Program Files (x86)\Hitachi\JP1CPA"
DataDir="v:\SHDISK\cpa"
Version="120000"
SchemaVersion="1"
GAServiceName="JP1/CPA"
StartupType="AUTO"
IESCSERVICEName="JP1/CPA Message"
IESCStartupType="AUTO"
# END
```

4. From Windows Administrative Tools, open **Services**. For both the job execution service and the message communication service, open the Properties window and set **Startup Type** to **Manual**.
5. From Windows Administrative Tools, open **Services**. Stop the job execution service and the message communication service.

### **(3) Registering the services with your cluster software**

Register the job execution service and the message communication service with your cluster software. For details on how to register the services, see documents for your cluster software.

## 3.4 Uninstallation

---

To uninstall CPA Option for AJS3:

1. As a member of the Administrators group, log in to the host on which you want to uninstall CPA Option for AJS3.
2. Exit all programs.  
Stop the running services and close the Job Design View, if any.
3. In Windows, open **Control Panel** and select **Programs and Features**.  
A dialog box appears prompting you to select the program you want to uninstall.
4. Select the program to uninstall and then click the **Uninstall** button.
5. When a dialog box appears prompting you to confirm the uninstallation, click the **Yes** button.  
A dialog box appears indicating that the uninstallation is in progress. The uninstallation is completed when the dialog box disappears. If you do not want to uninstall the program, click the **No** button. To suspend the uninstallation, click the **Cancel** button in the dialog box indicating that the uninstallation is in progress.
6. If necessary, manually delete the installation folder and data folder of CPA Option for AJS3.

### Important

- If you start the uninstaller while a file is opened, the file or folder might remain undeleted. You need to delete such a file or folder manually.
- If you created files by using **Save job file** or **Save as file** in the Job Design View or Item Design View, you need to delete the files manually, if necessary.
- The uninstaller deletes files and folders except for ones created by Windows users directly under the installation folder or data folder. To retain the configuration files or log files, you need to copy such files before uninstalling.

## 3.5 Port number

The following table describes the port number used by CPA Option for AJS3 and directions through the firewall.

Table 3–3: List of port numbers to be used

Service name	Port number	Used for	Direction through the firewall
cpamsgsrv	22260 <sup>#</sup>	Message communication service	--
		AJS3 linkage function	CPA Option for AJS3 (Custom jobs for CPA linkage) →CPA CPA →CPA Option for AJS3

Legend:

-- : Not applicable

#

- By default, the installer adds a definition to the `services` file so that `cpamsgsrv` can use port 22260. The port number must correspond with the port number that is defined for `cpamsgsrv` on the connected CPA host. If the `services` file already has a definition for port 22260, the installer adds another definition so that the service can use the smallest available port number greater than 22260, for example 22261. The installer might accidentally fail to add a definition to the `services` file. You need to ensure that a port number is defined and it corresponds with the port number that is defined on the connected CPA host.
- You can change this port number to any number to suit your environment. However, the number must not conflict with those for other programs.

In addition, the OS dynamically assigns some port numbers, which must not be used by the user. Different OSs assign different numbers depending on their types and versions, and the port numbers may conflict with some default values of the product. If this happens, change the values to avoid the conflict.

For details on the port numbers dynamically assigned by your OS, see the manual of the OS.

# 4

## Operation of CPA

This chapter, which describes the functions available for the operation of CPA, is intended for those responsible for operating the CPA system itself as well as for those responsible for operating jobs.

## 4.1 Operating jobs

---

This section, which describes how to operate CPA jobs by using the job operation functions provided by CPA, is intended for those responsible for the operation of CPA jobs.

If you have deleted the definition function by following the instructions given in [2.3.7 Suppressing the definition function in the operating environment](#), operate CPA jobs by following the procedures described in [4.1.2 Registering a job](#) and later subsections.

### 4.1.1 Defining an item

You can create an action flow by using the action items, action flow items, or standard items that have been created in the Item Design View.

#### (1) Defining an action flow

In the Job Design View, select the **Action flow** tab, and then by clicking the **New** button, open the **Action flow** dialog box.

By using action items, action flow items, or event monitor items, create a flow.

For details on how to operate the **Action flow** dialog box after opening it, see [6.3.1 Action flow dialog box](#).

#### (2) Changing the action flow definition

In the **Action flow** tab in the Job Design View, select a registered action flow item, and then click the **Edit** button to open the **Action flow** dialog box.

Change the definition of the registered action flow, and then save it under the same item name to overwrite the existing action flow.

For details on how to operate the **Action flow** dialog box after opening it, see [6.3.1 Action flow dialog box](#).

If you change the definition of an action flow item used by a registered job, keep the following in mind:

In the case of a job executed on time:

The new definition is applied the next time the job is executed. No particular operation is required.

In the case of a job executed on event:

The new definition is applied upon the reregistration of the job.

#### (3) Defining a calendar

In the Job Design View, select the **Calendar** tab, and then click the **New** button to open the **Calendar** dialog box.

For details on how to operate the **Calendar** dialog box after opening it, see [6.3.2 Calendar dialog box](#).

#### (4) Changing the calendar definition

In the **Calendar** tab in the Job Design View, select a registered calendar item, and then click the **Edit** button to open the **Calendar** dialog box.

Change the definition of the registered calendar, and then save it under the same item name to overwrite the existing calendar.

For details on how to operate the **Calendar** dialog box after opening it, see [6.3.2 Calendar dialog box](#).

If you change the definition of a calendar item used by a registered job, the change made to the calendar definition is automatically applied during operation. If you want to immediately apply the change made to the calendar definition, reregister the job that is using the calendar item.

## 4.1.2 Registering a job

Define when to execute the created item, and then execute it as a job.

### (1) Executing a job according to schedule

If you want to have an action executed according to the rules set out by a schedule (for example, if you want to have an action repeatedly executed according to the schedule set forth in a calendar), take the following steps:

In the Job Design View, select the **Exec. on time** tab, and then either click the **New** button or select a registered job definition and click the **Edit** button to open the **Job exec. on time** dialog box.

For details on how to operate the **Job exec. on time** dialog box after opening it, see [6.2.1 Job exec. on time dialog box](#).

You can also have the job executed within the specified valid time frame.

You can specify the time frame for determining job execution dates in the "from *yyyy/mm/dd* until *yyyy/mm/dd*" format. The following table gives an overview of the items you must specify to set a valid time frame.

Table 4–1: Items you must specify to set a valid time frame

No.	Type	Setting	Setting value	Valid range
1	Valid date (from when)	Year, month, and date	<i>yyyy/mm/dd</i>	Execution dates are determined within the time frame starting from the date specified with this setting.
2	Valid date (until when...) <sup>#</sup>	Year, month, and date	<i>yyyy/mm/dd</i>	Year: 2013 to 2037 Month: 01 to 12 Date: 01 to 31

#

Omitting this setting causes the system to create execution schedules for 367 days starting from the registration date of the job.

For example, if you register a job on January 1, 2019, the system creates execution schedules for a period of time until January 3, 2020. In this case, immediately after you registered the job, there is no schedule for January 4, 2020 or later. Such schedules will be created automatically on January 2, 2019 and later.

If no schedule is created when you register a job, the system will create the next schedule later during operation. That is, you do not need to re-register the job 367 days before the execution.

### (2) Executing a job after an event

If you want to have an action executed after a specific event such as the creation of a file or the arrival of an email, take the following steps:

In the Job Design View, select the **Exec. on event** tab, and then either click the **New** button or select a registered job definition and click the **Edit** button to open the **Job exec. on event** dialog box.

For details on how to operate the **Job exec. on event** dialog box after opening it, see [6.2.2 Job exec. on event dialog box](#).

### 4.1.3 Checking the state of a job

You can check whether the registered job is correctly scheduled for execution, whether it has been executed, and whether the executed job ended normally.

In the Job Design View, select the **Exec. on time** tab. In the displayed list of jobs to be executed on time, select the name of the job whose state you want to check, and then click the **Checking the status** button to open the dialog box that allows you to check the state of the job. If you want to check the state of a job to be executed on event, select the **Exec. on event** tab in the Job Design View, and then in the displayed list of jobs to be executed on event, select the name of the job whose state you want to check, and then follow the same step as the one used to check the state of a job executed on time to check the state of the selected job.

For details on how to operate the **Checking the status** dialog box after opening it, see [6.2.3 Checking the status dialog box](#).

You can also check the state of a job by using the `cpareport` command. For details on the `cpareport` command, see [cpareport](#) in [9. Commands](#).

For a list of displayed states and the meaning of each state, see [1.1.4 State management for jobs and actions](#).

### 4.1.4 Job operation function

The following table describes job operations. All operations are executed by commands. For details about the commands, see [9. Commands](#).

Table 4–2: List of job operations

No.	Function	Description	Command
1	Register <sup>#</sup>	Registers jobs.	<code>cpajobentry</code>
2	Delete <sup>#</sup>	Deletes an existing job.	<code>cpajobdelete</code>
3	Stop (Forced stop) <sup>#</sup>	Forcibly stops all running actions and then stops the job.	<code>cpajobstop -t</code>
4	Stop (Suspend)	Waits for all running actions to finish and then stops the job.	<code>cpajobstop -c</code>
5	Stop (Stop monitoring)	Stops monitoring the execution conditions for the job executed on event are met.	<code>cpajobstop -e</code>
6	Retry <sup>#</sup>	Reruns jobs.	<code>cpajobrun</code>

#

The following operations can be executed also from the Job design view: register, delete, stop (forced stop), and rerun.

If you want to resume monitoring for a job executed on event for which monitoring has been deactivated, you need to take one of the following steps:

- Overwrite and then register the job in question by using either the Job Design View or the `cpajobentry` command.
- Restart the job execution service.



## 4.2 Operating jobs that use automatic logon

---

This section, which describes how to operate jobs that use the Windows log-on function implemented on CPA, is intended for those who are responsible for operating the CPA system.

### 4.2.1 Windows log-on function

The Windows log-on function performs log-on to Windows or unlocks the screen.

For details on the setup, see [2.3.4 Using the Windows log-on function](#).

Users with the following accounts can log on to Windows by using the Windows log-on function:

- Local account
- Domain account

You can change the log-on user information (Windows user name and password) during operation by using the `cpasetumap` command. For details on the `cpasetumap` command, see [cpasetumap](#) in [9. Commands](#).

### 4.2.2 Notes on Windows log-on function

The Windows log-on function might not work properly depending on several conditions such as the machine settings, machine state, and log-on users. We therefore recommend that you keep the following precautions in mind when operating the Windows log-on function:

- Execute the Windows log-on action while the Windows log-on window is displayed.
- For example, suppose that you execute an action flow that involves the use of Windows event monitoring to monitor for log-off events followed by the execution of a Windows log-on action. In this case, a time lag between a log-off event and the subsequent display of the Windows log-on window can cause the subsequent Windows log-on action to end with an error. To prevent this from happening, you can insert a `sleep#` or other similar process before the execution of the Windows log-on action.

#

Time from log-off to the display of the Windows log-on window varies with each environment. You must therefore first verify that inserting a `sleep` does not cause any problems.

- Because screensaver is a state different from screen lock, you cannot cancel screensaver by using the unlock screen functionality.
- If **Interactive logon: Do not require CTRL+ALT+DEL** (the option you can access by opening the Local Security Policy console of Windows 10 and selecting **Local Policies**, and then **Security Options**) is set to **Disabled** (which, by default, is set to **Not Defined**), the Windows log-on function does not work. If you want to change this setting, you have to first check the security policy of the system you are using.
- If the log-on account for the job execution service is changed to an account other than the local system account, the Windows log-on action ends abnormally.

## 4.3 Using the AJS3 linkage function

This subsection, which describes the functions provided by CPA Option for AJS3, is intended for those responsible for operating jobs in CPA that is linked with AJS3.

### 4.3.1 List of CPA Option for AJS3 functions

The following table describes and lists CPA Option for AJS3 functions.

Table 4–3: List of CPA Option for AJS3 functions

No.	Function	Description
1	AJS3 linkage function	Enables a custom job for CPA linkage that is defined in a jobnet of AJS3 to trigger a job that is created in CPA.

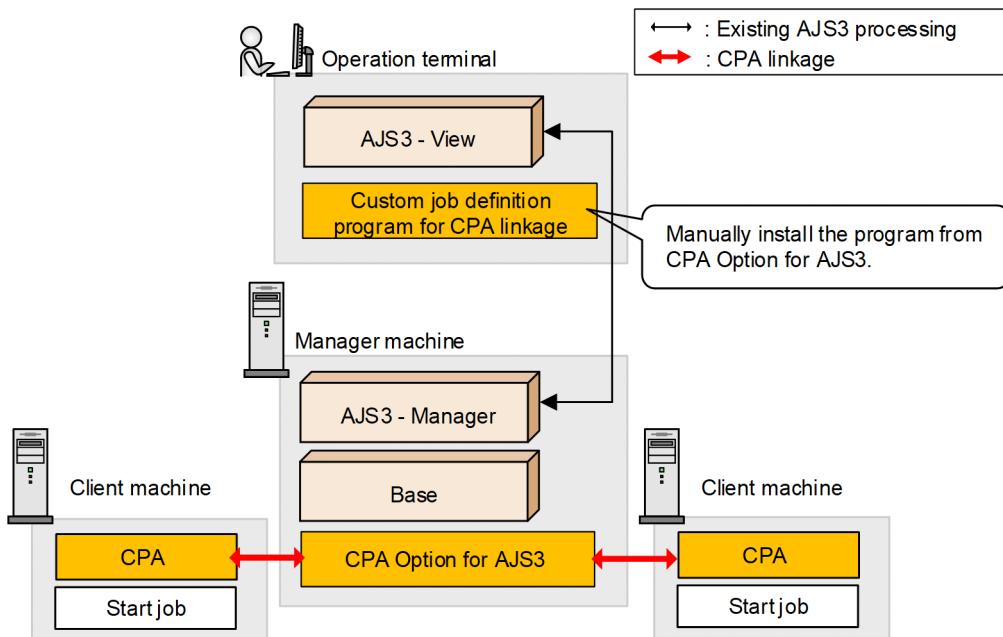
### 4.3.2 AJS3 linkage function

The AJS3 linkage function executes the jobs running on the client OS where CPA is installed through a jobnet of AJS3 - Manager.

### 4.3.3 System configuration

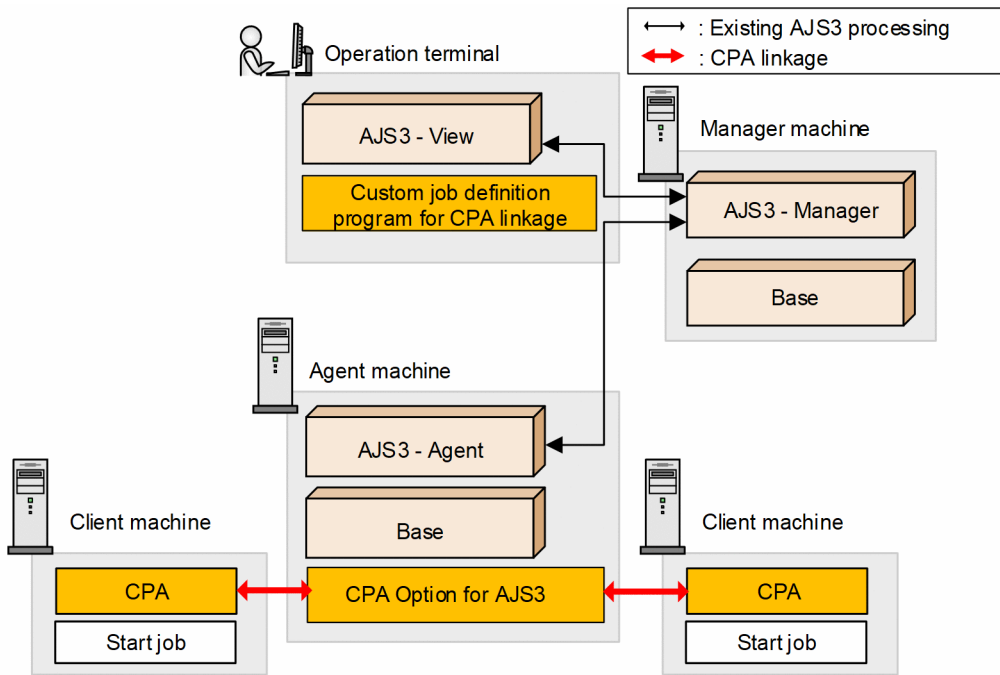
To use the AJS3 linkage function, you need to install CPA Option for AJS3 on the AJS3 - Manager host first. Then you install the custom job definition program for CPA linkage on the AJS3 - View host, and register a custom job for CPA linkage. For details on how to set up, see [2.3.3 Configurations for the AJS3 linkage function](#), and [3.3 Configurations](#).

Figure 4–1: System configuration (AJS3 - Manager linkage)



If CPA Option for AJS3 cannot be installed on the AJS3 - Manager host, you can use CPA Option for AJS3 by installing it on the AJS3 - Agent host.

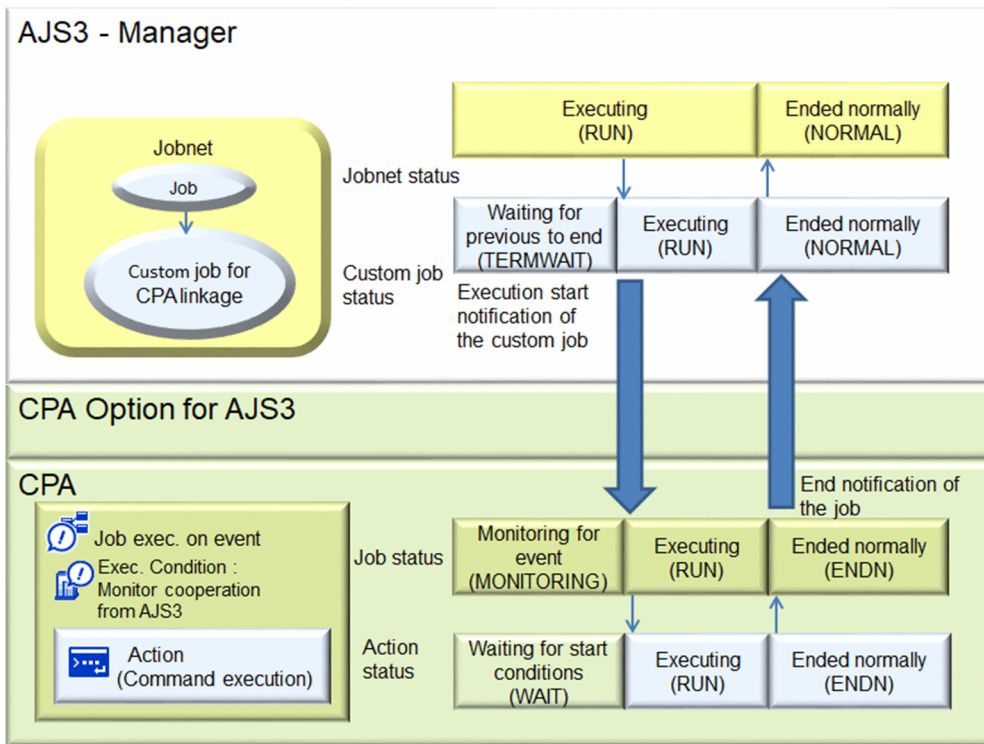
Figure 4–2: System configuration (AJS3 - Agent linkage)



### 4.3.4 Executing jobs

Registering and executing the custom job for CPA linkage defined in AJS3 allows you to execute the jobs that are registered in CPA as *jobs executed on event* with the execution condition of *Monitor cooperation from JP1AJS3*. The custom job for CPA linkage will finish after the CPA job finishes.

Figure 4–3: Overview of the function that executes CPA jobs from AJS3



### 4.3.5 Procedure to use the function

This section describes the procedure needed to use the AJS3 linkage function.

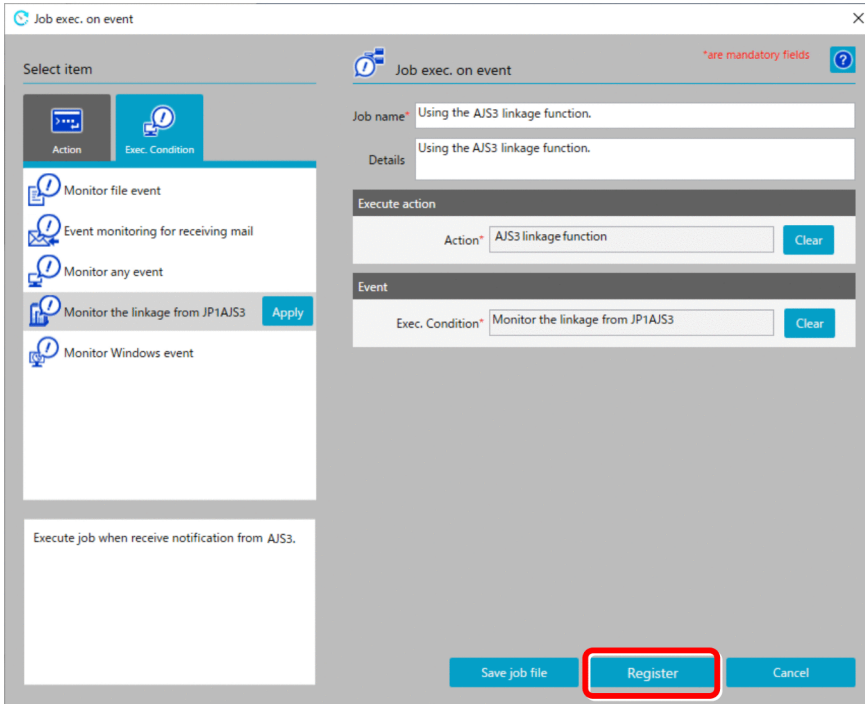
It is assumed that setups are completed on the relevant hosts. For details on how to set up, see [2.3.3 Configurations for the AJS3 linkage function](#), and [3.3 Configurations](#).

#### (1) Procedure to be done on the CPA host

You need to register a job for the AJS3 linkage function on the CPA host.

Register a job as follows:

1. In the Job Design View, select the **Exec. on event** tab, and then click the **New** button.  
In the displayed **Job exec. on event** dialog box, create a job for the AJS3 linkage function.

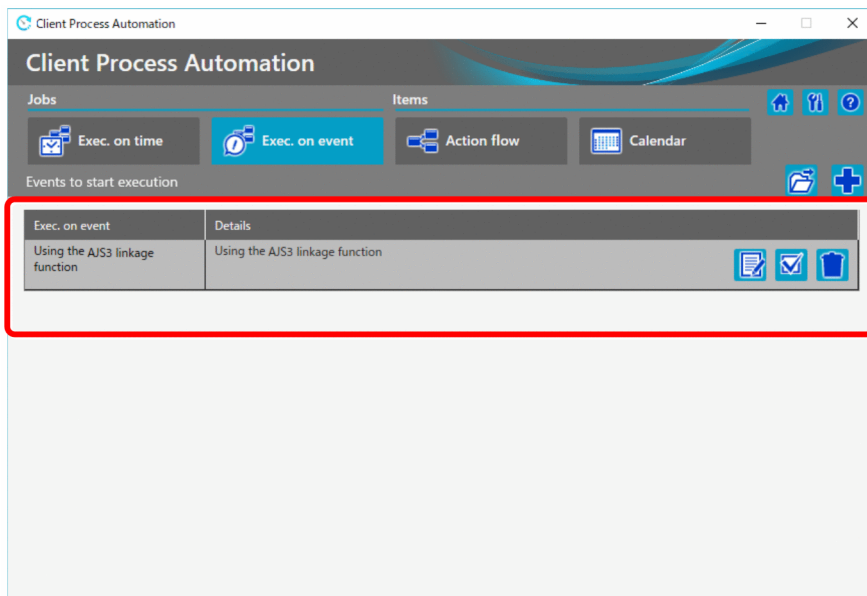


The following table lists and describes the setting items for the job you create.

Item	Description
Job name	Enter a name of the job.
Details	Enter a description.
Execute action	From the list of action items, select and apply an action item or an action flow you created.
Event	From the list of execution condition items, select and apply Monitor the linkage from JP1AJS3.

2. Click the **Register** button.

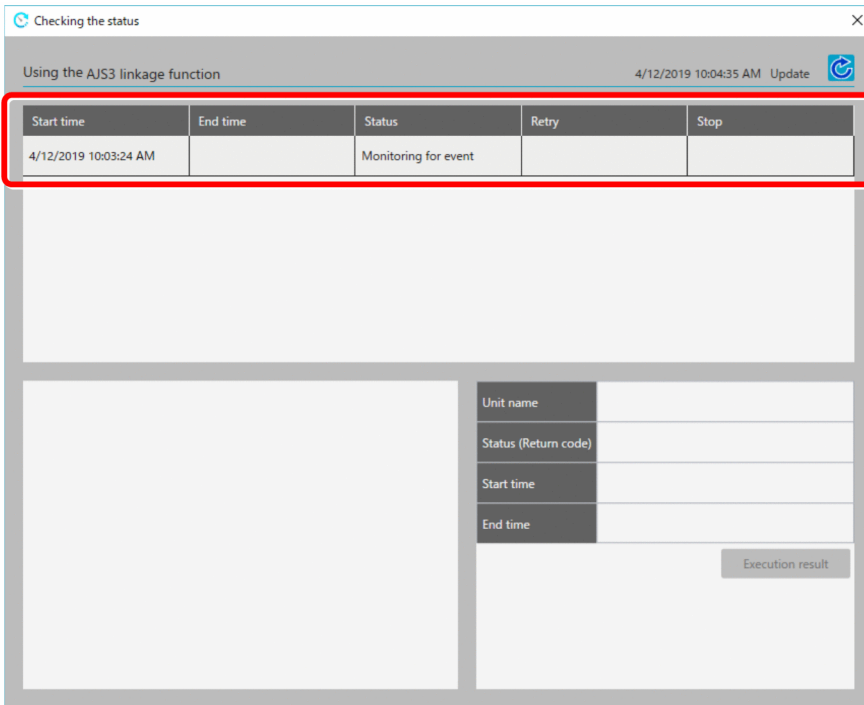
Register the job executed on event. When the event is registered, AJS3 starts event monitoring.



3. Click the **Checking the status** icon.

4. Operation of CPA

Open the **Checking the status** dialog box. In the **Checking the status** dialog box, ensure that the job is in the Monitoring for event state. Then, your configuration on the CPA host is completed.



**! Important**

If you re-execute a job, a new history of executed jobs is added for that job.

The **Start time** box displays the time at which the job was re-executed, whereas the **End time** box displays a copy of the end time of the job that was re-executed.

**(2) Procedure to be done on the AJS3 - View host**

You need to add a custom job for CPA linkage to a jobnet on AJS3 - View, and register it for execution.

The following procedure shows how to set up.

1. In the Jobnet Editor window, open the jobnet to which you want to add a custom job for CPA linkage. Open the jobnet you want to edit, and select **Exclusive edit**.
2. From the icon list, drag the icon of the custom job for CPA linkage and drop it onto the map area. This displays the Define Details - [Custom Job] dialog box.
3. Define attributes such as the unit name.

For **Exec-agent**, specify the execution agent name of the host where CPA Option for AJS3 is installed. The value is the name of the execution agent of the AJS3 - Agent host if CPA Option for AJS3 is installed on AJS3 - Agent.

For details on execution agents, see the manual *JP1 Version 12 JP1/Automatic Job Management System 3 Configuration Guide*.

When a CPA job ends with a warning, you might want to change the state of the custom job for CPA linkage to *Ended with warning* as well. In this case, specify **End judgment** in the **Definition** page as follows:

- **Rule: Judgment by threshold**
- **Warning:** 0

- **Abnormal:** 1

4. Click the **Details** button.

When the **Detailed** definition window appears, enter the following information.

Setting	Setting value
Client host name	The name of the host where CPA is installed
Job name	The name of the job for the AJS3 linkage function registered on CPA

The host names must be resolved between the host specified for the client host name and the execution host corresponding to the execution agent name of the custom job for CPA linkage. Network environment is also required for such hosts to communicate with each other.

5. In the Define Details - [Custom Job] dialog box, click the **OK** button.

### (3) Executing the job on AJS3

In *4.3.5(2) Procedure to be done on the AJS3 - View host*, you defined a custom job for CPA linkage in AJS3 - View. The custom job is triggered by a schedule or event that is registered for execution in AJS3.

When the custom job for CPA linkage is executed, it, in turn, triggers the CPA job you registered. After the CPA job finishes, the execution result (normal end, abnormal end, or others) is applied to the custom job for CPA linkage. Finally, you access the execution result details of the custom job for CPA linkage to view the execution log.

#### **!** Important

- The execution results of individual actions and action flows that are called from the jobs executed on CPA need to be checked on CPA.
- If the custom job for CPA linkage is forced to terminate on AJS3, the execution program remains running rather than ends on CPA. If you want to terminate the execution program, run the `cpajobstop` command on the CPA host.
- Custom jobs for CPA linkage are controlled by the setting on the maximum number of concurrently executable jobs in AJS3.

## 4.4 Job operation with the execution user specified

---

Intended for users who operate CPA jobs, this section describes how to operate jobs with the execution user setting function of CPA.

### 4.4.1 Execution user setting function

Using the execution user setting function, you can operate the following jobs:

- Execute commands that require a user profile or user environment variables.
- Execute commands that access files on the network.
- Monitor files on the network.

For details on available files on the network, see [4.4.1\(3\) Using files on the network](#).

With the execution user setting function, you can specify a virtual user name for the item to be executed.

The item is executed by the Windows user associated with the virtual user name.

The CPA virtual user to be associated with the information (Windows user name and password) of the user who executes the item must be added to the user mapping file (`usrprof.xml`).

To add the virtual user and the information of the user associated with the virtual user to the user mapping file (`usrprof.xml`), execute the `cpasetumap` command. For details on the `cpasetumap` command, see [cpasetumap](#) in [9. Commands](#).

#### Important

The Windows user associated with the virtual user name must have the following user rights:

- Local logon

Note that if the user is an administrator, the local logon rights may be granted.

### (1) Defining a virtual user name for an item

You can specify the name of a virtual user who executes an item in the item definition of the Job Design View or Item Design View. In the item definition of the Item Design View, you can also create an item with a virtual user defined for it.

For details on the item definition of the Job Design View, see [6.2 Job operation windows](#) and [6.3 Window used to define items](#).

For details on the item definition in the Item Design View, see [7.2 Item definition windows](#).

### (2) Executing an item as a specified user

If you specify a virtual user name, an item is executed with the Windows user associated with the virtual user name. If the virtual user name is not specified, the item is executed with the account for the CPA job execution service.



Depending on whether or not the virtual user name is specified, items are executed differently at command execution. For details, see *A.3 Different behaviors in different command execution environments*.

### (3) Using files on the network

When a Windows user executes an item, the user can have access to files on the network. The Windows user who executes it must be able to access the files on the network.

Note that you specify a file name in the format available without the need for assigning it as a network drive (a path starting with `\\computer-name`). A network drive name cannot be specified.

A file on the network can be specified in the parameter and the file name to be monitored. It can also be used in the file with the `.bat` or `.exe` extension specified for the execution command or the command name.

#### Important

If a file on the network temporarily becomes inaccessible, it may take time to execute an item or detect an event.

Use items with the names of files on the network in a stable network environment.

## 4.5 Migrating definitions

This section, which describes the functionality available to save and recover files containing the definitions of the items and jobs created with CPA so that you can migrate the definitions to another location, is intended for those responsible for operating the CPA system.

You can, for example, save definitions, distribute them to another host, and then recover the definitions there so that you can build an operating environment in that host. Note that you can only save files containing definitions of the jobs and items created on CPA. The saved information does not contain the execution results of jobs or the commands and icons used in items. You must therefore manually save and recover commands and icons.

### 4.5.1 Saving and recovering definition files

The Job Design View and the Item Design View are used for the saving and recovery of files containing definitions of jobs and items. You can also save files with the `cpajobprint` command and recover them with the `cpajobentry` command.

The profile information and virtual user information, which are used in action flows or for logging on to Windows, can be saved and recovered by using the `cpasetprof` command and the `cpasetumap` command, respectively.

For details on the Job Design View and the Item Design View, see [6. Job Design View](#) and [7. Item Design View](#), respectively. For details on commands, see [9. Commands](#).

The following table describes the functions used to save and recover definition files.

Table 4–4: Functions used to save and recover definition files

No.	Definitions to be saved and recovered	Functions used to save definitions	Functions used to recover definitions
1	Jobs	<ul style="list-style-type: none"><li>Job Design View Operation: <b>Save job file</b></li><li><code>cpajobprint</code> command</li></ul>	<ul style="list-style-type: none"><li>Job Design View Operation: <b>Open</b></li><li><code>cpajobentry</code> command</li></ul>
2	Custom items	<ul style="list-style-type: none"><li>Job Design View Operation: <b>Save as file</b></li><li><code>cpajobprint</code> command</li></ul>	<ul style="list-style-type: none"><li>Job Design View Operation: <b>Open</b></li><li><code>cpajobentry</code> command</li></ul>
3	Definitions created in the Job Design View: <ul style="list-style-type: none"><li>Action flows</li><li>Calendars</li></ul> Definitions created in the Item Design View: <ul style="list-style-type: none"><li>Actions</li><li>Action flows</li><li>Calendars</li><li>Execution date</li><li>Execution time</li></ul>	<ul style="list-style-type: none"><li>Item Design View Operation: <b>Save as file</b></li><li><code>cpajobprint</code> command</li></ul>	<ul style="list-style-type: none"><li>Item Design View Operation: <b>Open</b></li><li><code>cpajobentry</code> command</li></ul>
4	Profiles	<code>cpasetprof</code> command	<code>cpasetprof</code> command
5	Virtual users	<code>cpasetumap</code> command	<code>cpasetumap</code> command

When you recover definition files, the definitions used in jobs take effect when one of the actions described in the following table takes place. After confirming when the recovered definitions take effect, perform the necessary operations.

Table 4–5: When the definitions take effect

No.	Target definition	Where the definitions are used	When the definitions take effect
1	Action items, action flow items	Jobs executed on time	The next time the job is executed
2		Jobs executed on event	<ul style="list-style-type: none"> <li>• Reregistration of the job</li> <li>• Restart of the job execution service</li> </ul>
3	Calendar items, execution date items, execution time items	Jobs executed on time	<ul style="list-style-type: none"> <li>• Automatically updated during operation</li> <li>• Immediately applied upon the reregistration of the job</li> </ul>
4	Profiles	Action flows	The next time the unit is executed
5	Virtual users	Actions (Windows log-ons)	The next time the user logs on to Windows
6		Item with a virtual user name	The next time the item is executed

## 4.5.2 Notes on saving and recovering definitions

- To recover the saved definition file to another host, the destination host must have the same or a newer version of CPA as the one installed on the original host.
- When profiles and virtual user names are used in custom items, we recommend that you recover all information at the destination host because there is no way to confirm their dependencies.
- If the Send mail function, the Monitoring mail reception function, and the AJS3 linkage function are used, you have to set them up at the destination host as well.  
For details on the setup, see [2.3 Configurations](#).
- The save function outputs data to UTF-8-encoded xml files. Do not change the contents, character encoding, and file extension (.xml) of the output files. In particular, when you transfer, compress, or decompress definition files, be careful not to accidentally make changes to them.
- The language environment at the destination host must be the same as the one at the original host. However, jobs and items created in an English environment can be recovered to any language environment.

## 4.6 Migrating the CPA system

This section, which describes how to save and recover information regarding the entire CPA system, including the definitions and execution results of the jobs and items created with CPA, is intended for those responsible for operating the CPA system.

We recommend that you regularly save the information regarding the entire CPA system before and during the operation of jobs.

Note that you cannot select and recover a specific piece of information, such as a job definition, from the information you have saved by following the procedure described in this section. For details on how to save and recover the definitions of jobs and items created with CPA, see [4.5 Migrating definitions](#).

### 4.6.1 Data to be saved

You can save and recover the CPA system by copying the files managed by CPA.

The following table describes the data to be saved.

Table 4–6: CPA data to be saved

No.	File name / Folder name	Type	Description
1	Data folder The name of the data folder specified upon the installation of CPA Default data folder: %ProgramData%\HITACHI\jpl\jpl_default\JP1CPA	Folder	CPA data folder

### 4.6.2 Saving data

You can save data as follows:

1. If you are using the Job Design View or any other CPA window, close it.
2. Stop the following two CPA services:
  - Service name: JP1\_CPA
  - Service name: JP1\_CPA\_Message
3. Save the files containing CPA system data.  
(Example) To save C:\ProgramData\HITACHI\jpl\jpl\_default\JP1CPA:
  1. Using Explorer, display the data folder.  
If the C:\ProgramData folder is hidden from view, select the **Hidden items** check box in the **View** tab.
  2. Navigate to the JP1CPA folder you want to save, right-click it, and then select **Send to**, then **Compressed (zipped) folder**. The JP1CPA.zip file is generated inside the folder.
  3. Move the generated JP1CPA.zip file to the disk in which to save it.
4. Start the services you stopped in step 2.

### 4.6.3 Recovering data

You can recover the saved data as follows:

1. If you are using the Job Design View or any other CPA window, close it.

2. Stop the following two CPA services:

- Service name: JP1\_CPA
- Service name: JP1\_CPA\_Message

3. Recover the saved files.

To recover the saved files to C:\ProgramData\HITACHI\jpl\jpl\_default\JP1CPA:

1. Using Explorer, display the data folder.

If the C:\ProgramData folder is hidden from view, select the **Hidden items** check box in the **View** tab.

2. Navigate to the JP1CPA folder to which to recover the saved data, and delete any folders and files under it.

3. Recover the saved files to the JP1CPA folder.

4. Start the services you stopped in step 2.

### 4.6.4 Notes on saving and recovering data

- When you recover the saved files to a different machine environment, that environment must have the same or a newer version of CPA as the one installed on the environment in which the files to be recovered were saved. The names of the installation folder and data folder must also have the same environment.
- Recover all the saved files. Do not try to recover only some of the saved files.

# 5

## Business Operations Planning with CPA

This chapter, which describes business operations planning with CPA, is intended for those responsible for planning the business operations to be executed with the CPA system as well as for those responsible for building an environment for executing the business operations.

## 5.1 Building an item creation environment

---

You can create a custom item by using the Job Design View or the Item Design View. To use the Item Design View, however, you have to first set up an item creation environment. For details on how to set up an item creation environment, see [2.3.6 Creating a custom item](#).

After you set up an item creation environment, **Item Design View** is added to the **Start** menu. The rest of this chapter focuses on how to create a custom item mainly by using the Item Design View.

### Important

You can create action flow and calendar items in both the Job Design View and the Item Design View. However, an item created in the Item Design View cannot be edited in the Job Design View. In addition, an item created in the Job Design View can be edited in the Item Design View, but can no longer be edited in the Job Design View after the edit.

## 5.2 Common settings for items

---

This section describes the common settings for items.

### Item name

The name of the item you are creating. The item name you specify here will show up in the **Select item** area in both the Job Design View and the Item Design View, and will be displayed as the label of the tab in the Home window that shows information regarding the definition of the item.

Specify a name that clearly conveys what the item does while it is in use.

### Details

Specify the details of the item you are creating.

The details you specify here will be displayed in the list of items in the Home window or when the item is selected in the Job Design View.

Specify details that clearly convey what the item does as well as what to specify for the item.

### Set icon

This setting is available only when you create an action or an action flow item.

Specify the file name of the icon to be displayed in the **Select item** area of both the Job Design View and the Item Design View or in the flow map of the action flow definition window. If you do not specify this setting, it is assumed that the default icon provided by CPA is set.

As a file name, you have to specify either the relative path to the icon file in the current folder *data-folder\work* or the absolute path to the icon file.

### Apply this item to the item list in the Job Define View

This setting is available only when you create an action or an action flow item.

You can specify whether to display the item in the **Select item** area of the Job Design View.

When the item you are creating is to be used only by item designers (for example, when it is used only in action flows created from the Item Design View), clear this check box. Note that, even when you clear this check box, you can still execute the action flows or jobs that use this item.



## 5.3 Creating action items

---

With CPA, you can create original action items. Just like standard action items, the original action items that you create can be used as units of a job or an action flow.

Furthermore, by creating an action item appropriate to the purpose of the process by specifying the commands to be executed and their parameters, you can plan business operations without being bothered by the need to specify the necessary commands and their parameters.

For details on the standard action items, see [8.1 List of standard CPA items](#).

You can create action items by opening the **Action** dialog box in the Item Design View. For details on the **Action** dialog box, see [7.2.1 Action dialog box](#).

### 5.3.1 Basic settings for action items

#### (1) Command

Specify the file name of the command to execute. The file name can be specified as a relative path if it is on the `PATH` environment variable of the user who starts the job execution service. If it cannot be resolved by the `PATH` environment variable, specify it with an absolute path.

If specifying a file whose extension is other than `.exe` or `.bat`, you need to associate the file extension with an appropriate application.<sup># 1</sup>

To execute a command prompt (`cmd.exe`) command, specify `cmd.exe` for the execution command, and `/c command` parameter for the parameter. For details, see the specifications of `cmd.exe`.<sup># 2</sup>

#1

Use the following procedure to associate the file extension with an application.

1. Register the application to associate with using the `f t y p e` command.
2. Associate the extension with the application registered in step 1 using the `a s s o c` command.
3. Restart the OS.

For details on the settings of the `f t y p e` and `a s s o c` commands, see Windows Help.

#2

Only a local file name can be specified as the name of the execution command file.

#### (2) Parameter

Specify the parameter you want to specify for the command to be executed. You can also proceed to specify an inherited information variable, a profile variable, and an input item variable. For details on input item variables, see [5.7 Creating an input item](#).

#### (3) Execution type

As the execution environment for the command that you have specified for an action, select either **Do not show the window** or **Show the window**.

- **Do not show the window**

When you select this option, the command is executed in an environment that is equivalent to the the standard item *Command execution*. The command is executed by the account that started the job execution service (by default, the local system account), with no session allocated to it. The command can be executed without any user logged on to the Windows system. However, when the command is of an interactive nature, it can get stuck in the running state because there is no one to operate the window. (Use the Task Manager or a similar means to check if this is happening.)

- **Show the window**

When you select this option, the command is executed in an environment that is equivalent to the standard item *Command execution in window*. The current session of the user logged on to the Windows system is allocated for command execution, and the command is executed by the account belonging to the allocated user session. While the use of this option requires that a user be logged on to the Windows system during the execution of the command, it enables not only the execution of a command that displays a window on the display device assigned to a session of the logged-on user but also access to resources such as a shared drive to which the logged-on user is granted access.

Furthermore, selecting this option allows you to specify **Wait for end** and **Exclusive option**.

The command inside an action is executed in the execution environment described in the following table.

Table 5–1: Command execution environment

No.	Setting	Description
1	User account	<ul style="list-style-type: none"> <li>• If you select <b>Do not show the window</b> as the execution type: The account that started the job execution service (By default, the local system account)</li> <li>• If you select <b>Show the window</b> as the execution type: Windows user logged on to the session</li> <li>• If you specify a virtual user name: Windows user associated with the virtual user name</li> </ul>
2	Environment variables	<ul style="list-style-type: none"> <li>• CPADATAPATH=<i>data-path</i></li> <li>• CPAINSTALLPATH=<i>install-path</i></li> <li>• CPAJDS=<i>job-name</i></li> <li>• CPAJCS=<i>job-execution-ID</i></li> <li>• Environment variables of the user who starts the job execution service</li> <li>• If you specify a virtual user name: Environment variable for the Windows user associated with the virtual user name</li> </ul> <p>Do not use the environment variables starting with CPA, such as CPAxxx (xxx is any string), regardless of uppercase or lowercase letters. This is because such variables are reserved by the system.</p>
3	Current path	Work folder ( <i>data-path</i> \work)
4	Priority class of a process	NORMAL_PRIORITY_CLASS
5	User profile	<p>If you specify a virtual user name: User profile of the Windows user associated with the virtual user name</p>

The behavior of the command varies depending on whether it is defined inside an action and executed from CPA or it is executed in the command prompt. For details on the different behaviors of commands, see [A.3 Different behaviors in different command execution environments](#).

## (4) Wait for end

Select **Wait for the window showing process to terminate** or **Does not wait for the window showing process to terminate** to indicate whether to wait for the executed command to end.

## (5) Exclusive option

To prevent all the specified commands from being simultaneously executed (to ensure that one command is executed at a time), select the **Do not execute concurrently among selected actions** check box.

Select this check box when there is a window-display command that can cause a problem when executed concurrently with other commands.

## (6) Operation when terminates

As the method of determining the state of the action after the execution of the command, select **Abnormal end if the return code is other than 0**, **Warning end if the return code is other than 0**, or **Always normal end**.

Based on the return value of the executed command, define the state of the action and regulate the execution of the subsequent unit in the action flow. For details on action flows, see [5.4 Creating an action flow item](#).

The following table describes the behavior following the termination of the command.

Table 5–2: Operation when terminates

No.	Value	Description
1	<b>Abnormal end if the return code is other than 0</b>	<i>Ended normally</i> when the return value is 0, or <i>Ended abnormally</i> when the return value is other than 0.
2	<b>Warning end if the return code is other than 0</b>	<i>Ended normally</i> when the return value is 0, or <i>Ended with warning</i> when the return value is other than 0.
3	<b>Always normal end</b>	<i>Ended normally</i> regardless of the return value.

## 5.3.2 Option setting for an action item

### (1) Execution timeout period

Specify the timeout period for actions being executed. The valid values are 0 to 1,440 minutes. If omitted or set to 0, actions are not terminated.

### (2) Virtual user name

Specify the name of the virtual user who executes items with half-width alphanumeric characters from 1 to 64 bytes or a variable. If this is omitted, the item is executed with the account for the CPA job execution service.

If **Show the window** is selected as the command execution type, the virtual user name cannot be specified.

## 5.3.3 Setting an input item in an action item

For details on how to set an input item, see [5.7 Creating an input item](#).

## 5.4 Creating an action flow item

---

An action flow item, which consists of several different actions, executes as a single process.

In designing an action flow item, the following functions are available to ensure convenient use.

- **Action flow execution function**  
This function controls the order in which actions are executed. It also allows you to have the action flow branch into two or more actions based on the *Ended* state of the previous action. For details on *Ended* states, see [1.1.4 State management for jobs and actions](#).
- **Item execution condition**  
By specifying this condition, you can control whether to execute an item based on the return value of the preceding unit.
- **Inherited information**  
This information, which reflects the output of the preceding unit, can be used as an input item of the item.

### 5.4.1 Action flow execution function

This function controls the execution order of actions, event monitors, and action flows according to the settings of action flows.

#### (1) Execution order control function

This function executes actions, event monitors, and action flows in the order in which relation lines are connected. When the unit at the starting end of a relation line (which is indicated by the arrows in the figure below) ends either normally or with a warning, the unit at the destination end of the relation line is executed. There are two types of relation lines: one that takes effect when the preceding unit ends normally and one that takes effect when the preceding unit ends with a warning. If the preceding unit ends abnormally, the subsequent unit does not run.

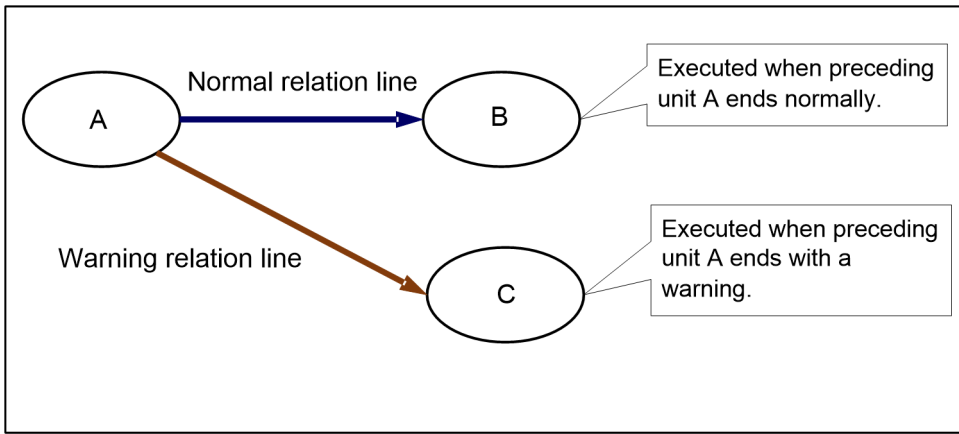
Each unit defined in an action flow must have a unique name.

#### (2) Termination status and relation lines

The *Ended* states of a unit can largely be categorized into *Ended normally*, *Ended with warning*, and *Ended abnormally*. Relation lines, on the other hand, are available in two types: normal relation lines and warning relation lines.

The following figure shows the possible *Ended* states of the preceding unit and the relation lines extending from it.

Figure 5–1: Possible Ended states of the preceding unit and the relation lines extending from it



The subsequent unit at the end of a normal relation line is executed when the *Ended* state of the preceding unit is *Ended normally*. In the example shown in Figure 5-1, unit B is executed when unit A ends normally.

The subsequent unit at the end of a warning relation line is executed when the *Ended* state of the preceding unit is *Ended with warning*. In the example shown in Figure 5-1, unit C is executed when unit A ends with warning.

Note that when the type of a relation line does not match the *Ended* state of the preceding unit, the unit at the destination end of that relation line is not executed. For the example in Figure 5-1, if unit A ends normally, unit C is not executed and enters a special state that is not the normal or abnormal state, namely, the *Condition not satisfied* state.

The behavior of a subsequent unit is determined by whether the type of the relation line matches the state of the preceding unit, whether the preceding unit has the *Condition not satisfied* state, and the execution conditions specified for the unit.

The following table describes the execution rules of the subsequent unit based on the above conditions.

Table 5–3: Rules concerning relation lines and the execution of units

No.	State(s) of the preceding unit(s)	Combinations of preceding units and relation lines	Behavior of the subsequent unit	See
1	At least one preceding unit is in the <i>Ended abnormally</i> state.	One or more of the preceding units ended abnormally.	The subsequent unit stops in the <i>Wait for start</i> state.	<a href="#">Figure 5-2</a>
2	None of the preceding units are in the <i>Ended abnormally</i> state or the <i>Condition not satisfied</i> state. (All of them are in either the <i>Ended normally</i> or <i>Ended with warning</i> state.)	The states of relation lines and the unit execution conditions all match those of its preceding unit.	The subsequent unit is executed.	<a href="#">Figure 5-3</a>
3		The state of at least one relation line does not match the state of its preceding unit, or the unit execution conditions are not met.	The subsequent unit is not executed (enters the <i>Condition not satisfied</i> state or the <i>Ended normally (unexec)</i> state).	
4	Although no preceding units are in the <i>Ended abnormally</i> state, one or more preceding units are in the <i>Condition not satisfied</i> state. (The rest of them are in either the <i>Ended normally</i> or <i>Ended with warning</i> state.)	The states of relation lines and the unit execution conditions match those of its preceding unit, except when the preceding unit is in the <i>Condition not satisfied</i> state.	The subsequent unit is executed.	<a href="#">Figure 5-4</a>
5		The state of at least one relation line does not match the state of its preceding unit, except when the preceding unit is in the <i>Condition not satisfied</i> state, or the unit execution conditions are not met.	The subsequent unit is not executed (enters the <i>Condition not satisfied</i> state or the <i>Ended normally (unexec)</i> state).	

No.	State(s) of the preceding unit(s)	Combinations of preceding units and relation lines	Behavior of the subsequent unit	See
6	Every preceding unit is in the <i>Condition not satisfied</i> state.	Every preceding unit is in the <i>Condition not satisfied</i> state.	The subsequent unit is not executed (enters the <i>Condition not satisfied</i> state).	

Figure 5–2: Behavior of the subsequent unit when the preceding unit ends abnormally

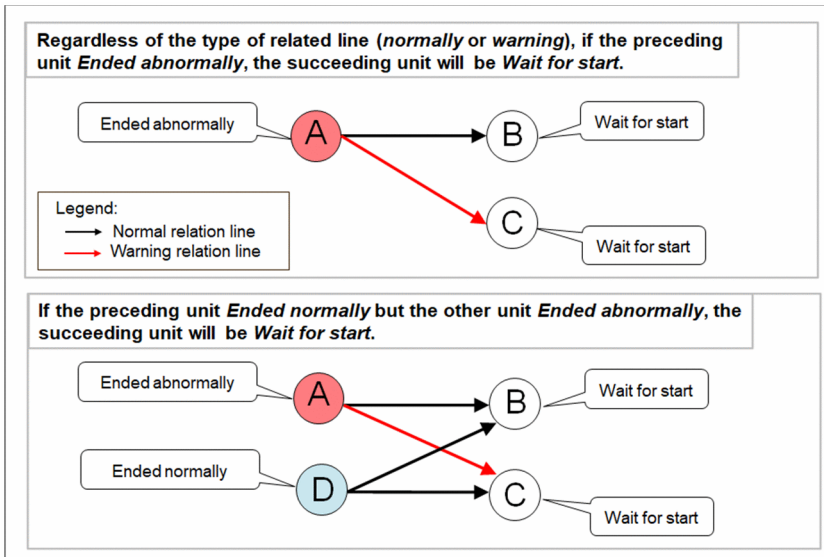


Figure 5–3: Behavior of the subsequent unit when the preceding unit ends normally or with warning

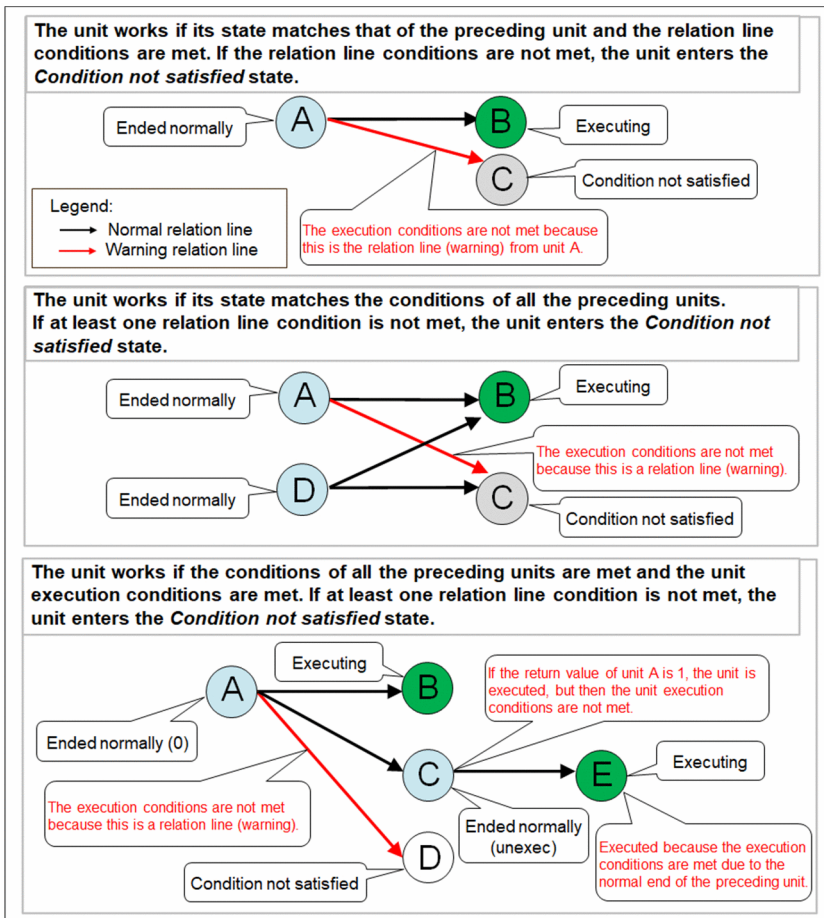
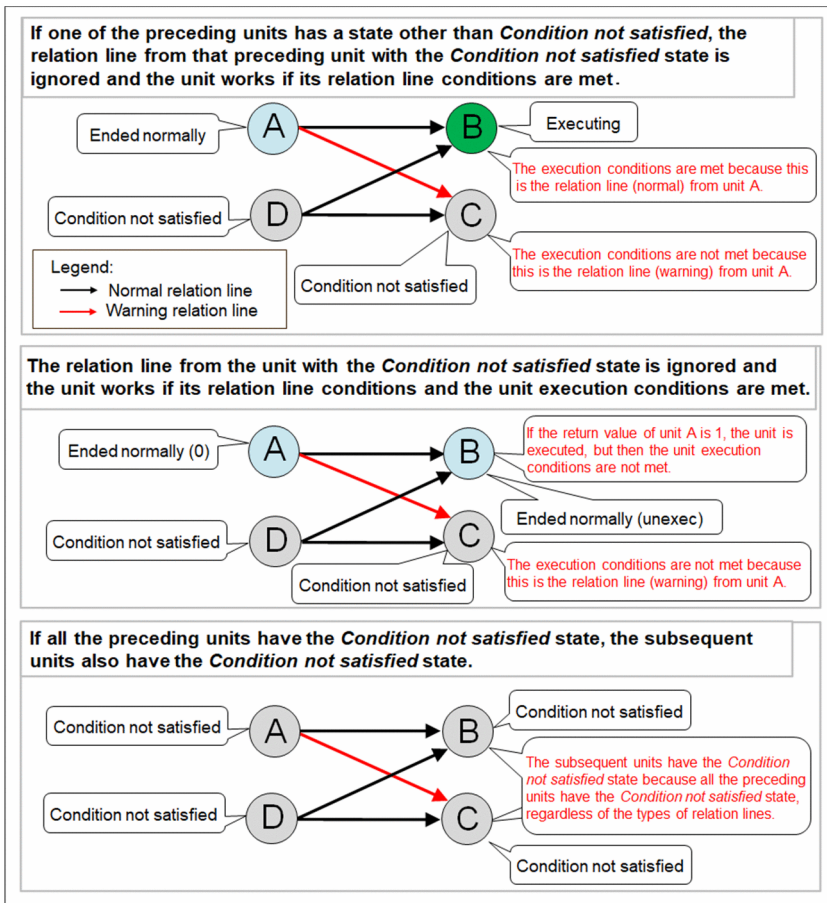


Figure 5–4: Behavior of the subsequent unit when a preceding unit has the Condition not satisfied state



The following figures show how the action flow works when it contains relation lines pertaining to the *Ended normally* and *Ended with warning* states.

Figure 5–5: Example of how the action flow works when unit A ends normally

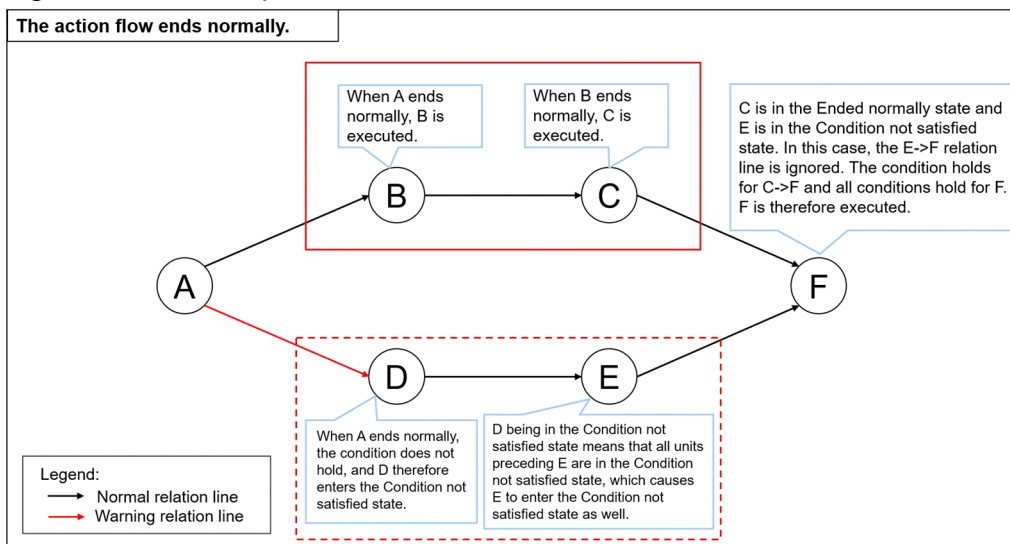
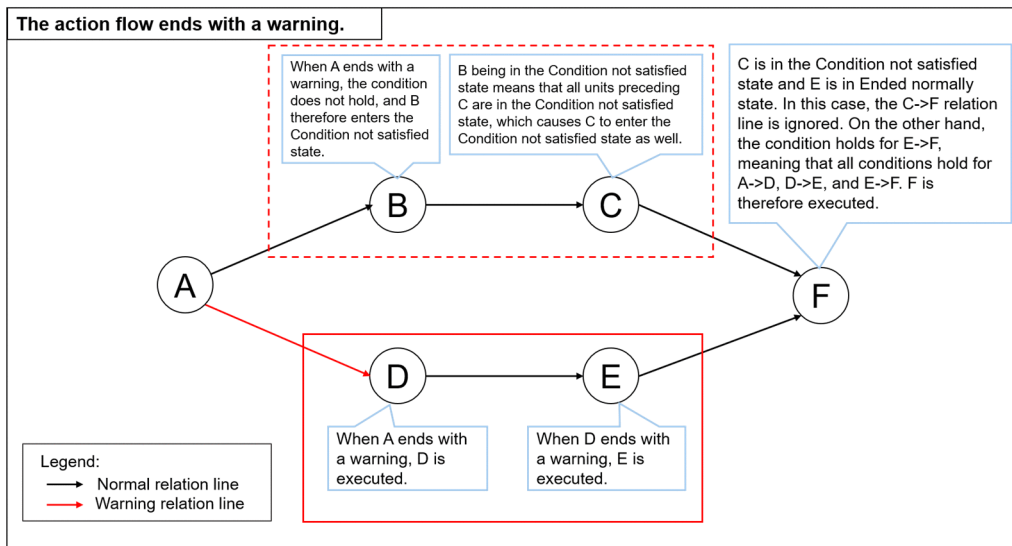




Figure 5–6: Example of how the action flow works when unit A ends with warning



Furthermore, the *Ended* state of an action flow is determined by the states of the actions it holds as shown in the following table.

Table 5–4: Ended state of the action flow

No.	States of the actions inside the action flow	Ended state of the action flow
1	At least one action ended abnormally.	Ended abnormally
2	No action ended abnormally but at least one action ended with a warning.	Ended with warning
3	No action ended abnormally or with a warning.	Ended normally

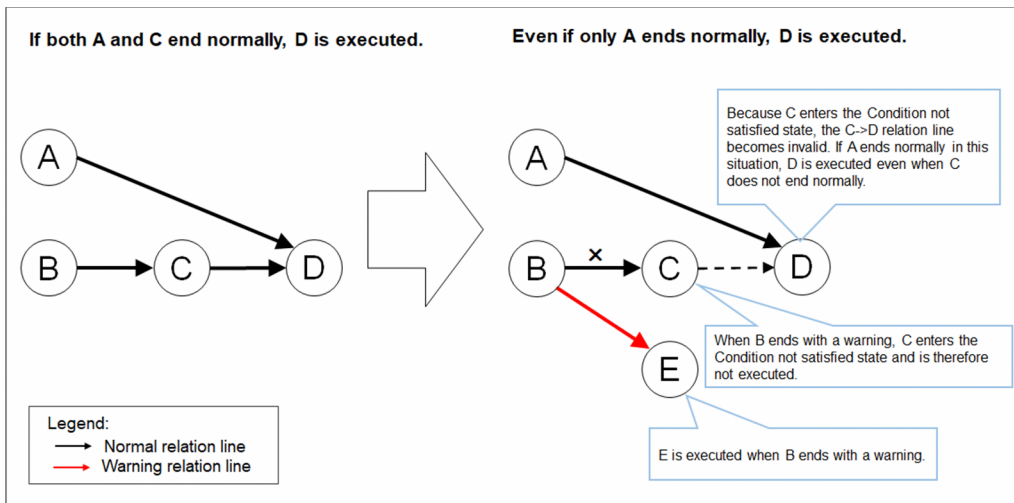
### (3) Notes on using a warning relation line

When you use a warning relation line, the subsequent unit as the destination of the normal relation line will have the *Condition not satisfied* state when the preceding unit ends with a warning. Note that it affects the behavior of the subsequent unit.

The following figure shows an example of how the use of a warning relation line gives rise to an unwanted behavior when the preceding unit does not end normally.



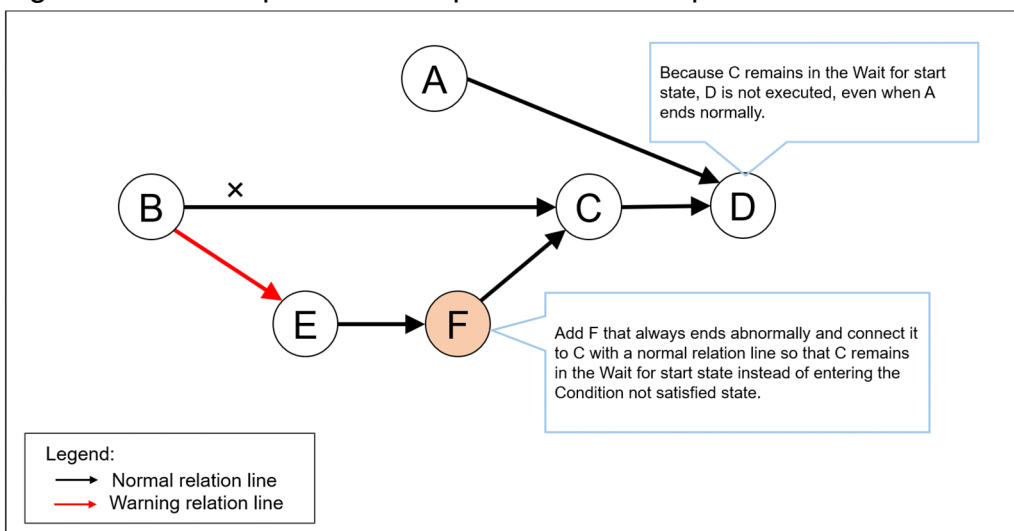
Figure 5–7: Example of using a warning relation line



In this case, you can avoid the behavior to add a process, which is end with warning explicitly, and which is connecting to the original execution order with normal relation line.

The following figure shows an example of how to prevent.

Figure 5–8: Example of how to prevent a subsequent unit from behaving in a certain manner



## 5.4.2 Item execution condition

When you edit an action flow from the Item Design View, you can specify an item execution condition for a unit.

Specify a condition that consists of a statement to the effect that the return value of the preceding unit is either equal to or not equal to the specified value. The item specified in the unit is executed when the condition is satisfied, and skipped when the condition is not satisfied. If the specified preceding unit does not exist, has not ended yet, was forcibly terminated, or ended when the timeout period was reached, the conditions are considered not to be met. When the execution of the item specified in the unit is skipped, the state of that unit would be *Ended normally (unexec)*.

## (1) Inherited information

When an action is executed, information about the execution is stored. This information is called *inherited information*. The inherited information can be referred to using the methods described in *Table 5-5 List of inherited information of actions*.

When you want to refer to the results of the unit in an action flow, specify the user-defined unit name in a method.

### Example

When you refer to a standard output message output by command execution in the action flow:

```
{JDS:unit-name.GetStdOut() }
```

Table 5–5: List of inherited information of actions

No.	Action	Method	Description
1	Command execution	<code>unit-name.GetStdOut()</code>	Returns a message output to the standard output. <sup>#</sup>
2		<code>unit-name.GetStdErr()</code>	Returns a message output to the standard error output. <sup>#</sup>
3		<code>unit-name.GetStdOutFile()</code>	Returns the name of the file that stores the output of messages to the standard output.
4		<code>unit-name.GetStdErrFile()</code>	Returns the name of the file that stores the output of messages to the standard error output.
5		<code>unit-name.GetExitCode()</code>	Returns the return code of the command when the monitoring condition was met.

#

The output from the `GetStdOut` or `GetStdErr` method includes line breaks, such as `\n`, `CR+LF`, and `LF`.

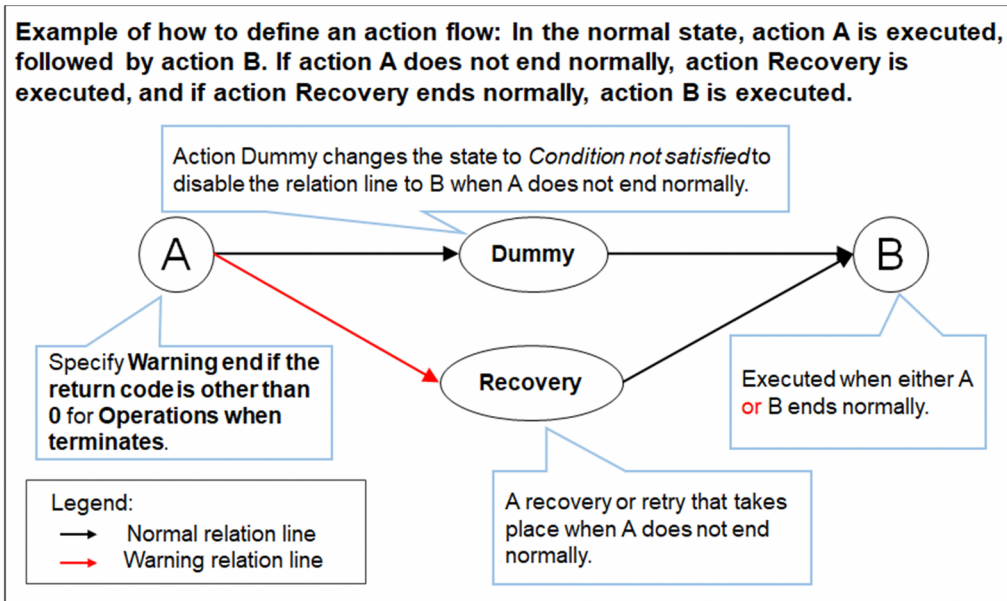
Thus, if you specify the `GetStdOut` or `GetStdErr` method as a parameter of an action item, the command line stops at the linefeed. If you are using Windows and want to output messages without linefeeds, we recommend using the `cpaecho` command. For details on the `cpaecho` command, see *cpaecho* in *9. Commands*. If output cannot be inherited correctly, make sure that no linefeed is included in the command output.

## 5.4.3 Examples of how to create an action flow

### (1) An action flow that allows for a recovery or retry

The following figure shows an action flow that allows for a recovery or retry.

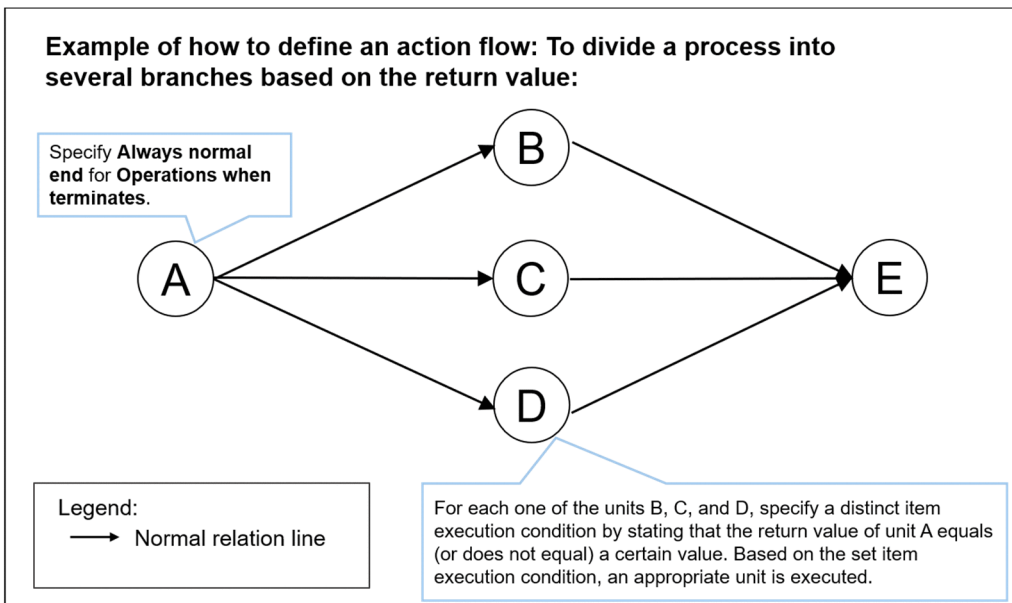
Figure 5–9: Example of a recovery that takes place upon the occurrence of an error



## (2) Action flow in which a process is divided into several branches based on the return value

The following figure shows an action flow in which a process is divided into several branches based on the return value.

Figure 5–10: Example of dividing a process into several branches based on the return value



## 5.5 Event monitor functions

These functions monitor event occurrence.

If these functions are used as execution conditions of the job executed on event, they execute the job whenever an event occurs. If the functions are used in an action flow, they start monitoring when an event monitor starts and end when a given event is detected. This way, a job can wait for event occurrence.

### 5.5.1 List of event monitor functions

Table 5–6: List of event monitor functions

No.	Function	Monitored event	Description
1	Monitor file event	Create	Monitors whether a file with a specified name was created and whether a file has been newly created. <sup># 1</sup>
2		Delete	Monitors whether a file with a specified name was deleted and whether the file has been just deleted. <sup># 1</sup>
3		Change size	Monitors whether the size of a file with a specified name is changed. <sup># 1</sup>
4		Change final write time	Monitors whether the update time of a file with a specified name is changed. <sup># 1</sup>
5	Monitor any event	The execution results of a command	Executes a command every minute, and monitors whether the result is the specified return value. The command name must be an absolute path.
6	Monitor the linkage from JP1AJS3 <sup>#2</sup>	Job execution-start notification from AJS3	Monitors whether the custom job for CPA linkage defined in AJS3 - Manager starts.
7	Event monitoring for receiving mail	Sender	Monitors reception of the emails from certain email addresses.
8		Subject	Monitors reception of the emails with certain strings in a subject.
9		Message text	Monitors reception of the emails with certain strings in message text.
10	Monitor Windows event <sup>#3</sup>	Log on	Monitors for log-ons by Windows users.
11		Log off	Monitors for log-offs by Windows users.
12		Screen lock	Monitors for screen locks by Windows users.
13		Screen lock release	Monitors for screen unlocks by Windows users.

#1

The file name of a local file with the absolute path can be specified as the name of the monitored file. Note that when you use the execution user setting function, the file name of a file on the network can be specified because the Windows user associated with the virtual user name can have access to network files. For details, see [4.4.1\(3\) Using files on the network](#).

#2

This function is used for execution conditions.

To enable this function, setting up AJS3 is required too. For details on how to set up, see [2.3.3 Configurations for the AJS3 linkage function](#), and [3.3 Configurations](#).

#3

To use this function, you need to start the required services. For details, see [2.3.5 Using the Windows event monitor function](#).

## 5.5.2 Inherited information

When an event monitor detects an event, it stores the information obtained. This information is called *inherited information*. The inherited information can be referred to using the methods described in [Table 5-7 List of inherited information of event monitors](#).

When you want to refer to the results of execution conditions for the job executed on event, use the *unit name used in execution conditions* in the following table to specify a unit name in a method. When you want to refer to the results of the unit in an action flow, specify the user-defined unit name in a method.

### Example

- To refer to the name of the *file containing file names* that is detected by the file monitor used as execution conditions:

```
{JDS:@Event.Monitor file.GetFlInfoList() }
```

- To refer to the *sender of the email* that is detected by the Monitoring mail reception item in an action flow:

```
{JDS:unit-name.GetMailFrom() }
```

Table 5–7: List of inherited information of event monitors

No.	Event monitor	Unit name used in execution conditions	Method	Description
1	File monitor	Monitor file	<code>unit-name.GetFlInfoList()</code>	Obtains the name of the file <sup>#1</sup> that contains the full name, size, and update time of the monitored files for which the file monitor detected an event.
2	Monitor any event	Monitor any event	<code>unit-name.GetStdOutFile()</code>	Returns the name of the file that stores output of messages to the standard output when the monitoring conditions were met.
3			<code>unit-name.GetStdErrFile()</code>	Returns the name of the file that stores output of messages to the standard error output when the monitoring conditions were met.
4			<code>unit-name.GetStdOut()</code>	Returns the message output to the standard output when the monitoring condition was met. <sup>#2</sup>
5			<code>unit-name.GetStdErr()</code>	Returns the message output to the standard error output when the monitoring condition was met. <sup>#2</sup>
6			<code>unit-name.GetExitCode()</code>	Returns the return code of the command when the monitoring condition was met.
7	Monitoring mail reception	Monitor mail	<code>unit-name.GetMailBodyFilePath()</code>	Obtains the path of the file that stores email message text.
8			<code>unit-name.GetMailSubject()</code>	Obtains the subject of the email.
9			<code>unit-name.GetMailFrom()</code>	Obtains the sender of the email.
10			<code>unit-name.GetMailReplyTo()</code>	Obtains the email address to reply to.

No.	Event monitor	Unit name used in execution conditions	Method	Description
11			<code>unit-name.GetMailTo()</code>	Obtains the email address in the To field.
12			<code>unit-name.GetMailCc()</code>	Obtains the email address in the CC field.
13			<code>unit-name.GetMailAttachFileList()</code>	Obtains the name of the file <sup>#3</sup> that contains a list of paths of attached files.
14	Windows event monitor	Monitor Windows event	<code>unit-name.GetStdOut()</code>	Obtains the name of the Windows user who triggered the detected Windows event.

#1

The file that contains the full name, size, and update time of the monitored files is in CSV format.

Information is output to this file in the following format:

```
"condition", "full-name-of-detected-file", "file-size (bytes)", "update-time-of-file (yyyy/mm/dd hh:mm:ss)"
```

Depending on the detected event, one of the following types is output as a *condition*:

- c: The file was created.
- d: The file was deleted.
- m: The update time of the file was updated.
- s: The file size was updated.

#2

The output from the `GetStdOut` or `GstStdErr` method includes line breaks, such as `\n`, `CR+LF`, and `LF`. Thus, if you specify the `GetStdOut` or `GetStdErr` method as a parameter of an action item, the command line stops at the linefeed. If you are using Windows and want to output messages without linefeeds, we recommend using the `cpaecho` command. For details on the `cpaecho` command, see [cpaecho](#) in [9. Commands](#). If output cannot be inherited correctly, make sure that no linefeed is included in the command output.

#3

Full paths to all the attached files are output to the file, with line feeds separating them.

## 5.6 Creating schedule items

By creating a schedule item, you can define a scheduled date and time for job execution.

### 5.6.1 Schedule functions

CPA uses the schedule functions to start the job at the scheduled dates and times by calculating its scheduled execution dates and times. The scheduled dates and times are determined based on execution date, execution time, and calendar, each of which can be defined as schedule items, namely, an execution date item, an execution time item, and a calendar item. You can use the defined schedule items during the registration of a job in the Job Design View.

#### (1) Execution date items

This item defines the dates on which to execute a job.

You can create an execution date item by using the **Exec. date** dialog box, which you can open from the Item Design View. For details on the **Exec. date** dialog box, see [7.2.4 Exec. date dialog box](#).

The following table describes the settings you can specify for an execution date item.

Table 5–8: Settings of execution date items

No.	Type	Setting	Setting type	Note
1	Set start day <sup>#1#2</sup>	Year, month, and date Day <sup>#3</sup> Open day <sup>#3</sup> Closed day <sup>#3</sup>	yyyy/mm/dd <sup>#4</sup> Number X Day Number Y Open day Number Z Closed day	The job is executed on the date specified here. Specify <b>Number X Day</b> to execute the job at a weekly, monthly, or yearly cycle, <b>Number Y Open day</b> to execute the job every open day specified, and <b>Number Z Closed day</b> to execute the job every closed day specified. Specify a number denoting how many days after the beginning of the specified year and month the job begins.
2	Execution cycle	No cycle	No cycle (One day only)	When this option is selected, the job is executed only on the date specified as the start date.
3		Everyday	Execute everyday	When this option is selected, the job is executed everyday starting from the date specified as the start date.
4		Every operation day	Execute in every operation day	When this option is selected, the job is executed every open day specified in the calendar, starting from the date specified as the start date.
5		Every closed day	Execute in every close day	When this option is selected, the job is executed every closed day specified in the calendar, starting from the date specified as the start date.
6		Every week	Execute in every X day	When this option is selected, the job is executed every Monday, Tuesday,

No.	Type	Setting	Setting type	Note
				Wednesday, Thursday, Friday, Saturday, or Sunday, starting from the date specified as the start date.
7		Every month	Execute the job on a specific date, a specific day of the week, the last day, the specified open day, or the specified closed day of each month.	When this option is selected, the job is executed every month starting from the date specified as the start date. When the job is to be executed on the specified open day or closed day of each month, the open day or closed day on which to execute the job is determined based on the calendar that defines the open days or closed days.
8		Every year	Execute the job on a specific date, a specific day of the week, <i>X</i> days prior to month-end, or Number <i>Y</i> Open day/Closed day of each year.	When this option is selected, the job is executed every year starting from the date specified as the start date. When the job is to be executed on the specified open day or closed day of each year, the open day or closed day on which to execute the job is determined based on the calendar that defines the open days or closed days.
9	Transfer setting	Not execute in close days	Not execute in close days	When any of the job execution dates determined based on the start date and the cycle happens to fall on a closed day, the job will not be executed on those days.
10		Can execute even in closed days	Can execute even in closed days	Even when some of the execution dates determined based on the start date and the cycle happen to fall on a closed day, the job will still be executed on those days.
11		Transfer in to a open day before the scheduled execute day and execute	One open day before a job execution date becomes the substitute job execution date.	In a case where some of the execution dates determined based on the start date and the cycle happen to fall on a closed day, one open day before a job execution date that falls on a closed day becomes the substitute job execution date.
12		Transfer in to a open day after the scheduled execute day and execute	One open day after a job execution date becomes the substitute job execution date.	In a case where some of the execution dates determined based on the start date and the cycle happen to fall on a closed day, one open day after a job execution date that falls on a closed day becomes the substitute job execution date.

#1

If this setting is omitted, the value of the registration date is used.

#2

The start date you specify here is automatically set as the start date of the valid time frame for job execution which you can set when you register a job from the Job Design View. You cannot change the automatically set start date of the valid time frame for job execution from the Job Design View. When you do not specify a start date here, the scheduled dates and times for job execution are determined based on the valid time frame for job execution specified during the registration of the job.



#3

Specify a date that exists for the start date. If a date that does not exist is specified as the start date of an execution date item and the item is used to register a job, the KNAO4025-E and KNAO4019-E errors occur, causing the scheduled execution date and time not to be determined.

#4

When English is selected as the language setting, is automatically converted to *mm/dd/yyyy*.  
For details on the language settings, see [2.6 Language setting](#).

## (2) Execution time items

This item defines the time at which to execute a job.

You can create an execution time item by using the **Exec. time** dialog box, which you can open from the Item Design View. For details on the **Exec. time** dialog box, see [7.2.5 Exec. time dialog box](#).

The following table describes the settings you can specify for an execution time item.

Table 5–9: Settings of execution time items

No.	Type	Setting	Setting type	Note
1	Set execute time	Scheduled start time	<i>hh:mm</i>	The job is executed at the time specified here. You can specify a value in the range from 00:00 to 23:59.
2		Shift time	Execute from start time of ope.time period <i>hh:mm</i> later than the scheduled job start time	The job is executed at the start time of the operation time period defined in the calendar.
3		Time across day	Previous day / Next day <i>hh:mm</i>	The job is executed at the set time but on the day before or after the determined job execution date.
4	Repetition setting	Execute in every fixed time after started execution	<i>hh:mm</i>	This setting specifies the interval for repeatedly executing the job that starts running at the time specified as the start time. The job is repeated during the operation time period. If a calendar is omitted, the job is repeated during a period from 00:00 to 23:59 on the job execution date. If the set interval is too short, the job does not end in time for the next interval to start, which can cause a scheduled run to be skipped or restrict the number of actions that can be executed at a time. For this reason, we recommend that you set an adequate interval so that it does not cause a problem during operation.

## (3) Calendar items

This item specifies open days, closed days, and an operation time period.

You can create a calendar item by using the **Calendar** dialog box, which you can open from the Item Design View. For details on the **Calendar** dialog box, see [7.2.3 Calendar dialog box](#).

You can also use the Job Design View to define or edit calendar items. Note, however, that a calendar item defined in the Item Design View cannot be edited in the Job Design View.

The following table describes the settings you can specify for a calendar item.

**Table 5–10: Settings of calendar items**

No.	Type	Setting	Setting type	Note
1	Open days and closed days	Week	<b>Mo, Tu, We, Th, Fr, Sa, Su</b>	This setting allows you to specify the day of the week as open days or closed days. The information you specify here can be used to control both the start date and the cycle of job execution.
2		Certain date	Specified dates on the calendar	This setting allows you to specify certain dates as open days or closed days. The information you specify here can be used to control both the start date and the cycle of job execution.
3	Operation time period	Start time	<i>hh:mm</i> of the previous, current, or next day	This setting allows you to specify the start time of the operation time period during which to execute a job in the range from 00:00 to 23:59 of the previous, current, or next day. The information you specify here can be used to control the start time of job execution.
		End time	<i>hh:mm</i> of the previous, current day, or next day	This setting allows you to specify the end time of the operation time period during which to execute a job in the range from 00:00 to 23:59 of the previous, current, or next day. The information you specify here can be used to control the range of job execution times determined based on the set cycle of job execution.

## 5.7 Creating an input item

You can set an input item in an action item or action flow item to accept user input. The user input can be used as a variable, which in turn can be set as a parameter of the command set inside an item.

For details on the Item Design View, see [7. Item Design View](#).

### 5.7.1 Text box input item

You can create a text box in an item, which allows you to use a user-input string as the parameter of a command or the like.

#### Example: Sending user input to standard output

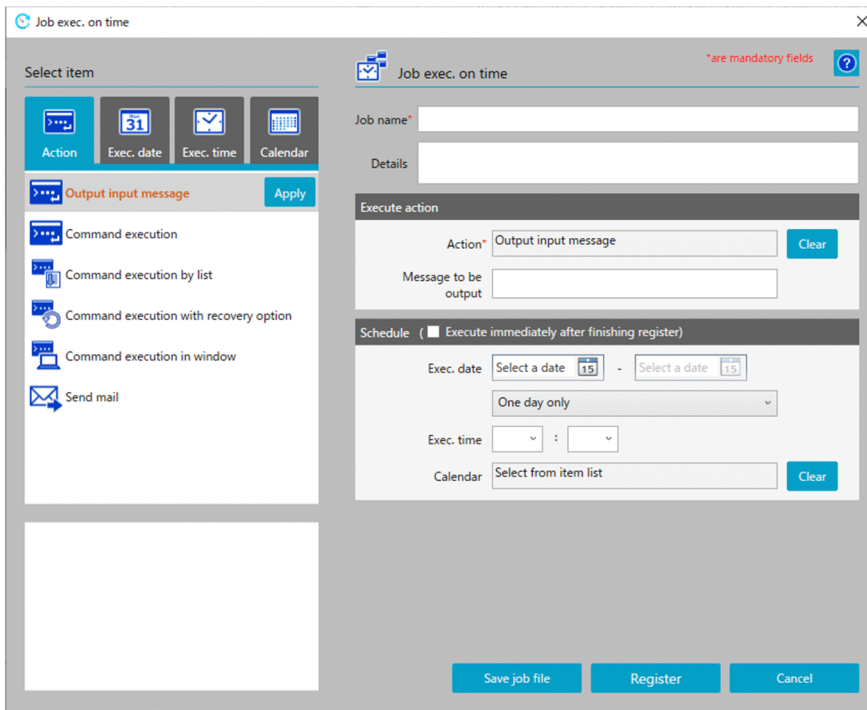
Open the **Action** dialog box from the Item Design View, specify the information described in the following table in an item, and then register the item.

Table 5–11: Information specified for the input item "Text box"

No.	Specifics of where each piece of information is defined			Information to be specified
	Main window	Where to transition to	Location where the information is specified	
1	Item Design View +-- <b>Action</b> dialog box	Basic settings	Command	cpaecho
2			Parameter	<code>\${JDS:msg}</code>
3			Execution type	<b>Do not show the window</b>
4		<b>Input item settings</b> +-- <b>Define input items</b> dialog box	Variable name	msg
5			Label	Message to be output
6			Input type	Text box

The following figure shows what the dialog box looks like when the created item is to be used in a job.

Figure 5–11: An item having a text box input item



The string entered in the **Message to be output** text box is sent to standard output.

## 5.7.2 List box input item

You can create a list box in an item, which allows you to use the value set in the user-selected entry as the parameter of a command or the like.

### Example: Sending a message appropriate to the selected information to standard output

Open the **Action** dialog box from the Item Design View, specify the information described in the following table in an item, and then register the item.

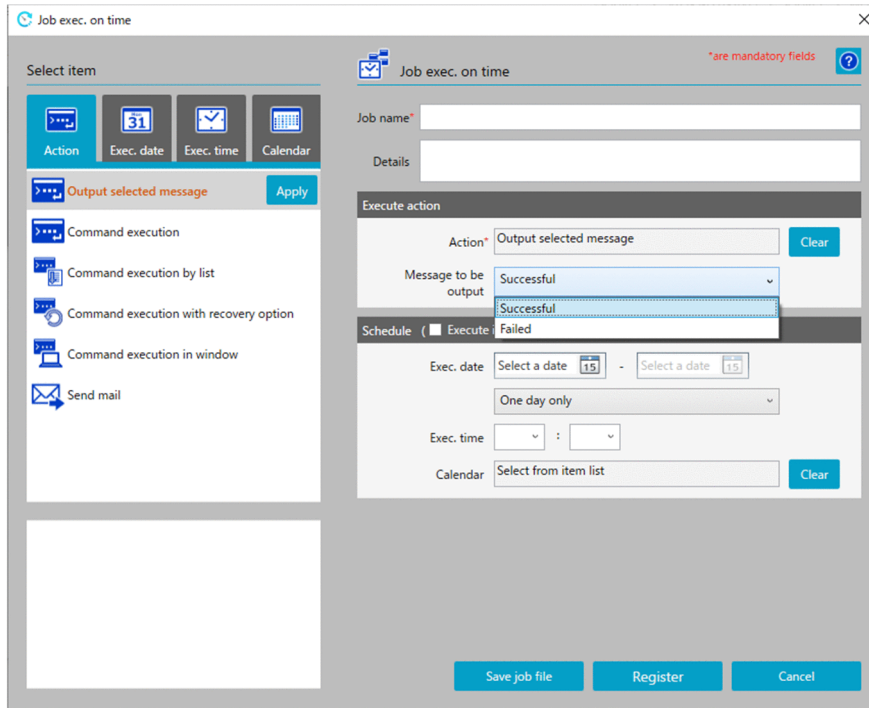
Table 5–12: Information specified for the input item "List box"

No.	Specifics of where each piece of information is defined			Information to be specified
	Main window	Where to transition to	Location where the information is specified	
1	Item Design View +-- <b>Action</b> dialog box	Basic settings	Command	cpaecho
2			Parameter	\${JDS:msg}
3			Execution type	<b>Do not show the window</b>
4		<b>Input item settings</b> +-- <b>Define input items</b> dialog box	Variable name	msg
5			Label	Message to be output
6			Input type	List box
7				List label: Successful Value: The process ended successfully.

No.	Specifics of where each piece of information is defined			Information to be specified
	Main window	Where to transition to	Location where the information is specified	
8				List label: Failed Value: The process ended in failure.

The following figure shows what the dialog box looks like when the created item is to be used in a job.

Figure 5–12: An item having a list box input item



When **Successful** is selected in the **Message to be output** list box, `The process ended successfully.` is sent to standard output, whereas when **Failed** is selected in the list box, `The process ended in failure.` is sent to standard output.

### 5.7.3 Check box input item

You can create a check box in an item, which allows you to use the value that takes effect when the user selects the created check box as the parameter of a command or the like.

#### Example: Outputting a message to standard output when the check box is selected

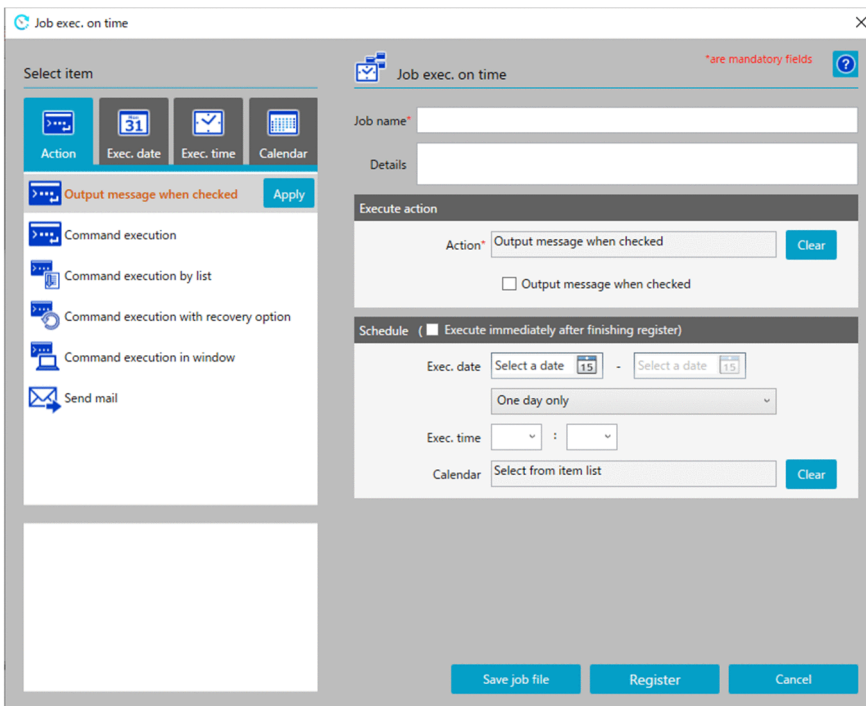
Open the **Action** dialog box from the Item Design View, specify the information described in the following table in an item, and then register the item.

Table 5–13: Information specified for the input item "Check box"

No.	Specifics of where each piece of information is defined			Information to be specified
	Main window	Where to transition to	Location where the information is specified	
1	Item Design View +-- <b>Action</b> dialog box	Basic settings	Command	cpaecho
2			Parameter	\${JDS:msg}
3			Execution type	<b>Do not show the window</b>
4		<b>Input item settings</b> +-- <b>Define input items</b> dialog box	Variable name	msg
5			Label	(Nothing is input.)
6			Input type	Check box
7				Value when checked: The check box was selected. Value when unchecked: (Nothing is input.) Details: The message is output when the check box is selected.

The following figure shows what the dialog box looks like when the created item is to be used in a job.

Figure 5–13: An item having a check box input item



When the **Send a message when the check box is selected** check box is selected, the message The check box was selected. is sent to standard output.

## 5.7.4 Enabling or disabling an input item

You can set an enable condition when you want the user to enter a string or select an option or check box only when the specified precondition is satisfied. This setting allows you to prevent an input error or other operational mistakes.

### Example: Outputting the entered value to standard output when User input allowed is selected

Open the **Action** dialog box from the Item Design View, specify the information described in the following table in an item, and then register the item.

Table 5–14: Information specified for the input item "Enable condition"

No.	Specifics of where each piece of information is defined			Information to be specified		
	Main window	Where to transition to	Location where the information is specified			
1	Item Design View +-- <b>Action</b> dialog box	Basic settings	Command	cpaecho		
2			Parameter	\${JDS:msg}		
3			Execution type	<b>Do not show the window</b>		
4		<b>Input item settings</b> +-- <b>Define input items</b> dialog box		Variable name	select	
5				Label	Whether user input is allowed	
6				Input type	List box	List label: User input not allowed Value: 0
7						List label: User input allowed Value: 1
8						
9		<b>Input item settings</b> +-- <b>Define input items</b> dialog box		Variable name	msg	
10				Label	Message to be output	
11				Input type	Text box	
12				Option Settings	Enable when item ID: 1 <sup>#</sup>	
13				+--Enable condition	Value: 1	

#  
When you first define an input item whose details are provided in No. 4 to 8 in the table above, and then proceed to set the other input item whose details are provided in No. 9 to 13 in the table above, you have to set the item ID to 2. In the **Input item settings** tab, check **Input item list for items** to find the item ID of the input item whose details are provided in No. 4 to 8 and then set an appropriate item ID in No. 12.

The following figure shows what the dialog box looks like when the created item is to be used in a job.

Figure 5–14: An item in which an input item is disabled

The screenshot shows the 'Job exec. on time' configuration window. On the left, the 'Select item' panel has 'Output input message' selected. The main area shows the 'Execute action' section with 'Action' set to 'Output input message' and 'Message to be output' set to 'User input not allowed'. The 'Message to be output' text box is disabled. The 'Schedule' section has 'Execute immediately after finishing register' checked. At the bottom are 'Save job file', 'Register', and 'Cancel' buttons.

Figure 5–15: An item in which an input item is enabled

The screenshot shows the 'Job exec. on time' configuration window. On the left, the 'Select item' panel has 'Output input message' selected. The main area shows the 'Execute action' section with 'Action' set to 'Output input message' and 'Message to be output' set to 'User input allowed'. The 'Message to be output' text box is enabled. The 'Schedule' section has 'Execute immediately after finishing register' checked. At the bottom are 'Save job file', 'Register', and 'Cancel' buttons.

When **User input not allowed** is selected, the **Message to be output** text box is disabled and cannot accept user input.

When **User input allowed** is selected, the **Message to be output** text box is enabled and is ready to accept user input. The message entered while the text box is enabled is sent to standard output.



## ! Important

When the disabled input item is enabled, users can enter a string in the text box, select an option from the list box, or select the check box.

If the enabled input item is edited, disabled, and then registered, the input item is registered in its default state. By default, the text box is empty, the first option is selected in the list box, and the check box is cleared.

### 5.7.5 Inserting an option letter before the value

When the specification of a command parameter requires that both the option and the value specified for the option be specified, if *option-letter input-value-variable* is specified as the value of the parameter in an item, the command ends in an error when there is no user-input value.

This entry can serve the following purpose: Whenever a user-input value is specified, it is prefixed with an option letter, and when no user-input value is specified, a variable to which an input value is assigned is used without option letters.

#### Example: Executing the `cpasetprof` command with `-d input-value` when there is a user-input value

Open the **Action** dialog box from the Item Design View, specify the information described in the following table in an item, and then register the item.

Table 5–15: Information specified for the input item "Option letters"

	Specifics of where each piece of information is defined			Information to be specified
	Main window	Where to transition to	Location where the information is specified	
1	Item Design View +-- <b>Action</b> dialog box	Basic settings	Command	<code>cpasetprof</code>
2			Parameter	<code>\${JDS:prof}</code>
3			Execution type	<b>Do not show the window</b>
4		<b>Input item settings</b> +-- <b>Define input items</b> dialog box	Variable name	<code>prof</code>
5			Label	Name of the profile to be deleted
6			Input type	Text box
7			Option letters	<code>-dΔ#</code>

#

Δ denotes a half-width space.

The following figure shows what the dialog box looks like when the created item is to be used in a job.

Figure 5–16: An item that has an option letter

When there is a user-input value, `cpasetprof -d input-value` is executed, and when there is no user-input value, `cpasetprof` is executed.

For details on the `cpasetprof` command, see [cpasetprof](#) in *9. Commands*.

### ! Important

Depending on the input type of the item, no option letter is inserted in the following cases:

- List box  
The list has no corresponding value.
- Check box  
The option of the check box has no corresponding value.

## 5.7.6 Making an input item a mandatory field

When the specification of a command parameter requires that a certain option be specified, failure to enter that option can cause a problem, such as the occurrence of an error upon the execution of the command. This input item can help prevent an operational error resulting from missing input by mandating the entry of text.

### Example: Sending the user-input value to standard output

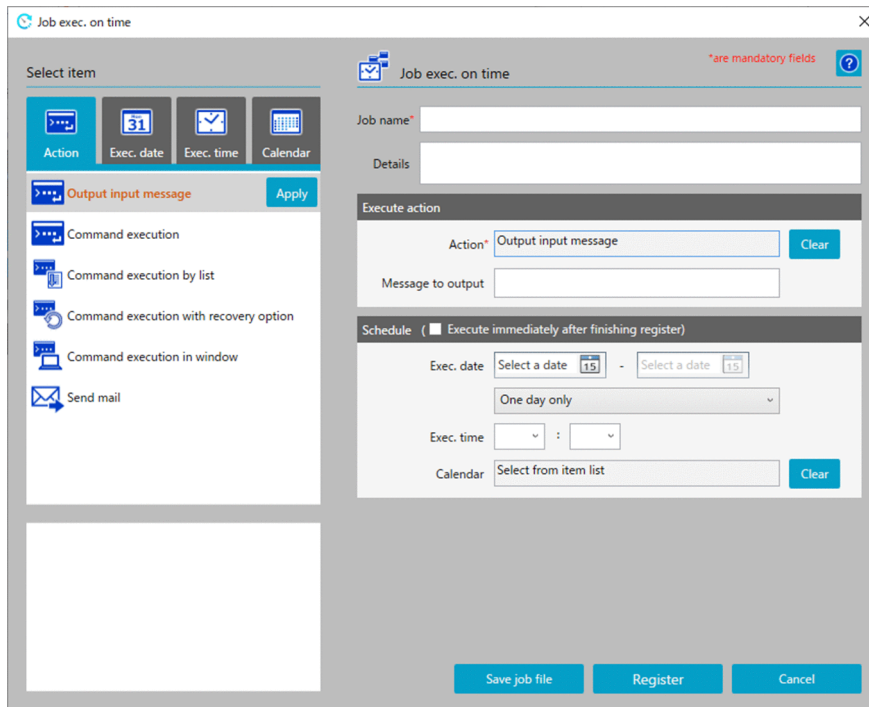
Open the **Action** dialog box from the Item Design View, specify the information described in the following table in an item, and then register the item.

Table 5–16: Information specified for the input item "Abbreviate"

No.	Specifics of where each piece of information is defined			Information to be specified
	Main window	Where to transition to	Location where the information is specified	
1	Item Design View +-- <b>Action</b> dialog box	Basic settings	Command	cpaecho
2			Parameter	\${JDS:msg}
3			Execution type	<b>Do not show the window</b>
4		<b>Input item settings</b> +-- <b>Define input items</b> dialog box	Variable name	msg
5			Label	Message to be output
6			Input type	Text box
7			Abbreviate	Select the check box.

The following figure shows what the dialog box looks like when the created item is to be used in a job.

Figure 5–17: Item that has a mandatory input item



The string entered in the **Message to be output** text box is sent to standard output. If you attempt to register the item without specifying a string in this text box, the KNAO4004-E dialog box appears, and the registration attempt fails.

# 6

## Job Design View

To open the Job Design View, go to the program menu on the machine on which CPA is installed, and select **JP1 Client Process Automation**, then **Job Design View**. A standard Job Design View allows you to register jobs and create custom items.

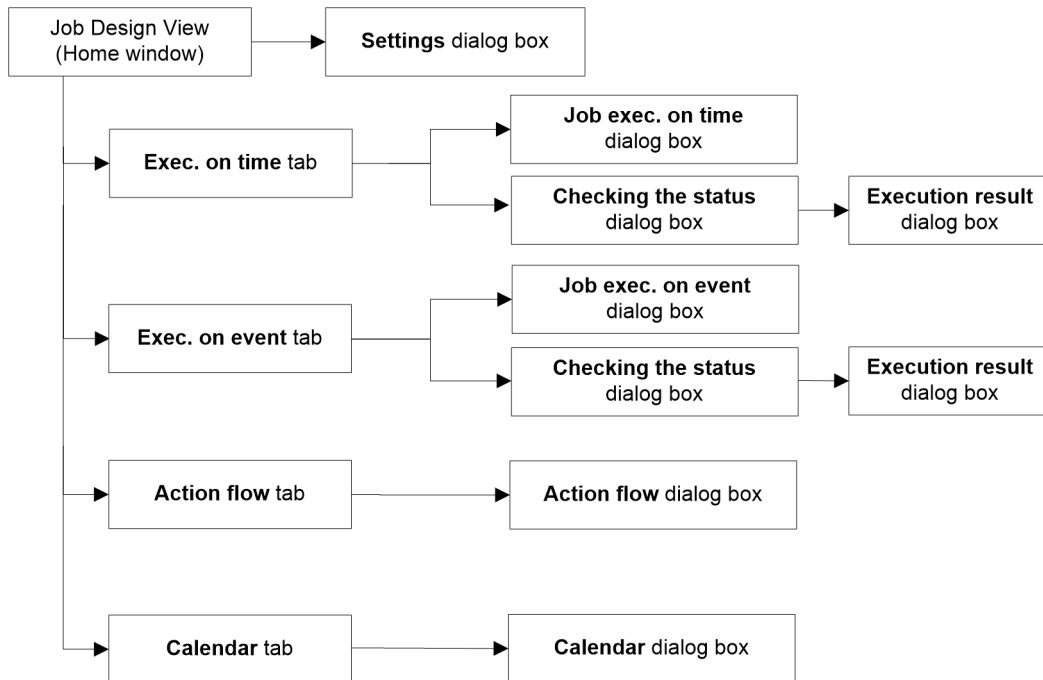
This chapter describes the Job Design View of CPA.

The Job Design View can be configured to display only the job registration functions (in other words, functions related to the definition of action flow and calendar items can be removed from the view), which may provide greater convenience to users who do not need to create or edit custom items. For details on this configuration, see [\*cpausrenvsetup\*](#) in [\*9. Commands\*](#).

## 6.1 List of windows

The following figure shows the windows you can navigate to from the Job Design View.

Figure 6–1: Windows you can navigate to from the Job Design View



The following table provides a list of windows you can navigate to from the Job Design View.

Table 6–1: List of windows

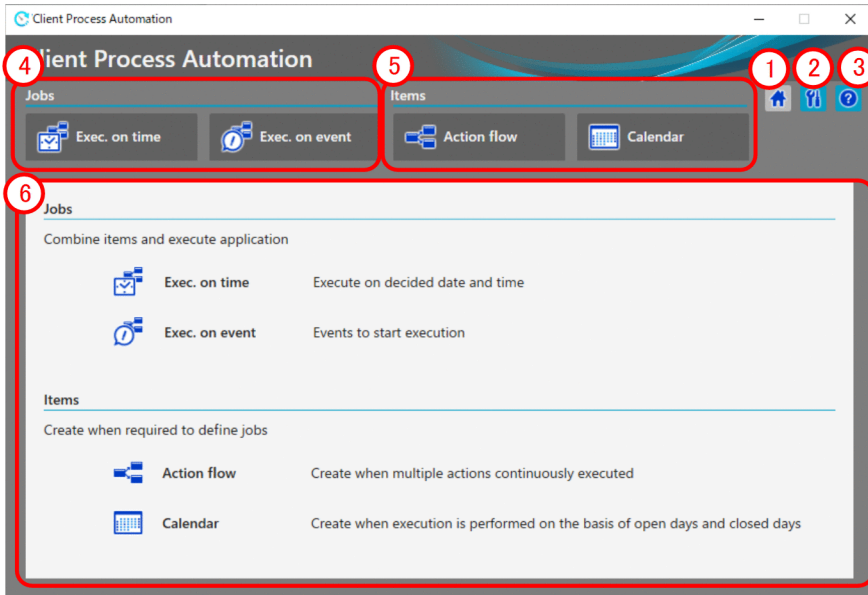
No.	Window name	Functions
1	Job Design View (Home window)	This window allows you to define jobs and items.
2	Settings dialog box	This dialog box allows you to set a language.
3	Exec. on time tab	This tab allows you to view a list of jobs executed on time.
4	Exec. on event tab	This tab allows you to view a list of jobs executed on event.
5	Action flow tab	This tab allows you to view a list of action flow items.
6	Calendar tab	This tab allows you to view a list of calendar items.
7	Job exec. on time dialog box	This dialog box allows you to define jobs executed on time.
8	Job exec. on event dialog box	This dialog box allows you to define jobs executed on event.
9	Checking the status dialog box	This dialog box allows you to check, re-execute, and stop jobs.
10	Execution result dialog box	This dialog box allows you to check job execution results in details.
11	Action flow dialog box	This dialog box allows you to define action flow items.
12	Calendar dialog box	This dialog box allows you to define calendar items.

## 6.1.1 Home window

The Home window is the first window you encounter when you start the Job Design View of CPA. It gives you access to the various functions provided by CPA.

The following figure shows the Home window.

Figure 6–2: Home window



The following items are displayed:

### ① Home button

This button displays the Home window.

### ② Settings button

This button displays the **Settings** dialog box. From the **Settings** dialog box, you can specify a display language for the Job Design View.

### ③ Help button

This button displays the **Help** dialog box. In the **Help** dialog box, you can quickly find out how to use each window.

### ④ Job operation-related tabs

Select one of these tabs when you want to perform job operation-related actions. When you select a tab, registered jobs show up in the lower part of the window.

### ⑤ Item definition-related tabs

Select one of these tabs when you want to perform item definition-related actions. When you select a tab, registered items show up in the lower part of the window.

When the use of item definition functions is disabled in the configuration, these tabs will not appear in the Home window.

### ⑥ Explanation of each tab

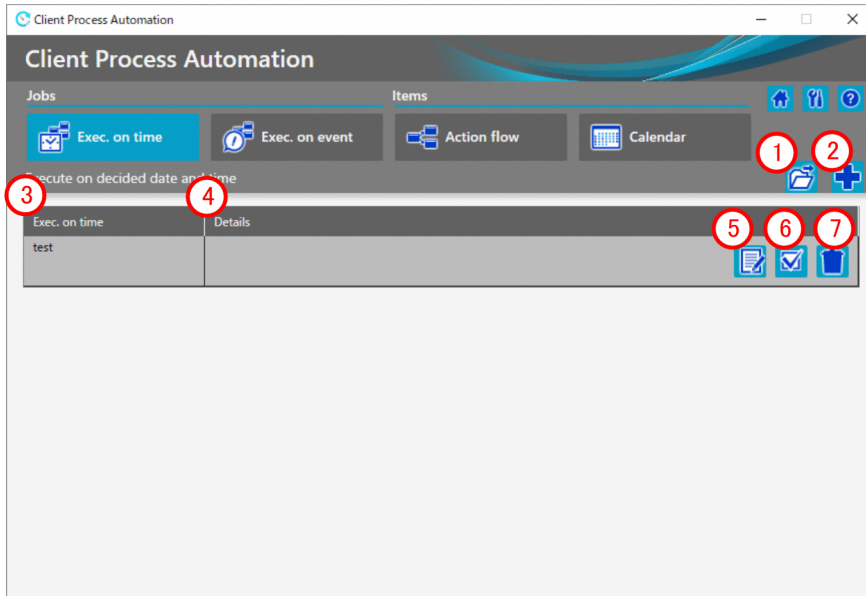
This area displays a list of jobs or items corresponding to the selected tab. When no tab is selected, it provides a brief explanation of each tab.

## 6.2 Job operation windows

The **Exec. on time** tab and the **Exec. on event** tab allow you to create, edit, check the status of, and delete jobs that are executed according to the set schedule and those that are triggered by the set event, respectively.

The following figure shows an example of the window displayed when one of these tabs is selected.

Figure 6–3: Exec. on time tab



The following items are displayed:

**1 Open button**

This button loads a job definition file. This job definition file is the one you defined in the **Job exec. on time** dialog box and then created with the **Save job file** button.

**2 New button**

This button creates a new job executed on time.

**3 Exec. on time**

This column displays the name of the registered job.

**4 Details**

This column provides an explanation of the registered job.

**5 Edit button**

This button allows you to edit the definition of the registered job. This button is useful when you want to change the behavior of the registered job.

**6 Checking the status button**

This button shows the status of the registered job (for example, whether it is currently executing or has already ended). If the job has already ended, you can check its execution result.

**7 Delete button**

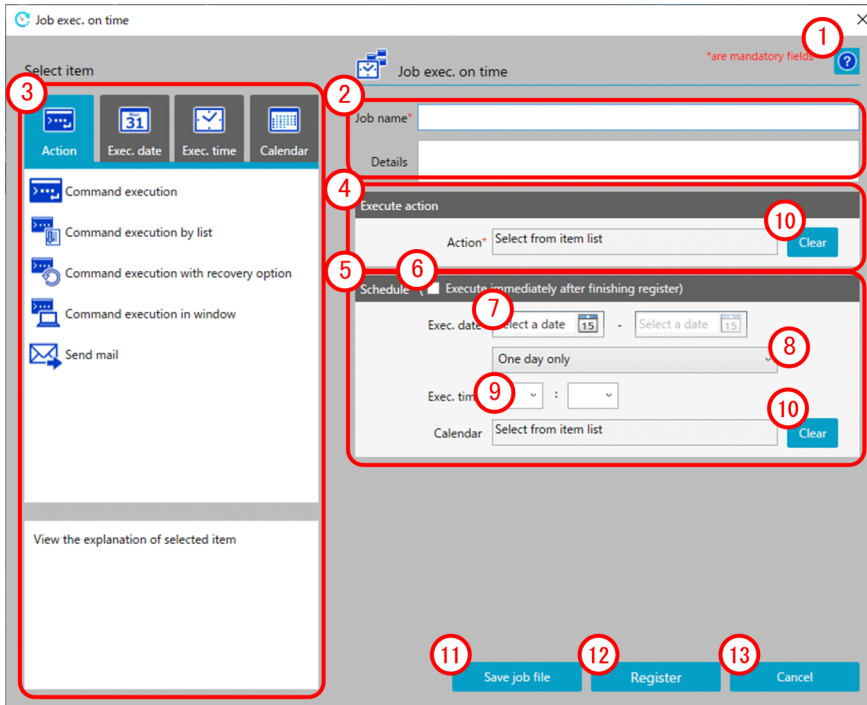
This button deletes the registered job.

## 6.2.1 Job exec. on time dialog box

You can display the **Job exec. on time** dialog box by selecting the **Exec. on time** tab. This dialog box allows you to define a job in which a schedule is set for job execution.

The following figure shows the **Job exec. on time** dialog box.

Figure 6–4: Job exec. on time dialog box



The following items are displayed:

### ① Help button

This button displays the Help window.

### ② Basic information area

This area allows you to enter a job name and provides an explanation of the job.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Job name	Text box	A 1- to 45-byte string. You can specify half-width alphanumeric characters, underscores (_), and multibyte characters.	Mandatory	None
2	Details	Text box	A character string not exceeding 1,024 bytes. You can use line breaks.	Optional	None

### Note

Each character in the string you enter in the text box is counted differently in bytes depending on the character. For the conversion rate of the characters you enter into bytes, see [C.2 Conversion rate of characters into bytes](#). Note that text boxes cannot contain only line breaks and/or half-width spaces.



### ③ Select item area

This area allows you to select an item you want to apply. When you select one of the **Action**, **Exec. date**, **Exec. time**, and **Calendar** tabs, selectable items show up. From the displayed items, either select the one you want to apply and click the **Apply** button or simply double-click the one you want to apply. The selected item is applied to the **Execute action** area and the **Schedule** area.

### ④ Execute action area

This area displays the items you can set for the item you have applied by using the **Action** tab in the **Select item** area.

### ⑤ Schedule area

This area allows you to set a schedule for executing the job. It reflects the items you have applied by using the **Exec. date**, **Exec. time**, and **Calendar** tabs in the **Select item** area. You can also directly specify the job execution date(s), job execution cycle, and job execution time.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Exec. date	Text box	Specify this item in <i>yyyy/mm/dd</i> format. You can specify a value in the range from 2013/01/01 to 2037/12/31.	Optional	None
2	(Cycle)	Pull-down menu	Select from one of the following options: <b>One day only</b> , <b>Execute everyday</b> , <b>Execute in every open day</b> , and <b>Execute in every closed day</b> .	Optional	One day only
3	Exec. time	Text box	Specify a value in the range from 00 to 23 for the hour, and a value in the range from 00 to 59 for the minute.	Optional	None

### ⑥ Execute immediately after finishing register check box

If you select this check box, the job is executed immediately after it is registered. Selecting this check box disables all schedule settings, which creates a schedule definition that enables immediate execution of the job. If you register a job with this check box selected, and then open the **Job exec. on time** dialog box again to edit it, the check box is cleared in the displayed dialog box.

### ⑦ Exec. date

Specify the start date and end date of job execution. You can specify the end date only when you select **Execute everyday**, **Execute in every open day**, or **Execute in every closed day** from the **Cycle** pull-down menu located immediately below.

You can also enter these dates by using the **Calendar** button displayed to right of each entry field.

### ⑧ Cycle

This pull-down menu allows you to specify the cycle for executing the job.

### ⑨ Exec. time

This pull-down menu allows you to specify the time at which to start executing the job.

### ⑩ Clear button

This button deletes the values that reflect the item applied in the **Select item** area.

**11 Save job file** button

This button saves the file containing the created job definition.

**12 Register** button

Registers jobs.

**13 Cancel** button

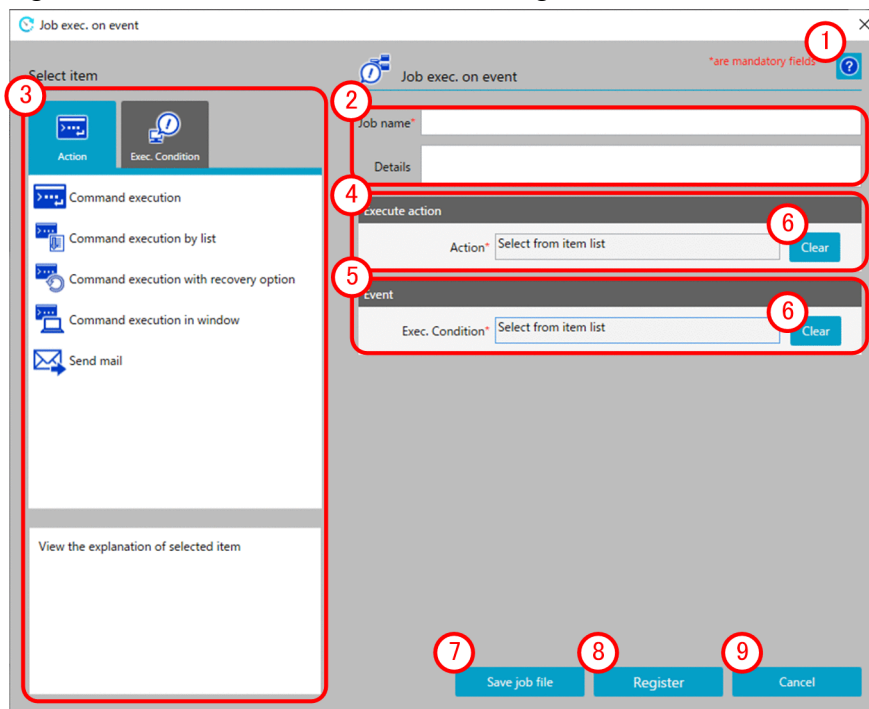
This button closes the window without registering the edited definition.

## 6.2.2 Job exec. on event dialog box

You can display the **Job exec. on event** dialog box by selecting the **Exec. on event** tab. This dialog box allows you to define a job in which an event is set to trigger job execution.

The following figure shows the **Job exec. on event** dialog box.

Figure 6–5: Job exec. on event dialog box



The following items are displayed:

**1 Help** button

This button displays the Help window.

**2 Basic information** area

This area allows you to enter a job name and provides an explanation of the job.

For details on the displayed items and the restrictions on the strings you can specify in this area, see *Basic information area* in [6.2.1 Job exec. on time dialog box](#).

### ③ **Select item** area

This area allows you to select an item you want to apply. When you select the **Action** tab or the **Exec. Condition** tab, selectable items show up. From the displayed items, either select the one you want to apply and click the **Apply** button or simply double-click the one you want to apply. The selected item is applied to the **Execute action** area and the **Schedule** area.

### ④ **Execute action** area

This area displays the items you can set for the item you have applied by using the **Action** tab in the **Select item** area.

### ⑤ **Evet** area

This area displays the items you can set for the item you have applied by using the **Exec. Condition** tab in the **Select item** area.

### ⑥ **Clear** button

This button deletes the values that reflect the item applied in the **Select item** area.

### ⑦ **Save job file** button

This button saves the file containing the created job definition.

### ⑧ **Register** button

Registers jobs.

### ⑨ **Cancel** button

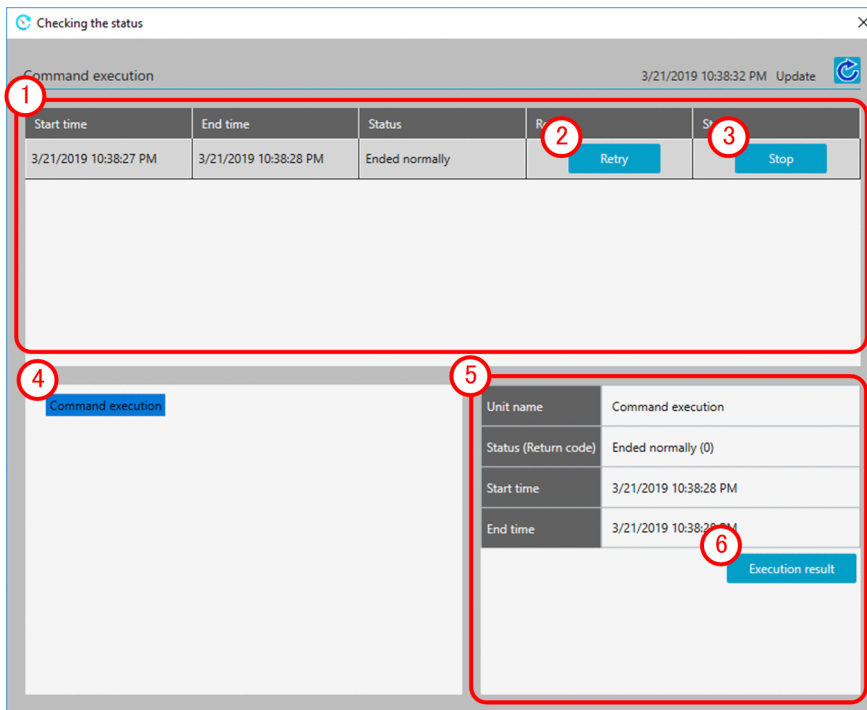
This button closes the window without registering the edited definition.

## 6.2.3 Checking the status dialog box

You can display the **Checking the status** dialog box by selecting the **Exec. on time** tab or the **Exec. on event** tab. This dialog box allows you to check the status of the job.

The following figure shows the **Checking the status** dialog box.

Figure 6–6: Checking the status dialog box



The following items are displayed:

① Area that displays a list of executed jobs

This area displays a list of executed jobs. It also shows you the start time, end time, and status of each job. Furthermore, buttons are provided to re-execute or stop each job.

② **Retry** button

Reruns jobs.

③ **Stop** button

This button stops the job that is currently being executed.

④ Area that displays a list of executed units

This area displays a list of units belonging to the job selected in the area that displays a list of executed jobs. When you select a unit, its status shows up in the unit status area.

⑤ Item status area

This area displays the status of the unit selected in the area that displays a list of executed units. Furthermore, if the unit is an action item, you can see the execution result of the item by clicking the **Execution result** button.

⑥ **Execution result** button

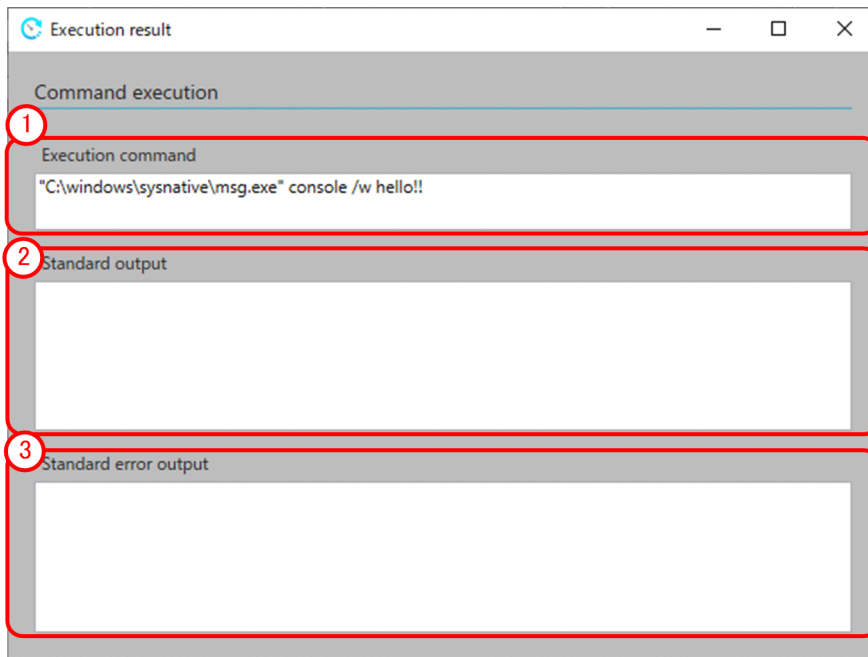
This button enables you to see the execution result of the unit (action or event monitor item) selected in the area that displays a list of executed units.

## (1) Execution result dialog box

In the **Execution result** dialog box, you can check the result of executing an item.

The following figure shows the **Execution result** dialog box.

Figure 6–7: Execution result dialog box



The following items are displayed:

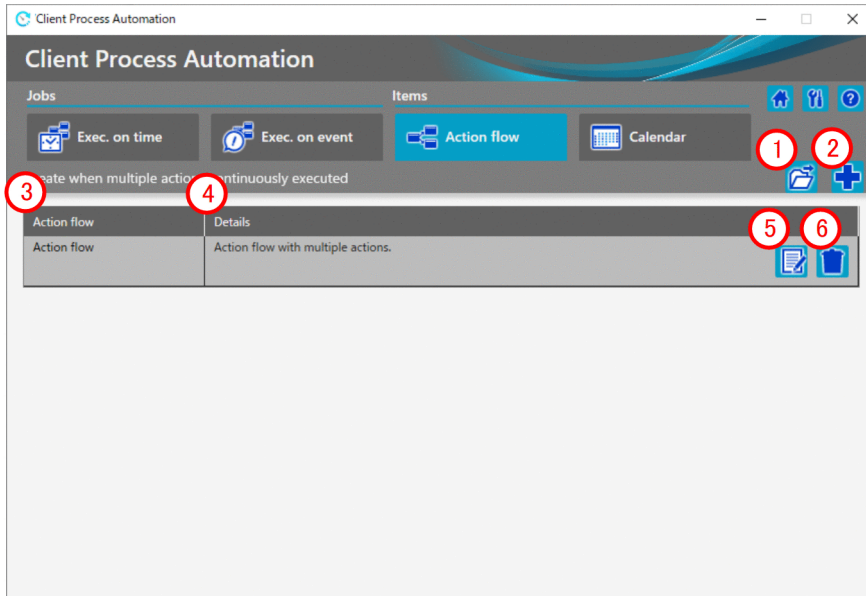
- 1 Execution command area**  
This area displays the executed command.
- 2 Standard output area**  
This area displays the standard output of the executed command.
- 3 Standard error output area**  
This area displays the standard error output of the executed command.

## 6.3 Window used to define items

The **Action flow** tab and the **Calendar** tab allow you to create, edit, and delete action flow items and calendar items, respectively, that are used in jobs.

The following figure shows an example of the window displayed when one of these tabs is selected.

Figure 6–8: Action flow tab



The following items are displayed:

① **Open** button

This button loads an action flow definition file. This action flow definition file is the one you defined in the **Action flow** dialog box and then created with the **Save as file** button.

② **New** button

This button creates a new action flow.

③ **Action flow**

This column displays the item name of the registered action flow.

④ **Details**

This column provides an explanation of the registered action flow.

⑤ **Edit** button

This button allows you to edit the definition of the registered action flow. This button is useful when you want to change the definition of the registered action flow.

⑥ **Delete** button

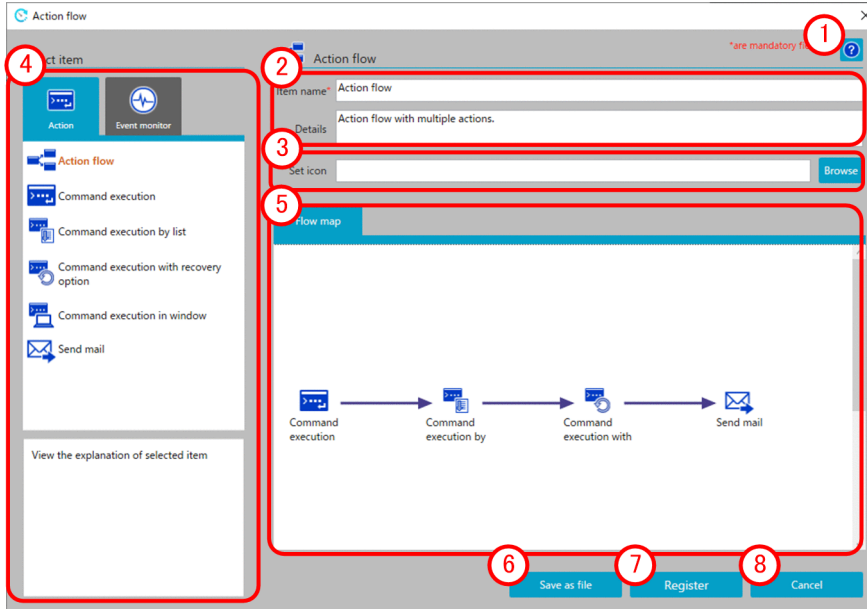
This button deletes the registered action flow.

## 6.3.1 Action flow dialog box

You can display the **Action flow** dialog box by selecting the **Action flow** tab. This dialog box allows you to define an action flow in which multiple actions are defined along with the order in which to execute them.

The following figure shows the **Action flow** dialog box.

Figure 6–9: Action flow dialog box



The following items are displayed:

### ① Help button

This button displays the Help window.

### ② Basic information area

This area allows you to enter an item name and provides an explanation of the item.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Item name	Text box	A 1- to 45-byte string. You can specify half-width alphanumeric characters, underscores (_), and multibyte characters.	Mandatory	None
2	Details	Text box	A character string not exceeding 1,024 bytes. You can use line breaks.	Optional	None

#### Note

Each character in the string you enter in the text box is counted differently in bytes depending on the character. For the conversion rate of the characters you enter into bytes, see [C.2 Conversion rate of characters into bytes](#). Note that text boxes cannot contain only line breaks and/or half-width spaces.

### ③ Set icon area

This area allows you to set the icon to be displayed in the **Select item** area. By default, nothing is displayed in this area. If no icon is specified, the default action flow icon is applied.

#### ④ Select item area

Select the items you want to assign to the action flow. When you select the **Action** tab or the **Event monitor** tab, the selectable items show up. From the displayed items, drag the one you want to assign to the action flow to the **Flow map** area, and then drop it. This assigns the item to the action flow.

#### ⑤ Flow map area

This area allows you to set the order in which to execute the actions assigned to the action flow. You can do this by selecting one of the actions assigned to this area and then drawing a relation line (arrow) from it.

By right-clicking the relation line (arrow), you can delete the execution order definition.

#### ⑥ Save as file button

This button saves the file containing the created action flow definition.

#### ⑦ Register button

Registers action flow.

#### ⑧ Cancel button

This button closes the window without registering the edited definition.

## 6.3.2 Calendar dialog box

You can display the **Calendar** dialog box by selecting the **Calendar** tab.

You can use the **Calendar** dialog box to set certain days of the week or certain dates as open days or closed days as well as define the operation time period during which to execute a job.

The following figure shows the **Calendar** dialog box.

Figure 6–10: Calendar dialog box (Set closed days tab)

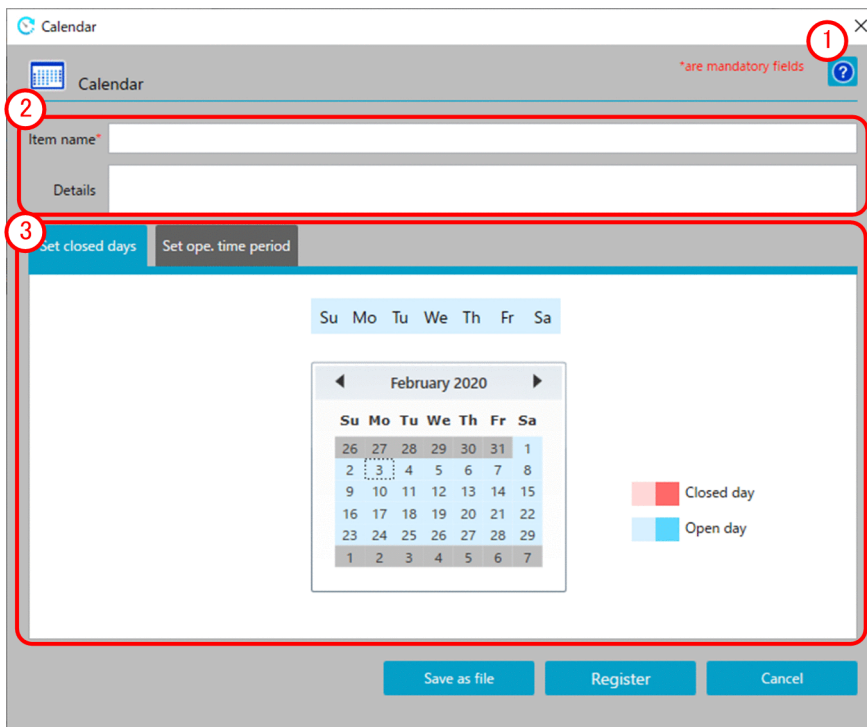
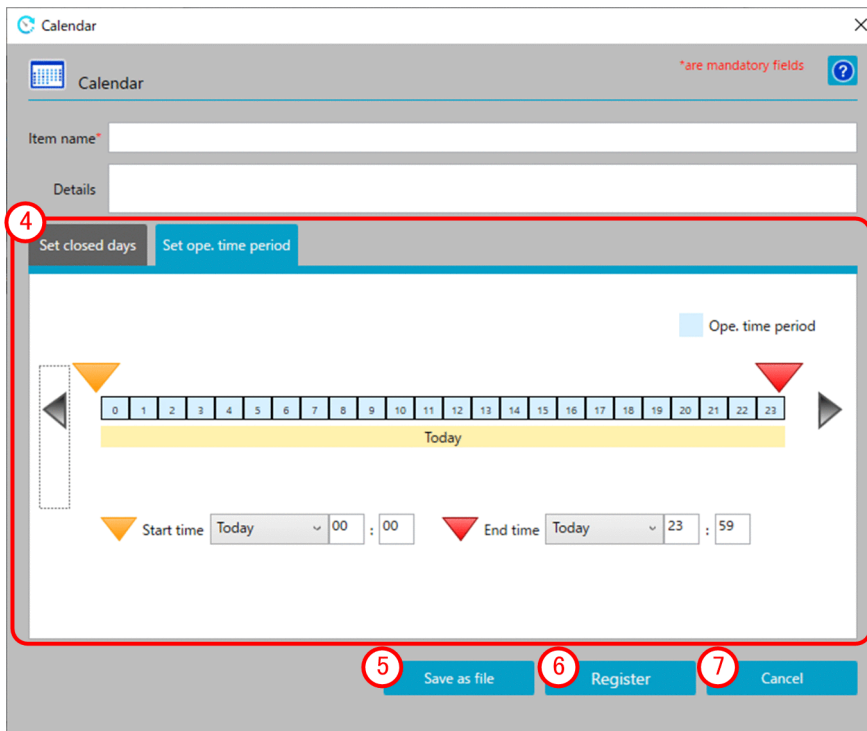




Figure 6–11: Calendar dialog box (Set ope. time period tab)



The following items are displayed:

① **Help** button

This button displays the Help window.

② **Basic information** area

This area allows you to enter the item name and specify an explanation of the item.

For details on the displayed items and the restrictions on the strings you can specify in this area, see *Basic information area* in 6.3.1 *Action flow dialog box*.

③ **Set closed days** area

This area allows you to set open days and closed days for the calendar you are creating. If you select the **Set closed days** tab, an operational calendar appears.

By selecting a day of the week displayed on top of the calendar, you can set whether that day of the week is usually an open day or a closed day. By selecting a date in the calendar, you can designate the selected date as the open day or closed day.

④ **Set ope. time period** area

This area allows you to set a time period during which to execute the job. If you select the **Set ope. time period** tab, the date and time of the operation time period show up.

By specifying the start time and end time of the displayed operation time period, you can set the time period during which to execute the job.

⑤ **Save as file** button

This button saves the file containing the created calendar definition.

⑥ **Register** button

Registers calendar.

**7** **Cancel** button

This button closes the window without registering the edited definition.

# 7

## Item Design View

To open the Item Design View, go to the program menu on the machine on which CPA is installed, and select **JP1 Client Process Automation**, then **Item Design View**.

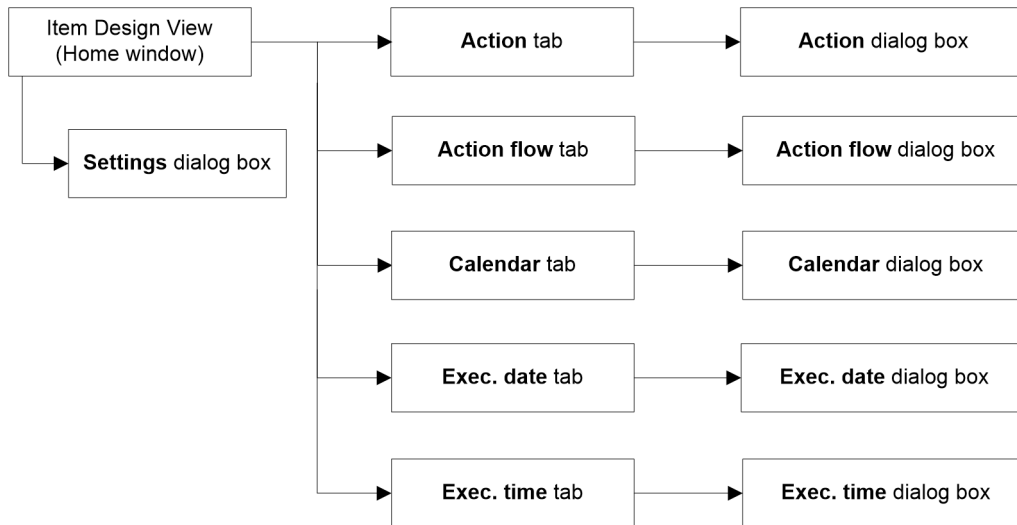
The Item Design View allows you to create or edit items that can be used during the registration of jobs.

This chapter describes the Item Design View of CPA.

## 7.1 List of windows

The following figure shows the windows you can navigate to from the Item Design View.

Figure 7–1: Windows you can navigate to from the Item Design View



The following table provides a list of windows you can navigate to from the Item Design View.

Table 7–1: List of windows

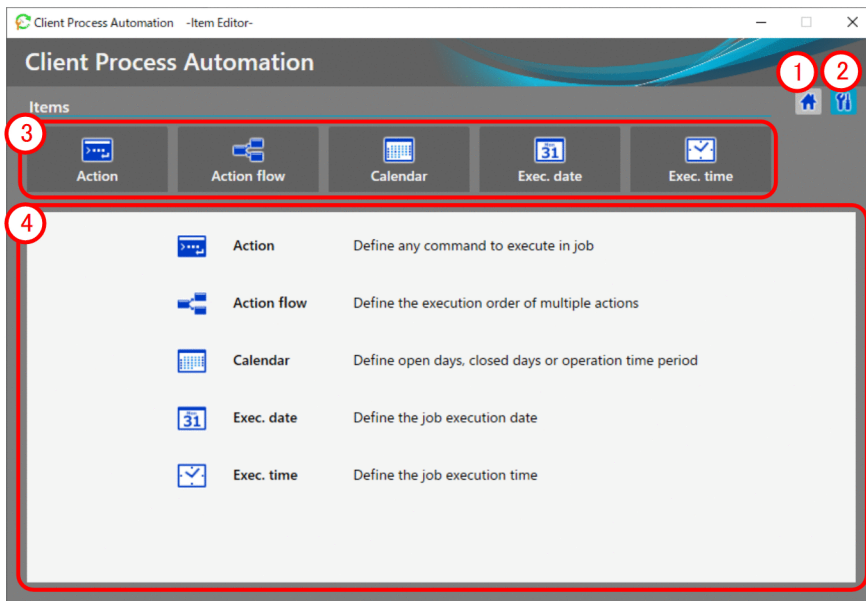
No.	Window name	Functions
1	Item Design View (Home window)	This window allows you to define items.
2	<b>Settings</b> dialog box	This dialog box allows you to set a language.
3	<b>Action</b> tab	This tab allows you to view a list of registered actions.
4	<b>Action flow</b> tab	This tab allows you to view a list of registered action flows.
5	<b>Calendar</b> tab	This tab allows you to view a list of registered calendars.
6	<b>Exec. date</b> tab	This tab allows you to view a list of registered execution dates.
7	<b>Exec. time</b> tab	This tab allows you to view a list of execution times.
8	<b>Action</b> dialog box	This dialog box allows you to define actions.
9	<b>Action flow</b> dialog box	This dialog box allows you to define action flows.
10	<b>Calendar</b> dialog box	This dialog box allows you to define calendars.
11	<b>Exec. date</b> dialog box	This dialog box allows you to define execution dates.
12	<b>Exec. time</b> dialog box	This dialog box allows you to define execution times.

### 7.1.1 Home window

The Home window is the first window you encounter when you start the Item Design View of CPA. It gives you access to the various functions provided by the Item Design View of CPA.

The following figure shows the Home window.

Figure 7–2: Home window



The following items are displayed:

① **Home** button

This button displays the Home window.

② **Settings** button

This button displays the **Settings** dialog box. From the **Settings** dialog box, you can specify a display language for the Item Design View.

③ **Item definition-related tabs**

These tabs allow you to select a type of item you want to edit. When you select a tab, registered items show up in the lower part of the window.

④ **Explanation of each tab**

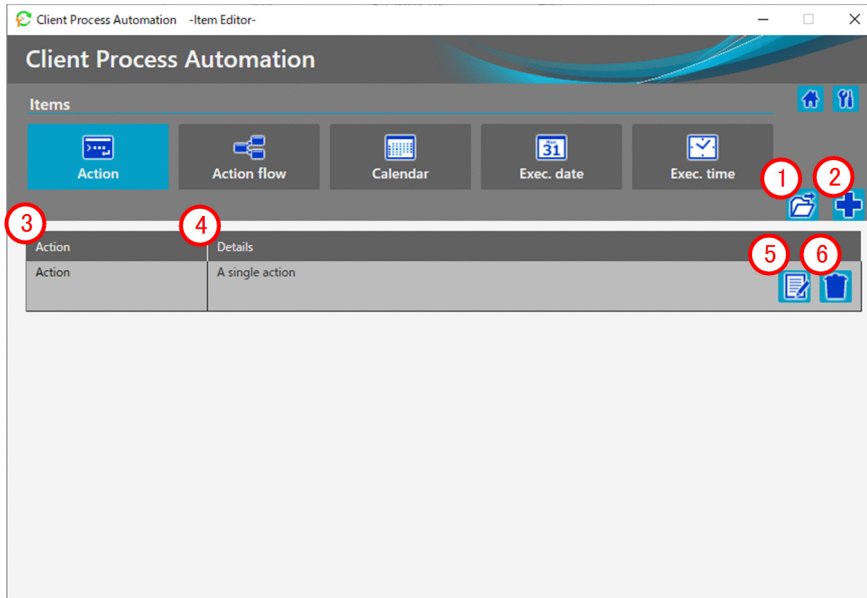
This area displays a list of jobs or items corresponding to the selected tab. When no tab is selected, it provides a brief explanation of each tab.

## 7.2 Item definition windows

You can create, edit, or delete an item used in a job. You can select an item you want to operate from the **Action**, **Action flow**, **Calendar**, **Exec. date**, and **Exec. time** tabs.

The following figure shows an example of the window displayed when one of these tabs is selected.

Figure 7–3: Action tab



The following items are displayed:

**① Open button**

This button loads an action definition file. This action definition file is the one you defined in the **Action** dialog box and then created with the **Save as file** button.

**② New button**

This button creates a new action.

**③ Action**

This column displays the name of the registered action.

**④ Details**

This column provides an explanation of the registered action.

**⑤ Edit button**

This button allows you to edit the registered action. This button is useful when you want to change the definition of the registered action.

**⑥ Delete button**

This button deletes the registered action.

## 7.2.1 Action dialog box

You can display the **Action** dialog box by selecting the **Action** tab. This dialog box allows you to define a single action.

The following figure shows the **Action** dialog box.

Figure 7–4: Action dialog box

The screenshot shows the 'Action' dialog box with the following elements highlighted by red circles and lines:

- 1**: A red box around the 'Item name\*' text input field.
- 2**: A red box around the 'Details' text input field.
- 3**: A red box around the 'Set icon' text input field and the 'Browse' button next to it.
- 4**: A red circle around the 'Save as file' button at the bottom.
- 5**: A red circle around the 'Register' button at the bottom.
- 6**: A red circle around the 'Cancel' button at the bottom.

Other visible elements include: 'Action' title bar, 'are mandatory fields' warning, 'Apply this item to the item list in the Job Define View' checkbox, 'Basic settings', 'Option settings', and 'Input item settings' tabs, 'Command\*' and 'Parameter' text boxes with 'Browse' buttons, and 'Command operation settings' section with 'Execution type' and 'Operations when terminates' dropdown menus.

The following items are displayed:

### ① Basic information area

This area allows you to enter an item name and provides an explanation of the item.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Item name	Text box	A 1- to 45-byte string. You can specify half-width alphanumeric characters, underscores (_), and multibyte characters.	Mandatory	None
2	Details	Text box	A character string not exceeding 1,024 bytes. You can use line breaks.	Optional	None

#### Note

Each character in the string you enter in the text box is counted differently in bytes depending on the character. For the conversion rate of the characters you enter into bytes, see [C.2 Conversion rate of characters into bytes](#). Note that text boxes cannot contain only line breaks and/or half-width spaces.

### ② Set icon area

This area allows you to set an icon to be displayed in the **Select item** area.

If no icon is set, the default action icon is applied.

**3 Apply this item to the item list in the Job Define View** check box

This check box allows you to select whether to apply the registered action to the item list in the Job Design View. By clearing this check box, you can prevent the registered action from showing up in the item list in the Job Design View used by operators. Even when the check box is cleared, the action is still visible from the Item Design View. If you just want to use the action as part of an action flow, clear this check box.

**4 Save as file** button

This button saves the file containing the created action definition.

**5 Register** button

Registers action flow.

**6 Cancel** button

This button closes the window without registering the edited definition.

## (1) Basic settings tab

The **Basic settings** tab allows you to specify basic settings for the action.

The following figure shows the **Basic settings** tab.

Figure 7–5: Basic settings tab

Figure 7–5 shows the 'Basic settings' tab of the 'Action' configuration window. The window title is 'Action' and it has a close button. A red asterisk indicates mandatory fields. Fields include 'Item name', 'Details', and 'Set icon' with a 'Browse' button. A checked checkbox reads 'Apply this item to the item list in the Job Define View'. Below are three tabs: 'Basic settings' (selected), 'Option settings', and 'Input item settings'. Under 'Basic settings', there are 'Command\*' and 'Parameter' fields, each with a 'Browse' button. A red box highlights these two fields and is labeled with a circled '1'. Below this is the 'Command operation settings' section with 'Execution type' (set to 'Do not show the window') and 'Operations when terminates' (set to 'Abnormal end if the return code is other than 0'). These two settings are labeled with circled '2' and '3' respectively. At the bottom are 'Save as file', 'Register', and 'Cancel' buttons.

The following items are displayed:



## ① Command specification area

This area allows you to enter the command to execute and parameter.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Command	Text box	A 1- to 256-byte string	Mandatory	None
2	Parameter	Text box	A string not exceeding 512 bytes	Optional	None

## ② Execution type

This pull-down menu allows you to specify a command execution type. The specifiable options are **Do not show the window** and **Show the window**. If you select **Show the window**, you can specify the following settings: **Wait for end** and **Exclusive option**.

## ③ Operation when terminates

This pull-down menu allows you to specify how the executed command should end. The specifiable options are **Abnormal end if the return code is other than 0**, **Warning end if the return code is other than 0**, and **Always normal end**.

### (a) Execution type is Show the window

If you selected **Show the window** as the execution type, you have to specify **Wait for end** and **Exclusive option**.

The following figure shows the window when **Show the window** is selected.

Figure 7–6: When Show the window

The screenshot shows the 'Action' configuration window with the following settings:

- Item name: [Text box]
- Details: [Text box]
- Set icon: [Text box] [Browse]
- Apply this item to the item list in the Job Define View
- Basic settings | Option settings | Input item settings
- Command: [Text box] [Browse]
- Parameter: [Text box]
- Command operation settings:
  - Execution type: Show the window
  - ① Wait for end: Wait for the window showing process to terminate
  - ② Exclusive option:  Do not execute concurrently among selected actions
  - Operations when terminates: Abnormal end if the return code is other than 0
- Buttons: Save as file, Register, Cancel

The following items are displayed:

## ① Wait for end

This pull-down menu allows you to specify whether to wait for the command displayed in the window to end. The specifiable options are **Wait for the window showing process to terminate** and **Do not wait for the window showing process to terminate**.

If you select **Do not wait for the window showing process to terminate**, you cannot select **Operations when terminates**.

## ② Exclusive option

If this option is selected, the specified actions are not executed concurrently (only one action is executed all the time) (which is the same as the **Exclusive option** for the **Command execution in window** item). The option is applied among all the items for which it is enabled. If some items should not be executed concurrently, select this option.

## (2) Option settings tab

This tab allows you to set when the executing action times out.

It also allows you to specify the name of the virtual user who executes the item.

The following figure shows the **Option settings** tab.

Figure 7–7: Option settings tab

The screenshot shows the 'Action' configuration dialog box. The 'Option settings' tab is selected. The 'Execution timeout period' field is marked with a circled 1, the 'Item status after timeout' dropdown is marked with a circled 2, and the 'Virtual user name' field is marked with a circled 3. The dialog also includes a 'Browse' button for the 'Set icon' field and a 'Save as file' button at the bottom.

The following items are displayed:

### ① Execution timeout period

This setting allows you to specify a timeout period for the executing action. If you do not specify this setting or specify 0, the action never times out. If you have selected the **Exclusive option** check box in the **Basic settings** tab, the period of time you specify here also includes the time spent waiting for other actions to end.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Execution timeout period	Text box	Specify a half-width number or a variable. You can specify a number in the range from 1 to 1440.	Optional	None

### ② Ended state of the action upon timeout

This pull-down menu allows you to specify the *Ended* state of the action upon timeout. The specifiable values are **Ended abnormally** and **Ended with warning**.

### ③ Virtual user name

This field specifies the virtual user name mapped to the name of the Windows user who executes the item. If the virtual user name is omitted, the item is executed with the account for the CPA job execution service.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

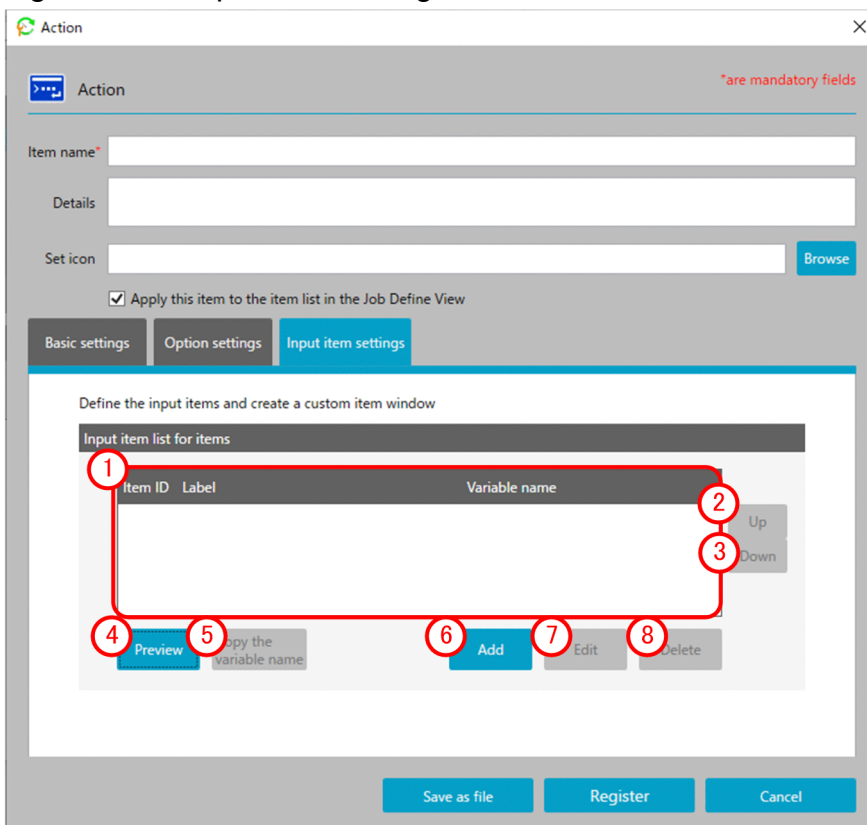
No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Virtual user name	Text box	A string not exceeding 64 bytes. Specify a half-width number or a variable.	Optional	None

## (3) Input item settings tab

The **Input item settings** tab allows you to set input items that are necessary to apply the action.

The following figure shows the **Input item settings** tab.

Figure 7–8: Input item settings tab



The following items are displayed:

① **Input item list for items**

This list displays the input items defined for the action. You can create a maximum of 10 input items.

② **Up button**

This button moves the selected input item up in the list.

③ **Down button**

This button moves the selected input item down in the list.

④ **Preview button**

This button displays a preview of the action input window.

With the **OK** button, you can check the operation of input items according to the specified settings (the **Abbreviate** option, the **Max. size** field, and the **Possible input letters** option). You can use the **Cancel** button to close the preview window.

⑤ **Copy variable name button**

This button copies the **Input point variable name** setting specified for the selected input item.

⑥ **Add browse button**

This button creates a new input item to be used in the action.

⑦ **Edit button**

This button edits the definition of a registered input item.

## 8 Delete button

This button deletes a registered input item.

### (a) Define input items dialog box

Clicking the **Add** button brings up the **Define input items** dialog box.

The following figure shows the **Define input items** dialog box.

Figure 7–9: Define input items dialog box

The screenshot shows the 'Define input items' dialog box. It is divided into two main sections: 'Basic settings' and 'Option settings'. The 'Basic settings' section includes fields for 'Variable name' (with a red asterisk indicating it is mandatory), 'Label', 'Input type' (set to 'Text box'), 'Max. size' (in bytes), and 'Possible input letters' (set to 'All characters'). There are also checkboxes for 'Enable line breaks' and 'Add browse button'. The 'Option settings' section includes an 'Enable condition' (Enable when item ID is [ ] value is [ ] only), 'Option letters' (Before the variable value, grant the option letters [ ] before the value), and an 'Abbreviate' checkbox (Specification of input item is mandatory). At the bottom right, there are 'OK' and 'Cancel' buttons. Red circles with numbers 1 through 8 are overlaid on the image to highlight specific elements: 1 points to the dialog title bar, 2 to the 'Input type' dropdown, 3 to the 'Basic settings' section header, 4 to the 'Enable condition' field, 5 to the 'Option letters' field, 6 to the 'Abbreviate' checkbox, 7 to the 'OK' button, and 8 to the 'Cancel' button.

The following items are displayed:

#### 1 Basic settings area

This area allows you to specify basic settings, including the input point variable name and the label.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Variable name	Text box	A variable-format string not exceeding 57 bytes. Only a string consisting of half-width alphanumeric characters can be used to specify a variable name.	Mandatory	None
2	Label	Text box	A string not exceeding 64 bytes. Multibyte characters can be used.	Optional	None
3	Input type	Pull down	Select from the following options: <b>Text box</b> , <b>List box</b> , and <b>Check box</b> .	Optional	None

#### Note

Each character in the string you enter in the text box is counted differently in bytes depending on the character. For the conversion rate of the characters you enter into bytes, see [C.2 Conversion rate of characters into bytes](#). Note that text boxes cannot contain only half-width spaces.

## ② Input type

This pull-down menu allows you to specify a data entry method for the input item to be added.

## ③ Option settings area

This area allows you to set options for the input item.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Enable condition	Text box	You can specify half-width numbers.	Optional	None
2	Option letters	Text box	A string not exceeding 32 bytes. This text box accepts a string consisting of half-width alphanumeric characters and symbols.	Optional	None
3	Abbreviate	Check box	--	Optional	Not selected

Legend:

--: Not applicable

## ④ Enable condition

These text boxes allow you to specify the condition for enabling the input item. Specify this option when you want a certain condition to be satisfied before data can be entered.

## ⑤ Option letters

This field specifies option letters preceding the variable value. If the variable does not have a value, the option letters are not added.

## ⑥ Abbreviate

This check box allows you to specify whether the input item can be omitted.

## ⑦ OK button

This button saves the edited input item.

## ⑧ Cancel button

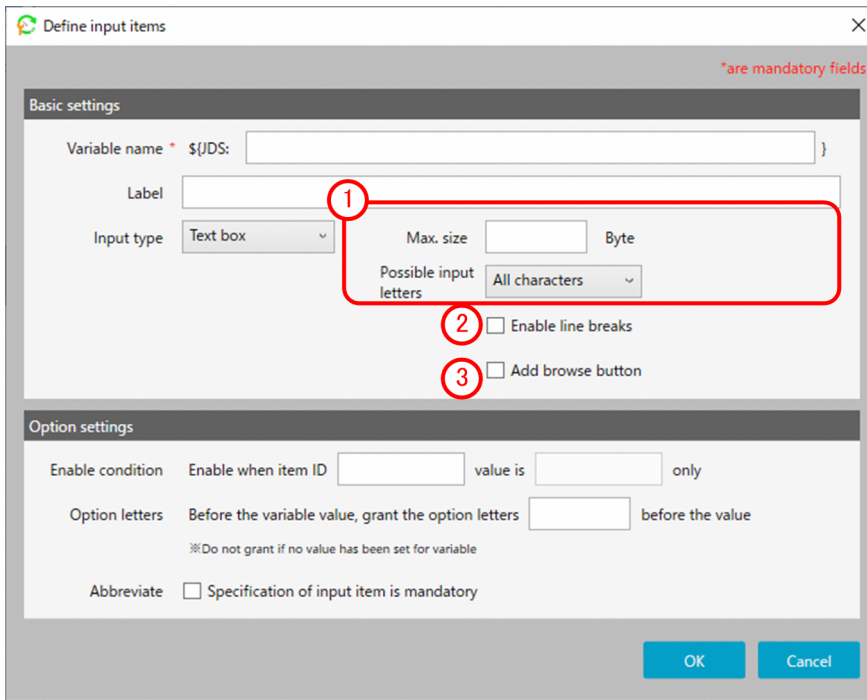
This button closes the dialog box without adding the edited definition.

### ■ Input type (Text box)

If you select **Text box** from the **Input type** pull-down menu, the text box setting area shows up in the dialog box.

The following figure shows what the window looks like when **Input type (Text box)** is selected as the execution cycle.

Figure 7–10: What the window looks like when Input type (Text box) is selected as the execution cycle



The following items are displayed:

**1** Text box setting area

This area allows you to specify the maximum size of data that can be entered in the text box as well as the types of characters that can be entered.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Max. size	Text box	Specify a half-width number in the range from 1 to 512.	Optional	512
2	Possible input letters	Pull down	Select from the following options: <b>All characters</b> , <b>Alphanumeric characters only</b> , <b>Alphabetical only</b> , <b>Numbers only</b> , and <b>Alphanumeric characters and symbols only</b> .	Optional	None

Note

Each character in the string you enter in the text box is counted differently in bytes depending on the character. For the conversion rate of the characters you enter into bytes, see [C.2 Conversion rate of characters into bytes](#).

**2** Possible break line check box

This check box allows you to specify whether to allow line breaks to be added in the text box.

**3** Append check box

This check box allows you to specify whether to add a button that can be used to display a reference window during the entry of data into the text box.

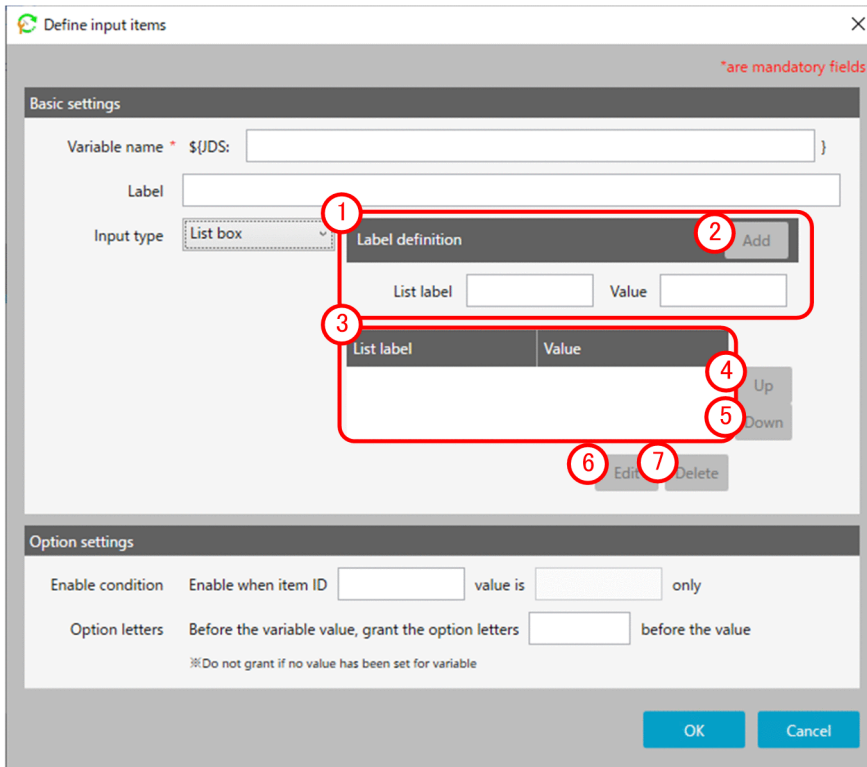
Select this check box when the text box is used to specify a path to an execution program or file.

## ■ Input type (List box)

If you select **List box** from the **Input type** pull-down menu, the **Label definition** area and the **List label** area show up in the dialog box.

The following figure shows what the window looks like when **Input type (List box)** is selected as the execution cycle.

Figure 7–11: What the window looks like when Input type (List box) is selected as the execution cycle



The following items are displayed:

### ① Label definition area

This area allows you to specify a label name to be added to the list box as well as its value.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	List label	Text box	A string not exceeding 64 bytes	Optional	None
2	Value	Text box	A string not exceeding 512 bytes	Optional	None

### ② Add browse button

This button adds the defined label to the list box.

### ③ List label area

This area displays a list of created list labels.

### ④ Up button

This button moves the selected label up.



### ⑤ Down button

This button moves the selected label down.

### ⑥ Edit button

This button allows you to edit the selected list label.

### ⑦ Delete button

This button deletes the selected list label.

## ■ Input type (Check box)

If you select **Check box** from the **Input type** pull-down menu, the check box setting area shows up in the dialog box.

The following figure shows what the window looks like when **Input type (Check box)** is selected as the execution cycle.

Figure 7–12: What the window looks like when Input type (Check box) is selected as the execution cycle

The screenshot shows a dialog box titled "Define input items". It has a close button (X) in the top right corner. A red asterisk indicates that certain fields are mandatory. The "Basic settings" section includes: "Variable name" (mandatory, with a text box containing "\$(JDS: )"), "Label" (text box), and "Input type" (dropdown menu set to "Check box"). A red circle with the number "1" highlights the "Value when checked", "Value when unchecked", and "Details" text boxes. The "Option settings" section includes: "Enable condition" (text box) "value is" (text box) "only", and "Option letters" (text box) "before the value". A note at the bottom of the "Option settings" section says "Do not grant if no value has been set for variable". "OK" and "Cancel" buttons are at the bottom right.

The following items are displayed:

### ① Check box setting area

You can set a value applicable when the check box is selected and a value applicable when the check box is cleared, as well as specify an explanation of the check box.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Value when checked	Text box	A string not exceeding 512 bytes	Optional	None
2	Value when unchecked	Text box	A string not exceeding 512 bytes	Optional	None
3	Details	Text box	A string not exceeding 64 bytes However, if the details contain any half-width character, all the characters	Optional	None

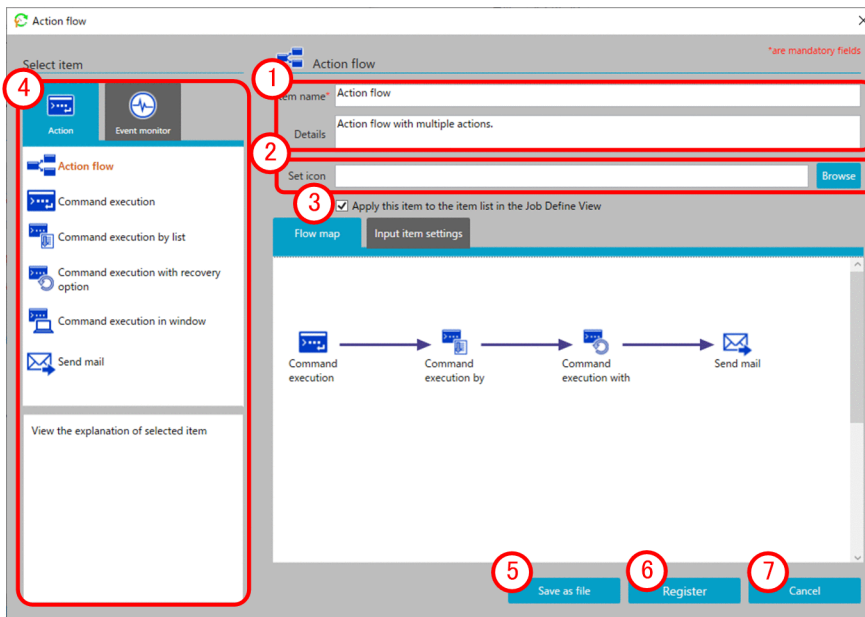
No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
			may not be shown. Use the preview window to see how they are shown.		

## 7.2.2 Action flow dialog box

You can display the **Action flow** dialog box by selecting the **Action flow** tab. This dialog box allows you to define an action flow in which multiple actions are defined along with the order in which to execute them.

The following figure shows the **Action flow** dialog box.

Figure 7–13: Action flow dialog box



The following items are displayed:

### ① Basic information area

This area allows you to enter an item name and provides an explanation of the item.

For details on the displayed items and the restrictions on the strings you can specify in this area, see *Basic information area* in [7.2.1 Action dialog box](#).

### ② Set icon area

This area allows you to set the icon to be displayed in the **Select item** area.

If no icon is specified, the default action flow icon is applied.

### ③ Apply this item to the item list in the Job Define View check box

This check box allows you to select whether to apply the registered action to the item list in the Job Design View. By clearing this check box, you can prevent the registered action from showing up in the item list in the Job Design View used by operators. Even when the check box is cleared, the action is still visible from the Item Design View. If you just want to use the action as part of an action flow, clear this check box.

#### ④ Select item area

Select the items you want to assign to the action flow. When you select the **Action** tab or the **Event monitor** tab, the selectable items show up. From the displayed items, drag the one you want to assign to the action flow to the **Flow map** area, and then drop it. This assigns the item to the action flow.

#### ⑤ Save as file button

This button saves the file containing the created action flow definition.

#### ⑥ Register button

Registers action flow.

#### ⑦ Cancel button

This button closes the window without registering the edited definition.

## (1) Unit dialog box

When you put an item you want to add to **Action flow** from the **Select item** area to the **Flow map** area, you see the **Unit** dialog box where you configure the settings necessary for executing the item.

As an example, the following figure shows the **Unit** dialog box when you put the *Command execution* standard item.

Figure 7–14: Unit dialog box

The screenshot shows the 'Unit' dialog box with the following components and callouts:

- 1**: Unit name\* text box containing 'Command execution'.
- 2**: 'conditions to execute item' section containing a 'Unit name' text box, 'Return code is' dropdown (set to 'equal to'), and a 'Value' text box.
- 3**: 'Items' section containing:
  - Item name: Command execution
  - Details: Execute command.
  - Execution command\*: text box with a 'Browse' button.
  - Parameter: text box.
  - Virtual user name: text box.
- 4**: OK button.
- 5**: Cancel button.

The following items are displayed:

#### ① Unit name area

Enter a unique name in the action flow.

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Unit name	Text box	A string from 1 to 45 bytes. You can specify half-width alphanumeric characters, underscores ( _ ), and multibyte characters.	Mandatory	Item name

**Note**

Each character in the string you enter in the text box is counted differently in bytes depending on the character. For the conversion rate of the characters you enter into bytes, see [C.2 Conversion rate of characters into bytes](#).

**② Conditions to execute item area**

This area is used to specify whether or not the item is executed depending on the return value of the preceding unit.

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Unit name <sup>#</sup>	Text box	A string from 1 to 45 bytes. You can specify half-width alphanumeric characters, underscores ( _ ), and multibyte characters.	Optional	None
2	(Condition)	Pull down	Select from either of <b>equal to</b> or <b>not equal to</b> .	Optional	<b>equal to</b>
3	Value	Text box	Specify a value from 0 to 4,294,967,295. The return value is treated as an unsigned 32-bit value. For example, if the application as the preceding unit ends with -1, the return value will be 4,294,967,295.	Optional	None

**Note**

Each character in the string you enter in the text box is counted differently in bytes depending on the character. For the conversion rate of the characters you enter into bytes, see [C.2 Conversion rate of characters into bytes](#).

#

In the **Unit name** field, you can specify only the unit name for an action or event monitor, which is found in the flow map.

**③ Item area**

This area displays the input items for the item you put.

**④ OK button**

This button places the input items you edited.

**⑤ Cancel button**

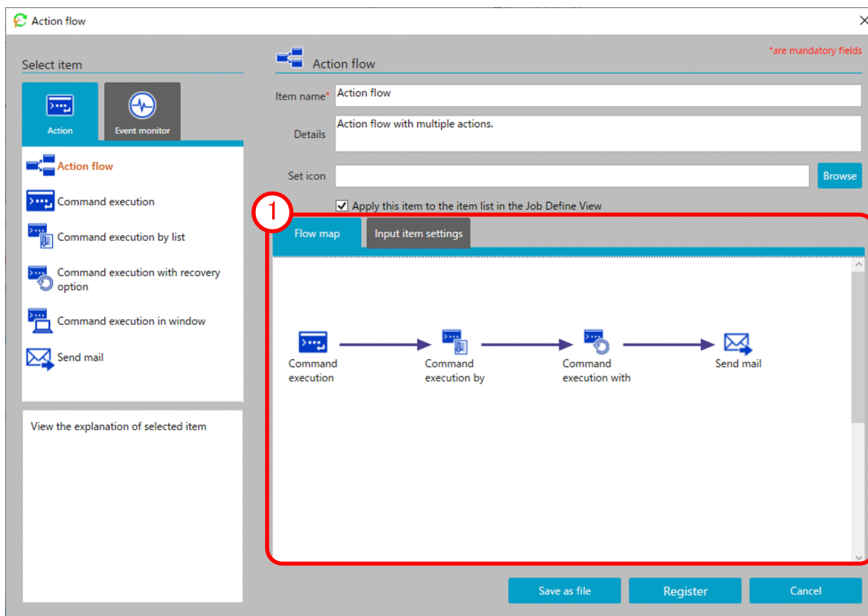
This button closes the window without registering the edited definition.

**(2) Flow map tab**

The **Flow map** tab allows you to set the order in which to execute the assigned actions.

The following figure shows the **Flow map** tab.

Figure 7–15: Flow map tab



The following items are displayed:

### ① Flow map area

This area allows you to select an action placed in the **Flow map** area and draw a relation line (arrow) from it to connect it with another action. This allows you to define the order in which to execute actions.

In addition, by right-clicking a relation line (arrow), you can change the connection method or delete the execution order definition.

You can change the connection method by defining the execution order of subsequent actions differently depending on how the preceding action ends (*Ended normally* vs. *Ended with warning*). For example, you can define that when action A is executed and ends normally, action B is executed, whereas when action A is executed and ends with a warning, action C is executed instead. In this case, you have to select **Change connection method** and **Ended normally** for the arrow extending from action A to action B, and **Change connection method** and **Ended with warning** for the arrow extending from action A to action C.

## (3) Input item settings tab

The **Input item settings** tab allows you to set the input items necessary to apply an action flow.

For details on the **Input item settings** tab, see [7.2.1 \(3\) Input item settings tab](#).

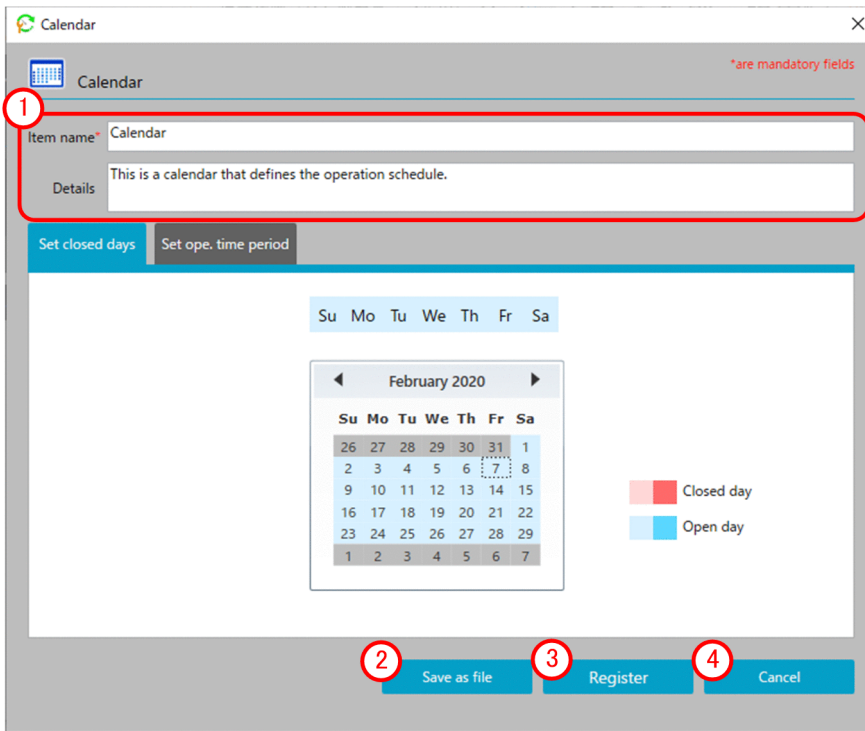
## 7.2.3 Calendar dialog box

You can display the **Calendar** dialog box by selecting the **Calendar** tab.

You can use the **Calendar** dialog box to set certain days of the week or certain dates as open days or closed days as well as define the operation time period during which to execute a job.

The following figure shows the **Calendar** dialog box.

Figure 7–16: Calendar dialog box



① Basic information area

This area allows you to enter the item name and specify an explanation of the item.

For details on the displayed items and the restrictions on the strings you can specify in this area, see *Basic information area* in *7.2.1 Action dialog box*.

② Save as file button

This button saves the file containing the created calendar definition.

③ Register button

Registers calendar.

④ Cancel button

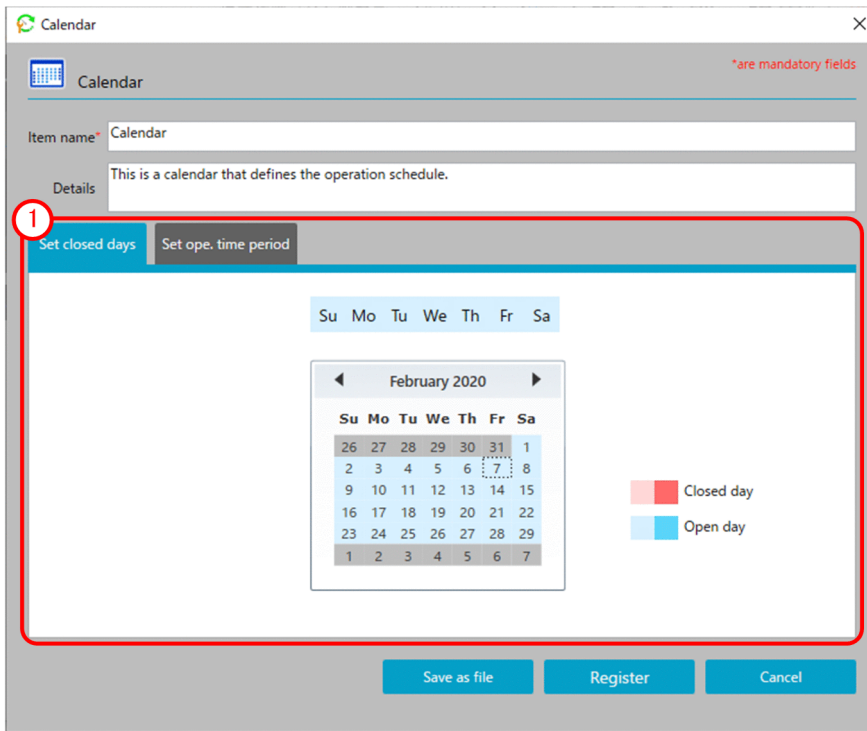
This button closes the window without registering the edited definition.

## (1) Set closed days tab

The **Set closed days** tab allows you to set open days and closed days for the calendar you are creating.

The following figure shows the **Set closed days** tab.

Figure 7–17: Set closed days tab



The following items are displayed:

**1** Set closed days area

This area allows you to set open days and closed days for the calendar you are creating. If you select the **Set closed days** tab, an operational calendar appears.

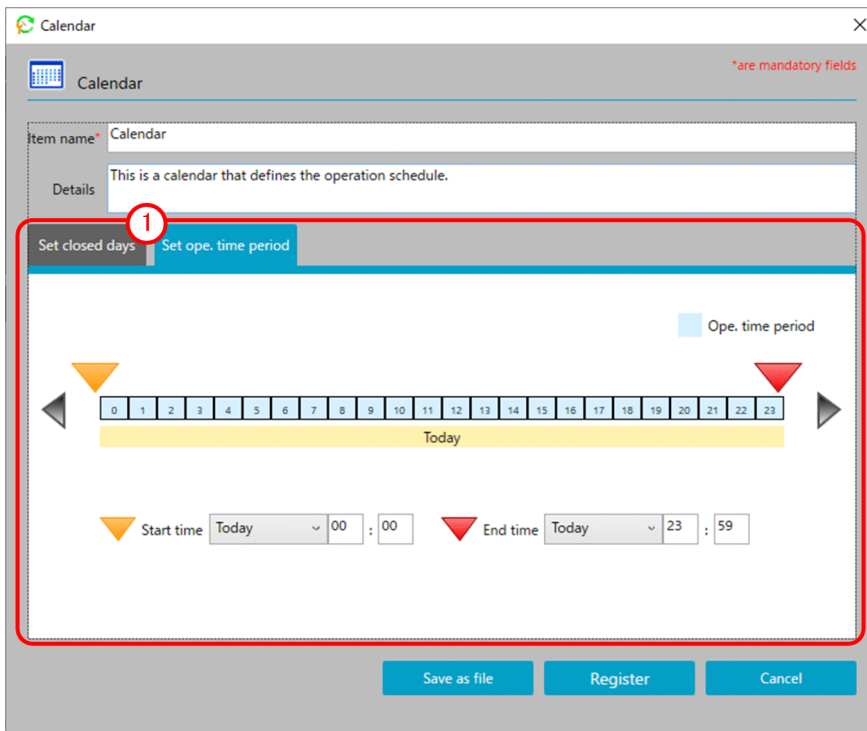
By selecting a day of the week displayed on top of the calendar, you can set whether that day of the week is usually an open day or a closed day. By selecting a date in the calendar, you can designate the selected date as the open day or closed day.

## (2) Set ope. time period tab

The **Set ope. time period** tab allows you to specify the duration of time during which to execute the job.

The following figure shows the **Set ope. time period** tab.

Figure 7–18: Set op. time period tab



The following items are displayed:

① Set op. time period area

This area allows you to set a time period during which to execute the job. If you select the **Set op. time period** tab, the date and time of the operation time period show up.

By specifying the start time and end time of the displayed operation time period, you can set the time period during which to execute the job.

## 7.2.4 Exec. date dialog box

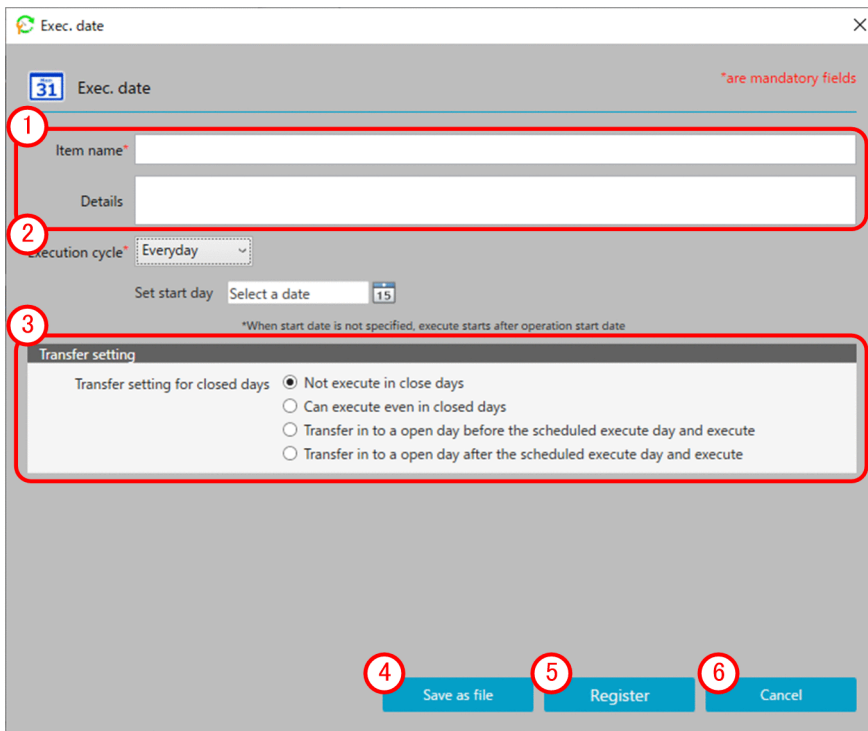
You can display the **Exec. date** dialog box by selecting the **Exec. date** tab.

The **Exec. date** dialog box allows you to define when to start executing the job as well as the processing cycle.

The following figure shows the **Exec. date** dialog box.



Figure 7–19: Exec. date dialog box



The following items are displayed:

① Basic information area

This area allows you to enter the item name and specify an explanation of the item.

For details on the displayed items and the restrictions on the strings you can specify in this area, see *Basic information area* in 7.2.1 *Action dialog box*.

② Execution cycle

This pull-down menu allows you to specify the job execution cycle.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Execution cycle	Pull down	Select from the following options: <b>Everyday</b> , <b>Every week</b> , <b>Every month</b> , <b>Every year</b> , <b>Every operation day</b> , <b>Every closed day</b> , and <b>No cycle</b> .	Mandatory	None

③ Transfer setting

This setting allows you to specify how to determine a substitute job execution date when the job execution date happens to fall on a closed day. By default, **Not execute in close days** is selected. When, however, **Execution cycle** is set to **Every closed day**, **Can execute even in closed days** is selected by default.

④ Save as file button

This button saves the file containing the created execution date definition.

⑤ Register button

Registers execution date.

## ⑥ Cancel button

This button closes the window without registering the edited definition.

## (1) Execution cycle (Everyday)

If you have selected **Everyday** as the execution cycle, set the start date for the job to be executed every day.

The following figure shows what the window looks like when **Everyday** is selected as the execution cycle.

Figure 7–20: What the window looks like when Everyday is selected as the execution cycle

The following items are displayed:

## ① Execution cycle (Everyday)

This area allows you to specify the conditions for starting the job execution cycle.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

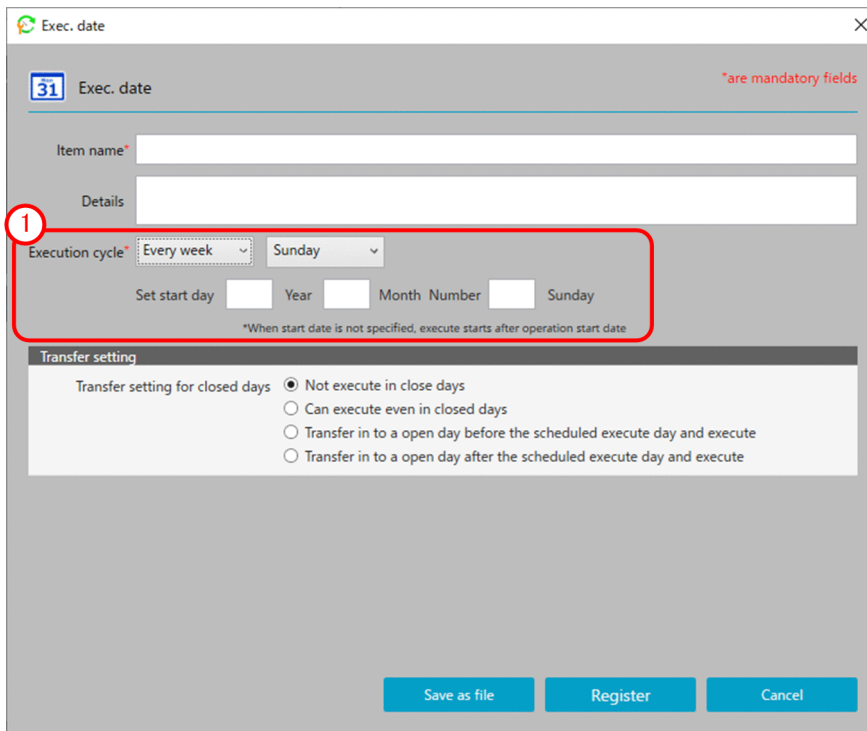
No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Set start day	Pull down	Specify the start date in <code>yyyy/mm/dd</code> format. You can specify a value in the range from 2013/01/01 to 2037/12/31.	Optional	None

## (2) Execution cycle (Every week)

If you have selected **Every week** as the execution cycle, set the day of the week for weekly execution of the job as well as the date on which to start job execution.

The following figure shows what the window looks like when **Every week** is selected as the execution cycle.

Figure 7–21: What the window looks like when Every week is selected as the execution cycle



The following items are displayed:

### ① Execution cycle (Every week)

This area allows you to specify the conditions for starting the job execution cycle.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

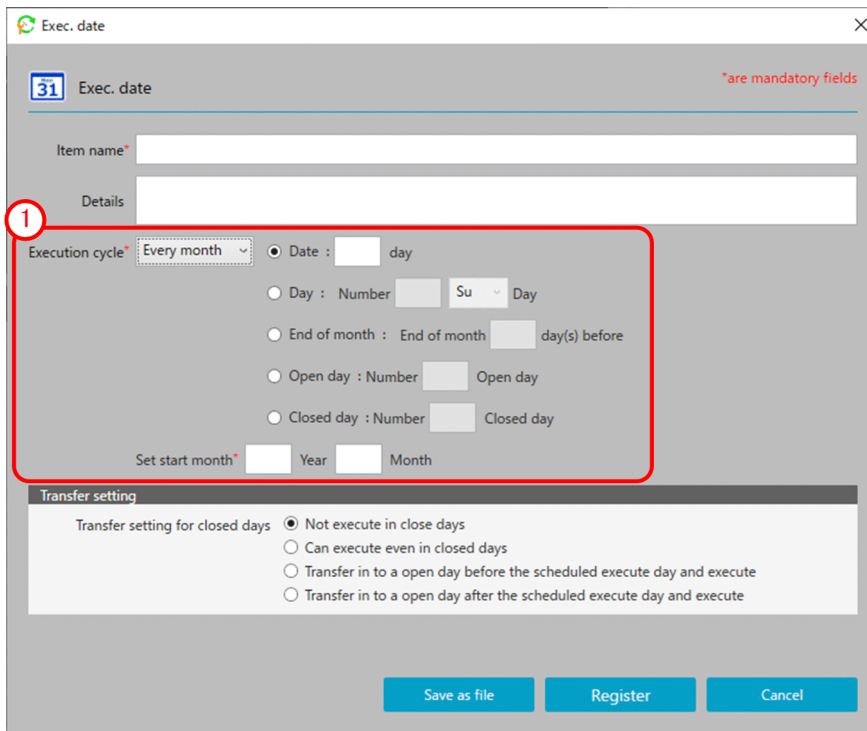
No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Day of the week	Pull down	Select from the following options: <b>Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, and Saturday.</b>	Optional	None
2	Set start day Year	Text box	Specify this setting by using a half-width number. You can specify a value in the range from 2013 to 2037.	Optional	None
3	Set start day Month	Text box	Specify this setting by using a half-width number. You can specify a value in the range from 1 to 12.	Optional	None
4	Set start day Number Day	Text box	Specify this setting by using a half-width number. You can specify a value in the range from 1 to 5.	Optional	None

### (3) Execution cycle (Every month)

If you have selected **Every month** as the execution cycle, set the date on which to execute the job every month as well as the month in which to start job execution.

The following figure shows what the window looks like when **Every month** is selected as the execution cycle.

Figure 7–22: What the window looks like when Every month is selected as the execution cycle



The following items are displayed:

**① Execution cycle (Every month)**

This area allows you to specify the conditions for starting the job execution cycle.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Execution date specification method	Radio button	Select from the following options: <b>Date, Day, End of month, Open day, and Closed day.</b>	Optional	None
2	Date	Text box	Specify this setting by using a half-width number. You can specify a value in the range from 1 to 31.	Optional	None
3	Day	Text box	Specify this setting by using a half-width number. You can specify a value in the range from 1 to 5.	Optional	None
4		Pull down	Select from the following options: <b>Su, Mo, Tu, We, Th, Fr, and Sa.</b>	Optional	None
5	End of month	Text box	Specify this setting by using a half-width number. You can specify a value in the range from 0 to 30.	Optional	None
6	Open day	Text box	Specify this setting by using a half-width number. You can specify a value in the range from 1 to 31.	Optional	None
7	Closed day	Text box	Specify this setting by using a half-width number. You can specify a value in the range from 1 to 31.	Optional	None
8	Set start month Year	Text box	Specify this setting by using a half-width number. You can specify a value in the range from 2013 to 2037.	Optional	None
9	Set start month Month	Text box	Specify this setting by using a half-width number. You can specify a value in the range from 1 to 12.	Optional	None

## (4) Execution cycle (Every year)

If you have selected **Every year** as the execution cycle, set the date on which to execute the job every year as well as the year in which to start job execution.

The following figure shows what the window looks like when **Every year** is selected as the execution cycle.

Figure 7–23: What the window looks like when Every year is selected as the execution cycle

The following items are displayed:

### ① Execution cycle (Every year)

This area allows you to specify the conditions for starting the job execution cycle.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Execution date specification method	Radio button	Select from the following options: <b>Date</b> , <b>Day</b> , <b>End of month</b> , <b>Open day</b> , and <b>Closed day</b> .	Optional	None
2	Date Month	Text box	Specify the month by using a value in the range from 1 to 12 and specify the date by using a value in the range from 1 to 31.	Optional	None
3	Day Month Number Day	Text box	Specify the month by using a value in the range from 1 to 12 and specify the day of the week by using a value in the range from 1 to 5.	Optional	None
4		Pull down	Select from the following options: <b>Su</b> , <b>Mo</b> , <b>Tu</b> , <b>We</b> , <b>Th</b> , <b>Fr</b> , and <b>Sa</b> .	Optional	None
5	End of month Month	Text box	Specify the month by using a value in the range from 1 to 12 and specify the day(s) by using a value in the range from 0 to 30.	Optional	None

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
	Number day(s) before				
6	Open day Month Number Open day	Text box	Specify the month by using a value in the range from 1 to 12 and specify the date by using a value in the range from 1 to 31.	Optional	None
7	Closed day Month Number Closed day	Text box	Specify the month by using a value in the range from 1 to 12 and specify the date by using a value in the range from 1 to 31.	Optional	None
8	Set start year Year	Text box	Specify this setting by using a half-width number. You can specify a value in the range from 2013 to 2037.	Optional	None

## (5) Execution cycle (Every operation day)

If you have selected **Every operation day** as the execution cycle, set the start date on which to start executing the job every open day.

The following figure shows what the window looks like when **Every operation day** is selected as the execution cycle.

Figure 7–24: What the window looks like when Every operation day is selected as the execution cycle

The following items are displayed:

### ① Execution cycle (Every operation day)

This area allows you to specify the conditions for starting the job execution cycle.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Set start day Year	Text box	Specify this setting by using a half-width number. You can specify a value in the range from 2013 to 2037.	Optional	None
2	Set start day Month	Text box	Specify this setting by using a half-width number. You can specify a value in the range from 1 to 12.	Optional	None
3	Set start day Number    Open day	Text box	Specify this setting by using a half-width number. You can specify a value in the range from 1 to 31.	Optional	None

## (6) Execution cycle (Every closed day)

If you have selected **Every closed day** as the execution cycle, set the start date on which to start executing the job every closed day.

The following figure shows what the window looks like when **Every closed day** is selected as the execution cycle.

Figure 7–25: What the window looks like when Every closed day is selected as the execution cycle

The following items are displayed:

### ① Execution cycle (Every closed day)

This area allows you to specify the conditions for starting the job execution cycle.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Set start day Year	Text box	Specify this setting by using a half-width number. You can specify a value in the range from 2013 to 2037.	Optional	None

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
2	Set start day Month	Text box	Specify this setting by using a half-width number. You can specify a value in the range from 1 to 12.	Optional	None
3	Set start day Number Closed	Text box	Specify this setting by using a half-width number. You can specify a value in the range from 1 to 31.	Optional	None

## (7) Execution cycle (No cycle)

If you have selected **No cycle** as the execution cycle, set the date on which to execute the job just once.

The following figure shows what the window looks like when **No cycle** is selected as the execution cycle.

Figure 7–26: What the window looks like when No cycle is selected as the execution cycle

The following items are displayed:

### ① Execution cycle (No cycle)

This area allows you to specify the conditions for starting the job execution cycle.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Set execution date	Pull down	Specify the start date in <i>yyyy/mm/dd</i> format. You can specify a value in the range from 2013/01/01 to 2037/12/31.	Optional	None



## 7.2.5 Exec. time dialog box

You can display the **Exec. time** dialog box by selecting the **Exec. time** tab.

The **Exec. time** dialog box allows you to define the time at which to execute the job.

The following figure shows the **Exec. time** dialog box.

Figure 7–27: Exec. time dialog box

The following items are displayed:

### ① Basic information area

This area allows you to enter the item name and specify an explanation of the item.

For details on the displayed items and the restrictions on the strings you can specify in this area, see *Basic information area* in [7.2.1 Action dialog box](#).

### ② Set execute time area

This area allows you to set the execute time.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Scheduled start time	Combo box	Specify the hour by using a value in the range from 00 to 23 and specify the minute by using a value in the range from 00 to 59.	Optional	None
2	Shift time	Combo box	Specify the hour by using a value in the range from 00 to 23 and specify the minute by using a value in the range from 00 to 59.	Optional	None
3	Time across day	Pull down	Select from the following options: <b>Previous day</b> and <b>Next day</b> .	Optional	None
4		Combo box	Specify the hour by using a value in the range from 00 to 23 and specify the minute by using a value in the range from 00 to 59.	Optional	None

### ③ Scheduled start time

If you select this radio button, specify the time at which the job is scheduled to start.

### ④ Shift time

If you select this radio button, specify the duration of time by which to delay the execution of the job relative to the start time of the operation time period specified in the calendar.

### ⑤ Time across day

By selecting this radio button, you can specify a certain time on the day before or after the day on which the job is scheduled for execution.

### ⑥ Repetition setting area

By selecting this check box, you can specify the interval at which to repeat the execution of the job after it is started.

The following table describes the items displayed and the restrictions on the strings you can specify in this area:

No.	Displayed item name	Input format	Restrictions on the strings you can specify	Mandatory or optional	Default value
1	Interval	Combo box	Specify the hour by using a value in the range from 00 to 23 and specify the minute by using a value in the range from 00 to 59.	Optional	None

### ⑦ Save as file button

This button saves the file containing the created execution time definition.

### ⑧ Register button

Registers execution time.

### ⑨ Cancel button

This button closes the window without registering the edited definition.

# 8

## Standard items

This chapter describes the standard items that you can use with CPA.

## 8.1 List of standard CPA items

The following tables describe and list standard CPA items.

Table 8–1: List of action items

No.	Item name	What the item does	Reference
1	Command execution	Executes actions using specified commands and parameters.	8.2.1
2	Command execution by list	Sets the strings extracted from the list file for the parameters of the execution command, and executes the command the number of times equivalent to the number of lines.	8.2.2
3	Command execution with recovery option	Executes the specified recovery execution command if the execution command fails with an error.	8.2.3
4	Command execution in window	Executes the command that displays a window.	8.2.4
5	Send mail	Sends emails to the specified email addresses.	8.2.5
6	Windows log on	Allows Windows users linked with the specified virtual users to log on to Windows or unlock the screen.	8.2.6

Table 8–2: List of event monitor items

No.	Item name	What the item does	Reference
1	Monitor file	Monitors file status.	8.3.1
2	Monitor any event	Executes a command periodically and monitors the return code of the command.	8.3.2
3	Monitoring mail reception	Monitors email reception.	8.3.3
4	Monitor Windows event	Monitors Windows events, including log-on, log-off, lock screen, and unlock screen events.	8.3.4

Table 8–3: List of execution condition items

No.	Item name	What the item does	Reference
1	Monitor file event	Executes the job when file status changes.	8.4.1
2	Monitor any event	Executes a command periodically and executes the job depending on the return code of the command.	8.4.2
3	Monitor the linkage from JP1AJS3	Executes jobs when the custom job for CPA linkage defined in AJS3 - Manager starts.	8.4.3
4	Event monitoring for receiving mail	Executes the job when an email that matches certain conditions is received.	8.4.4
5	Monitor Windows event	Executes jobs upon the occurrence of Windows events, including log-on, log-off, lock screen, and unlock screen events.	8.4.5

Table 8–4: List of execution date items

No.	Item name	What the item does
1	Execute in every Monday (later substitute)	Executes the job every Monday starting from the registered date. If a Monday falls on one of the closed days, the job is executed on an open day after the Monday.
2	Execute in every Friday (before substitute)	Executes the job every Friday starting from the registered date. If a Friday falls on one of the closed days, the job is executed on an open day before the Friday.

No.	Item name	What the item does
3	Execute in every Monday	Executes the job every Monday starting from the registered date. The job does not run on closed days.
4	Execute in every Tuesday	Executes the job every Tuesday starting from the registered date. The job does not run on closed days.
5	Execute in every Wednesday	Executes the job every Wednesday starting from the registered date. The job does not run on closed days.
6	Execute in every Thursday	Executes the job every Thursday starting from the registered date. The job does not run on closed days.
7	Execute in every Friday	Executes the job every Friday starting from the registered date. The job does not run on closed days.
8	Execute in every Saturday	Executes the job every Saturday starting from the registered date. The job does not run on closed days.
9	Execute in every Sunday	Executes the job every Sunday starting from the registered date. The job does not run on closed days.
10	Execute in end of every month	Executes the job on the last day of every month starting from the registered date. The job does not run on closed days.
11	Execute in first open day in all months	Executes the job on the first day of every month starting from the registered date.
12	Execute on last open day in every month	Executes the job on the last open day of every month starting from the registered date.

**Table 8–5: List of execution time items**

No.	Item name	What the item does
1	Execute from start time of ope. time period	Executes the job at the start time of the operation time period specified in the calendar.
2	Execute every hour within ope. time period	Executes the job at the start time of the operation time period specified in the calendar. Also, executes the job every hour until the end time has passed.
3	Execute every 2hours within ope. time period	Executes the job at the start time of the operation time period specified in the calendar. Also, executes the job every two hours until the end time has passed.
4	Execute every 6hours within ope. time period	Executes the job at the start time of the operation time period specified in the calendar. Also, executes the job every six hours until the end time has passed.

**Table 8–6: List of calendar items**

No.	Item name	What the item does
1	Saturday Sunday holidays	Specifies the calendar in which Saturdays and Sundays are holidays and the operation time period is set to 0:00 to 23:59 on that day.

## 8.2 Action items

---

### 8.2.1 Command execution

#### (1) Overview of the function

This item executes actions using specified commands and parameters.

#### (2) List of settings

Name	Required	Format	Can specify inherited information?
Execution command	Yes	String (1 to 512 bytes)	Yes
Parameter	No	String (0 to 512 bytes)	Yes
Virtual user name	No	String (0 to 512 bytes)	Yes (The profile information cannot be specified.)

#### (3) Settings

##### Execution command

Specify the file name of the command to execute.

A command is executed as a child process of the job execution service. When commands that require interactive operations are executed, those commands can be left running if you cannot perform operations in the window. (You can check this by the Task Manager or an alternative.) If you execute a command that requires interactive operations, use the command execution with the window item.

If specifying a file whose extension is other than `.exe` or `.bat`, you need to associate the file extension with an appropriate application.

##### Parameter

Specify the parameter for the execution command.

##### Virtual user name

Specify the name of the virtual user mapped to the name of the Windows user who executes the item. If the virtual user name is omitted, the item is executed with the account for the CPA job execution service.

To map the virtual user name, use the `cpasetumap` command. For details on the `cpasetumap` command, see [cpasetumap](#) in *9. Commands*.

#### (4) Return values

Value	Description
Other than 126	The return value of the execution command. A value other than 0 indicates an abnormal end.
126	The internal processing of CPA ended abnormally during startup of the execution command. Note: <ul style="list-style-type: none"><li>If an error occurs in the internal processing of CPA during startup of the execution command, 126 is set to the return value of the command. Check whether the specified execution command file exists. If it exists, check the message log and take action according to the message.</li></ul>

Value	Description
	<p>Note that if the execution command ends with the return value of 126, you cannot determine whether an error occurred in the internal processing or the execution command ended abnormally. We recommend that commands specified as the execution command return values other than 126.</p> <ul style="list-style-type: none"> <li>• If a virtual user name is specified for the item, fix the error by checking if: <ul style="list-style-type: none"> <li>- The virtual user name is specified properly.</li> <li>- The virtual user name is registered.</li> <li>- The Windows user associated with the virtual user name is correct.</li> <li>- The Windows user associated with the virtual user name has the correct user rights.</li> <li>- The password for the Windows user associated with the virtual user name has not expired.</li> <li>- The password for the Windows user associated with the virtual user name is correct (the account is not locked).</li> <li>- The Net Logon service has been running.</li> </ul> </li> </ul>

## (5) Example

1. Execute the `tasklist /V` command.

Execution command	<code>C:\Windows\System32\tasklist.exe</code>
Parameter	<code>/V</code>

2. Execute the `tasklist /V` command and output the results to `c:\temp\tasklist_work.txt`.

Execution command	<code>cmd.exe</code>
Parameter	<code>/C tasklist /V &gt;c:\temp\tasklist_work.txt</code>

## 8.2.2 Command execution by list

### (1) Overview of the function

This item sets the strings extracted from the list file for the parameters of the execution command, and executes the command the number of times equivalent to the number of lines.

### (2) List of settings

Name	Required	Format	Can specify inherited information?
Execution command	Yes	String (1 to 512 bytes)	Yes
Parameter	No	String (0 to 512 bytes)	Yes
List file	Yes	String (1 to 512 bytes)	Yes
Virtual user name	No	String (0 to 512 bytes)	Yes (The profile information cannot be specified.)

### (3) Settings

#### Execution command

Specify the file name of the command to execute.

A command is executed as a child process of the job execution service. When commands that require interactive operations are executed, those commands can be left running if you cannot perform operations in the window. (You can check this by the Task Manager or an alternative.) If you execute a command that requires interactive operations, use the command execution with the window item.

If specifying a file whose extension is other than `.exe` or `.bat`, you need to associate the file extension with an appropriate application.

### Parameter

Specify the parameter for the execution command.

When you specify values extracted from the list file as parameters, use the following variables.

Information	Setting value	Description
Entire line	<code>%r</code>	Expands the entire line.
The <i>n</i> th string in a comma-separated list	<code>%c<i>n</i></code>	Expands the <i>n</i> th element in a comma-separated list. Specify an element number for <i>n</i> (valid values are <code>%c1</code> to <code>%c9</code> ).
	<code>%~c<i>n</i></code>	Use this variable when you want to remove double quotation marks (") from element strings to be specified as parameters if they are enclosed in double quotations. (Strings do not necessarily contain double quotation marks.) Specify an element number for <i>n</i> (valid values are <code>%~c1</code> to <code>%~c9</code> ).

If you want to specify a percent sign (%), specify it as `%%`.

### List file

Specify the file name of the list file.

Use the following format for the list file:

- Line delimiter: Newline character (`\n`, `CR+LF` or `LF`)
- Element delimiter: Comma (`,`). Note that commas in double quotation marks are considered as a part of an element.
- The maximum number of elements in a line: 9. Any delimiters in the elements after the ninth element are considered as normal strings.
- Supported characters: Multibyte characters are also supported. Make sure that characters used match the OS locale.
- The maximum line length: 8,192 bytes
- The maximum number of lines: 1,024
- The maximum file size: 8,192 kilobytes

### Virtual user name

Specify the name of the virtual user mapped to the name of the Windows user who executes the item. If the virtual user name is omitted, the item is executed with the account for the CPA job execution service.

To map the virtual user name, use the `cpasetumap` command. For details on the `cpasetumap` command, see [cpasetumap](#) in *9. Commands*.

## (4) Return values

Value	Description
Other than 126	Return values of execution commands.
126	The internal processing of CPA ended abnormally during startup of the execution command.



Value	Description
	<p>Note:</p> <ul style="list-style-type: none"> <li>If an error occurs in the internal processing of CPA during startup of the execution command, 126 is set to the return value of the command. Check whether the specified execution command file exists. If it exists, check the message log and take action according to the message. Note that if the execution command ends with the return value of 126, you cannot determine whether an error occurred in the internal processing or the execution command ended abnormally. We recommend that commands specified as the execution command return values other than 126.</li> <li>If a virtual user name is specified for the item, fix the error by checking if: <ul style="list-style-type: none"> <li>The virtual user name is specified properly.</li> <li>The virtual user name is registered.</li> <li>The Windows user associated with the virtual user name is correct.</li> <li>The Windows user associated with the virtual user name has the correct user rights.</li> <li>The password for the Windows user associated with the virtual user name has not expired.</li> <li>The password for the Windows user associated with the virtual user name is correct (the account is not locked).</li> <li>The Net Logon service has been running.</li> </ul> </li> </ul>

## (5) Example

Execution command	cmd.exe
Parameter	/c xcopy /y %c1 c:\temp
List file	c:\temp\list.txt

Here are the contents of c:\temp\list.txt.

```
c:\1.txt,20150131,-wx
c:\2.txt,20150715,rw-
c:\3.txt,20151230,rwx
```

## (6) Command name that is output to log files

```
cpaactlist.exe
```

## 8.2.3 Command execution with recovery option

### (1) Overview of the function

This item executes the specified recovery execution command if the execution command fails with an error.

### (2) List of settings

Name	Required	Format	Can specify inherited information?
Execution command	Yes	String (1 to 512 bytes)	Yes
Parameter	No	String (0 to 512 bytes)	Yes
Recovery execution command	Yes	String (1 to 512 bytes)	Yes

Name	Required	Format	Can specify inherited information?
Parameter for recovery execution command	No	String (0 to 512 bytes)	Yes
Virtual user name	No	String (0 to 512 bytes)	Yes (The profile information cannot be specified.)

### (3) Settings

#### Execution command

Specify the file name of the command to execute.

A command is executed as a child process of the job execution service. When commands that require interactive operations are executed, those commands can be left running if you cannot perform operations in the window. (You can check this by the Task Manager or an alternative.) If you execute a command that requires interactive operations, use the command execution with the window item.

If specifying a file whose extension is other than `.exe` or `.bat`, you need to associate the file extension with an appropriate application.

#### Parameter

Specify the parameter for the execution command.

#### Recovery execution command

Specify the file name of the command to execute when the execution command ended abnormally.

For notes and other details, see the descriptions of *Execution command*.

#### Parameter for recovery execution command

Specify the parameter for the recovery execution command.

#### Virtual user name

Specify the name of the virtual user mapped to the name of the Windows user who executes the item. If the virtual user name is omitted, the item is executed with the account for the CPA job execution service.

To map the virtual user name, use the `cpasetumap` command. For details on the `cpasetumap` command, see [cpasetumap](#) in *9. Commands*.

### (4) Return values

Value	Description
Other than 126	Return values of execution commands. Return values of the recovery execution command if recovery is performed.
126	The internal processing of CPA ended abnormally during startup of the execution command or recovery execution command. Note: <ul style="list-style-type: none"> <li>If an error occurs in the internal processing of CPA during startup of the execution command or recovery execution command, 126 is set to the return value of the command. Check whether the specified execution command file or recovery execution command file exists. If it exists, check the message log and take action according to the message. Note that if the execution command or recovery execution command ends with the return value of 126, you cannot determine whether an error occurred in the internal processing or the execution command or recovery execution command ended abnormally. We recommend that commands specified as the execution command or recovery execution command return values other than 126.</li> <li>If a virtual user name is specified for the item, fix the error by checking if:</li> </ul>

Value	Description
	<ul style="list-style-type: none"> <li>- The virtual user name is specified properly.</li> <li>- The virtual user name is registered.</li> <li>- The Windows user associated with the virtual user name is correct.</li> <li>- The Windows user associated with the virtual user name has the correct user rights.</li> <li>- The password for the Windows user associated with the virtual user name has not expired.</li> <li>- The password for the Windows user associated with the virtual user name is correct (the account is not locked).</li> <li>- The Net Logon service has been running.</li> </ul>

## (5) Example

Execution command	cmd.exe
Parameter	/c if not exist c:\temp\hello.txt exit 1
Recovery execution command	cmd.exe
Parameter for recovery execution command	/c msg.exe# console /w Failure!

#  
 If you want to execute the command on a 64-bit operating system, specify the command as follows.  
 When the installation path of the Windows system is C:\Windows:  
 C:\Windows\sysnative\msg.exe

## (6) Command name that is output to log files

cpaactexec.exe
----------------

## 8.2.4 Command execution in window

### (1) Overview of the function

This item executes the command that displays a window.

### (2) List of settings

Name	Required	Format	Can specify inherited information?
Execution command	Yes	String (1 to 512 bytes)	Yes
Parameter	No	String (0 to 512 bytes)	Yes
Wait for end	No	Combo box ( <b>Wait for the window showing process to terminate</b> , or <b>Does not wait for the window showing process to terminate</b> )	No
Operation when terminates the window	No	Combo box ( <b>Always normal end</b> or <b>Normal end when the return code is 0</b> )	No
Timeout period(min.)	No	Numeric (0 to 1,440)	No

Name	Required	Format	Can specify inherited information?
Exclusive option	No	Check box ( <b>Do not execute concurrently among selected actions</b> )	No

### (3) Settings

#### Execution command

Specify the file name of the command that displays a window.

The command is executed with the privilege of the logged-on user. This means that the command ends abnormally if it is executed when no user logs on.

If specifying a file whose extension is other than `.exe` or `.bat`, you need to associate the file extension with an appropriate application.

#### Parameter

Specify the parameter for the command that displays a window.

#### Wait for end

Specify whether to wait for the completion of the command that displays a window.

#### Operation when the window is closed

Specify whether the command that displays a window ends normally regardless of the return value, or it ends normally if the return value is 0. For both options, the command will end abnormally if an error occurs in the internal processing of CPA.

This setting is available only if **Wait for end** is set to **Wait for the window showing process to terminate**.

#### Timeout period (min.)

Specify the timeout period for actions being executed. The valid values are 0 to 1,440 minutes. If omitted or set to 0, actions are not terminated. If **Do not execute the selected actions concurrently** is selected in **Exclusion option**, lock-release wait time is also included in the time period.

#### Exclusion option

Actions for which this check box is selected are never executed concurrently (only one of them is executed at a time). This exclusive option is the same as the one provided for the **Command execution in window** item, and exclusive control is commonly applied to all items for which one of these options is enabled.

Select this option if you use an item that should not be executed concurrently.

### (4) Return values

Value	Description
Other than 126	Return values of execution commands.
126	<p>The command ended abnormally in CPA internal processing when the execution command starts.</p> <p>Note:</p> <p>If an error occurs in CPA internal processing when the execution command starts, 126 is set to the return value of the command. Check if the file specified as the execution command exists. If it exists, check the message log and take actions according to the message.</p> <p>Note that if a command specified as the execution command ends with 126, you cannot determine whether an error occurred in internal processing or the execution command ended abnormally.</p> <p>We recommend that you use values other than 126 for return values of the command specified as the execution command.</p>

## (5) Example

Execution command	C:\Windows\system32\notepad.exe
Parameter	c:\temp\out.txt
Wait for end	<b>Wait for the window showing process to terminate</b>
Operation when the window is closed	<b>Always normal end</b>

## (6) Command name that is output to log files

cpaguiexec.exe

## 8.2.5 Send mail

### (1) Overview of the function

This item sends emails to the specified email addresses.

### (2) List of settings

Name	Required	Format	Can specify inherited information?
To:	Yes	Half-width alphanumeric characters and symbols (1 to 256 bytes)	Yes
Cc:	No	Half-width alphanumeric characters and symbols (0 to 256 bytes)	Yes
Subject	No	String (0 to 256 bytes)	Yes
Message text	No	String (0 to 512 bytes)	Yes
Attached file 1	No	String (0 to 255 bytes)	Yes
Attached file 2	No	String (0 to 255 bytes)	Yes
Attached file 3	No	String (0 to 255 bytes)	Yes
Attached file 4	No	String (0 to 255 bytes)	Yes
Attached file 5	No	String (0 to 255 bytes)	Yes
Attached file 6	No	String (0 to 255 bytes)	Yes
Attached file 7	No	String (0 to 255 bytes)	Yes
Attached file 8	No	String (0 to 255 bytes)	Yes

### (3) Settings

#### To:

Specify the To: address of the email.

**Cc:**

Specify the CC: address of the email.

**Subject**

Specify the subject of the email. If omitted, set to "(no subject)".

**Message text**

Write message text of the email.

Do not enter only line breaks and/or half-width spaces.

**(4) Return values**

Value	Description
-1	Forced termination.
0	Send email succeeded.
14	No attached file exists.
58	An incorrect value was specified. [Action to take] Use the valid values, and then configure the settings again.
71	Failed to access the mail server. [Action to take] Make sure that communications to the mail server can be made.
72	Access was denied by the mail server. [Action to take] Make sure that the settings are correct in firewalls or other equipment on the communication path to the mail server.
73	Name resolution failed for the host name of the mail server. [Action to take] Check if the host name of the mail server can be resolved using the <code>hosts</code> file or other means.
120	The size of the attached file is equal to or larger than 10 MB.
Others	Cannot access the system settings file ( <code>config.xml</code> ), insufficient resources, or high communication load. [Action to take] Confirm if CPA has been correctly installed, and investigate whether there could have been any problems with the OS environment when the item was executed.

**(5) Example**

To:	<code>XXX@YYY.com</code>
Cc:	<code>ZZZ@WWW.com</code>
Subject	Regular notification
Message text	This is a regular notification message for today.

**(6) Settings of send mail**

Before using the Send mail item, you need to create profiles used by the send mail function. For details on how to create profiles, see [2.3.1 Configurations for the Send mail item](#).

## (7) Error handling procedures

For details on error handling procedures, see the section describing return values. If the return value does not suggest any solutions, check the standard error output or error messages output to log files, and then take actions accordingly.

### 8.2.6 Windows log on

This item performs log-on to Windows or unlocks the screen. To use this item, you have to set it up.

To use this item, you have to first set up the Windows log-on function and map information regarding the users logging on to Windows (Windows user names and passwords) to virtual users. For details on how to set up, see [2.3.4 Using the Windows log-on function](#).

#### (1) List of settings

Name	Required	Format	Can specify inherited information?
Virtual user name	Yes	String (1 to 14 bytes)	No

#### (2) Settings

##### Virtual user name

Specify the virtual user name mapped to the name of the Windows user logging on to Windows. Use the `cpasetumap` command to map the virtual user name. For details on the `cpasetumap` command, see [cpasetumap](#) in [9. Commands](#).

#### (3) Return values

Value	Description
-1	Forced termination.
0	The log-on or unlock screen action was successful.
41	The log-on processing timed out. [Action to take] Check the state of the communication with the domain server as well as the availability of system resources, and then re-execute the action item.
46	Due to a password error or other mistake in the settings, the log-on action failed. [Action to take] Check if the password is correct, if it is valid, and if it has been set in the first place.
47	Another user was logged on. [Action to take] Re-execute the action item after the logged-on user logs off.
52	A permission error occurred. [Action to take] Check that <b>Local System account</b> is set to the account of the user who starts the job execution service.
76	The virtual user is not registered.

Value	Description
	[Action to take] Register a virtual user.
77	The Windows user does not exist. [Action to take] Check if the Windows user mapped with the virtual user exists.
125	A system error occurred. [Action to take] <ul style="list-style-type: none"> <li>• Check whether the machine was restarted after the setup of the Windows log-on function.</li> <li>• Check if any Windows user manually logged on to Windows after a failed attempt to execute the Windows log-on action.</li> <li>• Check if a Remote Desktop connection was established during the execution of the Windows log-on action.</li> <li>• Check if multiple Windows log-on actions were simultaneously executed.</li> </ul>

## (4) Example

In the following example, the Windows user name `UserA#` mapped to the virtual user name `logonuserA` is used to log on to Windows:

Virtual user name	<code>logonuserA</code>
-------------------	-------------------------

#

You have to first map the virtual user name `logonuserA` to a Windows user name by using the `cpasetumap` command.

## (5) Command name that is output to log files

<code>cpaalexec.exe</code>
----------------------------

## (6) How the item behaves in an environment used by multiple Windows users

In an environment where multiple Windows users are registered on a single machine, the item behaves as described in the table below depending on the timing of the execution of a Windows log-on action.

Table 8–7: How the item behaves when a Windows log-on action is executed in an environment used by multiple users

Status of the Windows user logging on to Windows	Status of other users	How the item behaves when a Windows log-on action is executed		
		Status (return value)	Behavior	Message ID
Log-off	Log-off	Normal end (0)	The user logs on.	--
	Log-on	Abnormal end (47)	The user does not log on.	KNAO2804-E
	Lock screen	Normal end (0)	The user logs on.	--
Log-on	Log-off	Normal end (0)	The user does not log on.	--
	Lock screen			



Status of the Windows user logging on to Windows	Status of other users	How the item behaves when a Windows log-on action is executed		
		Status (return value)	Behavior	Message ID
Lock screen	Log-off	Normal end (0)	Lock screen is canceled.	--
	Log-on	Abnormal end (47)	Lock screen is not canceled.	KNAO2804-E
	Lock screen	Normal end (0)	Lock screen is canceled.	--

Legend:

--: Not applicable

If an error occurs in any of the cases described in the table above, use the standard error output or the product log to check the message corresponding to the applicable message ID, confirm the log-on status of the Windows user, and then re-execute the Windows log-on action.

## (7) Log-on behavior

When using this item, keep the following in mind with regard to the Windows log-on behavior:

- When the user is already logged on to Windows with the Windows user name mapped to a virtual user name, a Windows log-on action ends successfully even when the password is incorrect.
- If a Windows log-on action fails for reasons other than the specification of a nonexistent virtual user name or Windows user name, the subsequent Windows log-on actions end abnormally without any log-on processing being executed. This precaution is taken to prevent the account from being locked or otherwise rendered inaccessible as a result of repeated execution of invalid log-on processing. After a failed Windows log-on action, all the subsequent Windows log-on actions end abnormally with a return value of 125 unless a Windows user succeeds in manually logging on to Windows. For this reason, after a failed Windows log-on action, a Windows user must manually log on to Windows and review the password setting before a Windows log-on action can be executed again.
- When the action item Command execution in window works normally, it means that the Windows log-on action has ended successfully.
- If a user belonging to a domain is used to log on to Windows, when a timeout occurs due to a network problem, the log-on action can fail with 41 set as a return value. When using such a user, it is necessary to first check the state of communication with the domain server.
- A log-on action executed while the Windows log-on window (tile) is not displayed fails. Set a schedule to ensure that the log-on action is executed while the log-on window is displayed.
- If a log-on action or lock screen action fails because the password of the user in question is expired or incorrect, a message indicating a failed log-on action shows up in the Windows log-on window (tile). This message, however, displays the name of the Windows user who executed the previous successful log-on action, instead of the name of the user whose log-on action or lock screen action just failed.
- If a Windows log-on action is executed either when there is a user using a Remote Desktop connection or in an environment where a Remote Desktop connection was used just moments before, the log-on action ends in failure. You have to use the Windows log-on function in an environment where no Remote Desktop connection is used.
- The Windows log-on function does not work normally if you use any one of the following functions to log on to Windows:
  - PIN
  - Picture password
  - Windows Hello
  - Azure AD account

- If the following maintenance information is output to a message displayed upon the execution of a Windows log-on action by a user having a Domain account, check if a connection is being made to the domain server or if the domain server is up and running.
  - Get Specified user SID error
  - Get Active Console user SID error

## 8.3 Event monitor items

### 8.3.1 Monitor file (event monitor item)

#### (1) Overview of the function

This item monitors file status.

#### (2) List of settings

Name	Required	Format	Can specify inherited information?
File name to be monitored	Yes	String (1 to 255 bytes) Note: This setting is case-insensitive, meaning that uppercase and lowercase characters are treated as if they are identical.	Yes
Monitoring condition	Yes	Combo box ( <b>Create, Delete, Change final write time, or Change size</b> )	No
Timeout period(min.)	No	Numeric (0 to 1440)	No
Virtual user name	No	String (0 to 512 bytes)	Yes (The profile information cannot be specified.)

#### (3) Settings

##### File name to be monitored

Specify the name of the file to be monitored.

##### Monitoring condition

Specify the monitoring conditions. The condition is met when the status of the monitored file changes to the one specified for this setting.

##### Timeout period(min.)

Specify the timeout period for the file monitor. If omitted or set to 0, the file monitor is not terminated.

When the timeout period is reached, the monitor ends abnormally (ENDE).

##### Virtual user name

Specify the name of the virtual user mapped to the name of the Windows user who executes the item. If the virtual user name is omitted, the item is executed with the account for the CPA job execution service.

To map the virtual user name, use the `cpasetumap` command. For details on the `cpasetumap` command, see [cpasetumap](#) in *9. Commands*.

#### (4) Return values

Value	Description
-1	Forced termination.
0	One of the following cases:

Value	Description
	<ul style="list-style-type: none"> <li>The monitored file is detected and file monitoring is finished.</li> <li>The timeout period is reached and thus monitoring is stopped.</li> <li>If a virtual user name is specified for the item, monitoring could not be performed because the virtual user name specified for the item is incorrect or it is not registered.</li> </ul> [Action to take] <ul style="list-style-type: none"> <li>Specify the correct virtual user name.</li> <li>Register the virtual user name.</li> </ul>
2	Insufficient resources or high processing load. [Action to take] Terminate any unnecessary applications, and execute again.
3	System call failed. [Action to take] Terminate any unnecessary applications, and execute again.
4	An incorrect value was specified. [Action to take] Use the valid values, and then configure the settings again.

## (5) Example 1

Detect an event when a certain file is created.

File name to be monitored	c:\temp\a.txt
Monitoring condition	<b>Create</b>

## (6) Example 2

Monitor updates of multiple files at a time

File name to be monitored	c:\temp\*.txt
Monitoring condition	<b>Change final write time</b>

## (7) Understanding when monitoring conditions are met

The file monitor obtains the status of the monitored file every 60 seconds, and compares the file status. When the file monitor finds a change that satisfies the monitoring conditions, it compares the status again 10 seconds later to check whether the update of the monitored file is completed. If no change is found in the monitored file at the time, then the file monitor detects the event. Thus, it can take up to 70 seconds from when the monitored file is changed until the event is detected. For deletion, it can take up to 60 seconds because the second comparison is not performed.

Suppose that monitoring starts without there being any monitored file and with **Change final write time** or **Change size** specified as a monitoring condition. In this case, even when a monitored file is created, the event is detected assuming that the monitoring condition has been met.

## (8) Monitoring intervals

As the file monitor checks the file status every 60 seconds, it obtains only the latest status if the monitored file has been changed multiple times within 60 seconds. For example, suppose you select **Create** for **Monitoring condition**. If a monitored file is created and immediately deleted within 60 seconds, the file monitor cannot find the file, and thus cannot detect it as an event.

## (9) Files to be monitored

Files are monitored by using the account of the user who is logged on to the CPA job execution service. This means that when the default local system account is used to log on to the job execution service, only local files can be monitored (network files cannot be monitored).

## (10) When specifying monitored files with wildcards

If monitoring is performed for the files specified with wildcards, only one event is raised regardless of how many files meet the condition.

If more than 10 files meet the condition, only the first 10 files are processed for event detection and other subsequent files are not processed. That is, the information of the subsequent files is not included in the inherited information. For details on inherited information, see [5.5.2 Inherited information](#). In addition, short file names are also monitored.

## (11) If files already exist when monitoring starts

When the file monitor job starts with **Monitoring condition** set to **Create**, it detects events even if the monitored files already exist.

### 8.3.2 Monitor any event (event monitor item)

#### (1) Overview of the function

This item executes a command periodically and monitors the return code of the command.

#### (2) List of settings

Name	Required	Format	Can specify inherited information?
Command name	Yes	String (1 to 255 bytes)	Yes
Parameter	No	String (0 to 512 bytes)	Yes
Return code regarding to monitor event	Yes	String (1 to 512 bytes) Format: <i>numeric-value</i> [ , <i>numeric-value</i> ... ] The valid values are 0 to 4,294,967,295.	Yes
Return code regarding to monitor abnormal end	No	String (0 to 512 bytes) Format: <i>numeric-value</i> [ , <i>numeric-value</i> ... ] The valid values are 0 to 4,294,967,295.	Yes
Timeout period(min.)	No	Numeric (0 to 1440)	No

Name	Required	Format	Can specify inherited information?
Virtual user name	No	String (0 to 512 bytes)	Yes (The profile information cannot be specified.)

### (3) Settings

#### Command name

Specify the file name of the command to be executed periodically for monitoring.

#### Parameter

Specify the parameter for the execution command.

#### Return code regarding to monitor event

Specify the return codes that are regarded as event detection. You can specify multiple return codes, separated by commas.

#### Return code regarding to monitor abnormal end

Specify the return codes that are regarded as abnormal ends. You can specify multiple return codes, separated by commas.

#### Timeout period(min.)

Specify the timeout period for the Monitor any event. If omitted or set to 0, the Monitor any event is not terminated. When the timeout period is reached, the monitor ends abnormally (ENDE).

#### Virtual user name

Specify the name of the virtual user mapped to the name of the Windows user who executes the item. If the virtual user name is omitted, the item is executed with the account for the CPA job execution service.

To map the virtual user name, use the `cpasetumap` command. For details on the `cpasetumap` command, see [cpasetumap](#) in *9. Commands*.

### (4) Return values

Value	Description
-1	Forced termination.
0	<p>One of the following cases:</p> <ul style="list-style-type: none"> <li>The monitored file is detected and file monitoring is finished.</li> <li>The timeout period is reached and thus monitoring is stopped.</li> <li>If a virtual user name is specified for the item, monitoring could not be performed because the virtual user name specified for the item is incorrect or it is not registered.</li> </ul> <p>[Action to take]</p> <ul style="list-style-type: none"> <li>Specify the correct virtual user name.</li> <li>Register the virtual user name.</li> </ul>
2	<p>Insufficient resources or high processing load.</p> <p>[Action to take]</p> <p>Check OS resources used when the item is executed.</p>
3	<p>System call failed.</p> <p>[Action to take]</p> <p>Confirm if CPA has been correctly installed, and investigate whether there could have been any problems with the OS environment when the item was executed.</p>

Value	Description
4	An incorrect parameter was specified. [Action to take] Use the valid values, and then configure the settings again.
21	The execution result of the command matched the setting values of the <b>Return code regarding to monitor abnormal end</b> , and the monitor event item ended.
22	An attempt to execute the specified command failed. [Action to take] <ul style="list-style-type: none"> <li>• Make sure that there is enough desktop heap available.</li> <li>• If a virtual user name is specified for the item, fix the error by checking if: <ul style="list-style-type: none"> <li>- The Windows user associated with the virtual user name has the correct user rights.</li> <li>- The password for the Windows user associated with the virtual user name has not expired.</li> <li>- The password for the Windows user associated with the virtual user name is correct (the account is not locked).</li> <li>- The Net Logon service has been running.</li> </ul> </li> </ul>

## (5) Example 1

Detect an event if a certain process exists. Otherwise, continue monitoring.

Command name	cmd
Parameter	/c tasklist findstr <i>process-name</i>
Return code regarding to monitor event	0
Return code regarding to monitor abnormal end	--

Legend:

-- : Not specified

## (6) Example 2

Detect an event if a certain process exists. Otherwise, end abnormally.

Command name	cmd
Parameter	/c tasklist findstr <i>process-name</i>
Return code regarding to monitor event	0
Return code regarding to monitor abnormal end	1

## (7) About return codes

If the same value is set to both of the **Return code regarding to monitor event** and the **Return code regarding to monitor abnormal end**, the item ends abnormally.

## 8.3.3 Monitoring mail reception (event monitor item)

### (1) Overview of the function

This item monitors reception of emails that match certain conditions.

### (2) List of settings

Name	Required	Format	Can specify inherited information?
Sender	No	String (0 to 256 bytes)	Yes
Subject	No	String (0 to 512 bytes)	Yes
Message text	No	String (0 to 512 bytes)	Yes
Attached file save destination	No	String (0 to 255 bytes)	Yes
Operations for received mail in the server	Yes	Combo box ( <b>Do Not Delete, Delete only mail that match with monitoring condition, or Delete</b> )	No
Mail reception interval (min.)	No	Numeric (1 to 1440)	No
Operations for attached files	Yes	Combo box ( <b>Receive attached files or Do not receive attached files</b> )	No
Timeout period(min.)	No	Numeric (0 to 1440)	No

### (3) Settings

#### Sender

Specify partial matches for the senders from whom emails are to be monitored. Regular expressions are not supported.

#### Subject

Specify partial matches for the subject lines with which emails are to be monitored. Regular expressions are not supported.

#### Message text

Specify partial matches for the message text with which emails are to be monitored. Regular expressions are not supported.

#### Attached file save destination

Specify the name of the folder where files attached to emails are stored.

If omitted, attached files are temporarily stored in the internal directory under the data path.

If **Operations for attached files** is set to **Do not receive attached files**, attached files are not stored even though you specify where to save attached files.

If the specified folder does not exist, it is created. If an attached file with the same name already exists, it is overwritten with a new attached file with that name.

#### Operations for received mail in the server

Specify whether to delete received emails from the mail server.



### Mail reception interval (min.)

Specify the interval in minutes with which the mail server is queried to check email reception. If omitted, set to 10 minutes.

### Operations for attached files

Specify whether to receive files attached to emails.

### Timeout period(min.)

Specify the timeout period for the monitoring mail reception. If omitted or set to 0, the monitoring mail reception is not terminated.

When the timeout period is reached, the monitor ends abnormally (ENDE).

## (4) Return values

Value	Description
-1	Forced termination.
0	Emails were received and the monitoring conditions were met, or the monitor timed out because timeout was reached.
1	Emails were received but the monitoring conditions were not met.
2	No email was received.
54	Cannot access the profile of mail reception ( <code>recvmailprof.ini</code> ). [Action to take] Make sure that the profile of mail reception ( <code>recvmailprof.ini</code> ) can be accessed.
58	An incorrect value was specified. [Action to take] Use the valid values, and then configure the settings again.
71	Failed to access the mail server. [Action to take] Make sure that communications to the mail server can be made.
72	Access was denied by the mail server. [Action to take] Make sure that the settings are correct in firewalls or other equipment on the communication path to the mail server.
73	Name resolution failed for the host name of the mail server. [Action to take] Check if the host name of the mail server can be resolved using the <code>hosts</code> file or other means.
75	The mail server denied the login. [Action to take] Make sure that the settings are configured so that you can log into the mail server.
105	The settings of the profile of mail reception ( <code>recvmailprof.ini</code> ) contain an error. [Action to take] Verify the settings of the profile of mail reception ( <code>recvmailprof.ini</code> ).
111	The profile of mail reception ( <code>recvmailprof.ini</code> ) does not exist. [Action to take] Create the profile of mail reception ( <code>recvmailprof.ini</code> ).
Others	Cannot access the system settings file ( <code>config.xml</code> ), insufficient resources, I/O error, or high communication load.

Value	Description
	[Action to take] Confirm if CPA has been correctly installed, and investigate whether there could have been any problems with the OS environment when the item was executed.

## (5) Example 1

Continue monitoring until an email is received from a certain mail sender.

Sender	testuser@example.com
Subject	--
Message text	--

Legend:

-- : Not specified

## (6) Example 2

Continue monitoring until an email with a certain subject and message text is received.

Sender	--
Subject	Important
Message text	Execute jobs.

Legend:

-- : Not specified

## (7) Behavior when settings are omitted

If all the settings of the sender, subject, and message text are omitted, all received emails are monitored.

## (8) Settings for receiving mail

Before using the Monitoring mail reception item, you need to create a profile to be used by the Monitoring mail reception item. For details on how to create a profile, see [2.3.2 Configurations for the Monitoring mail reception item](#).

### 8.3.4 Monitor Windows event (event monitor item)

#### (1) Overview of function

This item monitors Windows events.

#### (2) List of settings

Name	Required	Format	Can specify inherited information?
Monitor event	Yes	Combo box ( <b>Log on</b> , <b>Log off</b> , <b>Screen lock</b> , or <b>Screen lock release</b> )	No

Name	Required	Format	Can specify inherited information?
Windows User Name	No	String (0 to 276 bytes) You can use alphanumeric characters and the following symbols: _ - \ .	No
Timeout period(min.)	No	Numeric (0 to 1440)	No

### (3) Settings

#### Monitor event

Specify the types of Windows events to be monitored.

You can select from the following options: **Log on**, **Log off**, **Screen lock**, and **Screen lock release**.

#### Windows User Name

Specify the Windows user name to be monitored. If you do not specify this setting, all events are detected, regardless of who is triggering those events.

If a user belonging to a domain is to be monitored, specify *NetBIOS-domain-name\user-name*. If you omit the NetBIOS domain name, the item assumes that the local host name is set as the domain name.

If a Windows event is triggered by a user other than the specified user, monitoring continues without the event being detected.

The monitor does not distinguish between upper and lower case letters in the specified Windows user names.

#### Timeout period(min.)

Specify the timeout period for the Windows event monitor. If omitted or set to 0, the Windows event monitor is not terminated.

When the timeout period is reached, the monitor ends abnormally (ENDE).

### (4) Return values

Value	Description
-1	Forced termination.
0	The event being monitored was detected, and the Windows event monitor ended, or the monitor timed out because timeout was reached.
2	Insufficient resources or high processing load. [Action to take] Check OS resources used when the item is executed.
3	System call failed. [Action to take] Confirm if CPA has been correctly installed, and investigate whether there could have been any problems with the OS environment when the item was executed.
21	The event ended abnormally due to one of the following reasons: <ul style="list-style-type: none"> <li>• A permission denied error occurred.</li> <li>• The specified information is incorrect.</li> <li>• A system error occurred.</li> </ul>
22	An attempt to execute the Windows event monitor failed. [Action to take] Make sure that there is enough desktop heap available.

## (5) Example 1

Continue monitoring until a user logs on to Windows.

Monitor event	Log on
Windows User Name	--

Legend:

-- : Not specified

## (6) Example 2

Continue monitoring until the designated user (UserA) unlocks the Windows screen.

Monitor event	Screen lock release
Windows User Name	UserA

## (7) Notes

You can only use this item while the COM+ Event System service, the COM+ System Application service, and System Event Notification Service service are up and running. For details, see [2.3.5 Using the Windows event monitor function](#).

Note that, if you stop the above services while this item is being executed, events can no longer be detected, although event monitoring continues. In this state, events cannot be detected even when the services are restarted. You have to re-execute the item.

## 8.4 Execution condition items

---

### 8.4.1 Monitor file event (execution condition item)

#### (1) Overview of the function

This item executes jobs when file status changes.

#### (2) List of settings

Name	Required	Format	Can specify inherited information?
File name to be monitored	Yes	String (1 to 255 bytes)	No
Monitoring condition	Yes	Combo box ( <b>Create</b> , <b>Delete</b> , <b>Change final write time</b> , or <b>Change size</b> )	No
Virtual user name	No	String (0 to 512 bytes)	No

#### (3) Settings

##### File name to be monitored

Specify the name of the file to be monitored.

##### Monitoring condition

Specify the monitoring conditions. The condition is met when the status of the monitored file changes to the one specified for this setting.

##### Virtual user name

Specify the name of the virtual user mapped to the name of the Windows user who executes the item. If the virtual user name is omitted, the item is executed with the account for the CPA job execution service.

To map the virtual user name, use the `cpasetumap` command. For details on the `cpasetumap` command, see [cpasetumap](#) in *9. Commands*.

#### (4) Example 1

Execute the job when a certain file is created.

File name to be monitored	c:\temp\a.txt
Monitoring condition	<b>Create</b>

#### (5) Example 2

Execute the job when the last write time of one of multiple files changes.

File name to be monitored	c:\temp\*.txt
Monitoring condition	<b>Change final write time</b>

## (6) Understanding when monitoring conditions are met

The file monitor obtains the status of the monitored file every 60 seconds, and compares the file status. When the file monitor finds a change that satisfies the monitoring conditions, it compares the status again 10 seconds later to check whether the update of the monitored file is completed. If no change is found in the monitored file at the time, then the file monitor detects the event. Thus, it can take up to 70 seconds from when the monitored file is changed until the event is detected. For deletion, it can take up to 60 seconds because the second comparison after 10 seconds is not performed.

Suppose that monitoring starts without there being any monitored file and with **Change final write time** or **Change size** specified as a monitoring condition. In this case, even when a monitored file is created, the event is detected assuming that the monitoring condition has been met.

## (7) Monitoring intervals

As the file monitor checks the file status every 60 seconds, it obtains only the latest status if the monitored file has been changed multiple times within 60 seconds. For example, suppose you select **Create** for **Monitoring condition**. If a monitored file is created and immediately deleted within 60 seconds, the file monitor cannot find the file, and thus cannot detect it as an event.

## (8) Files to be monitored

Files are monitored by using the account of the user who is logged on to the CPA job execution service. This means that when the default local system account is used to log on to the job execution service, only local files can be monitored (network files cannot be monitored).

## (9) When specifying monitored files with wildcards

If monitoring is performed for the files specified with wildcards, only one event is raised regardless of how many files meet the condition.

If more than 10 files meet the condition, only the first 10 files are processed for event detection and other subsequent files are not processed. That is, the information of the subsequent files is not included in the inherited information. For details on inherited information, see [5.5.2 Inherited information](#). In addition, short file names are also monitored.

## (10) Monitoring after restart

If the file monitor restarts the job execution service during monitoring, the file monitor carries over the status of the files before the service is stopped to monitor the files.

## (11) If files already exist when monitoring starts

When the file monitor job starts with **Monitoring condition** set to **Create**, it detects events even if the monitored files already exist. After the job execution service is restarted, such events are no longer detected because the file monitoring job carries over the status of the files before the service is stopped.

## 8.4.2 Monitor any event (execution condition item)

### (1) Overview of the function

This item executes a command periodically and executes the job depending on the return code of the command.

## (2) List of settings

Name	Required	Format	Can specify inherited information?
Command name	Yes	String (1 to 255 bytes)	No
Parameter	No	String (0 to 512 bytes)	No
Return code regarding to monitor event	Yes	String (1 to 512 bytes) Format: <i>numeric-value</i> [ , <i>numeric-value</i> ...] The valid values are 0 to 4,294,967,295.	No
Does not detect when the return code is the same as last time	No	Check box	No
After detecting events, start next monitor immediately	No	Check box	No
Return code regarding to monitor abnormal end	No	String (0 to 512 bytes) Format: <i>numeric-value</i> [ , <i>numeric-value</i> ...] The valid values are 0 to 4,294,967,295.	No
Virtual user name	No	String (0 to 512 bytes)	No

## (3) Settings

### Command name

Specify the file name of the command to be executed periodically for monitoring.

### Parameter

Specify the parameter for the execution command.

### Return code regarding to monitor event

Specify the return codes that are regarded as event detection, which triggers action execution. You can specify multiple return codes, separated by commas.

### Does not detect when the return code is the same as last time

Select this check box if you do not want CPA to detect an event if the return code of the command is the same as that of the previous monitoring. CPA still detects an event if the return code of the first execution matched the *return code regarding to monitor event*. If you select this option, you cannot select the **After detecting events, start next monitor immediately** option.

### After detecting events, start next monitor immediately

Select this check box if you want CPA to execute the command immediately after detecting an event, without a given interval. If you select this option, you cannot select the **Does not detect when the return code is the same as last time** option.

### Return code regarding to monitor abnormal end

Specify the return codes with which the *job executed on event* ends abnormally. You can specify multiple return codes, separated by commas.

### Virtual user name

Specify the name of the virtual user mapped to the name of the Windows user who executes the item. If the virtual user name is omitted, the item is executed with the account for the CPA job execution service.

To map the virtual user name, use the `cpasetumap` command. For details on the `cpasetumap` command, see [cpasetumap](#) in *9. Commands*.

## (4) Example 1

Execute the job if a certain process exists. Otherwise, continue monitoring.

Command name	cmd
Parameter	/c tasklist findstr <i>process-name</i>
Return code regarding to monitor event	0
Return code regarding to monitor abnormal end	--
Does not detect when the return code is the same as last time	Cleared
After detecting events, start next monitor immediately	Cleared

Legend:

-- : Not specified

## (5) Example 2

Detect an event if a certain process exists. Otherwise, end abnormally.

Command name	cmd
Parameter	/c tasklist findstr <i>process-name</i>
Return code regarding to monitor event	0
Return code regarding to monitor abnormal end	1
Does not detect when the return code is the same as last time	Cleared
After detecting events, start next monitor immediately	Cleared

## (6) Example 3

Detect an event if a certain process exists. The item does not detect an event even if the process still exists during the second and later monitoring.

Command name	cmd
Parameter	/c tasklist findstr <i>process-name</i>
Return code regarding to monitor event	0
Return code regarding to monitor abnormal end	--
Does not detect when the return code is the same as last time	Checked
After detecting events, start next monitor immediately	Cleared

Legend:

-- : Not specified

## (7) Example 4

Execute the job every hour.

Command name	sleep
Parameter	3600



Return code regarding to monitor event	0
Return code regarding to monitor abnormal end	--
Does not detect when the return code is the same as last time	Cleared
After detecting events, start next monitor immediately	Checked

Legend:

-- : Not specified

## (8) About return codes

If the same value is set to both of the **Return code regarding to monitor event** and the **Return code regarding to monitor abnormal end**, the item ends abnormally.

## (9) About the setting of "Does not detect when the return code is the same as last time"

If **Does not detect when the return code is the same as last time** is selected, the latest return code is compared with the return code of the previous monitoring. This means that the return code is retained and carried over for each monitoring interval. The return code retained is reset when:

- You change the definition of the job in the Job design view.
- You register the job or perform overwrite registration for the job in the following order:
  - Select the **Does not detect when the return code is the same as last time** check box and register the job.
  - Clear the **Does not detect when the return code is the same as last time** check box and perform overwrite registration for the job.
  - Select the **Does not detect when the return code is the same as last time** check box and perform overwrite registration for the job.

## (10) About the setting of "After detecting events, start next monitor immediately"

With the Monitor any event item, commands are executed every 60 seconds. However, if **After detecting events, start next monitor immediately** is selected, a command runs again immediately after it ends. This means that if you specify a command that takes a very short time to finish for the Monitor any event item, the command runs consecutively when the return codes of the command always satisfy conditions. This imposes a high load on the machine. This also might cause the Job Design View to fail to update the status information in real time. The window might display inconsistent information, for example, a job is in the Executing state but the end time appears. In such a case, you should wait until the peak period of events passes and then update the status information in the window. Alternatively, you can use the [cpareport](#) command described in [9. Commands](#) to view the status information. We recommend that you select **After detecting events, start next monitor immediately** only if you use a command that does not finish until an event is detected, or if you want to perform monitoring repeatedly for a certain return code.

## 8.4.3 Monitor the linkage from JP1AJS3 (execution condition item)

### (1) Overview of the function

This item executes a job when a custom job (custom job for CPA linkage) defined in AJS3 - Manager starts. To use this item, see [4.3 Using the AJS3 linkage function](#).

### (2) List of settings

None

### (3) Settings

None

### (4) Command name that is output to log files

```
cpalink.exe
```

### (5) About the number of concurrent executions and the maximum number of connections

- The maximum number of custom jobs for CPA linkage that can be concurrently executed from AJS3: 16 jobs<sup>#</sup>
- The maximum number of concurrent connections from CPA Option for AJS3 to CPA: 64 hosts

#

If the names of the jobs executed on CPA are the same, up to 16 instances of the job can be executed. CPA can store up to 16 events and thus subsequent events are discarded.

### (6) Values returned to AJS3 custom jobs

Depending on the *Ended* state of the CPA job, one of the following values is returned to the AJS3 custom job (custom job for CPA linkage):

- Ended normally on CPA:0
- Ended with warning on CPA:1
- Ended abnormally on CPA:2 or greater value

## 8.4.4 Event monitoring for receiving mail (execution condition item)

### (1) Overview of the function

This item executes the job when an email that matches certain conditions is received.

## (2) List of settings

Name	Required	Format	Can specify inherited information?
Sender	No	String (0 to 256 bytes)	Yes
Subject	No	String (0 to 512 bytes)	Yes
Message text	No	String (0 to 512 bytes)	Yes
Attach file save destination	No	String (0 to 255 bytes)	Yes
Operations for received mail in the server	Yes	Combo box ( <b>Not Delete, Delete only mail that match with monitoring condition, or Delete</b> )	No
Mail reception interval (min.)	No	Numeric (1 to 1440)	No
Operations for attach files	Yes	Combo box ( <b>Receive attached files or Do not receive attached files</b> )	No

## (3) Settings

### Sender

Specify partial matches for the senders from whom emails are to be monitored. Regular expressions are not supported.

### Subject

Specify partial matches for the subject lines with which emails are to be monitored. Regular expressions are not supported.

### Message text

Specify partial matches for the message text with which emails are to be monitored. Regular expressions are not supported.

### Attach file save destination

Specify the name of the folder where files attached to emails are stored.

If omitted, attached files are temporarily stored in the internal directory under the data path.

If **Operations for attached files** is set to **Do not receive attached files**, attached files are not stored even though you specify where to save attached files.

If the specified folder does not exist, it is created. If an attached file with the same name already exists, it is overwritten with a new attached file with that name.

### Operations for received mail in the server

Specify whether to delete received emails from the mail server.

### Mail reception interval (min.)

Specify the interval in minutes with which the mail server is queried to check email reception. If omitted, set to 10 minutes.

### Operations for attach files

Specify whether to receive files attached to emails.

## (4) Example 1

The job is executed when an email from a certain user is received.

Sender	testuser@example.com
Subject	--
Message text	--

Legend:

-- : Not specified

## (5) Example 2

The job is executed when an email with a certain subject and message text is received.

Sender	--
Subject	Important
Message text	Execute jobs.

Legend:

-- : Not specified

## (6) Behavior when settings are omitted

If all the settings of the sender, subject, and message text are omitted, all received emails are monitored.

## (7) Settings for receiving mail

Before using the Monitoring mail reception item, you need to create a profile to be used by the Monitoring mail reception item. For details on how to create a profile, see [2.3.2 Configurations for the Monitoring mail reception item](#).

### 8.4.5 Monitor Windows event (execution condition item)

#### (1) Overview of function

This item executes a job upon the occurrence of a Windows event.

#### (2) List of settings

Name	Required	Format	Can specify inherited information?
Monitor event	Yes	Combo box ( <b>Log on</b> , <b>Log off</b> , <b>Screen lock</b> , or <b>Screen lock release</b> )	No
Windows User Name	No	String (0 to 276 bytes) You can use alphanumeric characters and the following symbols: _ - \ .	No

#### (3) Settings

##### Monitor event

Specify the types of Windows events to be monitored.

You can select from the following options: **Log on**, **Log off**, **Screen lock**, and **Screen lock release**.

After the detection of an event, event monitoring immediately resumes without any intervals between monitoring jobs.

### Windows User Name

Specify the Windows user name to be monitored. If you do not specify this setting, all events are detected, regardless of who is triggering those events.

If a user belonging to a domain is to be monitored, specify *NetBIOS-domain-name\user-name*. If you omit the NetBIOS domain name, the item assumes that the local host name is set as the domain name.

If a Windows event is triggered by a user other than the specified user, monitoring continues without the event being detected.

The monitor does not distinguish between upper and lower case letters in the specified Windows user names.

## (4) Example 1

Run the job when a user logs on to Windows.

Monitor event	Log on
Windows User Name	--

Legend:

-- : Not specified

## (5) Example 2

Run the job when the designated user (UserA) unlocks the Windows screen.

Monitor event	Screen lock release
Windows User Name	UserA

## (6) Notes

You can only use this item while the COM+ Event System service, the COM+ System Application service, and System Event Notification Service service are up and running. For details, see [2.3.5 Using the Windows event monitor function](#).

Note that, if you stop the above services while this item is being executed, events can no longer be detected, although event monitoring continues. In this state, events cannot be detected even when the services are restarted. You have to re-execute the item.

# 9

## Commands

This chapter describes commands that you can use with CPA and CPA Option for AJS3.

## Command notation conventions

---

The following table lists notation conventions to describe commands and parameters.

Table 9–1: Notation conventions to describe commands and parameters

Notation	Explanation and example
 (stroke)	Separates multiple items, any of which can be selected. Example: A   B   C means A or B or C.
{ } (curly braces)	Indicates that one of the items enclosed in the curly braces must be selected. The items are separated by the stroke ( ) Example: { A   B   C } means that either A or B or C must be specified.
[ ] (square brackets)	Indicates optional items (items whose specification may be omitted). Example: [ A ] means that A may be specified if needed (A may be omitted if not required). [ B   C ] means that B or C may be specified as needed or that B and C may both be omitted.
. . . (ellipsis)	The item preceding the ellipsis can be specified as many times as needed. When an item is specified multiple times, the blank is used as the delimiter. Example: A B . . . means that B may be specified as many times as necessary following A.
<u>    </u> (underline)	Indicates the default value that is assumed by the system if all of the items surrounded by square brackets are omitted. Example: [ <u>A</u>   B ] means that the system will use A if neither A nor B is specified.
( ( ) ) (double parentheses)	Indicates the range of specifiable values.

## List of commands

The following table lists commands provided with CPA.

Table 9–2: List of CPA commands

No.	Category	Command	Description
1	Job definition	<code>cpajobdesign</code>	Starts the Job Design View.
2		<code>cpajobentry</code>	Registers a job or items.
3		<code>cpajobdelete</code>	Deletes a job or items.
4	Job operation	<code>cpajobstop</code>	Stops a job.
5		<code>cpajobrun</code>	Reruns a job.
6		<code>cpareport</code>	Sends information to standard output.
7		<code>cpajobprint</code>	Outputs the definition information for a job or item to a file.
8	Configuration	<code>cpapasswd</code>	Encrypts passwords and adds them to the mail profiles so that the Send mail and Monitoring mail reception items can use them.
9		<code>cpasetprof</code>	Configures profile items in the profile.
10		<code>cpasetfirewall</code>	Configures or reverts the Windows Firewall settings.
11		<code>cpaalsetup</code>	Sets up the Windows log-on function.
12		<code>cpasetumap</code>	Registers the following information in the user mapping file ( <code>usrprof.xml</code> ): CPA virtual users used with the Windows log-on function or execution user setting function and information regarding the users to be mapped to the virtual users (Windows user names and passwords).
13		<code>cpausrenvsetup</code>	Deletes the tabs for defining action flows and calendars from the Job Design View so as to set up an operating environment in which only items are available for use.
14		<code>cpaitemsetup</code>	Sets up the custom item creation environment.
15	Other	<code>cpaecho</code>	Sends a string excluding line breaks to standard output.
16		<code>cpasleep</code>	Suspends the execution of the job for a specified time period.

The following table lists commands provided with CPA Option for AJS3.

Table 9–3: List of CPA Option for AJS3 commands

No.	Category	Command	Description
1	Configuration	<code>cpasetfirewall</code>	Configures or reverts the Windows Firewall settings.



# cpajobdesign

---

## Function

Starts the Job Design View.

## Format

```
cpajobdesign
```

## Permission for execution

None

## Arguments

None

## Return values

None

## Environment requirements

This command can run regardless of whether the job execution service is running or stopped.

## Command stored in:

*install-path*\bin

Note that the command is stored in the following folder by default.

In a 32-bit operating system:

```
%ProgramFiles%\Hitachi\JP1CPA\bin
```

In a 64-bit operating system:

```
%ProgramFiles(x86)%\Hitachi\JP1CPA\bin
```

## Notes

- Do not start more than one Job Design View to edit the same job or item concurrently. When concurrent changes are made to a job or item, the later change overwrites the earlier one.
- To run this command, you must first install .NET Framework 4.5 framework.

# cpajobentry

---

## Function

Registers a job or items.

## Format

```
cpajobentry -a definition-file-name [-u]
```

## Permission for execution

None

## Arguments

### -a *definition-file-name*

Specify the absolute path to the definition file you want to register.

Note that the definition file is a local file specified in **Save job file** or **Save as file** of the Job Design View, or a file that stores the output result of the `cpajobprint` command.

### -u

Updates an existing job or items.

An item created in the Item Design View can also be updated.

An update you made to a job is applied when the job runs next time.

This command returns an error when the specified job is running.

## Return values

0	Succeeded
40	Insufficient resources
52	Could not access the file.
58	Syntax error in arguments
63	Cannot update because the job is running.
66	There is already a job or item with the same name.
85	The number of items to be registered exceeds the maximum allowed number.
96	The specified XML file has an error.
111	The specified file does not exist.
125	System error
126	Program error

## Environment requirements

This command can run regardless of whether the job execution service is running or stopped.

## Command stored in:

*install-path*\bin

Note that the command is stored in the following folder by default.

In a 32-bit operating system:

```
%ProgramFiles%\Hitachi\JP1CPA\bin
```

In a 64-bit operating system:

```
%ProgramFiles(x86)\Hitachi\JP1CPA\bin
```

## Notes

If the job execution service terminates due to power shutdown or other reasons, you should not run this command with the `-u` option to update an existing job until the job execution service restarts (that is, while the service is stopped). Doing so might prevent the state restoration process of the service from working properly when the service restarts.

# cpajobdelete

---

## Function

Deletes an existing job or items.

## Format

```
cpajobdelete -c {jds -i job-name
| action_group -a action-flow-item-name
| calendar -a calendar-item-name
}
```

## Permission for execution

None

## Arguments

`-c jds -i job-name`

Deletes the specified job. If the job is running, the job is stopped forcibly and then deleted.

A job name can include multibyte characters, alphanumeric characters, and underscores ( `_` ) and the length must be from 1 to 45 bytes.

`-c action_group -a action-flow-item-name`

Deletes the definition of an action flow. If the specified action flow is referenced by an existing *job executed on time*, an error occurs when the job runs next time.

An item name can include multibyte characters, alphanumeric characters, and underscores ( `_` ) and the length must be from 1 to 45 bytes.

`-c calendar -a calendar-item-name`

Deletes the definition of a calendar. If the specified calendar is referenced by an existing *job executed on time*, an error occurs when the schedule is updated next time.

An item name can include multibyte characters, alphanumeric characters, and underscores ( `_` ) and the length must be from 1 to 45 bytes.

## Return values

0	Succeeded
40	Insufficient resources
52	Could not access the file.
58	Syntax error in arguments
63	Cannot delete because the job is running.
66	The specified job or item does not exist.
124	Failed to delete an internal file.
125	System error
126	Program error

## Environment requirements

This command can run regardless of whether the job execution service is running or stopped.

### Command stored in:

*install-path*\bin

Note that the command is stored in the following folder by default.

In a 32-bit operating system:

%ProgramFiles%\Hitachi\JP1CPA\bin

In a 64-bit operating system:

%ProgramFiles(x86)%\Hitachi\JP1CPA\bin

# cpajobstop

---

## Function

Suspend: Waits for all running actions to finish and then stops the job.

Forced stop: Forcibly stops all running actions and then stops the job.

Stop monitoring: Stops monitoring whether execution conditions are met for the job executed on event.

## Format

```
cpajobstop [{-c|-t|-e}] -i job-name [-o job-execution-ID]
```

## Permission for execution

None

## Arguments

-c

Waits for all running actions to finish and then stops the job.

This command cannot work unless the specified job is running.

-t

Forcibly stops all running actions and then stops the job.

-e

Stops monitoring whether execution conditions are met for a job executed on event when the job is in the Monitoring for event state (MONITORING) or the Overflowing events state (MONITORING).

This option is allowed only for jobs executed on event.

Regardless of whether a job execution ID is specified with -o, this option can stop monitoring whether execution conditions are met but cannot stop a running job. Specifying an invalid job execution ID results in an error.

To stop *jobs executed on event* that are in the Executing (RUN) state, use the -c or -t option with the -o option.

-i *job-name*

Specify the job name of the job to stop.

A job name can include multibyte characters, alphanumeric characters, and underscores (\_) and the length must be from 1 to 45 bytes.

-o *job-execution-ID*

Specify the ID of the job execution unit you want to stop. Omitting this option causes the system to use the ID of the running job execution unit.

A job execution ID can include alphanumeric characters and hyphens (-) and the length must be from 1 to 40 bytes.

You can use the `cpareport` command to see job execution IDs. For details, see the description in [cpareport](#).

## Return values

0	Succeeded
11	Nothing is done because the specified job is stopped.
40	Insufficient resources

45	System error (temporary exclusion error)
50	Not allowed to access the system settings file ( <code>config.xml</code> ).
52	There is no appropriate privilege for the executable file.
58	Syntax error in arguments
65	Cannot stop monitoring because the specified job is a job executed on time.
67	The specified job or job execution ID does not exist.
82	The job is not running.
94	The system settings file ( <code>config.xml</code> ) is invalid.
96	The job or item is invalid.
109	The system settings file ( <code>config.xml</code> ) does not exist.
111	System error (file open error)
119	System error (Too many connections were attempted for the message communication service.)
124	System error (I/O error)
125	System error or other error
126	Program error

## Environment requirements

This command can run only when the job execution service is running.

### Command stored in:

`install-path\bin`

Note that the command is stored in the following folder by default.

In a 32-bit operating system:

`%ProgramFiles%\Hitachi\JP1CPA\bin`

In a 64-bit operating system:

`%ProgramFiles(x86)\Hitachi\JP1CPA\bin`

## Notes

If the job status is already *Waiting for start* (QUEUEING), you cannot stop the job in question by running the `cpajobstop` command. In this case, wait until the job status changes to *Running* (RUN) and then use the `cpajobstop` command to stop the job in question, or delete the job by either operating from the Job Design View or executing the `cpajobdelete` command.

# cpajobrun

---

## Function

Reruns a finished job.

## Format

```
cpajobrun -r -i job-name [-o job-execution-ID-in-ended-state] {-t|-l|-n}
```

## Permission for execution

None

## Arguments

-r

Reruns a job that is in an Ended state. This option is allowed only for jobs executed on time.

-i *job-name*

Specify the job name of the job to rerun.

A job name can include multibyte characters, alphanumeric characters, and underscores (\_) and the length must be from 1 to 45 bytes.

-o *job-execution-ID-in-ended-state*

Specify the ID of the job execution unit you want to rerun. Omitting this option causes the system to rerun the last job execution unit.

A job execution ID can include alphanumeric characters and hyphens (-) and the length must be from 1 to 40 bytes. You can use the `cpareport` command to see job execution IDs. For details, see the description in [cpareport](#).

-t

Reruns the job from the top.

-l

Reruns the job from the action with the status of Ended `abnormally` or Ended `abnormally (Cancel)`.

-n

Changes the action with the status of Ended `abnormally` or Ended `abnormally (Cancel)` to the Ended `normally` status and reruns the job from the next action.

## Return values

0	Succeeded
11	Nothing is done because the specified job is stopped.
40	Insufficient resources
45	System error (temporary exclusion error)
50	Not allowed to access the system settings file ( <code>config.xml</code> ).
52	There is no appropriate privilege for the executable file.
58	Syntax error in arguments
65	Cannot rerun the specified job because it is a job executed on event.



67	The specified job or job execution ID does not exist.
79	Could not rerun the job because it was not in an Ended state.
80	Tried to rerun the job when it was already rerunning.
94	The system settings file ( <code>config.xml</code> ) is invalid.
96	The job or item is invalid.
109	The system settings file ( <code>config.xml</code> ) does not exist.
111	System error (file open error)
119	System error (Too many connections were attempted for the message communication service.)
124	System error (I/O error)
125	System error or other error
126	Program error

## Environment requirements

This command can run only when the job execution service is running.

### Command stored in:

*install-path*\bin

Note that the command is stored in the following folder by default.

In a 32-bit operating system:

`%ProgramFiles%\Hitachi\JP1CPA\bin`

In a 64-bit operating system:

`%ProgramFiles(x86)%\Hitachi\JP1CPA\bin`

## Notes

This rerun command uses the original job definition created when the job was registered. To rerun a job with a different job definition, you must register the job again or register a new job.

# cpareport

## Function

Sends information to standard output.

## Format

```
cpareport [-c { system [-f {csv|xml}] }  
| job [-t all] [-b start-date] [-e end-date] [-f {csv|xml}] [-i job-name]  
| job -t detail -i job-name [-o job-execution-ID] [-f {csv|xml}]  
| job -t output -i job-name [-o job-execution-ID] -p action-name  
[-q {stdout|stderr}]  
| jds  
| item  
}]
```

## Permission for execution

None

## Arguments

-c *system*

Exports system operating information in CSV or XML format.

- **CSV format (when -f *csv* is specified)**

No.	Row number	Field name	Value
1	1	Header	PRJNAME,STATUS,IESC
2	2 and later	Project name PRJNAME	The value of project name in the system settings file ( <i>config.xml</i> )
3		Status of the job execution service STATUS	UP (The job execution service is running.) DOWN (The job execution service is stopped.)
4		Status of the message communication service IESC	UP (The message communication service is running.) DOWN (The message communication service is stopped.)

<Example output>

```
PRJNAME, STATUS, IESC  
project, UP, DOWN
```

- **XML format (when -f *xml* is specified)**

No.	Row number	Element name	Value
1	1	xml	The XML element
2	2	PRJNAME	The same as the value in CSV format
3		STATUS	
4		IESC	

<Example output>

```
<?xml version="1.0" encoding="utf-8"?>
<PRJNAME>project</PRJNAME><STATUS>UP</STATUS><IESC>DOWN</IESC>
```

-c job

When you execute the command with this option specified, you have to specify the -t option as well.

If you execute this command without the -t option specified, the states of the job execution units run on the day the command is executed are displayed in a list, in the order of priority described in the table below. This list is not displayed when there is no state.

Priority	State	Order of priorities among the job execution units having identical states
1	Wait for start time	--
2	Executing	--
3	Ended normally	Job execution unit with the latest start time (The unit that started last)
4	Ended with warning	Job execution unit with the latest start time (The unit that started last)
5	Ended abnormally, Terminating event monitoring	Job execution unit with the latest start time (The unit that started last)
6	Ended abnormally (Cancel)	Job execution unit with the latest start time (The unit that started last)
7	Monitoring for event	--

Legend:

-- : Not applicable

The following tables describe the output format when you omit the -t option.

• **CSV format (when -f csv is specified)**

No.	Row number	Field name	Value
1	1	Header	JDSNAME, JCSID, STATUS, STARTTIME, ENDTIME
2	2 and later	Job name JDSNAME	The name of the existing job
3		Job execution ID JCSID	nnnnnnnnnnnnnnnn (String consisting of 40 characters or less) An identifier assigned when a job execution schedule is generated. For details, see <a href="#">1.3 Job execution unit and job execution ID</a> .
4		Job status STATUS#	<ul style="list-style-type: none"> <li>• RUN (Executing)</li> <li>• WAIT (Wait for start time)</li> <li>• MONITORING (Monitoring for event, Overflowing events)</li> <li>• QUEUEING (Wait for start)</li> <li>• ENDN (Ended normally)</li> <li>• ENDW (Ended with warning)</li> <li>• ENDE (Ended abnormally, Ended abnormally (Cancel), Terminating event monitoring)</li> <li>• ENDK (Ended abnormally (Cancel))</li> </ul>
5		Start time STARTTIME	YYYY/MM/DD HH:MM:SS.sss Different values are included depending on the job status as follows: <ul style="list-style-type: none"> <li>• WAIT: The scheduled time for the job to run next time.</li> <li>• MONITORING: The time when the system started monitoring whether execution conditions are met.</li> </ul>

No.	Row number	Field name	Value
			<ul style="list-style-type: none"> <li>• <b>QUEUEING</b>: The time when execution conditions are met and a new event is raised while the job is running.</li> <li>• <b>Others</b>: The start time of the job.</li> </ul>
6		End time ENDTIME	YYYY/MM/DD HH:MM:SS.sss Different values are included depending on the job status as follows: <ul style="list-style-type: none"> <li>• <b>ENDN, ENDE, ENDK</b>: The end time of the job.</li> <li>• <b>Others</b>: Blank</li> </ul>

#: For details on job status, see [1.1.4 State management for jobs and actions](#).

• **XML format (when `-f xml` is specified)**

No.	Row number	Element name	Attribute name	Value
1	1	xml		The XML element
2	2 and later	JDS	NAME	The same as the value in CSV format
3		JCSID		
4		STATUS#		
5		STARTTIME		
6		ENDTIME		

#: For details on job status, see [1.1.4 State management for jobs and actions](#).

`-t all`

Lists all execution units of existing jobs with their status.

Execution units of a job are sorted in descending order of the start time or scheduled start time (`STARTTIME` in output information) of the execution units.

The output format is the same as when you omit the `-t` option. You can use the `-b` and `-e` options to specify a period of time to filter job execution units based on the start time. If you omit the options, the output includes job execution units that started on the date when you run the command. For details, see the description on the `-b` and `-e` options.

`-i job-name`

Displays a list of statuses of the job specified by *job-name*.

The output format is the same as when the `-t all` option is specified.

`-t detail -i job-name`

Lists the status of the specified job and its actions.

When the `-o` option is not specified, the job state and the action state are displayed for the last job execution ID (`JCSID`) assigned.

A job name can include multibyte characters, alphanumeric characters, and underscores (`_`) and the length must be from 1 to 45 bytes.

The following tables describe the output format.

• **CSV format (when `-f csv` is specified)**

No.	Row number	Field name	Value
1	1	Header	The same as when you omit the <code>-t</code> option

No.	Row number	Field name	Value
2	2	Job name JDSNAME	
3		Job execution ID JCSID	
4		Job status STATUS	
5		Start time STARTTIME	
6		End time ENDTIME	
7	3	Header	TYPE, ACTNAME, STATUS, STARTTIME, ENDTIME
8	4 and later	Unit type	<ul style="list-style-type: none"> <li>ACT: The unit is an action or event monitor.</li> <li>FLW: The unit is an action flow, which is a group consisting of zero or more actions.</li> </ul>
9		Action name ACTNAME	The action name in the following format: [ /action-flow-name ] /action-name <u>The action names are displayed in a random order.</u>
10		Action status STATUS#	<ul style="list-style-type: none"> <li>RUN (Executing)</li> <li>WAIT (Wait for start)</li> <li>ENDN (Ended normally)</li> <li>ENDW (Ended with warning)</li> <li>ENDE (Ended abnormally)</li> <li>ENDK (Ended abnormally (Cancel))</li> </ul>
11		Start time STARTTIME	YYYY/MM/DD HH:MM:SS.sss <ul style="list-style-type: none"> <li>WAIT: Blank</li> <li>Others: The start time of the action.</li> </ul>
12		End time ENDTIME	YYYY/MM/DD HH:MM:SS.sss <ul style="list-style-type: none"> <li>ENDN, ENDE, ENDK: The end time of the action.</li> <li>Others: Blank</li> </ul>

#: For details on action status, see *1.1.4 State management for jobs and actions*.

• **XML format (when `-f xml` is specified)**

No.	Row number	Element name	Attribute name	Value
1	1	xml		The XML element
2	2	JDS	NAME	The same as the value in CSV format
3		JCSID		
4		STATUS#		
5		STARTTIME		
6		ENDTIME		
7	3 and later	ACT	NAME	

No.	Row number	Element name	Attribute name	Value
8		TYPE		
9		STATUS		
10		STARTTIME		
11		ENDTIME		

#: For details on action status, see [1.1.4 State management for jobs and actions](#).

Specifying this option causes the command to return a return code indicating the status of the specified job.

**Table 9–4: List of return codes**

Return code	Description
0	Executing
2	Wait for start, Monitoring for event, Terminating event monitoring
3	Ended

**-t** *output*

Displays the standard output or standard error output of the action.

**-o** *job-execution-ID*

Specify the ID of the job execution unit you want.

Omitting this option causes the system to use the job execution ID of the last execution unit.

A job execution ID can include alphanumeric characters and hyphens (-) and the length must be from 1 to 40 bytes.

**-p** *action-name*

Specify the desired action in the job in the following format (which you can see in the output of the command with **-t detail**):

[ /*action-flow-name* ] /*action-name*

An action name can include multibyte characters, alphanumeric characters, and underscores (\_) and the length must be from 1 to 256 bytes including the leading slash (/).

**-q** {*stdout|stderr*}

Specify which type of information to be included in the output.

Omitting this option causes the command to display the standard output followed by the standard error output.

Specify *stdout* to display only the standard output.

Specify *stderr* to display only the standard error output.

**-b** *start-date* **-e** *end-date*

Specify dates in *YYYYMMDD* format to indicate a period for the output list.

Acceptable values are from 19940101 to 20991231.

If you specify the start date and the end date, the output list includes job execution units that started between the start date and the end date.

If you omit the start date, the output list includes job execution units that started before or on the end date.

If you omit the end date, the output list includes job execution units that started on or after the start date.

If you omit the start date and the end date, the output list includes job execution units that started on the date when you run the command.

An argument error occurs when you specify both the start date and end date and the end date is earlier than the start date.

-f csv

Generates output in CSV format.

-f xml

Generates output in XML format.

-c jds

Lists existing jobs with their name and registration date.

-c item

Lists all existing items with their type, name, and registration date.

## Return values

0	Succeeded If <code>-t detail</code> was specified, this indicates that the job is in the Executing state.
2	If <code>-t detail</code> was specified, this indicates that the job is in the Wait for start, Monitoring for event, or Terminating event monitoring state.
3	If <code>-t detail</code> was specified, this indicates that the job is ended and the next execution is not scheduled.
40	Insufficient resources
50	Not allowed to access the system settings file ( <code>config.xml</code> ).
52	Could not access the file.
58	Syntax error in arguments
86	No record (Nothing can be listed because no record has been generated for the specified job or job execution ID.)
94	The system settings file ( <code>config.xml</code> ) is invalid.
109	The system settings file ( <code>config.xml</code> ) does not exist.
111	File access error
121	A system limit is exceeded. (The number of opened files reaches the maximum number allowed on the system.)
123	The execution environment is invalid. (You need to delete the file or reinstall the program, for example, because a directory does not exist or a file is broken.)
124	I/O error (Cannot open the file due to a disk error.)
125	System error (The API of the operating system failed.)
126	Other error

## Environment requirements

This command can run regardless of whether the job execution service is running or stopped.

### Command stored in:

`install-path\bin`

Note that the command is stored in the following folder by default.

In a 32-bit operating system:

`%ProgramFiles%\Hitachi\JP1CPA\bin`

In a 64-bit operating system:

```
%ProgramFiles(x86)%\Hitachi\JP1CPA\bin
```



# cpajobprint

---

## Function

Outputs the definition information for a job or item to a file.

The command output result is redirected and saved as a definition file. A job or item can be registered by specifying the definition file in the `cpajobentry` command or loading it from the Job Design View or the Item Design View.

Example:

```
cpajobprint -i Job1 > output-file-name.xml
```

## Format

```
cpajobprint
  {  -i job-name
    | -f action-flow-item-name
    | -c calendar-item-name
    | -a action-item-name
    | -d execution-date-item-name
    | -t execution-time-item-name  }
> output-file-name.xml
```

## Permission for execution

None

## Arguments

### **-i** *job-name*

Outputs the definition information for the specified job.

Only one job can be specified.

For the job name, you can use multibyte characters, half-width alphanumeric characters, and underscores (`_`), and the length of the name is from 1 to 45 bytes.

### **-f** *action-flow-item-name*

Outputs the definition information for the specified item.

Only one item can be specified.

For the item name, you can use multibyte characters, half-width alphanumeric characters, and underscores (`_`), and the length of the name is from 1 to 45 bytes.

### **-c** *calendar-item-name*

Outputs the definition information for the specified item.

Only one item can be specified.

For the item name, you can use multibyte characters, half-width alphanumeric characters, and underscores (`_`), and the length of the name is from 1 to 45 bytes.

### **-a** *action-item-name*

Outputs the definition information for the specified item.

Only one item can be specified.

For the item name, you can use multibyte characters, half-width alphanumeric characters, and underscores (`_`), and the length of the name is from 1 to 45 bytes.

**-d** *execution-date-item-name*

Outputs the definition information for the specified item.

Only one item can be specified.

For the item name, you can use multibyte characters, half-width alphanumeric characters, and underscores (`_`), and the length of the name is from 1 to 45 bytes.

**-t** *execution-time-item-name*

Outputs the definition information for the specified item.

Only one item can be specified.

For the item name, you can use multibyte characters, half-width alphanumeric characters, and underscores (`_`), and the length of the name is from 1 to 45 bytes.

**>** *output-file-name.xml*

Specify the output file name as a redirect destination. The extension must be `.xml`.



**Note**

When the `-c item` or `-c jds` option is used in the `cpareport` command to output the information of the item or job list, and the item name or job name that is output is specified as an option of the `cpajobentry` command, the definition information can be output by item or job type.

## Return values

0	Succeeded
40	Insufficient resources
58	Invalid argument
67	The specified job or item does not exist.
96	The job or item is invalid.
111	File access error
125	System error
126	Program error

## Environment requirements

This command can run regardless of whether the job execution service is running or stopped.

### Command stored in:

*install-path*\bin

Note that the command is stored in the following folder by default.

In a 32-bit operating system:

`%ProgramFiles%\Hitachi\JP1CPA\bin`

In a 64-bit operating system:

`%ProgramFiles(x86)\Hitachi\JP1CPA\bin`

## Notes

- The definition information is output in UTF-8 format.
- If the command is executed without redirection, the result is sent to the standard output.
- Do not modify the contents of the definition file that is output. If modified, they cannot be registered using the `cpajobentry` command and read from windows using the **Open** button.
- Job names and items names are case sensitive. Specify them correctly.
- If the name of a job or item that does not correspond to the argument option is specified as an argument, a command error occurs and the command terminates. However, if a job or item with the same name as the value specified in the argument exists, the definition information of the job or item is output.

# cpapasswd

---

## Function

Encrypts the authentication password and adds it to the mail profile so that the password can be used for SMTP-AUTH or the POP before SMTP authentication to send mail.

This command also can encrypt the password of the OS user to log in to the incoming mail server for monitoring mail reception and add the password to the profile of mail reception.

## Format

```
cpapasswd {-s SMTP-password|-p POP-before-SMTP-authentication-password|-r in  
coming-mail-server-login-OS-user-password}
```

## Permission for execution

None

## Arguments

-s *SMTP-password*

Encrypts the authentication password for SMTP-AUTH and adds it to the mail profile. When the profile already includes a password, running this command updates the password.

An SMTP password can be any string with a length of 1 to 127 bytes.

-p *POP-before-SMTP-authentication-password*

Encrypts the authentication password for the POP before SMTP authentication and adds it to the mail profile. When the profile already includes a password, running this command updates the password.

A POP3 password can be any string with a length of 1 to 127 bytes.

-r *incoming-mail-server-login-OS-user-password*

Encrypts the password of the OS user that will be used to log in to the incoming mail server for monitoring mail reception, and adds the password to the profile of mail reception. When the profile already includes a password, running this command updates the password.

A password of the OS user to log in to the incoming mail server can be any string with a length of 1 to 127 bytes.

## Return values

0	Succeeded
40	Insufficient resources
45	Exclusive processing error
54	Not allowed to access the profile.
58	Syntax error in arguments
64	Cannot find the profile.
125	System error

## Environment requirements

This command can run regardless of whether the job execution service is running or stopped.

Before you can run the command, ensure that the `conf` directory in the data path directory contains the mail profile (`mailprof.ini`) or the profile of mail reception (`recvmailprof.ini`). For details on the mail profile (`mailprof.ini`), see [2.3.1 Configurations for the Send mail item](#). For details on the profile of mail reception (`recvmailprof.ini`), see [2.3.2 Configurations for the Monitoring mail reception item](#).

### Command stored in:

`install-path\bin`

Note that the command is stored in the following folder by default.

In a 32-bit operating system:

`%ProgramFiles%\Hitachi\JP1CPA\bin`

In a 64-bit operating system:

`%ProgramFiles(x86)%\Hitachi\JP1CPA\bin`

# cpasetprof

---

## Function

Configures profile items in the profile. Jobs use the configured profile items as variables to run.

## Format

```
cpasetprof [{-a profile-key=value|-d profile-key|-c}]
```

## Permission for execution

None

## Arguments

**-a** *profile-key=value*

Adds a key-value pair of a profile item to the profile.

*profile-key*:

Specify a string consisting of alphanumeric characters, with a length of 32 bytes or less.

*value*:

Specify a string with a length of 80 bytes or less. A value must not include a comma (,) or space ( ) in the ASCII table or line breaks (\n, CR+LF, or LF). If the encoding of the operating system's locale is not UTF-8, the length limit is applied to the UTF-8 converted string.

Specifying an existing key overwrites the existing value. You cannot add a new profile item when the maximum allowed number (16) of items already exists.

You can repeat the **-a** option as many times as needed unless the maximum allowed number is exceeded.

**-d** *profile-key*

Deletes the specified profile item from the profile.

You can repeat the **-d** option. Note that, if a **-d** statement fails to delete a profile item, the process still continues with subsequent statements. In such a case, the command returns the highest value of all return values from failed statements.

**-c**

Deletes the entire profile.

No option

Running the command with no option lists all items from the profile. If there is no item to be listed, the command sends a message to standard output indicating that the profile has no item, and then returns the return value for success.

The output list is generated in the following format.

<Example output>

```
prof1=user1
prof2=D:\work
```

## Return values

0	Succeeded
---	-----------

16	The profile item does not exist.
40	Insufficient resources
58	Invalid argument
85	Exceeded the maximum allowed number of registered items.
94	The system settings file ( <code>config.xml</code> ) has an error.
104	The profile settings file is invalid.
109	The system settings file ( <code>config.xml</code> ) does not exist.
112	The profile settings file does not exist.
125	System error
126	Program error

## Environment requirements

This command can run regardless of whether the job execution service is running or stopped.

This command can run regardless of whether the message communication service is running or stopped.

## Command stored in:

*install-path*\bin

Note that the command is stored in the following folder by default.

In a 32-bit operating system:

`%ProgramFiles%\Hitachi\JP1CPA\bin`

In a 64-bit operating system:

`%ProgramFiles(x86)\Hitachi\JP1CPA\bin`

# cpasetfirewall

---

## Function

Adds the exceptions to Windows Firewall so that the AJS3 linkage function can enable CPA and CPA Option for AJS3 to interact with each other. Additionally, this command can remove the exceptions from Windows Firewall.

## Format

```
cpasetfirewall [.bat] {-s|-u}
```

## Permission for execution

Administrator privileges

## Arguments

-s

Adds the exceptions to Windows Firewall.

-u

Removes the exceptions from Windows Firewall.

## Return values

0	Succeeded
Other than 0	Failed

## Environment requirements

This command can run regardless of whether the job execution service is running or stopped.

This command can run regardless of whether the message communication service is running or stopped.

## Command stored in:

*install-path*\bin

Note that the command is stored in the following folder by default.

In a 32-bit operating system:

%ProgramFiles%\Hitachi\JP1CPA\bin

In a 64-bit operating system:

%ProgramFiles(x86)%\Hitachi\JP1CPA\bin



# cpaalsetup

---

## Function

Sets up the Windows log-on function. If the `-u` option is specified, this command unsets up the Windows log-on function.

Run this command in a Windows 10 environment. After setting up the Windows log-on function, restart the machine.

## Format

```
cpaalsetup [-u]
```

## Permission for execution

Administrator privileges

## Arguments

`-u`

Unsets up the Windows log-on function.

No option

Sets up the Windows log-on function.

## Return values

0	Succeeded
45	Temporary error (The file to be updated is temporarily in use.)
52	Permission denied
58	Invalid argument
111	File access error
119	System limit exceeded
125	System error

## Environment requirements

This command can run regardless of whether the job execution service is running or stopped.

This command can run regardless of whether the message communication service is running or stopped.

This command cannot set up the Windows log-on function from the Windows logon window.

## Command stored in:

`install-path\bin`

Note that the command is stored in the following folder by default.

In a 32-bit operating system:

`%ProgramFiles%\Hitachi\JP1CPA\bin`

In a 64-bit operating system:

```
%ProgramFiles(x86)%\Hitachi\JP1CPA\bin
```

## Notes

Even Windows users with general privileges can execute the Windows log on action item. This means that, when the Windows log-on function is set up in an environment where a single machine is shared by multiple Windows users, even a Windows user with general privileges can log on to Windows as another Windows user. If this can pose a problem in your operations, do not set up the Windows log-on function.

# cpasetumap

---

## Function

Registers the following information in the user mapping file (`usrprof.xml`): CPA virtual users used with the Windows log-on function or execution user setting function and information regarding the users to be mapped to the virtual users (Windows user names and passwords).

## Format

```
cpasetumap [{-a virtual-user-name -u Windows-user-name -p password|-d virtual-user-name}]
```

## Permission for execution

None

## Arguments

`-a virtual-user-name`

Specifies the value to be specified as the *virtual user name* for the *Windows log-on* action item or the item that uses the execution user setting function.

The virtual user name specified with this option is linked with the user information specified with the `-u` and `-p` options and then registered in the user mapping file.

You can specify 1- to 14-byte half-width alphanumeric characters as a virtual user name.

If you specify a virtual user name that has already been registered in the user mapping file, the registered values are updated with the values specified for the `-u` and `-p` options. If the user mapping file already contains the maximum number of virtual users that can be registered (10 virtual users) in it, you cannot add a new virtual user to it.

`-u Windows-user-name`

Specifies the Windows user name to be linked to the virtual user name specified by the `-a` option.

When applicable, a user name containing a domain name must be specified in *NetBIOS-domain-name\user-name* format. If you omit the NetBIOS domain name, it is assumed that the local host name.

You can specify a 1- to 276-byte Windows user name consisting of half-width alphanumeric characters and the following half-width symbols:

`_ - \ .`

If the specified Windows user name contains any of the above half-width symbols, enclose the Windows user name in double quotation marks (`\`). Any double quotation marks within the password must be denoted as `\\` format.

You can only use a backslash (`\`) to separate *domain-name* from *user-name*. Neither *domain-name* nor *user-name* can contain a backslash.

`-p password`

Sets a password for the Windows user name specified by the `-u` option.

You can specify a 1- to 20-byte password consisting of half-width alphanumeric characters and the following half-width symbols:

`` ~ ! @ # $ % ^ & * ( ) _ - + = { } [ ] \ | : ; " ' < > , . ? /`

If the specified password contains any of the above half-width symbols, enclose the password in double quotation marks (`"`). Any double quotation marks within the password must be denoted as `\"` format.

-d *virtual-user-name*

From the user mapping file, removes the specified virtual user name as well as the Windows user name and password linked to it.

No option

If you do not specify any options, virtual user names registered in the user mapping file as well as the Windows user names linked to them are displayed in the following list format:

*virtual-user-name=Windows-user-name*

When there is no item to be displayed, a message stating that no virtual user information has been registered is sent to the standard output. In this case, the return value of the command will indicate that the command was completed successfully.

## Return values

0	Succeeded
16	The specified virtual user is not registered.
40	Insufficient resources
58	Invalid argument
85	Exceeded the maximum allowed number of registered items.
94	The system settings file ( <code>config.xml</code> ) has an error.
104	The user mapping file ( <code>usrprof.xml</code> ) is invalid.
109	The system settings file ( <code>config.xml</code> ) does not exist.
112	File access error
125	System error
126	Program error

## Environment requirements

This command can run regardless of whether the job execution service is running or stopped.

This command can run regardless of whether the message communication service is running or stopped.

## Command stored in:

*install-path*\bin

Note that the command is stored in the following folder by default.

In a 32-bit operating system:

`%ProgramFiles%\Hitachi\JP1CPA\bin`

In a 64-bit operating system:

`%ProgramFiles(x86)\Hitachi\JP1CPA\bin`

## Notes

- Even Windows users with general privileges can use this command. We recommend that you confirm in advance whether it is acceptable to allow Windows user with general privileges to perform the following operations:

- Even Windows users with general privileges can view virtual user names and Windows user names registered in the user mapping file. For example, they can obtain such user information and then use it with the Windows log-on function to log on to Windows.
- Even Windows users with general privileges can update virtual user information registered in the user mapping file. For example, they can use the updated information to log on to Windows by using the Windows log-on function.
- Even Windows users with general privileges can have full control (read and write privileges) over the user mapping file.
- Password information specified with the ?p option is obfuscated so that it cannot be easily recovered, and then set in the user mapping file. However, encryption by means of AES, RSA, or SHA-2 will not be performed. For details about the location in which the user mapping file is placed, see *D.1 Files and folders in CPA*.

# cpausrenvsetup

---

## Function

Deletes the tabs for defining action flows and calendars from the Job Design View so as to set up or unset up the operating environment in which only items are available for use.

## Format

```
cpausrenvsetup [-u]
```

## Permission for execution

None

## Arguments

-u

Restores the setting that makes the Job Design View display the tabs for defining action flows and calendars so that the Job Design View can be used to define items.

No option

Applies the setting that prevents the tabs for defining action flows and calendars from showing up in the Job Design View so that items cannot be defined from the Job Design View.

## Return values

0	Succeeded
58	Invalid argument
111	An error occurred during an attempt to access the Job Design View settings file (jobdesign_config.xml). <ul style="list-style-type: none"><li>• The file contains invalid information.</li><li>• The file does not exist.</li><li>• A file access error occurred due to lacking permissions.</li></ul>
125	System error

## Environment requirements

This command can run regardless of whether the job execution service is running or stopped.

This command can run regardless of whether the message communication service is running or stopped.

## Command stored in:

*install-path*\bin

Note that the command is stored in the following folder by default.

In a 32-bit operating system:

%ProgramFiles%\Hitachi\JP1CPA\bin

In a 64-bit operating system:

%ProgramFiles(x86)%\Hitachi\JP1CPA\bin

## Notes

- If you execute this command while the Job Design View is open, the settings are enabled the next time you open the Job Design View again.
- If this command ends abnormally, follow the instructions provided in the output error message. If the problem persists, copy the file `jobdesign_config.xml.model` under *data-path*\conf, rename it as `jobdesign_config.xml`, and then re-execute the command.

# cpaitemsetup

---

## Function

Sets up or unsets up the custom item creation environment.

Executing this command creates a shortcut to the Item Design View in the Windows **Start** menu, which makes it easier for you to open and use the Item Design View.

For details on the functionality available to create items in the Item Design View, see *5. Business Operations Planning with CPA*.

## Format

```
cpaitemsetup [-u]
```

## Permission for execution

Administrator privileges

## Arguments

-u

Unsets up the custom item creation environment.

If you specify the option, the shortcut **Item Design View** is deleted from **JP1\_Client Process Automation** in the Windows **Start** menu so that no one can open and use the Item Design View from there.

No option

Sets up the custom item creation environment.

If you do not specify the option, the short cut **Item Design View** is registered in **JP1\_Client Process Automation** in the Windows **Start** menu so that users can open and use the Item Design View from there.

## Return values

0	Succeeded
52	Permission denied
58	Invalid argument
111	An error occurred during the attempt to access the Item Design View settings file (itemdesign_config.xml). <ul style="list-style-type: none"><li>• The file contains invalid information.</li><li>• The file does not exist.</li><li>• A file access error occurred due to lacking permissions.</li></ul>
123	The execution environment is invalid.
125	System error

## Environment requirements

This command can run regardless of whether the job execution service is running or stopped.

This command can run regardless of whether the message communication service is running or stopped.



## Command stored in:

*install-path*\bin

Note that the command is stored in the following folder by default.

In a 32-bit operating system:

%ProgramFiles%\Hitachi\JP1CPA\bin

In a 64-bit operating system:

%ProgramFiles(x86)%\Hitachi\JP1CPA\bin

## Notes

- If you start the Item Design View before setting up the custom item creation environment or after unsetting up the custom item creation environment, the following message appears:  
KNAO4103-E: Setup of building environment for customized items is not finished. Creating customized items view cannot be started.
- Even after you unset up the custom item creation environment by executing this command, you might still find the shortcut to the Item Design View in the Windows **Start** menu. If this happens, restart the OS on the machine on which you have unset up the custom item creation environment.
- If, after setting up the custom item creation environment by executing this command, you decide to change the OS language setting, you have to first unset up the custom item creation environment.
- If you decide to uninstall CPA after you set up the custom item creation environment by executing this command, you have to first unset up the custom item creation environment.
- If this command ends abnormally, follow the instructions provided in the output error message. If the problem persists, copy the file `itemdesign_config.xml.model` under *data-path*\conf, rename it as `itemdesign_config.xml`, and then re-execute the command.

# cpaecho

---

## Function

Sends a string excluding line breaks to standard output.

## Format

```
cpaecho [string]
```

## Permission for execution

None

## Arguments

*string*

Specify a string you want to send to standard output.

## Return values

None

## Environment requirements

This command can run regardless of whether the job execution service is running or stopped.

## Command stored in:

*install-path*\bin

Note that the command is stored in the following folder by default.

In a 32-bit operating system:

%ProgramFiles%\Hitachi\JP1CPA\bin

In a 64-bit operating system:

%ProgramFiles(x86)%\Hitachi\JP1CPA\bin

## Notes

To enable a Command execution item to send a message to standard error output, configure the item as follows:

Execution command	cmd.exe
Parameter	/C cpaecho <i>message-to-be-sent</i> 1>&2

# cpasleep

---

## Function

Suspends the execution of the job for a specified time period.

## Format

```
cpasleep seconds
```

## Permission for execution

None

## Arguments

*seconds*

Specify a value in the range from 0 to 86,400 seconds as the time period for which to suspend the execution of the job.

## Return values

Always 0

## Environment requirements

This command can run regardless of whether the job execution service is running or stopped.

## Command stored in:

*install-path*\bin

Note that the command is stored in the following folder by default.

In a 32-bit operating system:

```
%ProgramFiles%\Hitachi\JP1CPA\bin
```

In a 64-bit operating system:

```
%ProgramFiles(x86)%\Hitachi\JP1CPA\bin
```

# Appendixes

## A. Troubleshooting

---

This appendix describes how to troubleshoot problems and which data you need to collect.

### A.1 Steps of troubleshooting

Use the following steps to troubleshoot a problem with CPA operation:

#### Clarify what occurred

Clarify details about the following:

- The situation where the problem occurred
- Messages (if any)
- The product log and other logs

For details on the causes and solutions to each message, see the manual *JP1/Client Process Automation Messages* which is listed in *Message*. For what kind of log data CPA can generate, see the manual *JP1/Client Process Automation Messages* which is listed in *Logging*.

#### Collect data

Collect data required to identify the cause of the problem. To collect appropriate data, see [A.2 Collecting data](#).

#### Investigate the problem

Examine collected data to identify the cause and isolate the source and affected areas.

### A.2 Collecting data

Collect data that is stored in the data folder, which was specified during the installation of CPA and CPA Option for AJS3.

You can use the following steps to collect data from the default data folder.

#### Default data folder

```
%ProgramData%\HITACHI\jpl\jpl_default\JP1CPA
```

(For example, when the system drive is C:, the path results in C:\ProgramData\HITACHI\jpl\jpl\_default\JP1CPA.)

1. Start Windows Explorer to open the data folder.

If the %ProgramData% folder does not appear, click **View** tab and select the **Hidden items** check box.

2. Right-click the JP1CPA folder, and select **Send to** and then **Compressed (zipped) folder**.

3. Now, a file named JP1CPA.zip is created in the same folder. Save the file so that the collected data can be used later.

If an error occurred during the installation or uninstallation, collect the following data (installation log files (CPA\_Inst.log)):

- %WinDir%\Temp\HCDINST folder
- %WinDir%\Temp\jplcommon\JP1CPA folder

- %WinDir%\Temp\JP1CPA folder

In addition, if you set up the Windows log-on function and an error occurs in the function, collect the following data (trace log (CpaCredentialProvider.log) of the Windows log-on function) as well:

- %WinDir%\Temp\JP1CPA folder

Next, follow the steps described below to obtain a list of processes, which will be used to confirm that there are no remaining unnecessary processes.

To collect a list of processes:

1. Open a command prompt with administrator privileges.
2. Run the following command:

```
tasklist>file-path-to-output-destination-file-(file-name.txt)
```

3. A file titled *file-name*.txt is created in the file path specified in step 2. Save it for later reference.

### A.3 Different behaviors in different command execution environments

A command behaves differently depending on whether it is issued from CPA or issued manually in the command prompt. Check the following table if a command issued from CPA behaves differently from your expectation.

Table A–1: Differences between manually-issued commands and job-triggered commands

Item	Description	Issued from			
		Command prompt (manually)	Command execution#1	Command execution in window#2	Command execution as virtual user#3
User profile	Is the user profile, information stored in the registry key HKEY_CURRENT_USER, loaded?	Yes, the user profile is loaded.	No, the user profile is not loaded.	The system uses the profile of the user that is logged on to the console session or the Remote Desktop session.  Note Ensure that the user is the same OS user that is used to run commands manually.	Yes, the user profile is loaded (the profile of the Windows user associated with the virtual user name).
WOW64 environment	In a WOW64 environment, a 64-bit application located in %systemroot%\system32 cannot be called in the following cases: <ul style="list-style-type: none"> <li>• When a file under %systemroot%\system32 is specified as the executable file for the action, the call is redirected to the %systemroot%\syswow64 path at run</li> </ul>	No need to consider.	Need to consider. (The command runs in a WOW64 environment.)		

Item	Description	Issued from			
		Command prompt (manually)	Command execution#1	Command execution in window#2	Command execution as virtual user#3
	<p>time. This can cause the executable file to fail to start or work properly.</p> <ul style="list-style-type: none"> <li>If the access to the HKEY_LOCAL_MACHINE\Software or HKEY_CLASS_ROOT registry key is redirected to HKEY_LOCAL_MACHINE\Software\SysWow64, the executable file cannot get intended information. This can cause the executable file to fail to work properly.</li> </ul> <p>Action to take</p> <p>Specify the %systemroot%\sysnative path, instead of the %systemroot%\system32 path.</p>				
Excel automation	<p>In a 64-bit operating system, if a service program includes VBScript code to interact with an Excel object, calling Excel automation from the service might result in a file-open error.</p> <p>Action to take</p> <p>Check whether the following directories exist:</p> <ul style="list-style-type: none"> <li>%SystemRoot%\SysWOW64\config\systemprofile\Desktop</li> <li>%SystemRoot%\System32\config\systemprofile\Desktop</li> </ul> <p>If these directories do not exist, create them. Then retry your operation to check whether the problem persists.</p>	No need to consider.	Need to consider.	No need to consider.	Need to consider.
Runspace	<p>The CPA Command execution item runs in the service space. That is, the command cannot use environment variables, network shares, or other information that becomes available when the user is logged on.</p> <p>Action to take</p> <p>Use the Command execution in window item to run an interactive job or a job that needs to display a window.</p>	No need to consider. (The command runs in the user session.)	Need to consider. (The command runs in the service space.)	No need to consider. (The command runs in the user session.)	Need to consider. (The command runs in the service space.) However, the environment variables can be set.

Item	Description	Issued from			
		Command prompt (manually)	Command execution <sup>#1</sup>	Command execution in window <sup>#2</sup>	Command execution as virtual user <sup>#3</sup>
Environment variables	<p>User environment variables are not set for the execution environment of the CPA Command execution item. Available environment variables are limited to system environment variables, which are set when the service starts.</p> <p>Action to take</p> <p>Use the Command execution in window item or Command execution as virtual user to run a job that references a user environment variable.</p>	No need to consider.	Need to consider. (User environment variables are unavailable.)	Need to consider. (The system uses the environment variables of the user that is logged on to the console session or the Remote Desktop session.)  Note Ensure that the user is the same Windows user that is used to run commands manually.	No need to consider.
OS user to be used to run commands	--	The user that started the command prompt	The account of the CPA job execution service	The user that is logged on to the session	Windows user associated with the virtual user name.
File handles	<p>Standard output, standard error output, and standard input are different.</p> <p>The <code>ftp</code> and <code>timeout</code> commands (provided with the operating system) might fail.</p>	No need to consider.	Need to consider. (The <code>ftp</code> and <code>timeout</code> commands do not work.)	No need to consider.	Need to consider. (The <code>ftp</code> and <code>timeout</code> commands do not work.)
Network share	Can the command use a network drive letter that is assigned by <b>Map network drive</b> ?	Yes, the command can use the letter.	This letter is not available when the user starting the CPA job execution service has a local system account (default account). <sup>#3</sup>	This letter becomes available when access is granted to the user logged on to Windows at the time when the <i>Command execution in window</i> item is executed.	This letter becomes available when the UNC format (which starts with <code>\computer-name</code> ) is used. The Windows user associated with the virtual user name must be able to access the network drive.
Current directory	--	<p>The current directory when the user runs the command.</p> <p>When the command prompt starts, the current directory is set to a folder that is specified in <b>Start in</b> of the Command Prompt properties.</p>	<code>system-drive:\ProgramData\Hitachi\jpl\jpl_default\JP1CPA\work</code>	<code>system-drive:\ProgramData\Hitachi\jpl\jpl_default\JP1CPA\work</code>	<code>system-drive:\ProgramData\Hitachi\jpl\jpl_default\JP1CPA\work</code>



#1

*Command execution* includes not only the *Command execution with recovery option* and *Command execution by list* items but also the action items for which **Do not show the window** was selected as the execution type when the custom items were created.

#2

*Command execution in window* includes the action items for which **Show the window** was selected as the execution type when the custom items were created.

#3

*Command execution as virtual user* includes the commands specified in *Command execution with recovery option*, *Command execution by list*, and *Monitor any event*. It also includes the action items for which **Do not show the window** was selected as the execution type when the custom items were created.

#4

This letter is available when the account of the user starting the CPA job execution service is changed to an account of the user with administrator privileges who has access to the network drive. However, the *Command execution in window* and *Windows log on* items cannot be executed with an account other than the local system account.

## B. Tips to Operate Jobs

---

### B.1 Using inherited information and profile items

Inherited information, including the execution results of actions and the information items detected by event monitors, can be referenced from subsequent actions in the same job by using variables. Additionally, the profile items, which are set on CPA by using the `cpasetprof` command, also can be referenced as variables.

The formats of variables to reference inherited information are as follows:

- To inherit information from an event monitor that is contained in the execution condition of a job executed on event:  
`${JDS:@Event.unit-name-used-in-execution-conditions.method-name}`
- To inherit information from an event monitor or action that is contained in an action flow:  
`${JDS:unit-name.method-name}`
- To inherit a profile item that is set by using the `cpasetprof` command:  
`${PRF:profile-key}`

Referencing these information items is useful when you want to run the same job on different PCs without changing the definitions manually.

For example, you configure the profile to include PC-specific and PC-user specific information such as email addresses. A job can reference the profile item as a variable in order to send a job completion notification to a different destination. Note that an action will end abnormally unless referenced profile items are set properly at its run time.

For details on the *unit-name-used-in-execution-conditions* and *method-name* to be specified to inherit information from an event monitor, see [5.5.2 Inherited information](#).

For details on the *unit-name* and *method-name* to be specified to inherit information from an action, see [5.4.2\(1\) Inherited information](#).

### B.2 How to use environment variables when an item is executed

When an item in a job is executed, CPA can reference the information of the job through environment variables. This section describes how to use environment variables when an item is executed.

#### (1) Environment variables when an item is executed

When you execute commands from items, such as *Command execution* and *Command execution in window*, that are executed in a job of CPA, the environment variables listed in the following table can be used.

Table B–1: Environment variables when an item is executed

No.	Environment variable	What value is specified
1	CPAJDS	Job name A value specified in <b>Job name</b> when the job is registered.
2	CPAJCS	Job execution ID

No.	Environment variable	What value is specified
		A string of 36 half-width alphanumeric characters (hexadecimal numbers), assigned by CPA when an execution schedule is generated. The ID does not conflict with job execution IDs of other jobs and in the execution results in the past. Example: e66bf2f5-b920-4faf-b1f0-a56418af4682

The environment variables (CPAJDS and CPAJCS) can be referenced within commands that CPA executes. For example, if a batch file is executed as a command, the environment variables are specified in the batch file.

They can also be used by defining them in the parameter of the command to be executed. The following table lists the setting items where the environment variables (CPAJDS and CPAJCS) can be defined.

**Table B–2: Setting items where the environment variables (CPAJDS and CPAJCS) can be defined**

No.	Type	Item name	Possible setting item
1	Action	Command execution	Parameter
2		Command execution by list	Parameter
3		Command execution with recovery option	Parameter Parameter for the recovery execution command
4		Command execution in window	Parameter
5		Send mail	--
6		Windows log on	--
7	Event monitor	Monitor file	--
8		Monitoring mail reception	--
9		Monitor any event	Parameter
10		Monitor Windows event	--
11	Execution condition	Monitor file event	--
12		Event monitoring for receiving mail	--
13		Monitor any event	Parameter
14		Monitor the linkage from JP1AJS3	--
15		Monitor Windows event	--

Legend:

-- : There is no setting item that can be defined.

## (2) Usage example

In this example, the results from actions in an action flow are output to a temporary file, and subsequent actions reference the contents of the file.

At this time, use the environment variables (CPAJDS and CPAJCS) to create a file with a unique name for each job, so that the name of the temporary file used in the action flow does not conflict among multiple jobs that use the same action flow.

## Command execution in window

Execution command:

```
cmd
```

Parameter:

```
/c sample1.bat >c:\temp\%CPAJDS%_%CPAJCS%.txt
```

sample1.bat contains:

```
@echo off
echo %DATE% %TIME%
echo CPAJDS:%CPAJDS%
echo CPAJCS:%CPAJCS%
cpasleep 10
```

## Command execution

Execution command:

```
cmd
```

Parameter:

```
/c sample2.bat c:\temp\%CPAJDS%_%CPAJCS%.txt
```

sample2.bat contains:

```
@echo off
echo %DATE% %TIME%
echo FILE:%1
echo -----
type %1
echo -----
```

The following shows an example of executing the above action flow as a job with the job name of sampleA:

1. The first *Command execution in window* action creates the following file.

Job executed

Job name: sampleA

Job execution ID: cab71f3-d402-4869-a651-1084fe7b7ae5

Name of the file to be created

c:\temp\sampleA\_cab71f3-d402-4869-a651-1084fe7b7ae5.txt

Explanation

One variable %CPAJDS% is replaced with the job name sampleA, and another variable %CPAJCS% with the job execution ID cab71f3-d402-4869-a651-1084fe7b7ae5.

2. The output result of the following action *Command execution* shows the contents of the file created by the first action. Note that the output result can be viewed in the **Execution result** dialog box opened from the **Checking the status** dialog box in the CPA Job Design View.

Output result:

```
2020/09/25 15:19:46.03
FILE:c:\temp\sampleA_cab71f3-d402-4869-a651-1084fe7b7ae5.txt
-----
```

```
2020/09/25 15:19:35.59
CPAJDS:sampleA
CPAJCS:cabc71f3-d402-4869-a651-1084fe7b7ae5
-----
```

## **B.3 Expanding the number of custom jobs for CPA linkage that are executed simultaneously on AJS3**

When you execute a custom job for CPA linkage on AJS3, the system executes the command that instructs CPA to start the job and waits for the job to finish. To communicate with CPA, this command uses the message communication service of CPA Option for AJS3.

If you start more than one custom job for CPA linkage simultaneously, you can change the maximum number of connections so that the connections do not exceed the upper limit managed by the message communication service.

### **(1) Expanding the number of connections for the message communication service**

The maximum number of custom jobs for CPA linkage that can be executed simultaneously is 16 jobs by default. If the number of jobs exceeds 16, specify a value obtained by multiplying the number of concurrent jobs by 4 for the maximum number of connections for the message communication service.

For details on the definition file of the message communication service, see [E.2 Message communication service definition file](#).

## C. Usage Estimation

---

This appendix describes the usage estimation of CPA.

### C.1 Limit values

The following table lists limit values on CPA.

Table C–1: Limit values

No.	Item	Limit value
1	The number of units (actions, action flows, and event monitors) that an action flow can contain	64
2	Unit depth (The depth indicates the level of a nested action flow. Referencing another action flow increases the depth of the hierarchical structure.)	2
3	The number of jobs that can be registered with the job execution service <sup>#1</sup>	16
4	The number of actions that the job execution service can execute concurrently	32
5	The number of actions that can run concurrently for a job	8
6	The maximum number of results stored for a job executed on time	5
7	The maximum number of results stored for a job executed on event	5
8	The length (Unit: bytes) of the name of a job or item	1 to 45
9	The length (Unit: bytes) of the description of a job or item	0 to 1,024
10	Dates allowed for schedules	2013/01/01 to 2037/12/31
11	The acceptable delay (Unit: minutes) from the scheduled time for a job to start	60
12	The number of files monitored by the Monitor file item (when the monitored file name is specified with wildcards)	10
13	Timeout period (Unit: minutes) of an event monitor	0 to 1,440
14	The maximum size (Unit: bytes) of data sent to standard output	32,768
15	The maximum size (Unit: bytes) of data sent to standard error output	32,768
16	The length (Unit: bytes) of the name of a command to be executed	1 to 256
17	The length (Unit: bytes) of the parameter specified with a command to be executed	0 to 512
18	The timeout period (Unit: minutes) for the Command execution in window item	0 to 1,440
19	The maximum number of queued instances of a job executed on event <sup>#2</sup>	16
20	The maximum number of profiles that can be registered	16
21	The maximum number of virtual users that can be registered	10
22	The number of input items that can be created for an item	10

#1

You can expand the number of registered jobs up to 64 by specifying `job_ent_max` in the system settings definition file. For details on the definition file, see [E.1 System settings definition file](#).

#2

During a run of a job executed on event, if the execution condition of the job becomes satisfied, the job enters the *Wait for start* state and the event causes a new instance to be queued unless the queue is full. If the queue is full (that is, this maximum number is already reached), the event queue overflows and the event is discarded without triggering the job.

## C.2 Conversion rate of characters into bytes

Each character in the string you enter in the text box is counted differently in bytes depending on the character.

The following table describes the conversion rate of the characters into bytes.

Table C–2: Conversion rate of characters into bytes

Character		Bytes
Multibyte character		3
Newline character		2
Symbols	"	6
	'	6
	&	5
	<	4
	>	4

## C.3 Memory requirements

The memory requirements might change according to the settings status or use status of CPA. For details about the memory requirements, see the *Release Notes*.

## C.4 Disk space requirements

For details about the amount of disk space required when CPA runs, see the *Release Notes*.

## D. Files and Folders

This appendix lists files and folders in CPA.

The CPA installation folder varies depending on your operating system:

In a 32-bit operating system:

```
%ProgramFiles%\Hitachi\JP1CPA\
```

In a 64-bit operating system:

```
%ProgramFiles(x86)%\Hitachi\JP1CPA\
```

### D.1 Files and folders in CPA

The following table lists files and folders in CPA.

Table D–1: Files and folders in CPA

Folder	File	Description
<i>install-path</i>	ga.ini	Product information file
	ProductInfo.txt	Product type file
	Readme_en.txt	Readme in English
	Readme_ja.txt	Readme in Japanese
	Version.txt	Version information
<i>install-path</i> \bin\	--	Folder containing executable files
<i>install-path</i> \img\	--	Folder containing image files
%ProgramData%\HITACHI \jp1\jp1_default\JP1CPA\	--	Data folder
%ProgramData%\HITACHI \jp1\jp1_default\JP1CPA\addin\	--	<ul style="list-style-type: none"><li>Folder containing add-ins to link to other products</li><li>Folder containing add-ins for additional functions</li></ul>
	foral	Folder containing add-ins for the Windows log-on function
%ProgramData%\HITACHI \jp1\jp1_default\JP1CPA\conf\	--	Folder containing the system settings files
	usrprof.xml#	User mapping file
	agtprof.xml.model	Model file for the profile file
	config.xml.model	Model file for the system settings definition file
	config.xml	System settings definition file
	iesconf.xml.model	Model file for the message communication service definition file
	iesconf.xml	Message communication service definition file
	mailprof.ini	Mail profile



Folder	File	Description
	mailprof.ini.model	Model file for the mail profile
	recvmailprof.ini.model	Model file for the profile of mail reception
	itemdesign_config.xml	Item Design View settings file
	itemdesign_config.xml.model	Model file for the Item Design View settings file
	jobdesign_config.xml	Job Design View settings file
	jobdesign_config.xml.model	Model file for the Job Design View settings file
%ProgramData%\HITACHI\jpl\jpl_default\JP1CPA\data\	--	Folder containing data
%ProgramData%\HITACHI\jpl\jpl_default\JP1CPA\database\	--	Database folder
%ProgramData%\HITACHI\jpl\jpl_default\JP1CPA\log\	--	Folder containing log files
	msg.log{.n} {.n}: (Sequential number for each backup log file)	Product log file
	report.log{.n} {.n}: (Sequential number for each backup log file)	Report log file
	operate.log{.n} {.n}: (Sequential number for each backup log file)	Operation log file
%ProgramData%\HITACHI\jpl\jpl_default\JP1CPA\sys\	--	Folder containing system files
%ProgramData%\HITACHI\jpl\jpl_default\JP1CPA\work\	--	Work folder to be used by running actions

**Legend:**

-- : Not applicable

#

This file is created when user information is set with the cpasetumap command.

## D.2 Files and folders in CPA Option for AJS3

The following table lists files and folders in CPA.

Table D–2: Files and folders in CPA Option for AJS3

Folder	File	Description
<i>install-path</i>	ga.ini	Product information file
	ProductInfo.txt	Product type file
	Readme_en.txt	Readme in English
	Readme_ja.txt	Readme in Japanese

Folder	File	Description
	Version.txt	Version information
<i>install-path</i> \bin\	--	Folder containing executable files
<i>install-path</i> \img\	--	Folder containing image files
%ProgramData%\HITACHI \jp1\jp1_default\JP1CPA	--	Data folder
%ProgramData%\HITACHI \jp1\jp1_default\JP1CPA\addin\	--	Folder containing add-ins to link to other products
%ProgramData%\HITACHI \jp1\jp1_default\JP1CPA\addin\forajs \	--	Folder containing add-ins to link to AJS3
%ProgramData%\HITACHI \jp1\jp1_default\JP1CPA\addin\forajs \view\	--	Folder containing files for the custom job for CPA linkage
	cpacustomdef.exe	Program to define the custom job for CPA linkage
	ja-JP	Folder containing resource files for the program to define the custom job for CPA linkage
	ja-JP \cpacustomdef.resources.dll	Resource file for the program to define the custom job for CPA linkage
	CUSTOM_PC_CPALINK.gif	Icon of the custom job for CPA linkage
%ProgramData%\HITACHI \jp1\jp1_default\JP1CPA\conf\	--	Folder containing the system settings files
	iesconf.xml	Message communication service definition file
%ProgramData%\HITACHI \jp1\jp1_default\JP1CPA\data\	--	Folder containing data
%ProgramData%\HITACHI \jp1\jp1_default\JP1CPA\database\	--	Database folder
%ProgramData%\HITACHI \jp1\jp1_default\JP1CPA\log\	--	Folder containing log files
	msg.log{.n} {.n}: (Sequential number for each backup log file)	Product log file
	report.log{.n} {.n}: (Sequential number for each backup log file)	Report log file
%ProgramData%\HITACHI \jp1\jp1_default\JP1CPA\sys\	--	Folder containing system files
%ProgramData%\HITACHI \jp1\jp1_default\JP1CPA\work\	--	Work folder to be used by running actions

**Legend:**

-- : Not applicable

## E. Definition files

This appendix describes definition files required for CPA and CPA Option for AJS3 services. Before you can update a definition, ensure that all CPA and CPA Option for AJS3 services are stopped.

### E.1 System settings definition file

The following table describes the system settings definition file (*data-path\conf\config.xml*).

#### Important

- Do not edit elements that are not listed in the following table.
- Before you can update the system settings definition file, ensure that all services and programs provided by Client Process Automation are stopped.

Table E–1: System settings definition file

Element name	Attribute	Description	Acceptable value	Default value
program_option	--	Program options	--	--
+-- system	--	System settings	--	--
+-- project	--	Project name	char(30) Acceptable characters are limited to alphanumeric characters, underscores (_), and hyphens (-).	project
+-- job_ent_max <sup>#</sup>	--	The maximum number of jobs that can be registered	1 to 64 (Unit: the number of jobs)	16
+-- logfile_size	--	The size of the product log file	1 to 10,240 (Unit: KB)	512
+-- logfile_num	--	The number of backup product log files	1 to 9 (Unit: the number of files)	1
+-- operatelog	--	Operation log file settings	--	--
+-- logfile_size	--	The size of the operation log file	1 to 10,240 (Unit: KB)	512
+-- logfile_num	--	The number of backup operation log files	1 to 9 (Unit: the number of files)	1

Legend:

char(*n*): A string with a length of *n* bytes

--: Not applicable

#

Although this element does not exist in the system settings definition file upon installation, please remember to include the element name during setup.

When you want to edit the settings in the file for enhancement, we recommend that you first estimate the number of jobs executed per day and the number of concurrently executed actions to ensure that the changes you are planning to make do not adversely affect the operation.

Furthermore, consider increasing the size or the number of files of the product log file and the operation log file.

## E.2 Message communication service definition file

The following table describes the message communication service definition file (*data-path\conf\iesccconf.xml*).

### Important

Do not edit elements that are not listed in the following table.

Table E–2: Message communication service definition file

Element name	Attribute	Description	Acceptable value	Default value	Note
iesc_profile	--	Message communication service settings	--	--	--
+++ tcp	--	Communication settings	--	--	--
+++ port	--	Port number or service name (TCP/IP)	char(30)	cpamsgsrv	Required. The port number must correspond with the port number that is defined on the connected host. Setting value in the services file: cpamsgsrv 22260/tcp
+++ server	--	Service setting	--	--	--
+++ con_max	--	Maximum number of connections	64 to 256 (Unit: connections)	64	Specify the value obtained by multiplying the number of concurrent custom jobs for CPA linkage by 4.

Legend:

char(n): A string with a length of *n* bytes

--: Not applicable

## F. Processes

---

The following table lists the processes of CPA and CPA Option for AJS3.

Table F–1: Processes

Element name	Description
<code>cpaservice</code>	Links the job execution service to SCM <sup>#</sup> .
<code>+-- cpamanager</code>	Manages the job execution service.
<code>+-- cpaplayer</code>	Controls job executions. One process is generated for each existing job on executed on time (up to the maximum number of registered jobs) and one process for all jobs executed on event.
<code>+-- cpamonflchk</code>	Is generated at monitoring intervals if a virtual user name is specified for <code>Monitor file</code> (event monitor item) or <code>Monitor file event</code> (execution condition item).
<code>cpaiescscm</code>	Links the message communication service to SCM <sup>#</sup> .
<code>+-- cpaiescsvr</code>	Enables the message communication service to receive emails.

#

SCM: Service Control Manager provided with Windows

## G. Version Changes

---

This appendix describes the changes made in the manuals for each version.

### G.1 Changes in version 12-50

- The setup procedure for the AJS3 - View host was modified in the section for setting up the AJS3 linkage function.
- The execution user setting function was added, which enables a virtual user name to be specified as an execution user in the item definition.
- The `cpajobprint` command to output job or item definition information was added.
- The following environment variables were added to the table of the command execution environment for action items:
  - CPAJDS
  - CPAJCS

A description of how to use environment variables when an item is executed was also added.

- The virtual user name was added in the subsections of *List of settings* and *Settings* for the following items. The description of the return value 126 was also modified.
  - Action items
    - Command execution
    - Command execution by list
    - Command execution with recovery option
- The virtual user name was added in the subsections of *List of settings* and *Settings* for the following items. The description of the return value 0 was also modified.
  - Event monitor items
    - Monitor file
    - Monitor any event

The description of the return value 22 for the following item was also modified:

- Event monitor items
  - Monitor any event
- The virtual user name was added in the subsections of *List of settings* and *Settings* for the following items.
  - Execution condition item
    - Monitor file
    - Monitor any event
- The maximum number of concurrent connections from CPA Option for AJS3 to CPA was changed.
- The function as well as the description of argument `-a virtual-user-name` of the following command were modified:  
`cpasetumap` command
- The *Command execution as virtual user* column was added to the table for the command execution environment.
- The number of custom jobs for CPA linkage that can be concurrently executed from AJS3 can now be expanded.
- The following process was added
  - `cpamonflchk` process

- Changed how the product name is expressed in this manual.

## G.2 Changes in version 12-10

- Design modifications were made to each icon in the windows.
- The following statuses for jobs or actions were added.
  - Ended with warning
  - Ended normally (not executed)
  - Unmatched
- A function that allows users to create and use custom items suitable for their business operations is now supported. Along with this, the following command and files were added.
  - The `cpaitemsetup` command
  - A setting file for the Item Design View
  - A model file for the Item Design View settings file
- The following settings were added as installation settings.
  - Log-on account for the job execution service
  - Log-on account for the message communication service
- A function that allows the use of definition functions in the Job Design View to be restricted is now supported. Along with this, the following command and files were added.
  - The `cpausrenvsetup` command
  - A setting file for the Job Design View
  - A model file for the Job Design View settings file
- *PART 3: Operations* was added, which describes the functions necessary for the operation of jobs with JP1/CPA.
- *PART 4: Planning* was added, which describes the functions necessary for business operations planning with JP1/CPA.
- Information regarding the windows used with JP1/CPA was added to *PART 5: Reference*.
- In the *List of settings* for the *Send mail* action item, the value in the *Can specify inherited information?* column was changed in the following items:
  - To:
  - Cc:

In *Return values*, the description of *Others* in the *Value* column was also modified.
- In *Return values* for the following items, the descriptions of *3* and *Others* in the *Value* column were modified:
  - Event monitor items  
Monitor any event, Monitoring mail reception, Monitor Windows event
- The return values for the JP1/AJS3 custom job (custom job for JP1/CPA linkage) were added.
- The following command was added
  - `cpasleep` command
- The tag for setting the maximum number of registered jobs was added to the system settings definition file.

- For JP1 Version 12 JP1/Client Process Automation, the manual contents of the old edition *JP1 Version 12 JP1/Client Process Automation* (3021-3-D37-10(E)) have now been divided into the following two manuals.
  - *JP1 Version 12 JP1/Client Process Automation Configuration and Administration Guide* (3021-3-D37-20(E))
  - *JP1 Version 12 JP1/Client Process Automation Messages* (3021-3-D45(E))

The following table describes the correspondence between the *JP1 Version 12 JP1/Client Process Automation* (3021-3-D37-10(E)) and the *JP1 Version 12 JP1/Client Process Automation Configuration and Administration Guide* (3021-3-D37-20(E)):

JP1 Version 12 JP1/Client Process Automation (3021-3-D37-10(E))	JP1 Version 12 JP1/Client Process Automation Configuration and Administration Guide (3021-3-D37-20(E))
1. Overview of JP1/CPA	1. Overview of JP1/CPA
1.1 Features of JP1/CPA	1.1 Features of JP1/CPA
1.2 Overview of JP1/CPA functionality	--
1.3 List of JP1/CPA functions	--
1.4 Job components	1.2 Job components
1.5 Job execution unit and job execution ID	1.3 Job execution unit and job execution ID
1.6 Action functions	5.3 Creating action items
1.6.1 Action execution function	5.3.1 Basic settings for action items
1.6.2 Action flow execution function	5.4.1 Action flow execution function
1.6.3 State management function	1.1.4 State management for jobs and actions
1.6.4 Job operation function	4.1.4 Job operation function
1.7 Schedule functions	5.6 Creating schedule items
1.8 Event monitor functions	5.5 Event monitor functions
1.9 Standard items	8. Standard items
1.9.1(6) Windows log on	4.2.1 Using the Windows log-on function 8.2.6 Windows log on
2. Overview of JP1/CPA Option for AJS3	4.3 Using the JP1/AJS3 linkage function
2.1 JP1/CPA Option for AJS3 functions	4.3.1 List of JP1/CPA Option for AJS3 functions
2.2 JP1/AJS3 linkage function	4.3.2 JP1/AJS3 linkage function
3. Installation and Configuration of JP1/CPA	2. Installation and Configuration of JP1/CPA
4. Installation and Configuration of JP1/CPA Option for AJS3	3. Installation and Configuration of JP1/CPA Option for AJS3
5. Commands	9. Commands
6. Messages	Moved to the <i>JP1 Version 12 JP1/Client Process Automation Messages</i> (3021-3-D45(E)).
Appendix A. Troubleshooting	Appendix A. Troubleshooting
Appendix B. Tips to Operate Jobs	Appendix B. Tips to Operate Jobs
Appendix C. Usage Estimation	Appendix C. Usage Estimation
Appendix D. Files and Folders	Appendix D. Files and Folders
Appendix E. Logging	Moved to the <i>JP1 Version 12 JP1/Client Process Automation Messages</i> (3021-3-D45(E)).



JP1 Version 12 JP1/Client Process Automation (3021-3-D37-10(E))	JP1 Version 12 JP1/Client Process Automation Configuration and Administration Guide (3021-3-D37-20(E))
Appendix F. Definition files	Appendix E. Definition files
Appendix G. Processes	Appendix F. Processes
Appendix H. Version Changes	Appendix G. Version Changes
Appendix I. Reference Material for This Manual	Appendix H. Reference Material for This Manual

## G.3 Changes in version 12-01

- Windows Server 2019 was added as an applicable operating system.
- The Windows event monitor function that monitors the following Windows events was added:
  - Log-on
  - Log-off
  - Lock screen
  - Unlock screen

In response to this, the item Windows event monitor was added to the event monitor items and the execution condition items.

- Information regarding the length of job names and the unit were added.
- Information regarding one of the execution date item types Cycle was changed.  
Furthermore, a note was added to call attention to the specific ways in which to set the execution date item Year, month, and date.
- A note regarding one of the calendar item types Operation time period was added.
- Information regarding the following files, which are obtained when the event monitor detects an event, was added:
  - The file that contains the full name, size, and update time of the monitored files obtained by the file monitor
  - The file that contains a list of paths of the attached files obtained by the monitoring mail reception
- The Windows log-on function was added, which allows Windows users linked with virtual users on JP1/CPA to perform log-on and unlock screen operations.

In response to this, the item Windows log on was added to the action items.

Furthermore, the following commands and file were added:

- cpaalsetup command
- cpasetumap command
- User mapping file
- -1 (forced termination) was added as a return value of the following items:
  - Action item  
Send mail
  - Event monitor items  
Monitor file, Monitor any event, Monitoring mail reception

Information regarding the return value 0 of the following items was changed:

- Event monitor items

## Monitor file, Monitor any event, Monitoring mail reception

- A note regarding the File name to be monitored, which is specified for Monitor file (one of the event monitor items), was added.
- Information regarding satisfaction of monitoring conditions was added.
- Information regarding the history of executed jobs used with the JP1/AJS3 linkage function was added.
- Information regarding the mail profile used with the configurations for the Send mail item was changed.
- Information regarding the time zone of both the host on which JP1/CPA is installed and the one on which JP1/CPA Option for AJS3 is installed was added.
- A procedure for unsetting up JP1/CPA was added to the procedure for uninstalling the same.
- Information regarding the language setting in the Job Design View was added.  
In response to this, a migration procedure to be followed when modifying the language setting in the OS was added.
- Notes on the following commands have been added:

- `cpajobdesign` command
- `cpajobstop` command

The argument `-i job-name` was added to the `cpareport` command.

- The following messages were added.  
KNAO0913-E, KNAO0914-E, KNAO2710-W, KNAO2711-E, KNAO2712-I, KNAO2800-E, KNAO2801-E, KNAO2802-E, KNAO2803-E, KNAO2804-E, KNAO2805-E, KNAO2810-I
- Information regarding the action to be taken by the operator in response to the following message was changed:  
KNAO2426-E, KNAO2505-E, KNAO2507-E
- Information regarding the maximum number of profiles that can be registered was added.
- Folders for storing the add-in files for additional functions were added.
- Information regarding the character encoding used in the standard output and the standard error output was changed.

## H. Reference Material for This Manual

---

This appendix provides reference information, including various conventions, for this manual.

### H.1 Related publications

This manual is part of a related set of manuals. The manuals in the set are listed below (with the manual numbers):

- *JP1 Version 12 JP1/Automatic Job Management System 3 Configuration Guide* (3021-3-D24(E))
- *JP1 Version 12 JP1/Automatic Job Management System 3 Linkage Guide* (3021-3-D29(E))
- *JP1 Version 12 Job Management: Getting Started (Client Process Automation)* (3021-3-D36(E))
- *JP1 Version 12 JP1/Client Process Automation Messages* (3021-3-D45(E))

### H.2 Conventions: Abbreviations for product names

This manual uses the following abbreviations for product names:

Abbreviation		Full name or meaning
AJS3	AJS3 - Manager	JP1/Automatic Job Management System 3 - Manager
	AJS3 - Agent	JP1/Automatic Job Management System 3 - Agent
	AJS3 - View	JP1/Automatic Job Management System 3 - View
CPA	JP1/Client Process Automation	
CPA Option for AJS3	JP1/Client Process Automation Option for Automatic Job Management System 3	

### H.3 Conventions: Acronyms

This manual also uses the following acronyms:

Acronym	Full name or meaning
ASCII	American Standard Code for Information Interchange
API	Application Programming Interface
Azure AD	Azure Active Directory
CPU	Central Processing Unit
CR	Carriage Return
CSV	Comma Separated Values
DNS	Domain Name System
DTD	Document Type Definition
ID	IDentification

Acronym	Full name or meaning
IMAP	Internet Message Access Protocol
I/O	Input/Output
IP	Internet Protocol
LF	Line Feed
NetBIOS	Network Basic Input Output System
NFS	Network File System
NIC	Network Interface Card
NTFS	New Technology File System
OS	Operating System
PC	Personal Computer
PIN	Personal Identification Number
POP	Post Office Protocol
SCM	Service Control Manager
SMTP	Simple Mail Transfer Protocol
SMTP-AUTH	SMTP Authentication
TCP	Transmission Control Protocol
UTF	UCS Transformation Format
WOW64	Windows On Windows 64
XML	eXtensible Markup Language

## H.4 Conventions: KB, MB, GB, and TB

This manual uses the following conventions:

- 1 KB (kilobyte) is 1,024 bytes.
- 1 MB (megabyte) is 1,024<sup>2</sup> bytes.
- 1 GB (gigabyte) is 1,024<sup>3</sup> bytes.
- 1 TB (terabyte) is 1,024<sup>4</sup> bytes.

# Index

## A

- abbreviations for products 239
- acronyms 239
- action flow execution function (action function) 72
- action functions
  - action flow execution function 72
  - job operation function 52
- action items 144, 146
  - Command execution 146
  - Command execution by list 147
  - Command execution in window 151
  - Command execution with recovery option 149
  - Send mail 153
  - Windows log on 155
- actions, eventmonitors, and actionflows 14
- AJS3 linkage function 54
  - executing jobs 55
  - procedure to use function 56
  - system configuration 54

## C

- calendar items 86, 145
- calendar items (schedule function) 85
- clustering 44
- Command execution (action item) 146
- Command execution by list (action item) 147
- Command execution in window (action item) 151
- Command execution with recovery option (action item) 149
- command notation conventions 179
- configuration
  - CPA 28
  - CPA Option for AJS3 42
- configurations for AJS3 linkage function 30
- configurations for Monitoring mail reception 29
- configurations for Send mail item 28
- configurations of message communication service 42
- conventions
  - abbreviations for products 239
  - acronyms 239
  - fonts and symbols 6
  - KB, MB, GB, and TB 240
  - version numbers 7
- CPA

- configuration 28
- features of 14
- Installation 24
- Installing a new instance 24
- language setting 37
- notes on installation 26
- overview of 13
- Overwriting or upgrading an existing instance 25
- performing silent installation 26
- port number 36
- Requirements 23
- uninstallation 35
- cpaalsetup 205
- cpaecho 214
- cpaitemsetup 212
- cpajobdelete 184
- cpajobdesign 181
- cpajobentry 182
- cpajobprint 197
- cpajobrun 188
- cpajobstop 186
- CPA Option for AJS3
  - configuration 42
  - installation 40
  - installing a new instance 40
  - notes on installation 41
  - overwriting or upgrading an existing instance 41
  - port number 48
  - requirements 39
  - uninstallation 47
- CPA Option for AJS3 functions
  - list of 54
- cpapasswd 200
- cpareport 190
- cpasetfirewall 204
- cpasetprof 202
- cpasetumap 207
- cpasleep 215
- cpausrenvsetup 210
- custom job for CPA linkage 54, 174

## D

- definition files 231

## E

- event monitor functions 80
  - event monitoring for receiving mail 80
  - inherited information 81
  - list of 80
  - monitor any event 80
  - monitor file event 80
  - monitor the linkage from AJS3 80
  - monitor Windows event 80
- Event monitoring for receiving mail (execution condition item) 174
- event monitor items 144, 159
  - Monitor any event 161
  - Monitor file 159
  - Monitoring mail reception 164
  - Monitor Windows event 166
- execution condition items 144, 169
  - Event monitoring for receiving mail 174
  - Monitor any event 170
  - Monitor file event 169
  - Monitor the linkage from JP1AJS3 174
  - Monitor Windows event 176
- execution date/time and execution conditions 15
- execution date items 83, 144
- execution time items 85, 145
- Execution user setting function 60

## F

- features of CPA 14
- files and folders 228
- font conventions 6

## G

- GB meaning 240

## I

- inherited information (event monitor function) 81
- installation
  - CPA 24
  - CPA Option for AJS3 40
- installing a new instance
  - CPA Option for AJS3 40
- Installing a new instance
  - CPA 24

## J

- job components 20
- job execution unit and job execution ID 21
- job operation function (action function) 52

## K

- KB meaning 240

## L

- language setting
  - CPA 37
- list of commands 180
- list of CPA Option for AJS3 functions 54
- list of event monitor functions 80

## M

- MB meaning 240
- Monitor any event (event monitor item) 161
- Monitor any event (execution condition item) 170
- Monitor file (event monitor item) 159
- Monitor file event (execution condition item) 169
- Monitoring mail reception (event monitor item) 164
- Monitor the linkage from JP1AJS3 (execution condition item) 174
- Monitor Windows event (event monitor item) 166
- Monitor Windows event (execution condition item) 176

## N

- notes on installation
  - CPA 26
  - CPA Option for AJS3 41

## O

- overview of CPA 13
- overwriting or upgrading an existing instance
  - CPA Option for AJS3 41
- Overwriting or upgrading an existing instance
  - CPA 25

## P

- performing silent installation
  - CPA 26
- port number
  - CPA 36
  - CPA Option for AJS3 48

**R**

Reference Material for This Manual 239

requirements

CPA 23

CPA Option for AJS3 39

**S**

schedule function

Execution time items 85

schedule functions

Execution date items 83

schedule functions 83

calendar items 85

Send mail (action item) 153

setting up components for AJS3 linkage function 42

standard items 143

action items 144, 146

calendar items 86, 145

event monitor items 144, 159

execution condition items 144, 169

execution date items 83, 144

execution time items 85, 145

symbol conventions 6

**T**

TB meaning 240

tips to operate jobs 222

troubleshooting 217

**U**

understanding jobs 14

uninstallation

CPA 35

CPA Option for AJS3 47

usage estimation 226

using Windows event monitor function 33

using Windows log-on function 32

**V**

version number conventions 7

**W**

Windows log on (action item) 155

---

 **Hitachi, Ltd.**

6-6, Marunouchi 1-chome, Chiyoda-ku, Tokyo, 100-8280 Japan

---