

Command Reference

JobCenter

R15.2

-
- Windows, Windows Server, Microsoft Azure, Microsoft Excel and Internet Explorer are trademarks or registered trademarks of Microsoft Corporation in the United States and other countries.
 - UNIX is exclusively licensed by The Open Group which is a registered trademark in the United States and other countries.
 - HP-UX is a trademark of Hewlett-Packard.
 - AIX is a trademark of IBM Corporation.
 - Linux is a registered trademark of Linus Torvalds in the United States and other countries.
 - Oracle Linux, Oracle Clusterware and Java are registered trademarks of Oracle and/or its affiliates.
 - Red Hat is a registered trademark of Red Hat, Inc.
 - SUSE is a registered trademark of Novell, Inc.
 - NQS is a Network Queuing System that was developed by Sterling Software for NASA Ames Research Center.
 - SAP ERP, SAP NetWeaver BW and ABAP are trademarks or registered trademarks of SAP AG.
 - Amazon Web Services is a trademark of Amazon.com, Inc. or its affiliates registered in the United States and other countries.
 - iPad and Safari are trademarks of Apple, Inc. registered in the United States and other countries.
 - All other software and hardware products mentioned herein are trademarks or registered trademarks of their respective owners.

The symbols (R), TM, and (c) are omitted in this document.

Precautions on exporting

This product (software) falls under a category of technology where establishment specified in the foreign exchange currency law is regulated. Please file necessary applications, such as permission for trade, with the Japanese government prior to exporting this product. Contact the dealer where you purchased the product or our nearest sales office for information on documents necessary for filing such applications.

Introduction

This manual describes how to use commands of JobCenter.

Note that this document is subject to change without prior notice.

1. How to read this manual

If you are installing JobCenter or upgrading it

→ Read the JobCenter Installation Guide.

If you are using JobCenter for the first time

→ Read the Quick Start Guide in the order of its contents.

If you wish to understand the basic operating methods of JobCenter

→ Refer to the Basic Guide.

If you wish to understand the construction of environments and the setting of various functions

→ Refer to the Environment Guide.

If you wish to understand how to use commands of JobCenter




→ Read this manual in the order of its contents.

If you wish to know about other functions

→ Read the related manuals, referring to the manuals pertaining to particular functions to be used.

2. Legend

The legend used in this manual is explained below.

	Information that you should pay attention to.
	Supplementary information about a description in the text.
	Supplementary tip about a description in the text.
Note	Explanation of a note that appears in the text.
—	The underlined parts in the descriptions for the installation window for UNIX mean inputs from a keyboard.

3. Related manuals

The following are manuals related to JobCenter. They are contained in JobCenter Media.

For information on the latest manuals, refer to the download page of the JobCenter product site.

<https://www.nec.com/en/global/prod/masterscope/jobcenter/en/support.html>

Document title	Overview
JobCenter Installation Guide	Describes how to newly install JobCenter or upgrade it.
JobCenter Quick Start Guide (Japanese only)	Describes the basic functions of JobCenter, as well as a complete set of operations, for first-time users of JobCenter.
JobCenter Basic Guide	Describes the basic functions of JobCenter, as well as how to operate it.
JobCenter Environment Guide	Describes various setting methods on the construction of environments required for using JobCenter, transferring of environment, linkage with other products, etc.
JobCenter Guide for using NQS function (Japanese only)	Describes methods of using NQS functions, the basis of JobCenter, from JobCenter.
JobCenter Guide for using operation logs and execution logs	Describes the functions for acquiring operation logs and job network execution logs from JobCenter CL/Win, as well as how to set the functions.
JobCenter Command Reference	Describes JobCenter commands for performing job network entry and execution state referencing, etc. from the command line like a GUI.
JobCenter Guide for using Cluster function	Describes linkage methods for operating JobCenter in the cluster system.
JobCenter Guide for using Helper function (Japanese only)	Describes the following three functions that enable efficient operation of JobCenter using Excel: JobCenter Definition Helper (definition information maintenance), JobCenter Report Helper (business report creation), and JobCenter Analysis Helper (performance analysis).
JobCenter Guide for using SAP function (Japanese only)	Describes how to link JobCenter and SAP.
JobCenter Guide for using WebOTX Batch Server linkage function (Japanese only)	Describes how to link JobCenter and WebOTX Batch Server.
JobCenter Guide for Using the Web Function	Describes JobCenter CL/Web that is a function to monitor jobs by using a web browser.
JobCenter Guide for using the text definition function (Japanese only)	Describes how to define a job network, schedule, calendar, custom job template by using a text file.
JobCenter Guide for upgrading the version and applying the patches in a cluster environment (Japanese only)	Describes how to upgrade the version of JobCenter and apply the patches in a cluster environment.
JobCenter R15.2 Release Notes	Provides the information specific to JobCenter R15.2.

4. Revision history

Edition	Date of revision	Item	Format	Contents of revision
1	July 20, 2018	New	-	First Edition

Table of Contents

Introduction	iii
1. How to read this manual	iv
2. Legend	v
3. Related manuals	vi
4. Revision history	vii
1. Command List	1
1.1. List of commands that start up the site temporarily if the site stops	5
2. Commands of JobCenter CL/Win	6
2.1. nscl_open (Starting JobCenter CL/Win (GUI Screen) and Connecting to a Specified Host)	7
2.1.1. Function Description	7
2.1.2. Options	7
2.1.3. Return Values	8
2.1.4. Notes	8
2.2. nscl_term (Terminating JobCenter CL/Win (GUI Screen))	10
2.2.1. Function Description	10
2.2.2. Options	10
2.2.3. Return Values	10
2.3. Nscl_Submit (Submitting a Remote Host Job Network)	11
2.3.1. Function Description	11
2.3.2. Options	11
2.3.3. Return Values	14
2.3.4. Notes	15
2.4. Nscl_Refer (Referring to the Status of a Job Network of a Remote Host)	16
2.4.1. Function Description	16
2.4.2. Options	16
2.4.3. Return Values	18
2.4.4. Tracker ID Status Display	18
3. Commands of JobCenter MG/SV	20
3.1. jnwsummary (Outputting the Execution Results of Job Networks)	21
3.1.1. Function Description	21
3.1.2. Options	21
3.1.3. Return Values	27
3.1.4. Notes	27
3.1.5. Output Formats (CSV Format/Readable Format)	28
3.2. jnwschprt (Displaying the Calendar and Schedule Information of a Job Network)	40
3.2.1. Function Description	40
3.2.2. Options	40
3.2.3. Return Values	44
3.2.4. Related Files	45
3.2.5. Notes	45
3.2.6. CSV Format of Schedule Information	45
3.3. jnwop (control) (Performing an Operation on a Component of a Job Network)	48
3.3.1. Function Description	48
3.3.2. Options	49
3.3.3. Return Values	49
3.3.4. Notes	49
3.4. jnwsubmitcmd (Submitting a Job Network)	51
3.4.1. Function Description	51
3.4.2. Options	51
3.4.3. Return Values	55
3.4.4. Notes	55
3.5. nqsbg (Starting a Process Not Terminated at the End of a Job from within the Job)	56
3.5.1. Function Description	56

3.5.2. Options	56
3.5.3. Return Values	57
3.5.4. Notes	57
3.6. jnwevtmgr (Checking and Deleting Events)	58
3.6.1. Function Description	58
3.6.2. Options	59
3.6.3. Return Values	59
3.6.4. Notes	59
3.7. jnwevtcmd (Transmitting an Event)	60
3.7.1. Function Description	60
3.7.2. Options	60
3.7.3. Return Values	61
3.7.4. Notes	61
3.8. jnwdelete (Deleting a Job Network or Schedule)	62
3.8.1. Function Description	62
3.8.2. Options	62
3.8.3. Return Values	63
3.8.4. Notes	63
3.9. schctrl (Enabling or Disabling a Schedule)	64
3.9.1. Function Description	64
3.9.2. Options	64
3.9.3. Subcommands	64
3.9.4. Return Values	67
3.9.5. Notes	67
3.10. nqsstart (Starting Daemon Processes)	69
3.10.1. Function Description	69
3.10.2. Options	69
3.10.3. Return Values	69
3.10.4. Notes	70
3.11. nqsstop (Stopping Daemon Processes)	71
3.11.1. Function Description	71
3.11.2. Options	71
3.11.3. Return Values	71
3.11.4. Notes	71
3.12. nmapmgr (Managing the Network Configuration)	72
3.12.1. Function Description	72
3.12.2. Notes	73
3.12.3. Sub Commands	73
3.13. qmgr (Managing the Configuration and Operation)	77
3.13.1. Function Description	77
3.13.2. Notes	77
3.13.3. Privileges	77
3.13.4. Pager Function (UNIX Version)	77
3.13.5. Resource Limits	77
3.13.6. Sub Commands	79
3.14. jdh_download (Downloading Definition Information)	118
3.14.1. Function Description	118
3.14.2. Options	118
3.14.3. Return Values	122
3.14.4. Main Messages	122
3.14.5. Notes	122
3.15. jdh_upload (Uploading Definition Information)	124
3.15.1. Function Description	124
3.15.2. Options	124
3.15.3. Return Values	125
3.15.4. Main Messages	126

3.15.5. Notes	126
3.16. jdh_trkget (Getting Tracker Information)	127
3.16.1. Function Description	127
3.16.2. Options	127
3.16.3. Return Values	128
3.16.4. Main Messages	128
3.16.5. Notes	130
3.17. jpf_stat (Checking JPF File Information)	131
3.17.1. Function Description	131
3.17.2. Options	131
3.17.3. Return Values	131
3.17.4. Execution Result Items (Common)	132
3.18. jc_backup (Backing Up Configuration Information)	135
3.18.1. Function Description	135
3.18.2. Options	135
3.18.3. Return Values	135
3.18.4. Notes	135
3.19. jc_restore (Restoring Configuration Information)	136
3.19.1. Function Description	136
3.19.2. Options	136
3.19.3. Return Values	136
3.19.4. Notes	136
3.20. jdh_conv (Performing Mutual Conversion between JDF and JPF Files)	137
3.20.1. Function Description	137
3.20.2. Options	137
3.20.3. Return Values	138
3.20.4. Main Messages	138
3.20.5. Notes	139
3.21. sleep (sleep Command for Windows)	140
3.21.1. Function Description	140
3.21.2. Options	140
3.21.3. Return Values	140
3.21.4. Notes	140
3.22. spoolconv (Migrating User Definition Information of R12.10.x or Earlier)	141
3.22.1. Function Description	141
3.22.2. Options	141
3.22.3. Return Values	141
3.22.4. Notes	141
3.23. jc_iedata_conv (Converting Export Data to a JPF File)	142
3.23.1. Function Description	142
3.23.2. Options	142
3.23.3. Return Values	143
3.23.4. Main Messages	143
3.23.5. Notes	144
3.24. jc_perf (Analyzing the Execution Status)	145
3.24.1. Function Overview	145
3.24.2. Function Description	146
3.24.3. Return Values	161
3.24.4. Main Messages	161
3.24.5. Notes	161
3.25. trfdelete (Deleting Tracking Files)	162
3.25.1. Function Description	162
3.25.2. Return Values	162
3.25.3. Notes	162
3.26. JCSession (Running a Specified Command from a Unit Job as an Active Session)	163
3.26.1. Function Description	163

3.26.2. Options	163
3.26.3. Environment Variables	163
3.26.4. Return Values	164
3.26.5. How to Use This Command	164
3.26.6. Notes	166
3.27. setuserpwd (Updating the User Password)	168
3.27.1. Function Description	168
3.27.2. Options	168
3.27.3. Output Result	168
3.27.4. Return Values	169
3.28. paramctrl (Registering a Parameter in an Encrypted Database)	170
3.28.1. Function Description	170
3.28.2. Sub Commands	170
3.28.3. Options	170
3.28.4. Return Values	171
3.28.5. Notes	171
3.29. jgres (HTTP Daemon Dedicated for JobCenter MG/SV)	172
3.29.1. Function Description	172
3.29.2. Options	172
3.29.3. Configuration File	172
3.29.4. Access Log	173
3.29.5. Error Log	174
3.29.6. API	174
3.29.7. Return Values	175
3.29.8. Main Messages	175
3.29.9. Notes	176
3.30. jnwqctrl (Displaying or Deleting a Job Network Submission Reservation List Managed by the Trigger)	177
3.30.1. Function Description	177
3.30.2. Sub Commands	177
3.30.3. Options	177
3.30.4. Return Values	178
3.30.5. Example of the Execution Result	178
3.30.6. Notes	180
3.31. jc_usermgr (Managing the JobCenter users)	181
3.31.1. Description of functions	181
3.31.2. Subcommands	181
3.31.3. Options	181
3.31.4. Return Values	182
3.31.5. Main Messages	182
3.31.6. Notes	182
3.32. jc_usercfg (Creating the user environment, and browsing and changing the user environment setting)	184
3.32.1. Description of functions	184
3.32.2. Subcommands	184
3.32.3. Options	184
3.32.4. Output Result	185
3.32.5. Parameter List	186
3.32.6. Return Values	196
3.32.7. Main Messages	196
3.32.8. Notes	197
3.33. jc_syscfg (Browsing and changing the system environment setting)	199
3.33.1. Description of functions	199
3.33.2. Subcommands	199
3.33.3. Options	199
3.33.4. Output Result	199

3.33.5. Parameter List	200
3.33.6. Return Values	204
3.33.7. Main Messages	204
3.33.8. Notes	205
3.34. jc_permcfg (Browsing and Changing the Permission Settings)	206
3.34.1. Description of functions	206
3.34.2. Return Values	224
3.34.3. Main messages shared by commands	224
3.34.4. Notes	224
3.34.5. Restrictions	224
4. Commands for a Cluster Environment	225
4.1. cjcmksite (Creating a Site)	226
4.1.1. Function Description	226
4.1.2. Options	226
4.1.3. Return Values	226
4.1.4. Notes	226
4.2. cjcpcw (Starting, Monitoring, or Stopping a Daemon Process)	227
4.2.1. Function Description	227
4.2.2. Options	228
4.2.3. Maintenance Mode	228
4.2.4. Related Files	229
4.2.5. Return Values	229
4.2.6. Notes	229
4.3. cjcls (Listing the Sites Running on the Machine That Executed the Command)	230
4.3.1. Function Description	230
4.3.2. Options	231
4.3.3. Return Values	232
4.3.4. Notes	232
4.4. nqspath (Displaying a Database Path Name)	233
4.4.1. Function Description	233
4.4.2. Options	233
4.4.3. Return Values	233
4.5. nqsportkpr (Binding the Ports for JobCenter-related Services Temporarily)	234
4.5.1. Function Description	234
4.5.2. Options	234
4.5.3. Return Values	235
4.5.4. Notes	235
5. Commands for SAP ERP	236
5.1. sapclient (Establishing Linkage with SAP ERP)	237
5.1.1. Function Description	237
5.1.2. Options	237
5.1.3. Return Values	237
5.1.4. Related Files	237
5.2. sapcmd (Generating SAP ERP Connection Parameters)	238
5.2.1. Function Description	238
5.2.2. Options	238
5.2.3. Return Values	238
5.2.4. Related Files	238
6. Configuration Check Commands	239
6.1. jc_check (Checking JobCenter Configuration)	240
6.1.1. Function Description	240
6.1.2. Options	240
6.1.3. Notes	240
6.1.4. Example of the Execution Result (Windows Version)	241
6.2. depend_tool (Checking JobCenter Definition Validity)	243
6.2.1. Function Description	243

6.2.2. Options	243
6.2.3. Main Messages	243
6.2.4. Notes	245
6.3. jc_ping (Checking Communication with a Specified Site)	246
6.3.1. Function Description	246
6.3.2. Options	246
6.3.3. Return Values	247
6.3.4. Notes	247
6.3.5. Output Formats (Readable Format/JSON Format)	247
6.3.6. Main Messages	250
7. Commands for Collecting Information	254
7.1. jc_getinfo (Collecting All Primary Information Necessary to Identify the Cause of a Failure in JobCenter)	255
7.1.1. Function Description	255
7.1.2. Options	255
7.1.3. Notes	256
7.1.4. Example of the Execution Result (Windows Version)	256
7.2. clweb_getinfo (Collecting All Primary Information Necessary to Identify the Cause of a CL/Web Server Failure)	258
7.2.1. Function Description	258
7.2.2. Options	258
7.2.3. Notes	258
8. Command for the Analysis Helper Function	259
8.1. jdh_trkupdate (Updating the Content of the Local Data Folder of Analysis Helper)	260
8.1.1. Function Description	260
8.1.2. Options	260
8.1.3. Return Values	260
8.1.4. Main Messages	260
8.1.5. Example of the Execution Result	260
8.1.6. Notes	261

List of Tables

1.1. Command list	1
1.2. List of commands that start up the site temporarily if the site stops	5
2.1. Relationship between the specified options and the screen display	7
3.1. Command exit values and evaluation results	26
3.2. Item type names	28
3.3. Readable format of job network execution result information	38
3.4. Item type name	45
3.5. Character strings that can be used in \$command	48
3.6. Event list names and descriptions	58
3.7. Event list output formats	58
3.8. Messages displayed after an event is deleted	59
3.9. Messages displayed when the jnwevtcmd command is executed	60
3.10. Messages displayed when the jnwdelete command is executed	62
3.11. Database elements of the nmapmgr command	72
3.12. Examples of the time limit specification	78
3.13. Examples of the size limit specification	78
3.14. Support of conversion by the jc_iedata_conv command for Windows	144
6.1. Readable output format	248
6.2. Parameter names of the JSON format	250

1 Command List

Table 1.1, "Command list" lists the commands provided for JobCenter.

It is indicated whether an individual command is supported (✓) or not supported (N/A) for JobCenter CL/Win, JobCenter SV/MG, and each operating system.

If a command requires that an environment variable be set before it is executed for a cluster site, the "NQS_SITE settings" column shows either "S" (NQS_SITE needs to be set in advance) or "DB" (NQS_SITEDB needs to be set in advance) or both.

If "!" is shown in this column, the command does not work properly when NQS_SITE or NQS_SITEDB is set in advance.

"-" indicates that the command does not require setting NQS_SITE or any other environment variable in advance or is not affected by an environment variable setting.

For details, see the cautions given later.

Table 1.1. Command list

Command name	Function	Support for CL/Win	Support for MG/SV		NQS_SITE settings	See
			Windows	UNIX		
nscl_open	Starts JobCenter CL/Win (GUI screen) and connects to a specified host.	✓	N/A	N/A	-	2.1
nscl_term	Terminates JobCenter CL/Win (GUI screen).	✓	N/A	N/A	-	2.2
Nscl_Submit	Submits a job network.	✓	N/A	N/A	-	2.3
Nscl_Refer	Displays tracker information.	✓	N/A	N/A	-	2.4
jnwsummary	Displays the execution status of a job network.	N/A	✓	✓	S,DB	3.1
jnwschprt	Displays the calendar and schedule information.	N/A	✓	✓	S,DB	3.2
jnwop (control)	Performs an operation on a part of a job network.	N/A	✓	✓	S,DB	3.3
jnwsubmitcmd	Submits a user-defined job network.	N/A	✓	✓	S,DB	3.4
nqsbg	Starts a command as a background process that does not receive a signal from a unit job when that unit job ends.	N/A	N/A	✓	-	3.5
jnwevtmgr	Displays or deletes an event.	N/A	✓	✓	S,DB	3.6
jnwvtcmd	Sends an event.	N/A	✓	✓	S,DB	3.7
jnwdelete	Deletes a job network or schedule.	N/A	✓	✓	S	3.8
schctrl	Enables or disables a schedule.	N/A	✓	✓	S,DB	3.9
nqsstart	Starts a JobCenter daemon process.	N/A	N/A	✓	-	3.10
nqsstop	Stops a JobCenter daemon process.	N/A	N/A	✓	-	3.11

Command List

Command name	Function	Support for CL/Win	Support for MG/SV		NQS_SITE settings	See
			Windows	UNIX		
nmapmgr	Manages the JobCenter network configuration.	N/A	✓	✓	S	3.12
qmgr	Manages the JobCenter configuration and operation.	N/A	✓	✓	S	3.13
jdh_download	Downloads definition information.	N/A	✓	✓	-	3.14
jdh_upload	Uploads definition information.	N/A	✓	✓	-	3.15
jdh_trkget	Gets tracker information.	N/A	✓	✓	S,DB	3.16
jpf_stat	Displays JPF file information.	N/A	✓	✓	-	3.17
jc_backup	Backs up configuration information.	N/A	✓	✓	-	3.18
jc_restore	Restores configuration information.	N/A	✓	✓	-	3.19
jdh_conv	Performs mutual conversion between JDF and JPF files.	N/A	✓	✓	-	3.20
sleep	Windows-version sleep command	N/A	✓	N/A	-	3.21
spoolconv	Migrates user definition information of R13.1 or earlier.	N/A	N/A	✓	S	3.22
jc_iedata_conv	Converts export data to a JPF file.	N/A	✓	✓	-	3.23
jc_perf	Collects and analyzes data about the execution status of JobCenter.	N/A	✓	✓	S	3.24
trfdelete	Deletes the tracking file.	N/A	✓	✓	S,DB	3.25
JCSession	Runs a specified command from a unit job in an active session.	N/A	✓	N/A	-	3.26
setuserpwd	Updates the user password.	N/A	✓	N/A	-	3.27
paramctrl	Registers a parameter in an encrypted database.	N/A	✓	✓	S	3.28
jcres	HTTP daemon dedicated for JobCenter MG/SV	N/A	✓	✓	-	3.29
jnwqctrl	Displays or deletes a job network submission reservation list managed by the trigger function.	N/A	✓	✓	S	3.30
jc_usermgr	Manages the JobCenter users.	N/A	✓	N/A	-	3.31
jc_usercfg	Creates the user environment, and displays and changes the user environment settings.	N/A	✓	✓	S	3.32
jc_syscfg	Displays and changes the system environment settings.	N/A	✓	✓	S	3.33
jc_permcfg	Displays and changes the permission settings.	N/A	✓	✓	S	3.34

Command List

Command name	Function	Support for CL/Win	Support for MG/SV		NQS_SITE settings	See
			Windows	UNIX		
cjcmksite	Creates a cluster site.	N/A	✓	✓	(-)	4.1
cjcpw	Starts a local site or cluster site. The command monitors the JobCenter daemon process to check that it is running.	N/A	✓	✓	(-)	4.2
cjcls	Lists cluster sites.	N/A	✓	✓	(-)	4.3
nqspath	Displays the database path name used by a site.	N/A	N/A	✓	-	4.4
nqsportkpr	Binds nqs ports temporarily.	N/A	N/A	✓	-	4.5
sapclient	Checks that the user is logged in to SAP ERP. The command is used to set the audit level.	N/A	✓	✓	S	5.1
sapcmd	Creates a sample file of the connection parameter file. The command creates an encrypted SAP user password to be written in the parameter file.	N/A	✓	✓	S	5.2
jc_check	Checks whether there is any problem with the current environment.	N/A	✓	✓	!	6.1
depend_tool	Checks the validity of definition information.	N/A	✓	✓	S	6.2
jc_ping	Checks communication with a specified site.	N/A	✓	✓	S,DB	6.3
jc_getinfo	When a failure occurs, the command collects all the primary information necessary to identify its cause.	N/A	✓	✓	!	7.1
clweb_getinfo	When the CL/Web server fails, the command collects all the primary information necessary to identify the cause of the failure.	N/A	N/A	N/A	-	7.2
jdj_trkupdate	Updates the content of the local data folder of Analysis Helper.	N/A	N/A	N/A	-	8.1



■ Of the commands listed in the above table, those for which "S" is shown require that a site name be set in environment variable NQS_SITE immediately before the command is executed for a cluster site. In the case of the Windows version JobCenter, a command for which "DB" is shown requires that a database path of the site be added to environment variable NQS_SITEDB.

- A command for which "!" is shown does not work properly when NQS_SITE or NQS_SITEDB is set before the command is executed. Before executing the command, be sure to check that neither of these environment variables is set.
- In NQS_SITE, set the value of \$site-name, which is specified when a cluster site is created with cjcmmksite.
- In NQS_SITEDB, set the value of \$db_directory as a full path, which is specified when a cluster site is created with cjcmmksite. (Windows version only)
- In the case of the UNIX version, do not set NQS_SITE as a fixed user-specific environment variable in .login or .cshrc. Doing so may cause an unexpected malfunction. Set NQS_SITE each time immediately before you execute the command.
- In the case of the Windows version, do not set NQS_SITE or NQS_SITEDB as a fixed system environment variable, because doing so may cause an unexpected malfunction. Set the environment variable immediately before you execute the command.
- The command paths and user data storage paths of the Windows version are different from those of R12.7 and earlier.

1.1. List of commands that start up the site temporarily if the site stops

The following commands start up the DB process temporarily if these commands are executed while the target site has not started up.

If these commands are executed for the local site while `local_daemon=OFF` is set in `daemon.conf`, temporary startup of the DB process is suppressed, and an error occurs in execution of the commands. Change the settings to `local_daemon=SITE` or `local_daemon=COMPAT` before executing the commands for the local site.

Table 1.2. List of commands that start up the site temporarily if the site stops

Command name	Parameter to which to specify the site	Chapter number
<code>jdh_trkget</code>	<code>NQS_SITE</code> (environment variable)	3.16
<code>jc_backup</code>	<code>-c</code> option	3.18
<code>jc_restore</code>	<code>-c</code> option	3.19
<code>spoolconv</code>	<code>NQS_SITE</code> (environment variable)	3.22
<code>trfdelete</code>	<code>NQS_SITE</code> (environment variable)	3.25
<code>jc_check</code>	<code>NQS_SITE</code> (environment variable)	6.1

2 Commands of JobCenter CL/Win

This chapter describes the commands that are available on a machine in which JobCenter CL/Win is installed.

2.1. nscl_open (Starting JobCenter CL/Win (GUI Screen) and Connecting to a Specified Host)

```
%InstallDirectory%\nscl_open [-server $server] [-user $user [-password $password | -pw $passwordfile | -epw $EncryptPasswordfile]] [-mode { edit | view | ref }] [-d] [-h]
```

2.1.1. Function Description

- The nscl_open command starts JobCenter CL/Win (GUI screen) with the option-specified items set in the connection confirmation dialog box.
- To execute this command, you need to enter the password directly as a command argument (-password option), use the file containing the password (-pw option), or use the file created with the password file creation command (-epw option).

Use the password file creation command if you cannot enter the password directly in a command argument or create a file containing the password for security reasons. For details about the password file creation command, see [Section 2.3, “Nscl_Submit \(Submitting a Remote Host Job Network\)”](#).

Table 2.1. Relationship between the specified options and the screen display

Condition	Display
The -server, -user, and -password (-pw or -epw) options are all specified, and the -d option is not specified.	The post-connection window appears without displaying the connection confirmation dialog box.
None of the -server, -user, and -password (-pw or -epw) option is specified.	The connection confirmation dialog box appears with the connection destination server name, user name, and password fields blank.
Either the -server or -user option is not specified.	The connection confirmation dialog box appears with the connection destination server name or user name field blank.

2.1.2. Options

-server \$server

Specify the connection destination machine name (site name).

-user \$user

Specify the connection destination user name.

-password \$password

This option is valid only when -user is specified.

In \$password, specify the password of the connection destination user.

-pw \$passwordfile

This option is valid only when -user is specified.

Specify a password file using a full path.

The user password is written in the password file as is (without encryption).

-epw \$EncryptPasswordfile

This option is valid only when `-user` is specified.

Specify the file created with the password file creation command (`-m` option of `Nscl_Submit`), using a full path. For details, see [Section 2.3, "Nscl_Submit \(Submitting a Remote Host Job Network\)"](#).

`-mode {edit|view|ref}`

Specify the connection mode.

■ `-mode edit`

(Edit mode) You can create, delete, and change job networks, as well as control job networks and jobs. This is the default value assumed when the `-mode` option is omitted.

■ `-mode view`

(View mode) You cannot create, delete, or change job networks. You can control job networks and jobs.

■ `-mode ref`

(Ref mode) You cannot create, delete, or change job networks, nor can you control job networks or jobs.

`-d`

The connection confirmation dialog box is displayed.

If you specify this option when the `-server`, `-user`, and `-password` (`-pw` or `-epw`) options are all specified, the connection confirmation dialog box is displayed.

`-h`

Help is displayed. If you specify this option when any other options are specified, the other options are ignored.

2.1.3. Return Values

Return value	Description
0	Normal end
1	The specified option is invalid.
2	A server name is not specified in the <code>-server</code> option.
3	A user name is not specified in the <code>-user</code> option.
4	A password is not specified in the <code>-password</code> option. The password file specified in the <code>-pw</code> (or <code>-epw</code>) option cannot be found, or a read error occurred.
5	The <code>-password</code> (<code>-pw</code> or <code>-epw</code>) option is specified without the <code>-user</code> option.
6	A connection mode is not specified in the <code>-mode</code> option.
7	The specified connection mode is invalid.

2.1.4. Notes

- If the `-server`, `-user`, and `-password` (`-pw` or `-epw`) options are all specified, you cannot verify the validity of the server name, user name, and password specified in the arguments until the connection operation is actually performed.

Therefore, even if any of these arguments is invalid, the CL/Win (GUI) screen starts and a connection error is indicated on the CL/Win screen.

2.2. nscl_term (Terminating JobCenter CL/Win (GUI Screen))

```
%InstallDirectory%\nscl_term [-server $server [-user $user]] [-h]
```

2.2.1. Function Description

- The nscl_term command terminates all the JobCenter CL/Win connected to the machine that issues this command or the machine specified in the option.

2.2.2. Options

No option specified

All the JobCenter CL/Win running on the machine that issues the nscl_term command are terminated.

-server \$server

JobCenter CL/Win connected to the specified machine (site) is terminated.

-user \$user

Use this option with the -server option. JobCenter CL/Win connected to the machine specified in \$server by the user specified in \$user is terminated.

-h

Help is displayed. If you specify this option when any other options are specified, the other options are ignored.

2.2.3. Return Values

Return value	Description
0	Normal end
1	The specified option is invalid.
2	A server name is not specified in the -server option.
3	A user name is not specified in the -user option.

2.3. Nscl_Submit (Submitting a Remote Host Job Network)

■ Job network submission command

```
%InstallDirectory%\nscl_submit -s $server -u $user {-pw $passwordfile | -epw $EncryptPasswordfile}
-j $jnname [-t $YYYYMMDDhhmmss] [-to $SocketTimeoutSeconds] [-c $comment] [-pa $parameter] [-e $key=
$value | -i $inputfile] [-l] [-h]
```

■ Password file creation command

```
%InstallDirectory%\nscl_submit -m $password -o $outfile [-h]
```

2.3.1. Function Description

- The Nscl_Submit command submits a specified job network of a specified user of a local or remote host.
- The command returns a tracker ID when it ends successfully or an error message when it fails.
- The command is stored in the installation folder of CL/Win.
- To submit a job network, you need to use the file containing the password (-pw option) or the file created with the password file creation command (-epw option).

Use the password file creation command if you cannot create a file containing the password for security reasons.

2.3.2. Options

■ Job network submission command

-s \$server

Specify the connection destination machine name (site name).

In \$server, you need to specify a site name with a domain added to it in principle. (This does not apply when the site name is resolved without a domain and JobCenter has been set up using that name.)

If the connection destination remote machine has a number other than 611 assigned to jccombase, you can explicitly specify the connection destination port number in the format of <site-name:port-number>.

See details about port numbers given in <Environment Guide>Section 2.1, "TCP Ports Used by the JobCenter".

-u \$user

Specify the user name. If any of the following characters is used in the user name:

```
!"#$%&'()*+,-./:;<=>?@[\\]^_{|}~\t
```

the user name is judged to include an invalid character and an error results.



The maximum length of a user name that can be used for JobCenter is 15 bytes.

-pw \$passwordfile

Specify a password file using a full path.

The user password is written in the password file as is (without encryption).

`-epw $EncryptPasswordfile`

Specify the file created with the password file creation command, using a full path.

`-j $jnwname`

Specify the name of the job network to be submitted.



The job network name conventions are as follows.

- The length must be 40 bytes or less.
- At least one character must be specified.
- You must not use: !, \, ", #, \$, %, &, ', (,), *, (.), (.), /, :, ;, <, =, >, ?, @, [, \,], ^, `, {, |, }, ~, and \t.

`-t $YYYYMMDDhhmmss`

Specify the submission date and time. The format is <YYYYMMDDhhmmss>. Specify the date and time according to the local time zone (default: JST-9).

If you omit this option or specify a past date and time, the current time is used by default.

The date and time conventions are as follows.

- The length must be 14 bytes. (If the value of the month, day, hour, minute, or second is a single-digit number, add 0 as a prefix to make it a double-digit number.)
- The specified date must be within the ranges shown below, and the date and time combination must be appropriate.

1997 - 2037 (year), 1 - 12 (month), 1 - 31 (day); the time must be represented in 24-hour format.

This option is equivalent to the "Submission Time" setting described in in <Basic Guide>Section 3.2.1, "To Submit a Job Network Immediately".

`-to $SocketTimeoutSeconds`

Specify the length of time before the communication with the JobCenter server times out.

Specify the timeout value in the range between [0 - 3599] seconds.

If you do not specify this option, the default value of 30 seconds is assumed.

`-c $comment`

Specify a comment. If omitted, the option is treated as a null. The comment must not include any linefeed code.

If you want to include a space character in the comment, enclose the comment string in double quotation marks, as in `-c "comment1 comment2"`. Otherwise, only the first part of the comment (comment1 in this example) is treated as the comment string.

Note that there is no way to escape a double quotation mark itself.

The other restrictions are the same as those for "Comment" described in <Basic Guide>Section 3.2.1, "To Submit a Job Network Immediately". For more information, see this document as well.



The maximum length is 256 bytes.

`-pa $parameter`

The character string specified in `$parameter` can be set as the environment variable `NSJNW_PARAM` at execution of the unit job.

The parameter must not include any linefeed code.

Any space included in the parameter must be enclosed in double quotation marks, as in the `-c` option.

This option is equivalent to the parameter that can be entered in the dialog during the submission operation of CL/Win as described in "Environment variable" - "Setting the job parameter (NSJNW_PARAM)".

`-e $key=$value`

Set `$key` (environment variable name) = `$value` (environment variable value) as the environment variable when executing the unit job.

To set more than one environment variable, specify this option several times.

```
-e key1=value1 -e key2=value2 ...
```

To specify a character string including a space, enclose `$key=$value` entirely with double quotation marks.

```
-e "key1=a b c"
```

This option is equivalent to the parameter that can be entered in the dialog during the submission operation of CL/Win as described in "Environment variable" - "Setting more than one environment variable".



Specify the environment variable name of up to 51 bytes.

Spaces, tabs, special characters, `=`, `#`, `'`, and `"` cannot be used for the environment variable name.

Tabs, special characters, `#`, `'`, and `"` cannot be used for the environment variable value.

This option cannot be used at the same time as the `-i` option.

`-i $inputfile`

Set the data in the file specified by `$inputfile` as the environment variable used when the unit job is executed. Specify `$inputfile` in the format of `key (environment variable name) = value (environment variable value)`, and add a linefeed to specify more than one environment variable.

```
key1=value1
key2=value2
```

...

This option is equivalent to the parameter that can be entered in the dialog during the submission operation of CL/Win as described in "Environment variable" - "Setting more than one environment variable".



Specify the environment variable name of up to 51 bytes.

Spaces, tabs, special characters, =, #, ', and " cannot be used for the environment variable name.

Tabs, special characters, #, ', and " cannot be used for the environment variable value.

One line in the file can contain up to 8191 bytes.

This option cannot be used at the same time as the -e. option.

-l

The tracker ID returned when the command ends successfully is displayed in local time. If you omit this option, the tracker ID is displayed in GMT. The tracker ID is followed by (GMT) or (Local time).

(Example)

If you submit job network TEST1 with the submission date and time set to September 15, 2005, 14:00:00, the tracker ID is displayed as follows.

Without the -l option: TEST1.20090915050000 (GMT)

With the -l option: TEST1.20090915140000 (Local time)

-h

Help is displayed. If you specify this option when any other options are specified, the other options are ignored.

■ Password file creation command

-m \$password

Specify the user password.

-o \$outfile

Specify the output destination of the password file to be created, using a full path or relative path.

-h

Help is displayed. If you specify this option when any other options are specified, the other options are ignored.

2.3.3. Return Values

Return value	Description
0	Normal end. A tracker ID is output.
1	The specified option is invalid.

2	The server name specified in the -s option exceeds 256 bytes, or the connection destination port number is not within the range of 1 to 65535.
3	The user name in the -u option includes one or more invalid characters.
4	The password file cannot be found, or a read error occurred.
6	The job network name format is invalid.
7	The date and time format is invalid, or the length of the comment exceeds 40 bytes.
10	Network failure, communication error, invalid resolution of the specified server name, or password check error. The error message from the server is displayed in the standard error output.
11	An error occurred during logout from the remote server. The error message from the server is displayed in the standard error output.
20	The specified job network cannot be found, or some other error has been detected on the server side. The error message from the server is displayed in the standard error output.
22	A read error has occurred in the environment variable setting file.
30	An error other than the above occurred.

2.3.4. Notes

■ You may not be able to get the submission result as Nscl_Submit times out for reasons, such as because a network problem prevents communication data returned from the specified remote host from being received or because it takes time to submit the job network due to a high machine load. In this case, although the job network has successfully been submitted on the remote host side, the return value of the Nscl_Submit command does not reflect it.

Therefore, in order to judge whether Nscl_Submit has succeeded in submitting the job network, you need to check for tracker information by using Nscl_Refer. Nscl_Refer requires specifying a tracker ID, which consists of a job network name and a submission date and time. Therefore, you need to specify the submission date and time of the job network in the -t option of Nscl_Submit in advance.

(Example)

```
Nscl_Submit.exe -s server1 -u user1 -pw passwd.txt -j Jnw1 -t 20121010123456
```

```
Nscl_Refer.exe -s server1 -u user1 -pw passwd.txt -tr Jnw1.20121010123456 -l
```

If you do not specify the -l option in Nscl_Refer, the tracker ID is displayed in GMT.

2.4. Nscl_Refer (Referring to the Status of a Job Network of a Remote Host)

```
%InstallDirectory%nscl_refer -s $server -u $user {-pw $passwordfile | -epw $EncryptPasswordfile} -tr $trackerID [-l] [-to $SocketTimeoutSeconds] [-h]
```

2.4.1. Function Description

- The Nscl_Refer command lets you refer to specified tracker information of a specified user of a local or remote host.
- The command returns the tracker status when it ends successfully or an error message when it fails.
- The command is stored in the installation folder of CL/Win.
- To execute this command, you need to use the file containing the password (-pw option) or the file created with the password file creation command (-epw option).

Use the password file creation command if you cannot create a file containing the password for security reasons. For details about the password file creation command, see [Section 2.3, "Nscl_Submit \(Submitting a Remote Host Job Network\)"](#).

2.4.2. Options

-s \$server

Specify the connection destination machine name (site name).

In \$server, you need to specify a site name with a domain added to it in principle. (This does not apply when the site name is resolved without a domain and JobCenter has been set up using that name.)

If the connection destination remote machine has a number other than 611 assigned to jccombase, you can explicitly specify the connection destination port number in the format of <site-name:port-number>.

See details about port numbers given in <Environment Guide>Section 2.1, "TCP Ports Used by the JobCenter".



You can also specify an IP address as the site name. Here is how you specify an IP address.

- When not specifying a port number

Specify an IP address in the format of <IP-address>.

- When specifying a port number

In the case of an IPv4 address, specify it in the format of <IPv4-address:port-number>.

In the case of an IPv6 address, specify it in the format of <[IPv6-address]:port-number>.

-u \$user

Specify the user name. If any of the following characters is used in the user name:

```
!\"#$%&'()*+,-./:;<=>?@[\\]^_{|}~\t
```

the user name is judged to include an invalid character and an error results.



The maximum length of a user name that can be used for JobCenter is 15 bytes.

`-pw $passwordfile`

Specify a password file using a full path.

The user password is written in the password file as is (without encryption).

`-epw $EncryptPasswordfile`

Specify the file created with the password file creation command (`-m` option of `Nscl_Submit`), using a full path. For details, see [Section 2.3, "Nscl_Submit \(Submitting a Remote Host Job Network\)"](#).

`-tr $trackerID`

Specify the tracker ID to identify the tracker.

A tracker ID consists of a job network name and a submission date and time. The format is `<job-network-name:YYYYMMDDhhmmss>`.

Specify the tracker ID returned by the `nscl_submit` command.



■ The job network name conventions are as follows.

- The length must be 40 bytes or less.
- At least one character must be specified.
- You must not use: `!, \, ", #, $, %, &, ', (,), *, (,), (.), /, :, ;, <, =, >, ?, @, [, \,], ^, `, {, |, }, ~, and \t`.



■ The submission date and time conventions are as follows. Depending on the setting of the `-l` option, the date and time is displayed in either GMT or local time.

- The length must be 14 bytes. (If the value of the month, day, hour, minute, or second is a single-digit number, add 0 as a prefix to make it a double-digit number.)
- The specified date must be within the ranges shown below, and the date and time combination must be appropriate. 1997 - 2037 (year), 1 - 12 (month), 1 - 31 (day); the time must be represented in 24-hour format.

`-l`

Use this option if you want to specify the tracker ID in local time that is returned when the `Nscl_Submit` command successfully ends.

If you omit this option, specify the tracker ID in GMT.

(Example)

Suppose that you submit job network TEST1 with the submission date and time set to September 15, 2005, 14:00:00 and the `-l` option (local time display) specified in `Nscl_Submit`. When you set the returned tracker ID (TEST1.20090915140000) in the `-tr` option of this command, specify it as follows.

Without the -l option (specify in GMT): -tr TEST1.20090915050000

With the -l option (specify in local time): -tr TEST1.20090915140000 -l

-to \$SocketTimeoutSeconds

Specify the length of time before the communication with the JobCenter server times out.

Specify the timeout value in the range between [0 - 3599] seconds.

If you do not specify this option, 30 seconds is assumed.

-h

Help is displayed. If you specify this option when any other options are specified, the other options are ignored.

2.4.3. Return Values

Return value	Description
0	Normal end. The status of the specified tracker ID is displayed in the standard output. The content that is displayed as the tracker ID status is the same as that displayed in the tracker list of CL/Win. Note, however, that information on whether the tracker is [ARCHIVED] and the sub job network status is not displayed. For details about the status display, see Section 2.4.4, "Tracker ID Status Display" .
1	The specified option is invalid.
2	The server name specified in the -s option exceeds 256 bytes, or the connection destination port number is not within the range of 1 to 65535.
3	The user name in the -u option includes one or more invalid characters.
4	The password file cannot be found, or a read error occurred.
5	The tracker ID format is invalid.
6	The job network name format is invalid.
7	The date and time format is invalid.
10	Network failure, communication error, invalid resolution of the specified server name, or password check error. The error message from the server is displayed in the standard error output.
11	An error occurred during logout from the remote server. The error message from the server is displayed in the standard error output.
20	Reading the tracker data failed.
21	The specified tracker ID cannot be found.
30	An error other than the above occurred.

2.4.4. Tracker ID Status Display

Tracker ID status display

The character string shown in the "Status" column is displayed in the standard output.

The "Label" column shows the character string that jnwsummary displays as JNW STATUS when the same tracker is specified. This indicates the correspondence with the "Status" character strings displayed by NscI_Refer.

Status	Label
WAIT	WAIT
PREPARE	PREPARE
STOP(ERROR)	STOP(ERROR)
DIALOG	DIALOG
HOLD	HOLD
RUN[ERROR][WAIT]	RUN[ERROR][WAIT]
RUN(ERROR)	RUN(ERROR)
RUN[WARNING][WAIT]	RUN[WARNING][WAIT]
RUN[WARNING]	RUN[WARNING]
RUN[WAIT]	RUN[WAIT]
RUN	RUN
ABORT	ABORT
SKIPPED ON BRANCH	SKIPPED
DONE[SKIP]	SKIPPED
KILLED	KILLED
DONE(ERROR)	DONE(ERROR)
WARNING	DONE[WARNING]
SKIPPED[RUNLIM]	SKIPPED[RUNLIM]
DONE	DONE
RUN<RERUN>	RUN(RERUN)
RUN[ERROR]<RERUN>	RUN(ERROR)(RERUN)
DONE<RERUN>	DONE(RERUN)
DONE(ERROR)<RERUN>	DONE(ERROR)(RERUN)
UNKNOWN	UNKNOWN
[SKIP]	[SKIP]
[HOLD]	[HOLD]
[ARCHIVED]	[ARCHIVED]

3 Commands of JobCenter MG/SV

This chapter describes the commands that are available on a machine in which JobCenter MG/SV is installed.

3.1. jnwsummary (Outputting the Execution Results of Job Networks)

■ UNIX version

```
/usr/lib/nqs/gui/bin/jnwsummary [-o $output] [-u $user | -a] [-c $component] [-j $JNWsearch
[, $JNWsearch ...]] [-p $print_format [, $print_format ...]] [-t $testp] [-h]
```

■ Windows version

```
%InstallDirectory%\bin\jnwsummary [-o $output] [-u $user | -a] [-c $component] [-j $JNWsearch
[, $JNWsearch...]] [-p $print_format [, $print_format...]] [-t $testp] [-h]
```

3.1.1. Function Description

- The jnwsummary command searches and acquires the tracker information (execution results) of any given user for whom you have the access privilege in relation to a local or cluster site. It then displays the acquired tracker information or performs a status test.
- There are two types of options: general options that take effect for all forms of the command and function options that are used to change the command behavior.
- To specify the job network to be displayed or tested, use general option -j (job network search option).
- To display the detailed result information of the target tracker, specify function option -p (detailed display mode).
- The execution times and end statuses of the tracker and each related job are reported.
- To test the status (run, done, etc.) of the target tracker, specify function option -t (test mode).
- If you omit both the -p and -t options, only the list of target trackers is displayed (list display mode).

3.1.2. Options

1. General options

These options take effect for all forms of the jnwsummary command.

-o \$output

The result information is output to the file of the specified name.

If you omit this option, the information is output to the standard output.

-u \$user | -a

Specify the user whose tracker information you want to search for.

■ **-u \$user**

The command is targeted at the tracker of the user specified in \$user. Note that you need the privilege to access the data of that user.

■ **-a**

The command is targeted at all users.

■ If -u and -a are both specified

-a takes precedence.

■ If -u and -a are both omitted

The command is targeted at only the tracker of the user who executed it.

-c \$component

Specify the job network (JNW), sub job network (sub_JNW), and component directly at which this command is targeted. This option is valid only when the -p or -t option is specified.

To specify a component, use the format of "JNW:subJNW...subJNW/component-name". If the target component is a job network or sub job network, omit the part after "/". If the job network has a sub job network, use the format of "JNW/component-name".

(Example)

-c "ABC00:ABC01/JOB1": Component named JOB1 in the sub job network named ABC01 in job network ABC00

-c "ABC00:ABC01": Sub job network named ABC01 in job network ABC00

-c "ABC00/JOB1": Component named JOB1 in the job network named ABC00

-j \$JNWsearch

This is the job network search option.

By using this option in combination with the sub options, you can explicitly specify the tracker to search for. An error occurs if the command cannot find any job network that matches the specified conditions.

If you omit this option, the command is targeted at the latest tracker when there are multiple trackers having the same job network name (this is the same condition that occurs when you specify -j last).



If you specify the -t option, an error occurs when the search result is not only one job network. By specifying this option in combination with the sub options, you can control the output format of the command.

■ -j sub options

The -j option requires that you specify one or more of the sub options described below. You can specify multiple sub options by delimiting them with ",".

-j name=\$JNWname

In \$JNWname, specify the tracker (job network name) to search for. As the job network name, specify the root job network name.

If you omit this sub option, the command searches for all job network names.



You cannot use this sub option together with the trk sub option.

-j run

The command searches only for the currently running trackers.

If you omit this sub option, the command searches for job networks in all statuses.



You cannot use this sub option together with the `trk`, `prep`, or `last` sub option.

`-j last`

The command searches only for the latest execution results of all the search target job networks.

If you omit this sub option, the command searches for all the execution results of the search target job networks.



You cannot use this sub option together with the `trk`, `prep`, `run`, `date`, or `today` sub option.

`-j date=[YYYYMMDD[-yyyymmdd] | [YYYY/][M]M/DD[-[yyyy/][m]m/dd]]`

The command searches for all the job networks that started within the specified period. In `YYYY(/)MM(/)DD` and `yyyy(/)mm(/)dd`, specify the start date and end date of the search period, respectively.

If you omit `"-yyyymmdd"` or `"-[yyyy/][m]m/dd"` at the end, the specified period is only 1 day specified in `YYYY(/)MM(/)DD`.

(Example)

`-j date=20060925-20061001` (7 days from September 25 to October 1, 2006)

`-j date=20061001` (1 day of October 1, 2006)

If you omit this sub option, the command searches for job networks regardless of the period.



You cannot use this sub option together with the `trk`, `last`, or `today` sub option.

`-j today`

The command searches for all the job networks that started on the day when the command was executed.

If you omit this sub option, the command searches for job networks regardless of the period.



You cannot use this sub option together with the `trk`, `last`, or `date` sub option.

`-j prep`

The trackers in the prepare status are included in the search target.

If you omit this sub option, the command does not search for trackers in the prepare status.



You cannot use this sub option together with the run or last sub option.

`-j trk=JNWname.YYYYMMDDHHmmSS`

Specify a job network ID (tracker name) directly.

YYYYMMDDHHmmSS is the (expected) job network submission date and time expressed in GMT. If the value of the month, day, hour, minute, or second is a single-digit number, add 0 as a prefix.

If a job network is submitted using `jnwsubmitcmd`, the information is displayed in the standard output used at the time of submission.

You can use this sub option in combination with the prep sub option.



You cannot use this sub option together with the name, run, last, date, or today sub option. You cannot use it together with the `-a` option, either.

`-h`

Help is displayed. If you specify this option when any other options are specified, the other options are ignored.

2. Function options

When using the `jnwsummary` command, you can specify the option described below. If you omit this option (default), the command runs in list display mode.

Default (list display mode)

The current statuses of the target trackers are listed.

`-p` option (detailed display mode)

The detailed information of the target trackers is displayed.



By specifying this option in combination with the sub options, you can control the output format of the command.

■ `-p` sub option

The `-p` option requires that you specify one or more of the sub options described below. You can specify multiple sub options by delimiting them with `","`.

`-p format=csv | rd`

Specify the output format (CSV or readable).

- `-p format=csv`

The information is output in CSV format.

- `-p format=rd`

The information is output in readable format.

If you omit this sub option, the information is output in readable format.

For details about the output format, see "Output Formats (CSV Format/Readable Format)" later in this section.

For the -p sub options other than the format sub option, the information to be output differs depending on the output format specified in the format sub option.

[For the CSV format (format=csv)]

-p style={0|1}

Specify the information to be output. Here is how you specify this sub option.

- 0

The individual component information is not output. Only the tracker summary is displayed.

- 1

All the information is output.

If you omit this sub option, the command behaves as if 0 is set in style.

-p ujsonly

Only the individual component information is output, while the summary and job network component information are omitted.

This sub option is valid only when 1 is set in style.

-p erronly

The individual component information of only error components is output.

This sub option is valid only when 1 is set in style.

-p notitle

The output of the title line is suppressed.

[For the readable format (format=rd)]

-p style={0|1|2}

Specify the information to be output. Here is how you specify this sub option.

- 0

The individual component information is not output. Only the tracker summary is displayed.

- 1

Part of the individual component information is omitted.

- 2

All the information is output.

If you omit this sub option, the command behaves as if 0 is set in style.

`-p ujonly`

Only the summary and individual component information are output, while the job network component information is omitted.

This sub option is valid only when 1 (or 2) is set in style.

`-p erronly`

The individual component information of only error components is output.

This sub option is valid only when 1 (or 2) is set in style.

`-p output`

The command outputs the following: unit jobs; ERP jobs; WOBS jobs; STDOUT (spool) and STDERR (joblog) of custom jobs; IPKG STATUS of BI jobs; CHAIN LOG and CHAIN PROCESS of PC jobs.

This sub option is valid only when 1 (or 2) is set in style.

`-t option (test mode)`

The command evaluates the status of the search target tracker by using a specified test expression (pattern). You can judge the evaluation result from the exit value of this command.

The following table shows the command exit values and the evaluation results they indicate.

Table 3.1. Command exit values and evaluation results

Command value	exit	Evaluation result
0		The test expression is true.
1		The test expression is false.
2		The target component does not exist.
3		Some other error occurred.



An error occurs when the search result is not only one tracker. In that case, specify the target tracker explicitly by using the `-j` option.

■ `-t` sub option

Status test option `testp` is the sub option of `-t`. Be sure to specify a test expression specific to this option.

`-t testp="test-expression"`

In "test-expression", write an expression for evaluating the status of a job network or its job component. Write the expression in "Reverse Polish Notation".

The expression consists of a "pattern" and an "operator". When specifying two patterns in a row, delimit them with `"/"`.

- Formats of the test expression


```
expression="pattern operator"
expression="pattern / expression operator"
expression="pattern"
```

The operator is "+" (meaning "or"), "*" (meaning "and"), or "!" (meaning "not"). Each of these operators acts on the evaluation value.

After the specified component is evaluated, the pattern has a true or false evaluation value.

- Evaluation values for a test based on the component attribute

hold True when "hold" is specified for the component
 skip True when "skip" is specified for the component; this value is also true when "skip" is set due to a conditional branch.

- Evaluation values for a test based on the component status

The component must be in one of the following statuses.

wait True when the component is in the wait status
 submit True when the component is in the submit status
 run True when the component is in the run status
 done True when the component is in the done status (including when the component ended with an error)

- Evaluation values for a test based on other ancillary attributes

error True when an error has occurred in the component (and the component is in the done status at the same time)
 stopped True when an error has occurred in the component and the job network has stopped running (and the component is in the done status and in the error status at the same time)
 complete True when the component is in the done status and not in the error status
 archived The job network to which the component belongs is archived and cannot be re-executed (and the component is in the done status at the same time).

(Example)

testp="error!done*": The component is done and not in the error status.

testp="complete": The component is done and not in the error status.

testp="hold/skip+wait*": The component is in the wait status and in the hold or skip status.

3.1.3. Return Values

Return value	Description
0	Normal end (except when the -t option is used)
Other than 0	Abnormal end. An error message is output to the standard error output.

3.1.4. Notes

- If you have changed the standard output destination in the [Result] tab of Unit Job parameter setting, you cannot refer to results.

- When you execute the command for a cluster site, you need to set the site name in environment variable NQS_SITE in advance. (For Windows, you also need to set the database name in NQS_SITEDB.)

For details, see the cautions given in [Chapter 1, 「Command List」](#) .

■ In version R13.1 or later, this command cannot be used while JobCenter is stopped.

3.1.5. Output Formats (CSV Format/Readable Format)

1. CSV format (-p format=csv)

The format to be used when jnwsummary outputs job network execution result information in CSV format is described below.

The data to be output by jnwsummary consists of the following types of items.

- Job network summary
- Job network component information
- Unit job component information
- ERP job component information
- BI job component information
- PC job component information
- WOBS job component information



Note that a custom job component is handled as a unit job component.

[Item types]

The job network summary is output once in the entire command output. As for the other items, the number of times they are output is determined according to the items of the displayed component.

In a CSV file, the information is output consecutively for each individual type of item. Therefore, every unit job component belongs to a job network. However, this structure needs to be judged from the parent job network names written in the columns of the CSV file, not from the order of the rows.

You can select whether to output these items, by using the command options.

When the item of each type is output for the first time, the format is as follows.

```
#TYPE=Item type name
Item name (example: jnwname,jnwdate, ...)
Item value (example: rootJNW,19991220133000, ...)
```

Table 3.2. Item type names

Item type name	Description
SUMMARY	Job network summary
JNW	Job network component information
UNITJOB	Unit job component information
ERPJOB	ERP job component information
BIJOB	BI job component information
PCJOB	PC job component information

WOBSJOB	WOBS job component information
---------	--------------------------------

When the item of another type is output, a blank row is inserted. You can suppress the output of the title row, by using the relevant command option.

For the individual item types, the item names shown below are output with comma delimiters. The data of each item is output on a column-by-column basis.

If there is no information for the item, null data is set.

■ #TYPE=SUMMARY

Item name	Description
jnwname	Job network name
jnwdate	Job network submission date and time (GMT) (YYYYMMDDHHmmSS)
user	User
machine	Execution machine
starttime	Actual start time Common time format (YYYYY/MM/DD HH:mm:SS)
endtime	Actual end time
amounttime	Value of the execution time (in seconds)
expstarttime	Expected start time
expendtime	Expected end time
status	Current status As the status, one of the following is output. <ul style="list-style-type: none"> ■ For Windows <ul style="list-style-type: none"> prepare Waiting in the set status wait Waiting in the unset status wait_hold Waiting in the hold status wait_skip Waiting in the skipped status submit Submitting run Running normally run_warn Running in the warning status run_error Running in the error status run_dialog Waiting for a dialog box done Done successfully or done with a warning done_skip Aborted by the skip process error Done with an error errorstop Done with an error and stopped ■ For UNIX <ul style="list-style-type: none"> wait Prepared run Submitting, running, waiting for a dialog box, or done with an error and stopped

Commands of JobCenter MG/SV

	done Done successfully, done with a warning, done with an error, or aborted by the skip process
totalcomponent	Total number of components
totalwait	Total number of waiting components
totalwaithold	Total number of waiting components that in the hold status
totalwaitskip	Total number of waiting components to be skipped
totalrun	Total number of running components
totaldone	Total number of done components
totaldoneskip	Total number of done components that are skipped
totalerror	Total number of error components
ujcomponent	Total number of unit jobs
ujwait	Total number of waiting unit jobs
ujwaithold	Total number of waiting unit jobs that are held
ujwaitskip	Total number of waiting unit jobs to be skipped
ujrun	Total number of running unit jobs
ujdone	Total number of done unit jobs
ujdoneskip	Total number of done unit jobs that are skipped
ujerror	Total number of error unit jobs
jnwcomponent	Total number of sub job networks
jnwwait	Total number of waiting sub job networks
jnwwaithold	Total number of waiting sub job networks that are held
jnwwaitskip	Total number of waiting sub job networks to be skipped
jnwrun	Total number of running sub job networks
jnwdone	Total number of done sub job networks
jnwdoneskip	Total number of done sub job networks that are skipped
jnwerror	Total number of error sub job networks
ERPcomponent	Total number of ERP jobs
ERPwait	Total number of waiting ERP jobs
ERPwaithold	Total number of waiting ERP jobs that are held
ERPwaitskip	Total number of waiting ERP jobs to be skipped
ERPrun	Total number of running ERP jobs
ERPdone	Total number of done ERP jobs
ERPdoneskip	Total number of done ERP jobs that are skipped
ERPerror	Total number of error ERP jobs
BIcomponent	Total number of BI jobs
BIwait	Total number of waiting BI jobs
BIwaithold	Total number of waiting BI jobs that are held
BIwaitskip	Total number of waiting BI jobs to be skipped
BIrun	Total number of running BI jobs
BIdone	Total number of done BI jobs
BIdoneskip	Total number of done BI jobs that are skipped

Blerror	Total number of error BI jobs
othercomponent	Number of other components
otherwait	Number of other waiting components
otherwaithold	Number of other waiting components that are held
otherwaitskip	Number of other waiting components to be skipped
otherrun	Number of other running components
otherdone	Number of other done components
otherdoneskip	Number of other done components that are skipped
othererror	Number of other error components

■ #TYPE=JNW (job network)

The number of components does not include sub job networks in a job network.

Item name	Description
type	Fixed to "JNW".
parentjnw	Parent job network name (JNW:subJNW...)
jnwdate	Job network submission date and time (GMT) (YYYYMMDDHHmmSS)
name	Job network name
starttime	Actual start time Common time format (YYYYY/MM/DD HH:mm:SS)
endtime	Actual end time
amounttime	Execution time (in seconds)
expstarttime	Expected start time
expendtime	Expected end time
status	Current status As the status, one of the following is output. <ul style="list-style-type: none"> ■ For Windows <ul style="list-style-type: none"> prepare Waiting in the set status wait Waiting in the unset status wait_hold Waiting in the hold status wait_skip Waiting in the skipped status submit Submitting run Running normally run_warn Running in the warning status run_error Running in the error status run_dialog Waiting for a dialog box done Done successfully or done with a warning done_skip Aborted by the skip process error Done with an error errorstop Done with an error and stopped ■ For UNIX

Commands of JobCenter MG/SV

	wait	Prepared
	run	Submitting, running, waiting for a dialog box, or done with an error and stopped
	done	Done successfully, done with a warning, done with an error, or aborted by the skip process
totalcomponent	Total number of components	
totalwait	Total number of waiting components	
totalwaithold	Total number of waiting components that in the hold status	
totalwaitskip	Total number of waiting components to be skipped	
totalrun	Total number of running components	
totaldone	Total number of done components	
totaldoneskip	Total number of done components that are skipped	
totalerror	Total number of error components	
ujcomponent	Total number of unit jobs	
ujwait	Total number of waiting unit jobs	
ujwaithold	Total number of waiting unit jobs in the hold status	
ujwaitskip	Total number of waiting unit jobs to be skipped	
ujrun	Total number of running unit jobs	
ujdone	Total number of done unit jobs	
ujdoneskip	Total number of done unit jobs that are skipped	
ujerror	Total number of error unit jobs	
jnwcomponent	Total number of sub job networks	
jnwwait	Total number of waiting sub job networks	
jnwwaithold	Total number of waiting sub job networks in the hold status	
jnwwaitskip	Total number of waiting sub job networks to be skipped	
jnwrun	Total number of running sub job networks	
jnwdone	Total number of done sub job networks	
jnwdoneskip	Total number of done sub job networks that are skipped	
jnwerror	Total number of error sub job networks	
ERPcomponent	Total number of ERP jobs	
ERPwait	Total number of waiting ERP jobs	
ERPwaithold	Total number of waiting ERP jobs in the hold status	
ERPwaitskip	Total number of waiting ERP jobs to be skipped	
ERPrun	Total number of running ERP jobs	
ERPdone	Total number of done ERP jobs	
ERPdoneskip	Total number of done ERP jobs that are skipped	
ERPerror	Total number of error ERP jobs	
BIcomponent	Total number of BI jobs	
BIwait	Total number of waiting BI jobs	
BIwaithold	Total number of waiting BI jobs in the hold status	
BIwaitskip	Total number of waiting BI jobs to be skipped	

Blrun	Total number of running BI jobs
Bldone	Total number of done BI jobs
Bldoneskip	Total number of done BI jobs that are skipped
Blerror	Total number of error BI jobs
othercomponent	Number of other components
otherwait	Number of other waiting components
otherwaithold	Number of other waiting components in the hold status
otherwaitskip	Number of other waiting components to be skipped
otherrun	Number of other running components
otherdone	Number of other done components
otherdoneskip	Number of other done components that are skipped
othererror	Number of other error components

■ #TYPE=UNITJOB (unit job component information)

Item name	Description
type	Fixed to "UNITJOB".
parentjnw	Parent job network name (JNW:subJNW...)
jnwdate	Job network submission date and time (GMT) (YYYYMMDDHHmmSS)
name	Unit job component name
starttime	Actual start time Common time format (YYYYY/MM/DD HH:mm:SS)
endtime	Actual end time
amounttime	Execution time (in seconds)
expstarttime	Expected start time
expendtime	Expected end time
status	Current status As the status, one of the following is output. <ul style="list-style-type: none"> ■ For Windows <ul style="list-style-type: none"> wait Waiting wait_hold Waiting and held wait_skip Waiting and skipped submit Submitting run Running run_warn Done with a warning done Done successfully done_skip Done by the skip process error Done with an error errorstop Done with an error and stopped ■ For UNIX <ul style="list-style-type: none"> wait Waiting

	submit Submitting run Running done Done successfully, done with a warning, or done by the skip process errorpass Done with an error errorstop Done with an error and stopped
exitcode	Job exit value (No value is set in the case of an internal error.)
errcode	Error code
requestid	Request ID
machine	Job execution machine name

■ #TYPE=ERPJOB (ERP job component information)

Item name	Description
type	Fixed to "ERPJOB".
parentjnw	Parent job network name (JNW:subJNW...)
jnwdate	Job network submission date and time (GMT) (YYYYMMDDHHmmSS)
name	ERP job component name
starttime	Actual start time Common time format (YYYYY/MM/DD HH:mm:SS)
endtime	Actual end time
amounttime	Execution time (in seconds)
expstarttime	Expected start time
expendtime	Expected end time
status	Current status As the status, one of the following is output. <ul style="list-style-type: none"> ■ For Windows <ul style="list-style-type: none"> wait Waiting wait_hold Waiting and held wait_skip Waiting and skipped submit Submitting run Running done Done successfully done_skip Done by the skip process error Done with an error errorstop Done with an error and stopped ■ For UNIX <ul style="list-style-type: none"> wait Waiting submit Submitting run Running done Done successfully or done by the skip process

	errorpass Done with an error errorstop Done with an error and stopped
machine	Job execution machine name

■ #TYPE=BIJOB (BI job component information)

Item name	Description
type	Fixed to "BIJOB".
parentjnw	Parent job network name (JNW:subJNW...)
jnwdate	Job network submission date and time (GMT) (YYYYMMDDHHmmSS)
name	BI job component name
starttime	Actual start time Common time format (YYYYY/MM/DD HH:mm:SS)
endtime	Actual end time
amounttime	Execution time (in seconds)
expstarttime	Expected start time
expendtime	Expected end time
status	Current status As the status, one of the following is output. <ul style="list-style-type: none"> ■ For Windows <ul style="list-style-type: none"> wait Waiting wait_hold Waiting and held wait_skip Waiting and skipped submit Submitting run Running done Done successfully done_skip Done by the skip process error Done with an error errorstop Done with an error and stopped ■ For UNIX <ul style="list-style-type: none"> wait Waiting submit Submitting run Running done Done successfully or done by the skip process errorpass Done with an error errorstop Done with an error and stopped

■ #TYPE=PCJOB (PC job component information)

Item name	Description
type	Fixed to "PCJOB".
parentjnw	Parent job network name (JNW:subJNW...)

jnwdate	Job network submission date and time (GMT) (YYYYMMDDHHmmSS)
name	PC job component name
starttime	Actual start time Common time format (YYYYY/MM/DD HH:mm:SS)
endtime	Actual end time
amounttime	Execution time (in seconds)
expstarttime	Expected start time
expendtime	Expected end time
status	Current status As the status, one of the following is output. <ul style="list-style-type: none"> ■ For Windows <ul style="list-style-type: none"> wait Waiting wait_hold Waiting and held wait_skip Waiting and skipped submit Submitting run Running done Done successfully done_skip Done by the skip process error Done with an error errorstop Done with an error and stopped ■ For UNIX <ul style="list-style-type: none"> wait Waiting submit Submitting run Running done Done successfully or done by the skip process errorpass Done with an error errorstop Done with an error and stopped

■ #TYPE=WOBSJOB (WOBS job component information)

Item name	Description
type	Fixed to "WOBSJOB".
parentjnw	Parent job network name (JNW:subJNW...)
jnwdate	Job network submission date and time (GMT) (YYYYMMDDHHmmSS)
name	WOBS job component name
starttime	Actual start time Common time format (YYYYY/MM/DD HH:mm:SS)
endtime	Actual end time
amounttime	Execution time (in seconds)
expstarttime	Expected start time


```
e,totaldoneskip,totalerror,ujcomponent,ujwait,ujwaithold,ujwaitskip,ujrun,ujdone,
ujdoneskip,ujerror,jnwcomponent,jnwwait,jnwwaithold,jnwwaitskip,jnwrun,jnwdone
e,jnwdoneskip,jnwerror,ERPcomponent,ERPwait,ERPwaithold,ERPwaitskip,ERPrun,ERPdone,
ERPDoneskip,ERPerror,BIcomponent,BIwait,BIwaithold,BIwaitskip,BIrun,BIdone,
BIdoneskip,BIerror,othercomponent,otherwait,otherwaithold,otherwaitskip,otherrun,
otherdone,otherdoneskip,othererror
JNW,rootJNW,20060818050507,subJNW,2006/08/18 14:05:08,,0,2006/08/18 14:05:07,2006/08/18
14:05:07,run,2,0,0,0,1,1,0,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
JNW,"",20060818050507,rootJNW,2006/08/18 14:05:08,,0,2006/08/18 14:05:07,2006/08/18
14:05:07,run,4,1,1,0,1,2,0,0,1,0,0,0,0,1,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,0,1,0
,0,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
#TYPE=UNITJOB
type,parentjnw,jnwdate,name,starttime,endtime,amounttime,expstarttime,expendtime,
status,exitcode,errcode,requestid,machine
UNITJOB,rootJNW:subJNW,20060818050507,JOB1,2006/08/18 14:05:12,2006/08/18 14:05:13,1,2006/08/18
14:05:07,2006/08/18 14:05:07,done,0,0,3.ORGAN,ORGAN
UNITJOB,rootJNW,20060818050507,JOB1,2006/08/18 14:05:10,2006/08/18 14:05:10,0,2006/08/18
14:05:07,2006/08/18 14:05:07,done,0,0,2.ORGAN,ORGAN
#TYPE=ERPJOB
type,parentjnw,jnwdate,name,starttime,endtime,amounttime,expstarttime,expendtime,
status,machine
ERPJOB,rootJNW:subJNW,20060818050507,JOB2,2006/08/18 13:33:24,,0,2006/08/18 14:05:07,2006/08/18
14:05:07,run,-
ERPJOB,rootJNW,20060818050507,JOB2,2006/08/18 13:33:22,2006/08/18 13:33:23,1,2006/08/18
14:05:07,2006/08/18 14:05:07,done,-
#TYPE=BIJOB
type,parentjnw,jnwdate,name,starttime,endtime,amounttime,expstarttime,expendtime,
status,
BIJOB,rootJNW,20060818050507,JOB3,,0,2006/08/18 14:05:07,2006/08/18 14:05:07,wait_hold,
```

2. Readable format

The format to be used when jnwsummary outputs job network execution result information in readable format is described below.

The information listed below is output in a predetermined format. If there is no information for an item, "-" is shown.

Table 3.3. Readable format of job network execution result information

Item name	Description
Tracker ID (TRACKER-ID)	Tracker ID to identify a job network
User name (USER)	Name of the user who submitted the job network
Actual start time (START TIME)	Time when the processing actually started
Actual end time (END TIME)	Time when the processing actually ended
Expected start time (START TIME(EXP.))	Start time expected from the job network submission time
Expected end time (END TIME(EXP.))	End time expected from the job network submission time
Execution time (in seconds) (AMOUNTTIME)	Time from the actual start to the actual end

Status (STATUS)	Current status of the component
Number of components (COUNT)	<p>Total number of components defined in the job network</p> <p>The components of sub job networks are also counted.</p> <p>The information counted for each component and each status is also output.</p> <p>The target components are as follows.</p> <ul style="list-style-type: none"> ■ All information (ALL) ■ Unit job (UJ) ■ ERP job (ERP) ■ BI job (BI) ■ Sub job network (JNW) ■ PC job (PC) ■ WOBS job (WOBSJ) ■ Custom job (CSTMJ) ■ Other (OTHER) <p>Regarding these components, the following statistics are output.</p> <ul style="list-style-type: none"> ■ Total (TOTAL) ■ Waiting (WAIT) ■ Waiting components in the hold status (WAIT(hold)) ■ Waiting components to be skipped (WAIT(skip)) ■ Running (RUN) ■ Done (DONE) ■ Done components skipped (DONE(skip)) ■ Error components (ERROR)
Exit code (EXIT CODE)	Exit value or error code of the unit job (shell)
Execution machine name (EXEC MAC)	Name of the machine on which the job was executed
Request ID (REQUEST-ID)	Request ID used to process the unit job
Standard output (STDOUT)	Output only when specially specified
Standard error output (STDERR)	Output only when specially specified

3.2. jnwschprt (Displaying the Calendar and Schedule Information of a Job Network)

■ UNIX version

```
/usr/lib/nqs/gui/bin/jnwschprt [-lNote1] [-lNote2] [-o $output] [-u $user | -a] [-d $date] [-p $starttime]

/usr/lib/nqs/gui/bin/jnwschprt -s [-lNote2] [-o $output] [-f {csv | rd}] [-u $user] [-c | +c] [-C | +C] -j $jnwnname -d $date [-p $starttime]

/usr/lib/nqs/gui/bin/jnwschprt -s [-lNote2] [-o $output] {-f {csv | rd}} [-u $user] [-c | +c] [-C | +C] -i $jnwnname.YYYYMMDDHHmmSS
```

Note1 "l" in lowercase

Note2 "l" in uppercase

■ Windows version

```
%InstallDirectory%\bin\jnwschprt [-lNote1] [-lNote2] [-o $output] [-u $user | -a] [-d $date] [-p $starttime]

%InstallDirectory%\bin\jnwschprt -s [-lNote2] [-o $output] [-f {csv | rd}] [-u $user] [-c | +c] [-C | +C] -j $jnwnname -d $date [-p $starttime]

%InstallDirectory%\bin\jnwschprt -s [-lNote2] [-o $output] [-f {csv | rd}] [-u $user] [-c | +c] [-C | +C] -i $jnwnname.YYYYMMDDHHmmSS
```

Note1 "l" in lowercase

Note2 "l" in uppercase

3.2.1. Function Description

- The jnwschprt command displays the scheduled tracker information for a given job network of a given user for whom you have the access privilege in relation to the local or cluster site in which the command is executed.
- In the first syntax, the command lists the trackers scheduled to be executed on the specified date. If you do not specify a date, the command lists the trackers scheduled to be executed on the day when the command is executed.
- In the second syntax, if there are multiple trackers scheduled to be executed at times on the date specified for the specified job network, the command displays the unit job information concerning the tracker scheduled for the oldest time point, taking into consideration the calendar branch on the specified date. However, if there are multiple running trackers on the specified date and they include unit jobs yet to be executed, the command selects the tracker whose submission date is the newest of all the running trackers and displays its unit job information.
- In the third syntax, the command lists the unit jobs of a specified tracker ID, taking into consideration the calendar branch on the (scheduled) execution date. A tracker ID of a past date and time can also be specified.

3.2.2. Options

-l (Lowercase l)

Job networks are listed.

-s

The schedule information of the jobs of a specified job network is output.

-l (Uppercase i)

Reading the configuration file is prohibited.

-o \$output

The execution result is output to the file specified in output. If you specify "-", the result is output to the standard output.

-u \$user | -a

Specify the user whose job network you want to search for.

■ -u \$user

The job network search is performed for the user name specified in user. Note that you cannot execute the command if you do not have the file access privilege.

■ Both -a and -u are omitted at the same time.

The job network search is performed for the user executing the command.

■ -a

The job network search is performed for all users. Note that you cannot execute the command if you do not have the file access privilege.

-j \$jnwnname

In \$jnwnname, specify the name of the job network to search for. As the job network name, specify the root job network name.

-d \$date

In date, specify the job network search range or schedule display date.

To specify this option, use the following format.

-d [YYYY/]MM/DD

-p \$starttime

Specify the time when the specified date starts.

If you omit this option, the date starts at 00:00 AM local time.

The option is used to get the correct list of daily processing for batch job work.

To specify this option, use the following format.

-p HH:MM

-f csv | rd

The information is output in CSV or readable format.

■ -f csv

The information is output in CSV format.

■ -f rd

The information is output in readable format.

If you omit this option, the information is output in readable format.

-c | +c

The information of the last job or waiting part is output to the information output.

■ -c

The information is output.

■ +c

The output of the information is suppressed.

If you omit this option, the information is output.

-C | +C

The information of all preceding jobs or waiting components is output to the schedule information output.

■ -C

The information is output.

■ +C

The output of the information is suppressed.

If you omit this option, the output is suppressed.

-i \$jnwname.YYYYMMDDHHmmSS

Specify the tracker ID to identify the tracker.

A tracker ID consists of a job network name and a submission date and time. As the submission date and time, specify the year, month, day, hour, minute, and second in GMT.

As the tracker ID to specify in this option, it is recommended to use the one displayed when the command of the first syntax is executed.

(Example)

-i jnw1.20090501130500

Tracker ID of job network jnw1 submitted (or scheduled) on May 1, 2009, 22:05

3.2.2.1. Configuration File

By writing a frequently specified option in the configuration file, you can use it as a default value. Still, you can change this set value by specifying the relevant option on the command line.

Create the configuration file with the file name jnwschprt.f under the installation directory shown below or the data directory of the user.

■ Configuration file common to the local site and cluster site

UNIX	/usr/lib/nqs/gui/bin/jnwschprt.f
Windows	%InstallDirectory%\spool\conf\JNWSCHPRT.F

■ Configuration file under the data directory of the user

UNIX	/usr/spool/nqs/gui/<user-name>/jnwschprt.f
Windows	%InstallDirectory%\spool\users\<user-name>\JNWSCHPRT.F



In the case of a cluster environment, replace /usr/spool or %InstallDirectory% with <cluster-database-path>.

In the file, write the option just as it is specified on the command line (the command name itself is not necessary). Note that a special character (environment variable or quote character) used by the shell is passed to the jnwschprt command as is without being interpreted.

(Setting example) In this example, the command is intended to output the following information.

- In the job network list, the date changes at 8:00 AM.
- The job networks of all users are searched and output.
- All preceding jobs are output to the schedule information.

```
% cat /usr/lib/nqs/gui/bin/jnwschprt.f
-s -p 8:00 -a -C +c
```

If an option of the same type is specified both on the command line and in each jnwschprt.f file mentioned above and if different parameters are specified respectively, the option specified on the command line takes precedence. Options of different types are all interpreted.

(Setting example) In this example, the command is executed with different options set in each file.

```
% cat /usr/lib/nqs/gui/bin/jnwschprt.f
-a -p 9:00
% cat /usr/spool/nqs/gui/<user-name>/jnwschprt.f
-s -p 8:00
% /usr/lib/nqs/gui/bin/jnwschprt -C -p 7:00
```

In this case, the command line takes precedence for the overlapping -p option specification, they are interpreted to be the same as executing the following:

```
% /usr/lib/nqs/gui/bin/jnwschprt -a -s -C -p 7:00
```

If the -p option is not specified on the command line, since the common configuration file under the installation directory takes precedence, they are interpreted to be the same as executing the following:

```
% /usr/lib/nqs/gui/bin/jnwschprt -a -s -C -p 7:00
```

3.2.2.2. Job Network Information List

The following information is output.

- Search date (search start time - search end time)
- User name
- Job network name
- Expected submission date and time
- Job network ID name (JNW-name.YYYYMMDDHHmmSS)

You will be notified if there is any user whose file cannot be accessed when all users are searched.

3.2.2.3. Schedule List Information

There are two output formats: CSV format and readable format. For details about the CSV format, see [Section 3.2.6, "CSV Format of Schedule Information"](#) later in this section.

As the schedule list information, the information about the job network (summary) and the schedule information of the individual jobs (unit job, ERP job, BI job, PC job, WOBS job, and custom job) are output as described below.

- Job network summary
 - Job network name
 - User name
 - Expected start time
 - Tracker name (JNW.YYYYMMDDHHmmSS)
 - Number of job components (number of unit jobs, number of ERP jobs, number of BI jobs, number of PC jobs, number of WOBS jobs, and number of custom jobs)
 - Number of job components skipped by a calendar branch
- Schedule information
 - Component name
 - Parent job network
 - Whether there are jobs held or skipped in advance
 - Last job or waiting component
 - All preceding jobs or waiting parts

The output of part of the information may be suppressed by options.

3.2.2.4. Treatment of a Calendar Branch

When the schedule information list is output, a calendar branch is judged according to the specified date condition. When it is judged that a job is not to be executed, that job is not output in the schedule information.

3.2.3. Return Values

Return value	Description
0	Normal end

1	Abnormal end
---	--------------

3.2.4. Related Files

jnwschprt.f

3.2.5. Notes

■ You can also display past entries in the schedule information. Note that, since the command always references the currently defined job network flow information, the displayed schedule information may be different from that of the actually executed schedule.

■ When you execute the command for a cluster site, you need to set the site name in environment variable NQS_SITE in advance. (For Windows, you also need to set the database path of the site in NQS_SITEDB.)

For details, see the cautions given in [Chapter 1, 「Command List」](#) .

■ In the Windows version, the information of executed job networks is also output. In the UNIX version, only the information of scheduled job networks is output.

■ In version R13.1 or later, this command cannot be used while JobCenter is stopped.

3.2.6. CSV Format of Schedule Information

The detailed format to be used when jnwschprt outputs schedule information in CSV format is described below.

[Item types]

The schedule information consists of the following two types of items.

■ Job network summary

The job network summary is output only once at the beginning of the output.

■ Schedule information of job components (unit job, ERP job, or BI job)

The schedule information is output repeatedly for each of the components.

Table 3.4. Item type name

Item type name	Description
SUMMARY	Job network summary
SCHEDULE	Schedule information of job components

■ When #TYPE is set to SUMMARY, the summary is output in the format below.

```
#TYPE=SUMMARY
jnw,user,start,jnwid,njob,ncskip
Value,...
```

■ The items of the summary output are described below.

Item name	Description
jnw	Job network name
user	User name
start	Expected start time of the job network

	The time is expressed in local time as follows. "YYYY/MM/DD HH:mm:SS"
jnwid	ID managed by JobCenter job-network-name.YYYYMMDDHHmmSS (YYYYMMDDHHmmSS is the submission date and time expressed in GMT.)
njob	Number of jobs to be executed by this job network Jobs skipped by a calendar branch are not included.
ncskip	Number of jobs skipped by a calendar branch

■ #TYPE=SCHEDULE (schedule information of job components)

The schedule information is output in the format below.

The schedule line is output after the summary line, with a blank line between them. (The item of initskip is output in JobCenter R12.5.4 and later.)

```
#TYPE=SCHEDULE
index,name,parentjnw,inithold,initskip
SCHEDULE, Value
PREV, Job name...
ALLPREV, Job name...
```

The first and second lines are the title lines and output only once.

The third, fourth, and fifth lines are output for individual job components. The output may be suppressed by the command options.

The first items of the third, fourth, and fifth lines (SCHEDULE, PREV, and ALLPREV) are fixed. They show what is written on each of these lines.

The items are described below.

■ Third line: SCHEDULE

Item name	Description
index	A fixed character string "SCHEDULE" is set.
name	Job component name
parentjnw	Name of the job network to which the job component belongs. In the case of a sub job network, it is written as follows. "rootJNW[:subJNW1...]:subJNWx"
inithold	Whether the Hold in advance status is set is indicated. When the status is set, 1 is shown. Otherwise, 0 is shown.
initskip	Whether the Skip in advance status is set is indicated. When the status is set, 1 is shown. Otherwise, 0 is shown.
	This item is output in JobCenter R12.5.4 and later.

■ Fourth line: PREV

Of the preceding jobs and waiting parts of the job component, only the last ones are displayed. One preceding part is displayed in each column.

- The preceding component information for up to 50 columns is output on one line.

- If there are more than 50 preceding parts, two PREV lines are output.

Parts are displayed as follows.

- Unit job or ERP job:

"J rootJNW[:subJNW...]/job-name"

- Time waiting part (waiting time HH:MM):

"T YYYY/MM/DD HH:MM"

The time in GMT is displayed in the format of "YYYY (year), MM (month), DD (day), HH (hour), and MM (minute)". If the value of the month, day, hour, or minute is a single-digit number, add 0 as a prefix.

- File waiting component

"F file-name"

- Fifth line: ALLPREV

All the preceding jobs and waiting components of the job component are displayed.

This line is identical to the PREV line, except that all preceding jobs are displayed.

3.3. jnwop (control) (Performing an Operation on a Component of a Job Network)

■ UNIX version

```
/usr/lib/nqs/gui/bin/jnwop control [-u $user] [-t $YYYYMMDDHHmmSS] [-tid $machine:$user:$JNWname.YYYYYMMDDHHmmSS(GMT).nstrk] $command $JNWname [$JOBname]
```

■ Windows version

```
%InstallDirectory%\bin\jnwop control [-u $user] [-t $YYYYMMDDHHmmSS] [-tid $machine:$user:$JNWname.YYYYYMMDDHHmmSS(GMT).nstrk] $command $JNWname [$JOBname]
```

3.3.1. Function Description

- The jnwop command lets you perform an operation either on a unit job, a job network or a sub job network that exists in a running job network (including one waiting to be archived) or a scheduled job network. The available operations are as follows.

Hold, Hold Release, Skip, Skip Release, Rerun, Forced Stop, Forced Start, Rerun from Error Point, Skip Error, Change to Done, Delete Tracker, Archive Track

- The character strings shown in the following table can be used in \$command.

Table 3.5. Character strings that can be used in \$command

Character string	Description
hold	The specified component is "held". If the component is already held, the command is ignored.
release	The specified "held" component is released. If the holding of the specified component has already been released, the command is ignored.
skip	The specified component is "skipped". If the component has already been skipped, the command is ignored.
noskip	The "skipping" of the specified component is released. If the skipping of the specified component has already been released, the command is ignored.
rerun	The specified component is "rerun".
estop	The specified component is "forced to stop".
forcestart	The specified component is "forced to start".
delete	The specified tracker information is "deleted".
before	If the specified tracker is in the STOP(ERROR) status, the processing is rerun from the unit job stopped due to an error. The unit jobs after that unit job in the flow are also rerun.
after	If the specified tracker is in the STOP(ERROR) status, the unit job stopped due to an error is skipped and the processing is rerun from the next unit job. The unit jobs after that unit job in the flow are also rerun.
statusdone	If the specified component is in the DONE(ERROR) or DONE[WARNING] status, its status is changed to DONE.
archive	If the specified tracker is in the DONE, DONE(ERROR) or WARNING status, it is archived. One minute or more needs to pass from the submission time of the tracker.

- In \$JNWname, specify the name of the job network to be operated. If the target component is a sub job network, delimit the job network name by ":" to represent the hierarchy.

(Example)

"rootJNW:subJNW"

- In \$JOBname, specify the name of the unit job network to be operated.

If you omit the unit job name, the operation is targeted at the job network or sub job network specified in \$JNWname.

3.3.2. Options

-u \$user

Specify the name of the user who owns the job network to be operated. If you do not specify this option, the operation is targeted at the job network owned by the user who executes this command.

For the environment variables used when a job network is submitted, see [Section 3.3.4, "Notes"](#).

This option is available only to the JobCenter administrator user.

-t \$YYYYMMDDHHmmSS

Use this option to specify the target tracker.

The job network submission time is expressed in local time as follows (this is different from GMT used in some tracker names).

The format is (YYYYMMDDHHmmSS).

If you omit this option, the last submitted tracker becomes the target.

-tid \$machine:\$user:\$JNWname.YYYYMMDDHHmmSS(GMT).nstrk

Use this option to specify the target tracker in GMT.

Specify the option as follows.

machine-name:user-name:job-network-name:submission-time(YYYYMMDDHHmmSS(GMT)).nstrk

3.3.3. Return Values

Return value	Description
0	Normal end
Other than 0	Abnormal end. An error message is output to the standard error output.

3.3.4. Notes

- If you execute the jnwop command with the -u option as nsumsmgr (or root) in the UNIX version, the message "Failed: mkdir () failure..." may appear. Use the return value to judge whether the operation is complete. If you log in as nsumsmgr (or root) from CL/Win, this message no longer appears.

- To use the jnwop command in the Windows version requires the Administrator privileges.

- If you execute hold/release, skip/noskip, etc., it is ignored when the operation is disabled.

- If you attempt to operate a nonexistent tracker with `-t` or `-tid` specified, the operation itself fails. Note, however, that the specified tracker is created at the date and time specified in the "PREPARE" status. (If the tracker is created, delete it immediately by using `delete`.)
- When you execute the command for a cluster site, you need to set the site name in environment variable `NQS_SITE` in advance. (For Windows, you also need to set the database path of the site in `NQS_SITEDB`.)

For details, see the cautions given in [Chapter 1, 「Command List」](#) .

- If you change the "WAIT" status of a tracker to "PREPARE" by executing the `jnwop` command (by specifying a value other than `statusdone` and `archive` in `$command`), the tracker uses the environment variables mentioned below.
 - When the target is the job network owned by the user who executes the command

The tracker uses the environment variables applied when the command is executed. Particularly, be careful about environment variables `TZ`, `LOGNAME`, `HOME`, `LANG`, and `NQS_SITE` to set the correct values.
 - When the target is the job network of another user

The tracker uses the environment variables applied when the job network is submitted according to schedule.

3.4. jnwsubmitcmd (Submitting a Job Network)

■ UNIX version

```
/usr/lib/nqs/gui/bin/jnwsubmitcmd [-p $parameter] [-e $key=$value | -i $inputfile] [-c $comment] [[-b] -t $YYYYMMDDHHmmSS(localtime) | $HH:MM(localtime)] [-d] [-h] [-s] [-r $remotehostname] [-u $user] [-w|-k|-f] $JNWname
```

■ Windows version

```
%InstallDirectory%\bin\jnwsubmitcmd [-p $parameter] [-e $key=$value | -i $inputfile] [-c $comment] [[-b] -t $YYYYMMDDHHmmSS(localtime) | $HH:MM(localtime)] [-d] [-h] [-s] [-r $remotehostname] [-u $user] [-w|-k|-f] $JNWname
```

3.4.1. Function Description

■ The jnwsubmitcmd command submits the job network specified in \$JNWname and generates a tracker. In \$JNWname, specify the job network defined in the environment of the user who uses this command.

■ If the submission of the job network is completed successfully, the generated tracker name is output in the following format.

```
machine-name:user-name:job-network-name.YYYYYMMDDHHmmSS.nstrk
```

YYYYMMDDHHmmSS is the submission date and time expressed in GMT.

■ If an error occurs, a message is output to the standard error output.

3.4.2. Options

-p \$parameter

The character string specified in \$parameter is set in environment variable NSJNW_PARAM when the unit job is executed.

This option is equivalent to the parameter that can be entered in the dialog during the submission operation of CL/Win as described in "Environment variable" - "Setting the job parameter (NSJNW_PARAM)".

-e \$key=\$value

Set \$key (environment variable name)=\$value (environment variable value) as the environment variable when executing the unit job.

To set more than one environment variable, specify this option several times.

```
-e key1=value1 -e key2=value2 ...
```

To specify a character string including a space, enclose \$key=\$value entirely with double quotation marks.

```
-e "key1=a b c"
```

This option is equivalent to the parameter that can be entered in the dialog during the submission operation of CL/Win as described in "Environment variable" - "Setting more than one environment variable".



Specify the environment variable name of up to 51 bytes.

Spaces, tabs, special characters, =, #, ', and " cannot be used as the environment variable name.

Tabs, special characters, #, ', and " cannot be used for the environment variable value.

This option cannot be used at the same time as the -i option.

-i \$inputfile

Set the data in the file specified by \$inputfile as the environment variable for execution of the unit job. Specify \$inputfile in the format of key (environment variable name)=value (environment variable value), and add a linefeed to specify more than one environment variable.

```
key1=value1
key2=value2
...
```

To contain a multi-byte character, set the character code in the file as described below.

■UNIX version: Use the code appropriate for the language in which the JobCenter MG/SV was set up.

■Windows version: Enter the character in SJIS.

This option is equivalent to the parameter that can be entered in the dialog during the submission operation of CL/Win as described in "Environment variable" to "Setting more than one environment variable".



Specify the environment variable name of up to 51 bytes.

Spaces, tabs, special characters, =, #, ', and " cannot be used as the environment variable name.

Tabs, special characters, #, ', and " cannot be used for the environment variable value.

One line in the file can contain up to 8191 bytes.

This option cannot be used at the same time as the -e option.

-c \$comment

The comment to be displayed in the tracker.

This is the same as the comment that can be entered in the dialog box of CL/Win for submitting a job network immediately.



The maximum length is 128 bytes in the UNIX version and 40 bytes in the Windows version.

-t \$YYYYMMDDHHmmSS(localtime) | \$HH:MM(localtime)

■-t \$YYYYMMDDHHmmSS

Specify the submission time of the job network (YYYYMMDDHHmmSS Local time).

This is the same as the time that can be entered in the dialog box of CL/Win for submitting a job network immediately.

■ -t \$HH:MM

Specify the submission time of the job network (HH:MM Local time (hour:minute)).

In YYYYMMDD, the year, month, and day of command execution in local time is set.

In SS (second), 00 is set.

In HH (hour) of the submission time, you can specify an integer between 0 and 23.

In MM (minute) of the submission time, you can specify an integer between 0 and 59.

If the value of the month, day, hour, minute, or second is a single-digit number, 0 is automatically added as a prefix.

-b

The job network can be submitted at the time specified in the -t option even if that submission time is earlier than the current time.

If the submission time is earlier than the current time when this option is not specified, the job network is submitted with the submission time corrected to the current time.

-d

The job network is not submitted if there is any other tracker submitted (or scheduled to be submitted) on the same date for a job network having the same name as that specified in \$JNWname.

The submission date is judged to be the same when the "YYYYMMDD" part of the tracker name is identical.

This option is used to prevent the same daily job from being submitted twice on the same date.

-h

If more job networks than the value set as the limit of the number of job networks that can be executed simultaneously have been submitted and are waiting, the job network specified in \$JNWname receives top priority in execution.

If there are multiple job networks for which this option is specified, they are executed in the order in which they are submitted.

-s

If there is any other tracker submitted (or scheduled to be submitted) at the same time (YYYYMMDDHHmmSS) for a job network having the same name as that specified in \$JNWname, the job network is submitted one second later.

■ Specification of the submission time

You can specify the submission time (YYYYMMDDHHmmSS (GMT)) in environment variable JNW_SUBMITTIME.

If this option is used together with the -t option, the time set in the -t option takes precedence. This value is reflected in the tracker name.

-r \$remotehostname

Specify this option to submit a job network that exists in a remote server. To specify this option, you need to specify the `-u` option described later as well. If you do not specify the option, the job network on the local server is submitted.



If you specify the option, execute the command with environment variable `NQS_SITE` not set. If environment variable `NQS_SITE` is set, you may not be able to execute the command due to an error such as failure to access a nonexistent path.

In the case of the Windows version, this option is available only to the administrator user.

`-u $user`

Specify the name of the user who owns the job network to be submitted. If you do not specify this option, the command submits the job network owned by the user who executes the `jnwsubmitcmd` command.

The environment variables used for job network submission are as follows.

■ When submitting a job network created by the user who executes the command

The environment variables applied when the command is executed are used.

■ When submitting a job network of another user or submitting a job network on a remote server by using the `-r` option

The same environment variables as those applied when a job network of another user or on a remote server is submitted according to schedule.



For example, even if a job network is submitted with this option specified immediately after an additional system environment variable is set in the OS of the server in which the job network exists, that additional environment variable is not reflected. (This is because the login environment of the user is not used when a job network is submitted according to schedule.)

To reflect the environment variables in the user environment, log in to the server in which the job network exists from CL/Win as the user who owns the job network and create and save a dummy schedule rule at least once. This reflects changes to the environment variables in JobCenter. You may delete dummy rules and dummy schedules after saving them.

In the case of the Windows version, this option is available only to the administrator user.

`-w`

The job network is submitted in the HOLD status, ignoring the preliminary settings of JNW parameters.

This option is available starting with R12.8.1 for the Windows version. For the UNIX version, it is available starting with R12.7.

`-k`

The job network is submitted in the SKIP status, ignoring the preliminary settings of JNW parameters.

This option is available starting with R12.8.1 for the Windows version. For the UNIX version, it is available starting with R12.7.

-f

The job network is submitted so that the JNW will be executed immediately, ignoring the preliminary settings of JNW parameters.

This option is available starting with R12.8.1 for the Windows version. For the UNIX version, it is available starting with R12.7.

3.4.3. Return Values

Return Value	Description
0	The submission of the job network is successfully completed.
1	The submission of the job network failed. It may succeed if you retry.
3	The way the command was used is invalid; e.g., a specified option is invalid.
4	A read error has occurred in the environment variable setting file.
Other	There is some problem in JobCenter.

3.4.4. Notes

- When you execute the command for a cluster site, you need to set the site name in environment variable NQS_SITE in advance. (For Windows, you also need to set the database path of the site in NQS_SITEDB.)

For details, see the cautions given in [Chapter 1, 「Command List」](#) .

- The environment variables used when this command is executed are inherited as those for job network execution (except when the -r or -u option is specified). Particularly, be careful about environment variables TZ, LOGNAME, HOME, LANG, and NQS_SITE to set the correct values.
- In the case of the UNIX version, make sure that environment variable LANG used when the command is executed matches the setup language of JobCenter. In the case of the Windows version, there is no particular need to pay attention because the setup language specified during installation is used instead of environment variable LANG.

When you specify a job network on a remote server by using the -r option, EUC and SJIS may be used in a mix. Note, however, that an error occurs if one is the Unicode environment and the other is not.

- When you start another job network from the unit job by using this command, you need to clear environment variables STANDARD_ERROR and STANDARD_OUTPUT before executing the command. If you do not clear these environment variables, the output result and error output result of the unit job of the job network started by this command are not recorded in the tracker.
- Even if the same user executes more than one command simultaneously with the same job network name specified or executes the command multiple times with the same time specified in the -t option, multiple trackers cannot be started with the same job network name at the same time. The first executed command takes effect and creates a tracker, and the other commands end without doing anything.

In this case, specify the -s option.

3.5. nqsbg (Starting a Process Not Terminated at the End of a Job from within the Job)

■ UNIX version

```
/usr/lib/nqs/nqsbg [-w] [ -n $subcommand ] $maincommand
```

■ Windows version

(None)

3.5.1. Function Description

- The nqsbg command provides a function that starts the user command specified in \$maincommand as a background process that does not receive the above signal (TERM/KILL) from a unit job script at the end of a unit job.

JobCenter normally waits until a command process started from a job request ends. When it ends the processing of the job request after the command process ends, JobCenter sends a signal (TERM/KILL) to a child process belonging to the same process group started from the command process so that the process started from the job request does not remain after the end of the job.

For example, when a user command returns control after starting a mail transmission process in the background or a unit job is started with a resident process such as a daemon process written in the unit job script, the background process that has been started is ended as the unit job ends successfully on the tracker. This command allows you to avoid this situation.

- The command starts the command specified in \$maincommand as a background process.
- Since the control of the started process is separated from the job, the end status of this process cannot be monitored by JobCenter. Also, ending JobCenter does not end the started process.
- This command is implemented only for the UNIX version. (For the Windows version, use the start or other appropriate command of the OS.)

3.5.2. Options

(The command is normally used with no options.)

nqsbg ends immediately after starting a sub process.

The started sub process in turn executes the user command specified in \$maincommand. Therefore, the process ID of nqsbg is different from that of the process started in the background.

-w


A sub process is not started.

After separating the terminal and the process group, nqsbg directly starts the command without starting the sub process to be actually started in the background. In this case, while the process ID of nqsbg becomes identical to that of the started command, the process itself does not become a background process.

-n \$subcommand

The character string specified in \$subcommand is used by specifying it in argv[0] of the command to be started (\$maincommand). Use this option when you need to change the process name displayed by the ps or other command.

3.5.3. Return Values

Return value	Description
0	<p>The return value is other than 1.</p> <div style="border: 1px solid red; padding: 5px;">  <ul style="list-style-type: none"> ■ Note that, even if the command specified in <code>\$maincommand</code> fails to be started because the corresponding command file does not exist, <code>nqsb</code> outputs 0 as the return value. ■ Even if the command specified in <code>\$maincommand</code> ends abnormally, <code>nqsb</code> outputs 0 as the return value, because it separates the process group immediately after starting <code>\$maincommand</code>. </div>
1	<p>The process group of the process cannot be separated.</p> <p>The sub process or command process cannot be generated.</p>



If you start the command directly by specifying `-w`, the exit code of the started program is returned.

3.5.4. Notes

- Set the process that is started by `nqsb` so that the standard output and standard error output are redirected to `/dev/null`.

If the process is not set to redirect the standard output and standard error output to `/dev/null`, the content written to the standard output and standard error output of the process started by this command is recorded in the file that constitutes the tracker data of the job. This may cause an increase in the tracker data size or an error during data reference or some other operation.

3.6. jnwevtmgr (Checking and Deleting Events)

■ UNIX version

```
/usr/lib/nqs/gui/bin/jnwevtmgr {-l | -d $hostname:$eventid }
```

■ Windows version

```
%InstallDirectory%\bin\jnwevtmgr {-l | -d $hostname:$eventid }
```

3.6.1. Function Description

■ The `jnwevtmgr` command displays the current status of event transmission/reception components. It also deletes events that are held. Unlike the operations in CL/Win that require the privilege to "operate other users' trackers", this command can display and delete even those events being tracked by other users.

■ Event check

The event list is output in the format shown below.

```
***Event list name***
Index Tracker name.YYYYMMDDHHmmSS Part name Host name Event ID Timeout
```

Details of the individual items are given in [Table 3.6, "Event list names and descriptions"](#) and [Table 3.7, "Event list output formats"](#).

Table 3.6. Event list names and descriptions

Event list name	Description
EventWaitList	List of reception parts waiting to be received
EventSendList	List of transmission parts waiting for the reception check
EventHoldList	List of held events

Table 3.7. Event list output formats

Name	Event list output formats
Index	The number for indexing is output.
Tracker name	The name of the tracker to which the part belongs and the time (GMT) are output in the format of tracker-name.time. In the case of the list of held events, the name of the tracker to which the part that transmitted the event belongs is output.
Part name	Name of the part. In the case of the list of held events, the name of the part that transmitted the event is output.
Host name	Source or destination host name specified in the part.
Event ID	Event ID specified in the part
Timeout	Timeout time. If the timeout time is infinite, "-" is displayed.



For the name of a component that transmitted an event using `jnwevtcmd`, `PIDjnwevtcmd-process-number.jnwevtcmd.execution-time (local time)` is displayed as the tracker name and `jnwevtcmd` is displayed as the component name.

■ Deletion of an event

After an event is deleted, one of the messages shown below appears.

Table 3.8. Messages displayed after an event is deleted

Message	Description
Invalid eventhostname or eventide.	The specified host name or event ID is invalid.
Invalid eventide.	The event ID is invalid.
Not found target event.	The specified target cannot be found.
Success.	Deletion was successful.

3.6.2. Options

-l
A list of events is displayed.

-d \$hostname:\$eventid
The held event having the specified source host name and event ID is deleted.
\$hostname: Event transmission source
\$eventid: Event ID

3.6.3. Return Values

Return value	Description
0	Normal end
1	Abnormal end

3.6.4. Notes

- An event can be deleted on the machine that is holding the event (the destination machine for event transmission).
- When you execute the command for a cluster site, you need to set the site name in environment variable NQS_SITE in advance. (For Windows, you also need to set the database path of the site in NQS_SITEDB.)

For details, see the cautions given in [Chapter 1, 「Command List」](#) .

3.7. jnwevtcmd (Transmitting an Event)

■ UNIX version

```
/usr/lib/nqs/gui/bin/jnwevtcmd -h $hostname -i $eventid [ -m $msg ] [ -l {off | on} -t $YYYYMMDDHHmmSS  
| -c $YYYYMMDDHHmmSS | -a [+]$HH:MM } } ] [-s]
```

■ Windows version

```
%InstallDirectory%\bin\jnwevtcmd -h $hostname -i $eventid [ -m $msg ] [ -l {off | on} -t  
$YYYYMMDDHHmmSS | -c $YYYYMMDDHHmmSS | -a [+]$HH:MM } } ] [-s]
```

3.7.1. Function Description

- The jnwevtcmd command allows any user to transmit an event to an event reception part of a job network at any time. Except when the -s option is specified, an event can be transmitted to all event waiting components that are waiting for an event having the specified event ID.
- The following table lists and describes the messages displayed when the jnwevtcmd command is executed.

Table 3.9. Messages displayed when the jnwevtcmd command is executed

Display	Description
FOUND	The event was received at the destination.
NOTFOUND	The event was not received at the destination.
FAILED	An error occurred at the destination.
***** is too long	The specified parameter ***** is too long.
Need parameter. (-* *****)	The parameter ***** is required.
Invalid Option	An invalid option is specified.
Lack of Option	One or more options are missing.
Invalid format	The format of a specified parameter is invalid.

3.7.2. Options

-h \$hostname

Specify the name of a destination host.

-i \$eventid

Specify an event ID.

-m \$msg

Specify a message.

-l {off | on}

Specify whether an event is held.

-t \$YYYYMMDDHHmmSS | -c \$YYYYMMDDHHmmSS | -a [+]\$HH:MM

Specify the holding period when the option is set to on to hold an event.

■ -t \$YYYYMMDDHHmmSS

The time is expressed in GMT as follows.

YYYY: Year, MM: Month, DD: Day, HH: Hour, mm: Minute, SS: Second

If the value of the month, day, hour, minute, or second is a single-digit number, add 0 as a prefix.

■ -c \$YYYYMMDDHHmmSS

The time is expressed in local time as follows.

YYYY: Year, MM: Month, DD: Day, HH: Hour, mm: Minute, SS: Second

If the value of the month, day, hour, minute, or second is a single-digit number, add 0 as a prefix.

■ -a [+]\$HH:MM

Specify the absolute or relative time. The maximum value is 99:59.

+: Relative time specification (calculated relative to the time the command was issued)

HH: Hour, MM: Minute

-s

Of the parts having the same event ID that are waiting for an event, the command transmits the event only to the part whose start time is the earliest.

3.7.3. Return Values

Return value	Description
0	Normal end
1	Abnormal end
2	Socket communication was attempted but failed to be connected.
255	Communication ended abnormally.

3.7.4. Notes

- The jnwevtcmd command does not support reception waiting.
- If the same user or different users transmit an event multiple times from the same machine to the same destination host by using the same \$eventid, the last transmitted event overwrites the previous ones on the destination host.
- When you transmit an event from a cluster site by using this command, you need to set the site name in environment variable NQS_SITE in advance. (For Windows, you also need to set the database path of the site in NQS_SITEDB.)

For details, see the cautions given in [Chapter 1, 「Command List」](#) .

3.8. jnwdelete (Deleting a Job Network or Schedule)

■ UNIX version

```
/usr/lib/nqs/gui/bin/jnwdelete [-u $user] -j $jnw
/usr/lib/nqs/gui/bin/jnwdelete [-u $user] [-f] -g $jnwgroup
/usr/lib/nqs/gui/bin/jnwdelete [-u $user] -s $schedule
```

■ Windows version

```
%InstallDirectory%\bin\jnwdelete [-u $user] -j $jnw
%InstallDirectory%\bin\jnwdelete [-u $user] [-f] -g $jnwgroup
%InstallDirectory%\bin\jnwdelete [-u $user] -s $schedule
```

3.8.1. Function Description

- The jnwdelete command deletes a job network or schedule.
- The command of the first syntax deletes a job network, the command of the second syntax deletes a job network group, and the command of the third syntax deletes a schedule. Note that -j, -g, and -s cannot be specified at the same time. You need to specify one of these options.
- The following table lists and describes the messages displayed when the jnwdelete command is executed.

Table 3.10. Messages displayed when the jnwdelete command is executed

Display	Description
DELETE: user=\$user, JNW=\$jnw	The specified job network was successfully deleted.
DELETE: user=\$user, Group=\$jnwgroup	The specified job network group was successfully deleted.
DELETE: user=\$user, Schedule=\$schedule	The specified schedule was successfully deleted. (This message is also output if a nonexistent schedule is specified.)
Specified object not exists.	The specified job network does not exist.
LANG is not equal to the language used by JobCenter	Environment variable LANG does not match the setup language used by JobCenter.
cannot get the database path	The site name specified in NQS_SITE is invalid.
Permission denied.	You do not have the permission to delete the target user's definition.
invalid option	An invalid option is specified.



The error codes displayed at the beginning of the message are not shown.

3.8.2. Options

-u \$user

Specify the name of the user whose definition you want to delete. If you do not specify this option, the command deletes the job network or schedule of the user who executes the command.

-j \$jnw

The specified job network is deleted.

-g \$jnwgroup

The specified job network group is deleted. Specify a group name from the top level group as follows.

(Example) Normal deletion of Grp1 in Grp1 of user1

```
jnwdelete -u user1 -g .Grp1.Grp2
```

When you specify \$jnwgroup, "." at the beginning indicates the Root group.

-f

Specify forced deletion or normal deletion when deleting a job network group. If you do not specify -f, normal deletion is assumed. Specifying -f enables forced deletion.

-s \$schedule

The specified schedule is deleted.

3.8.3. Return Values

Return value	Description
0	Normal end
1	Returned when the command ended abnormally. The error is output to the standard error output.
2	Returned when a command option is invalid or there is a problem in the execution environment. The error is output to the standard error output.

3.8.4. Notes

- To perform deletion using this command, the user who executes the command needs to be given the permission to "create, modify, and delete job networks" in the permission settings of JobCenter. See details about the permission settings in <Environment Guide>Chapter 9, 「User Authority (Permission)」 .
- Note that, while job networks being edited with CL/Win cannot be deleted, schedules can be deleted.
- Environment variable LOGNAME needs to match the user who executes this command. To change the user with the su command, use su -.
- Environment variable LANG needs to be identical to the setup language used by JobCenter.
- When you execute the command for a cluster site, you need to set the site name in environment variable NQS_SITE in advance.

For details, see the cautions given in [Chapter 1, 「Command List」](#) .

3.9. schctrl (Enabling or Disabling a Schedule)

■ UNIX version

```
/usr/lib/nqs/gui/bin/schctrl [-u $user] $subcommand { $schedule... | $group... | all }
```

■ Windows version

```
%InstallDirectory%\bin\schctrl [-u $user] $subcommand { $schedule... | $group... | all }
```

3.9.1. Function Description

- The schctrl command enables or disables a schedule or groups schedules.
- You can start the operation of a registered schedule or stop a schedule in operation. Also, by grouping schedules, you can control whether all those schedules are enabled or disabled.
- To operate grouped schedules or grouped schedule groups, specify individual schedules and groups by using colons as delimiters in the formats shown below.

Target to specify	Format
Schedule specification	\$group:\$schedule Example) GROUP01:GROUP02:GROUP03:GROUP04:Schedule01
Group specification	\$group:\$group Example) GORUP01:GROUP In some sub commands, you can use the specifier ":" that is used to specify the Root group.

- The maximum number of characters is 40 bytes for a schedule group name and 128 bytes for a schedule name. Up to five levels of groups can be created, and these groups have parent-child relationships.
- An operation performed on a parent group affects its child group or groups. Note that you cannot create two or more child groups having identical names in the same group.

3.9.2. Options

-u \$user

Specify the user whose schedule is to be operated.

If you omit this option, the schedule of the user who executes the command is operated.

This option can be used by the following users.

- In the UNIX version, the option can be used only by the root user.
- In the Windows version, the option can be used by a user having the Administrator privileges.

3.9.3. Subcommands

Each of these commands must be entered on a single line.

1. Subcommand list

Character string	Description
------------------	-------------

on sched	The specified schedule is enabled. If the schedule is already enabled, the command is ignored.
on group	All the schedules of the specified group are enabled. If the schedules are already enabled, the command is ignored.
on all	All schedules are enabled. If the schedules are already enabled, the command is ignored.
off sched	The specified schedule is disabled. If the schedule is already disabled, the command is ignored.
off group	All the schedules of the specified group are disabled. If the schedules are already disabled, the command is ignored.
off all	All schedules are disabled. If the schedules are already disabled, the command is ignored.
create	A group is created.
delete	A group is deleted.
rename	A group is renamed.
regist	The specified schedule is registered in a group.
release	The specified schedule is released from a group.
move sched	The specified schedule is moved to a group.
move group	The specified group is moved to a group.
show sched	Whether the specified schedule is enabled or disabled is shown.
show group	All the schedules and all the groups of the specified group are shown in a tree structure. Whether each schedule is enabled or disabled is shown.
show all	All schedules and all groups are displayed in a tree structure. Whether each schedule is enabled or disabled is shown.

2. Command syntax

schctrl on sched \$schedule...

The schedule specified in \$schedule is enabled.

You can specify \$schedule multiple times. (Maximum number of times: 100)

schctrl on group \$group...

All the schedules of the group specified in \$group are enabled.

You can specify \$group multiple times. (Maximum number of times: 100)

In \$group, you can use the Root group specifier ":".

schctrl on all

All the schedules of the user are enabled.

schctrl off sched \$schedule...

The schedule specified in \$schedule is disabled.

You can specify `$schedule` multiple times. (Maximum number of times: 100)

`schctrl off group $group...`

All the schedules of the group specified in `$group` are disabled.

You can specify `$group` multiple times. (Maximum number of times: 100)

In `$group`, you can use the Root group specifier ":".

`schctrl off all`

All the schedules of the user are disabled.

`schctrl create $group...`

A new group is created.

You cannot create two or more groups having identical names in the same group. You cannot create groups of more than five levels, which is the upper limit.

You can specify `$group` multiple times. (Maximum number of times: 100)

`schctrl delete $group...`

A group is deleted. If there is a schedule or child group belonging to the deleted group, the schedule no longer belongs to any group or the child group is deleted. The schedule that no longer belongs to any group remains enabled or disabled.

You can specify `$group` multiple times. (Maximum number of times: 100)

`schctrl rename $group $group`

The change target in the first `$group` parameter is renamed to the new name in the second `$group` parameter. In the last command parameter, specify the new group name.

You cannot rename the group if there is another group having the identical name in the same group.

`schctrl regist $schedule... $group`

The schedule specified in `$schedule` that no longer belongs to any group is registered to the group specified in `$group`. In the last command parameter, specify the name of the group to which to register the schedule.

You can specify `$schedule` multiple times. (Maximum number of times: 100)

`schctrl release $schedule...`

The schedule specified in `$schedule` is released from the group to which it belongs.

The released schedule no longer belongs to any group while remaining enabled or disabled.

You can specify `$schedule` multiple times. (Maximum number of times: 100)

`schctrl move sched $schedule... $group`

The schedule specified in \$schedule is moved to the group specified in \$group. In the last command parameter, specify the group to which to register the schedule.

The moved schedule belongs to the specified group while remaining enabled or disabled.

You can specify \$schedule multiple times. (Maximum number of times: 100)

In \$group, you can use the Root group specifier ":".

`schctrl move group $group... $group`

The group specified in \$group is moved to the destination group specified in \$group. In the last command parameter, specify the name of the group to which to move the group or groups. If the number of group levels exceeds the upper limit (five levels) after the move, you cannot move the groups.

The moved group belongs to the destination group while remaining enabled or disabled.

You can specify the source group parameter \$group multiple times. (Maximum number of times: 100)

In the source group parameter \$group, you can use the Root group specifier ":".

`schctrl show sched $schedule...`

Whether the schedule specified in \$schedule is enabled or disabled is shown in the following format.

[SCHEDULE] \$schedule : <on|off>

You can specify \$schedule multiple times. (Maximum number of times: 100)

`schctrl show group $group...`

All the schedules and all the groups of the group specified in \$group are shown in a tree structure.

Whether each schedule is enabled or disabled is shown.

You can specify \$group multiple times. (Maximum number of times: 100)

In \$group, you can use the Root group specifier ":".

`schctrl show all`

All schedules and all groups are displayed in a tree structure.

Whether each schedule is enabled or disabled is shown.

3.9.4. Return Values

Return value	Description
0	Normal end
1	Abnormal end

3.9.5. Notes

- In this command, special characters (!"#%&'()*+,-./:;<=>?@[\\]^_{|}~) are considered invalid.
- Even if a schedule is disabled, a tracker already submitted according to that schedule continues to be processed without being forced to end. Also, even if you disable a schedule for a tracker whose

status is changed from WAIT to PREPARE due to the hold or other operation, the tracker in the PREPARE status is not deleted.

- Even if you change the schedule parameter from CL/Win, the enable or disable status set using this command is maintained.
- The operations for enabling/disabling and grouping schedules are the same as those for [Schedule List] of CL/Win.
- Even if you enable or disable a schedule by using this command, Auto Refresh of Tracker List of CL/Win does not reflect this status change. To check the status change of the schedule from CL/Win, execute [Refresh (R) F5] in Tracker List.
- In version R13.1 or later, this command cannot be used while JobCenter is stopped.

3.10. nqsstart (Starting Daemon Processes)

■ UNIX version

```
/usr/lib/nqs/nqsstart [-f] [$sitename]
```

■ Windows version

(None)

3.10.1. Function Description

- The nqsstart command starts JobCenter. Note that the JobCenter daemon can be started only by the root account (super user). This command is implemented only for the UNIX version.
- This command starts the JobCenter daemon (/usr/lib/nqs/nqsdaemon) and the daemon processes related to event transmission and others.
- This command loads the daemon startup configuration file (/usr/lib/nqs/rc/daemon.conf) and reads the options for starting the daemon.

3.10.2. Options

\$sitename

Specify the name of the local site or cluster site to be started in site mode.

-f

The JobCenter daemon (/usr/lib/nqs/nqsdaemon) and the daemon processes related to event transmission and others are started.



If JobCenter did not end normally or was terminated improperly the last time, an error may occur for the command of nqsstart [\$sitename]. In that case, check that the daemon is not running in the host or site you are going to start, and then execute the command with the -f option to force the daemon to start.



How to check that the daemon is not running in the host or site

1. Use the grep search command by specifying names such as jnwxxx, nqsxxx, and NQSxxx with the ps -ef command, and check the PIDs and PPIDs of the daemons of JobCenter.
2. Check the nqsdaemon PIDs in the cluster site by executing the cjcls command. (SHUT indicates that JobCenter is not running in the site shown on the same line of the output results of the cjcls command.)
3. To check the daemons in the host, subtract the daemons for which the nqsdaemon PIDs and PPIDs in the results of step 2 match the PID of the nqsdaemon in the host from the results of step 1, and check whether the remaining daemons exist. To check the daemons in the site, check the results of step 1 to find the daemons for which the nqsdaemon PIDs and PPIDs in the results of step 2 match the PID of the nqsdaemon in the site.

3.10.3. Return Values

Return value	Description
--------------	-------------

0	Normal end
1	Abnormal end

3.10.4. Notes

- While JobCenter is usually started at system startup, the system administrator can start it as well by using this command. Note that, in that case, the information output by the individual daemons is displayed on the console or terminal unless you have the standard output of the command redirected to a file or other appropriate location.
- If you start the JobCenter daemon directly without using this command, some of the necessary resident processes that have dependencies fail to be started, making some of the functions unavailable.
- When starting a local site, you do not need to specify `$site-name`. When starting a cluster site, use `cjcpw` under normal operation.
- Pay attention to the `umask` value of the root account used when starting `nqsstart`. For example, if you start JobCenter by using `nqsstart` with `umask` set to `002`, that `umask` value is inherited to all the resident and other processes in JobCenter. Therefore, the `umask` value of `002` is also applied to the data directly managed by JobCenter, permissions of created files, and any user-created files executed from unit job scripts, possibly resulting in an unexpected condition.

For this reason, in order to manage `umask` strictly, you need to exercise control by such means as explicitly setting `umask` at the beginning of a unit job script or setting `umask` in each user's `.cshrc` by explicitly specifying `/bin/csh` in the "script execution shell command" of the unit job parameter.

3.11. nqsstop (Stopping Daemon Processes)

■ UNIX version

```
/usr/lib/nqs/nqsstop [$sitename]
```

■ Windows version

(None)

3.11.1. Function Description

- The nqsstop command stops JobCenter. Note that the JobCenter daemon can be stopped only by the root account (super user). This command is implemented only for the UNIX version.
- This command stops the JobCenter daemon and the related daemon processes.
- JobCenter automatically stops when the system is shut down. Use this command when you want to stop JobCenter at a desired time without shutting down the system.

3.11.2. Options

```
$site-name
```

When stopping the daemon started in site mode, specify the same local site name or cluster site name used at the time of startup.

3.11.3. Return Values

Return value	Description
0	Normal end
1	Abnormal end

3.11.4. Notes

- If you do not use this command and stop the daemon by using the shutdown or kill command of qmgr, some JobCenter-related processes do not end normally.

Therefore, if those processes remain when the JobCenter daemon is restarted, they may fail to run normally.

- After execution, this command may end with or without the message "jnwengine: shutdown." displayed.

This message may not be displayed if the message output source process ends earlier before the message is displayed on the terminal. However, this is not due to an internal processing error, and there is no problem in particular.

3.12. nmapmgr (Managing the Network Configuration)

■ UNIX version

```
/usr/bin/nmapmgr
```

■ Windows version

```
%InstallDirectory%\bin\qcmd\nmapmgr
```

3.12.1. Function Description

- The nmapmgr command builds and configures the network database to be used by JobCenter. This network database includes the six basic elements listed below.

Table 3.11. Database elements of the nmapmgr command

Element	Description
mid	This is the machine ID uniquely assigned to each of the machines comprising the JobCenter network. JobCenter uses this mid to identify a specific machine. The minimum value of the ID is 1. The maximum value of the ID is $2^{31} - 1$.
principal-name	This is a unique host name in the network to which JobCenter belongs. The network is accessed by using this name.
alias	This is the alias of principal-name of the machine in the network. The alias is understood only by the local JobCenter host. (It is not interpreted as the host for communication on the network.) Use it in such cases as when you want to identify a machine by a work role, not by its original machine name. The alias can be a character string of no more than 255 characters.
uid	This is the user ID. The user ID in the remote host can be mapped to any desired user ID of the local host.
gid	This is the group ID. The group ID in the remote host can be mapped to any desired group ID of the local host.
type	This is the supported JobCenter type. Currently, the following types are available. <ul style="list-style-type: none"> ■ nec JobCenter running on UNIX or Linux or NQS with functions extended for NEC ■ necnt JobCenter running on a Windows OS ■ cos NQS for COSMIC (when the linked machine is ACOS)

- The operation of changing the network configuration by using this command can be performed only by the root account (super user) in a UNIX environment or the JobCenter administrator in a Windows environment.
- Other users can execute only the show and get sub commands.
- When you execute the nmapmgr command, the "NMAPMGR>:" prompt appears. In this status, you can enter the sub commands described below. To return to the shell or command prompt, use the exit or quit sub command.

3.12.2. Notes

- When you configure a cluster site, you need to set the site name in environment variable NQS_SITE in advance.

For details, see the cautions given in [Chapter 1, 「Command List」](#) .

3.12.3. Sub Commands

The letters in upper case in the description show the abbreviated form of the command. For example, in the case of CHange Name, the command can be abbreviated as ch n. Note that, if this command is abbreviated as c n, it cannot be distinguished from the CReate command, thus resulting in an error.

While the actual commands can be written in both uppercase and lowercase, some parameters such as the machine name are case sensitive.

The maximum character string length of a sub command is 256 bytes per line (UNIX) or 254 bytes per line (Windows).

Every command needs to be entered on a single line. The line continuation character "\" cannot be used.

Add Mid \$mid \$principal-name

The machine name information is added to the network configuration by using \$mid and \$principal-name in combination.

\$principal-name is a name determined (at JobCenter setup) to uniquely identify the machine corresponding to the mid.

Add Name \$alias \$to-mid

An alias (\$alias) of the machine identified by the machine ID of \$to-mid is added.

Multiple \$alias names can be assigned to a single mid from both this sub command and CL/Win. However, you cannot delete only an alias from CL/Win while leaving a principal name undeleted.

To delete only an alias, use the Delete Name sub command.

Add Uid \$from-mid \$from-uid \$to-uid

The user mapping setting of the user account indicated by \$from-uid in the remote machine identified by the machine ID of \$from-mid and the user account indicated by \$to-uid in the local machine is added.

Note that, when you perform the user mapping operation from CL/Win, you connect to the machine (job request transfer source) indicated by \$from_mid. When setting the user mapping by executing this sub command, in contrast, you perform the operation on the mapping destination machine (job request transfer destination).

If you set the user mapping in the request transfer source machine by executing this sub command, it is not interpreted.

Note that the transfer source user ID and transfer destination user ID have a mapping relationship of n:1. A mapping relationship of 1:n cannot be set, because the job request execution user cannot be determined uniquely.

CHange Name \$mid \$principal-name

A principal-name (\$principal-name) of the machine identified by the machine ID of \$mid is changed.

CReate

The network mapping database is cleared.

When you execute this sub command, the following confirmation message appears.

```
Are you sure to initialize the Network mapping database? (y/[n]) :
```

To clear the network mapping database, type y or Y. To cancel the command, type n or N.



When you execute this sub command, the network mapping databases of all the machines including your own are cleared. Therefore, do not use it except in an emergency.

Note that, if you clear the database, you become unable to start or stop JobCenter.

If you clear the database by mistake, register your machine name and machine ID that had been set before the database was cleared in your machine again and have them recognized again, by using the Add Mid and SEt Local Name sub commands. After that, register your machine name to the machine group by using the set machine_group sub command of qmgr, and this makes it possible to start JobCenter.

Delete DEFUId \$from-mid

The default user mapping setting to the user account in the remote machine identified by the machine ID of \$from-mid is deleted.

See also the SEt DEFUId sub command.

Delete Gid \$from-mid \$from-gid

The mapping relationship related to the group ID indicated by \$from-gid in the remote machine identified by the machine ID of \$from-mid is deleted.

Delete Mid \$mid

The information of the machine identified by the machine ID of \$mid is deleted from the mapping database.

See also the Add Mid sub command.

Delete Name \$alias

An alias (\$alias) is deleted from the mapping database.

See also the Add Name sub command.

Delete Uid \$from-mid \$from-uid

The user mapping setting related to the user account indicated by \$from-uid in the remote machine identified by the machine ID of \$from-mid is deleted.

See also the Add Uid sub command.

Exit

The nmapmgr command is terminated.

Get Gid \$from-mid \$from-gid

The group ID in the local machine mapped to the group ID indicated by \$from-gid in the machine identified by the machine ID of \$from-mid is displayed.

Get Mid \$name

When a principal name or alias is specified in \$name, the machine ID assigned to that machine is displayed.

Get Name \$mid

The principal name of the machine identified by the machine ID of \$mid is displayed.

Use this sub command in combination with the Get Mid sub command to identify the principal name when you cannot tell whether the registered machine name is a principal name or alias.

Get Uid \$from-mid \$from-uid

The user ID in the local machine mapped to the user ID indicated by \$from-uid in the machine identified by the machine ID of \$from-mid is displayed.

Help

A list of sub commands is displayed.

Quit

The nmapmgr command is terminated.

SEt DEFUid \$from-mid \$def-uid

The user ID indicated by \$def-uid as the default user mapping setting to the user account in the remote machine identified by the machine ID of \$from-mid is set.

SEt Local Name \$localname

The principal name set in JobCenter on your machine is changed to \$localname.



If you have changed the principal name of your machine (not another machine) by using the CHange Name sub command, register your machine's principal name again by using this sub command. If you do not register it again, JobCenter may fail to operate normally.

SEt Mail_address \$mid \$address

The mail address of the machine identified by the machine ID of \$mid is set to \$address.

SEt Type \$mid \$type

The JobCenter machine type specified in \$type is set in the machine identified by the machine ID of \$mid. (Normally, when you add a machine icon from CL/Win, the appropriate machine type is automatically selected.)

In \$type, you can specify one of the following.

■nec

Specify this when the machine of \$mid is UNIX.

■necnt

Specify this when the machine of \$mid is Windows.

■cos

Specify this when the machine of \$mid is ACOS.



When linking with an ACOS machine, you cannot register the machine from CL/Win. Use the Add Name sub command and this sub command to register the machine information.

SHow State [\$name]

When a principal name or alias is specified in \$name, the mapping state, machine ID, and machine type of that machine are displayed.

If you omit \$name, all machine information is displayed.

3.13. qmgr (Managing the Configuration and Operation)

■ UNIX version

```
/usr/bin/qmgr
```

■ Windows version

```
%InstallDirectory%\bin\qcmd\qmgr
```

3.13.1. Function Description

The qmgr command is a program that system administrators and operators use for NQS-related controls in the JobCenter system on the local machine.

When you start the qmgr command, it returns the "Mgr:" prompt after which you can enter the sub commands described later. To end the qmgr command, use the exit command.

Note that, in the Windows version, the qmgr command can be started only by the user specified as the JobCenter administrator during setup, regardless of the settings.

3.13.2. Notes

- When you configure a cluster site, you need to set the site name in environment variable NQS_SITE in advance.

For details, see the cautions given in [Chapter 1, 「Command List」](#).

3.13.3. Privileges

The following two types of privilege are provided for the qmgr sub systems.

- The JobCenter administrator (a user who is shown with :m by the show managers sub command) is permitted to use all the sub commands.
- The JobCenter operator (a user who is shown with :o by the show managers sub command) is permitted to use only a limited set of sub commands, excluding setting, addition, and deletion of queues, forms, devices, destination queues, and administrator user accounts.

The required privileges are indicated in the description of each sub command given below.

A user with no privileges can use only the help and show sub commands.

3.13.4. Pager Function (UNIX Version)

You can specify the program to be used for paging in environment variable PAGER. The output of the help and show sub commands is used as the standard input of this program. For example, suppose that you set the environment variable as follows.

```
PAGER=/usr/bin/more
export PAGER
```

Then, the output of help is paged to the program and displayed.

If the specified program cannot be executed, the output is displayed without being paged.

3.13.5. Resource Limits

The formats of the limits specified in the set and modify sub commands are as follows.

3.13.5.1. Time Limit

Specify a time-related limit in the format below.

[[hours:]minutes:]seconds[.fraction]

If there is no limit, specify the character string "unlimited".

Table 3.12. Examples of the time limit specification

Specification example	Description
1234:58:21.29	1234 hours, 58 minutes, 21.29 seconds
59:01	59 minutes, 1 second
12345	12345 seconds
121.1	121.1 seconds

3.13.5.2. Size Limit

Specify a size-related limit in the format below.

integer[.fraction][units]

The units that can be specified in units are as follows.

Notation	Unit
B	Bytes
kb	Kilobytes
mb	Megabytes
gb	Gigabytes

If you do not specify units, the unit of bytes is assumed. If there is no limit, specify the character string "unlimited".

The limit may be converted to a value that is convenient for the machine.

Table 3.13. Examples of the size limit specification

Specification example	Description
1234	1234 bytes
1234kb	1234 kilobytes
1234.5gb	1234.5 gigabytes

3.13.5.3. Nice Value

As the nice value, specify a value directly. A request is connected to a queue only when all the limits of that request are below the corresponding limits of the queue. If "unlimited" is specified for any limit in a request, "unlimited" must be specified for the corresponding limit of the queue as well. If there is any limit not specified in a request, the corresponding limit of the queue is used as the limit of that request.

Once a request is connected to a queue, the limits of the request remain unchanged even if the limits of the queue are changed.

In the UNIX OS for which SVR4.2MP is introduced, the memory size limit of each process must be larger than the stack segment size limit of each process. The default stack segment size limit is 256 MB. The memory size limit needs to be larger than that.

Note that the shell script that executes a request is subject to the resource limits as well.

3.13.6. Sub Commands

The letters in uppercase in the description show the abbreviated form of the sub command. For example, ADd DESTination can be abbreviated as "ad des". Note that, if this command is abbreviated as "ad de", it cannot be distinguished from the ADd DEvIce sub command, thus resulting in an error.

While the actual sub commands can be written in both uppercase and lowercase, object names such as queue and user names are case sensitive. "... " indicates that the preceding part enclosed in brackets ([]) can be repeated as many times as you want. The maximum character string length of a sub command is 256 bytes per line (UNIX) or 254 bytes per line (Windows). When you enter a sub command that is multiple lines long, insert the line continuation character "\" before each line break.

A request ID is uniquely assigned within NQS when a request is submitted and consists of a sequence number and a submission machine name. If you specify only the sequence number part, the machine name part is interpreted to be the local site.

ABort Queue \$queue [\$seconds]

All the requests currently running in the queue specified in \$queue are aborted as described below.

First, the SIGTERM signal is sent to each process of the requests currently running in the queue. Next, after the number of seconds specified in \$seconds elapses, the SIGKILL signal is sent to the remaining processes of each request running in the specified queue.

In \$seconds, you can specify a value between 0 and 600. If you do not specify this parameter, the default is 60 (seconds).

All the requests that are aborted by this command are deleted from the queue, and the result files of those requests are returned.



When a network queue is specified in \$queue

All the network requests that are running in the specified queue are deleted. In this case, the result file that is originally supposed to be transferred by each deleted network request is placed in the home directory of the owner of the request on the execution machine.

The JobCenter operator privileges are required.

ADd Queues = (\$queue[, \$queue]...) \$complex

One or more queues specified in \$queue are added to the batch queue complex specified in \$complex.

The batch queue complex specified in \$complex needs to be created in advance by using the Create Complex sub command.

The JobCenter administrator privileges are required.

ADd DESTination = \$destination \$queue \$position

ADd DESTination = (\$destination[, \$destination]...) \$queue \$position

The destination specified in \$destination is added as a valid destination of the pipe queue specified in \$queue. If \$destination is specified multiple times, the request is transferred to the destinations in order from the top of the set list.

In \$position, specify whether the destination you are going to set is to be placed before or after the destinations already set in the pipe queue, by using the formats described below. If you do not

specify this parameter, the new destination is added at the end of the destination list already set in the pipe queue.

■ **after \$queue_name**

The new destination is placed after the destination specified in \$queue_name.

■ **before \$queue_name**

The new destination is placed before the destination specified in \$queue_name.

■ **first**

The new destination is placed before all the destinations.

■ **last**

The new destination is placed after all the destinations.

The JobCenter administrator privileges are required.

ADd DEvice = \$device \$queue

ADd DEvice = (\$device [, \$device] ...) \$queue

(This function is not supported in R12.7 and later.)

The device specified in \$device is added as a resource of the queue specified in \$queue. You must specify an existent device (see the Create DEVICE sub command).

The JobCenter administrator privileges are required.

ADd Forms \$form-name [\$form-name]...

The form specified in \$form-name is added to the valid form list.

The JobCenter administrator privileges are required.

ADd Groups = \$group \$queue

ADd Groups = (\$group [, \$group] ...) \$queue

The group specified in \$group is added to the access permission list of the queue specified in \$queue. This allows the users who belong to that group to submit requests to the queue specified in \$queue.

You can specify a group in the following two formats.

■ **\$group-name**

■ **[\$group-id]**

For example, when \$group-name is "group1" and \$group-id is "2001", specify either "group1" or "[2001]".

The queue access restriction function is enabled by the SEt NO_Access sub command and disabled by the SEt UNrestricted_access sub command. Once the function is disabled, the queue access permission list is initialized.

Note that, if a group definition is deleted from the OS, the access permission list is not synchronized with that change. In that case, delete the group by using the DElete Groups sub command with [\$group-id] specified.

In the case of Windows, since every JobCenter user is required to participate in a JobCenter group, a group is not displayed in the access permission list even if it is set with this function.

The JobCenter administrator privileges are required.

ADd Managers \$manager [\$manager]...

The account specified in \$manager is added to the list of authorized JobCenter administrators.

You can specify \$manager in the following two formats.

■ \$local_account_name:m or \$local_account_name:o

■ [\$local_user_id]:m or [\$local_user_id]:o

For example, when \$ local_account_name is "user1" and \$local_user_id is "1001", specify either "user1:m" or "[1001]:m".

■ When :m is added to an account name or [user_id]

This means to add the privileges of a JobCenter administrator who is permitted to use all the qmgr commands.

■ When :o is added to an account name or [user_id]

This means to add the privileges of a JobCenter operator who is permitted to use some of the qmgr commands.

For details about privileges, see [Section 3.13.3, "Privileges"](#) earlier in this section.

In the case of UNIX, the nsumsmgr and root (super user) accounts are registered with the JobCenter administrator privileges in advance.

In the case of Windows, JobCenter administrators are registered with the JobCenter administrator privileges in advance.

The JobCenter administrator privileges are required.

ADd Users = \$user \$queue

ADd Users =((\$user [,\$user]...) \$queue

The user specified in \$user is added to the access permission list of the queue specified in \$queue. This allows that user to submit requests to the queue specified in \$queue.

You can specify \$user in the following two formats.

■ \$user-name

■ [\$user-id]

For example, when \$user-name is "user1" and \$user-id is "1001", specify either "user1" or "[1001]".

The queue access restriction function is enabled by the SEt NO_Access sub command and disabled by the SEt UNrestricted_access sub command. Once the function is disabled, the queue access permission list is initialized.

In the case of Windows, a user account can be added to the queue access permission list only when that user account belongs to a JobCenter group and is registered in the user list of [Server Environment Setting].

When a user definition is deleted from the OS or a user is cleared from [Server Environment Setting] of Windows, the access permission list is not synchronized with that change. In that case, delete the user by using the DElete Users sub command with [\$user-id] specified.

The JobCenter administrator privileges are required.

Create Batch_queue \$queue PRiority=\$p [Plpeonly] [Run_limit= \$n]

A batch queue having the name specified in \$queue is created with the queue priority specified in \$p.

The lowest priority of \$p is 0, and the highest is 63.

If you specify Plpeonly, the requests to be registered in the queue specified in \$queue are limited to those that are passed from the pipe queue.

In \$n, specify the number of requests that can be run simultaneously in each individual batch queue. If you do not specify Run_limit, the default is 1.

By default, the maximum value that can be specified in Run_limit is 220 in the UNIX version and 190 in the Windows version. If you have set the limit on the number of requests that can be run simultaneously by using the SEt RUn_limit sub command, the upper limit is equal to that value.

The JobCenter administrator privileges are required.

Create Complex =(\$queue[\$queue]...) \$complex

A queue complex comprising a set of batch queues specified in \$queue is created.

The set of batch queues grouped by JobCenter as a queue complex can have run-limit for the entire complex. See the SEt COMplex sub commands.

The JobCenter administrator privileges are required.

Create DEVICE \$device FOrms=\$forms FULLname=\$filename Server =(\$server)

(This function is not supported in R12.7 and later.)

The JobCenter device belonging to the form specified in \$forms is defined and associated with the server specified in \$server.

In \$server, specify the absolute path name of the program binary and several arguments (e.g., options).

\$filename is the absolute path name of the device (special file). For example, /dev/device may be specified in the UNIX version.

The JobCenter administrator privileges are required.

Create DEVICE_queue \$queue PRiority=\$p [Device=\$device] [Device =(\$device [, \$device]...)] [Plpeonly]

(This function is not supported in R12.7 and later.)

A device queue having the name specified in \$queue is created with the queue priority specified in \$p.

The lowest priority of \$p is 0, and the highest is 63.

If you specify Plpeonly, the requests to be registered in this device queue are limited to those that are passed from another pipe queue.

In \$device, list one or more JobCenter devices that service this queue.

The JobCenter administrator privileges are required.

```
Create Network_queue $queue Destination=$dest-name Priority = $p [Server = ($server)] [Run_limit = $n]
```

The network queue \$queue whose destination is the host specified in \$dest-name is created with the queue priority specified in \$p. (Not supported in the Windows version.)

The network queue is used to transfer request execution result files. If there is no network queue that corresponds to the destination host, the default network queue (Default-NetQue) is used (for UNIX only; this queue does not exist in Windows).

While multiple network queues can be created for a host, the only network queue that is actually used is the one with the highest queue priority and the other queues are ignored. You cannot specify more than one host for a network queue. Also, once a queue is created, its host cannot be changed.

You can specify \$dest-name in the following two formats.

■ \$machine-name

■ [\$machine-id]

For example, when \$machine-name is "machine1" and \$machine-id is "100", specify either "machine1" or "[100]".

The lowest priority of \$p is 0, and the highest is 63.

In \$server, specify the name of the server program to be used in this queue. If you do not specify this parameter, the server program specified in the SET NETWORK Client sub command is used. To change the server program after creating a queue, use the SET NETWORK_client sub command.

In \$n, specify the number of requests that can be transferred simultaneously by each individual network queue. If you do not specify Run_limit, the default is 1.

The default upper limit value that can be specified in Run_limit is 220. If you have set the limit on the number of requests that can be run simultaneously by using the SET RUN_limit sub command, the upper limit is equal to that value.

The JobCenter administrator privileges are required.

```
Create Pipe_queue $queue PRiority=$p SErver=($server) [Destination = $destination] | [Destination = ($destination[, $destination]... )] [Pipeonly] [Run_limit = $n ] [STaywait] [Check]
```

A pipe queue having the name specified in \$queue is created with the queue priority specified in \$p of PRiority and associated with the server specified in \$server.

The lowest priority of \$p is 0, and the highest is 63.

In \$server, specify the absolute path name of the transfer program, options, etc. (You do not need to specify these in the Windows version of R12.8 and later.)

In \$destination, list one or more destination queues to which requests are transferred from this pipe queue.

If you specify Pipeonly, the requests to be registered in this pipe queue are limited to those that are passed from another pipe queue.

In \$n, specify the number of requests that can be transferred simultaneously by each individual pipe queue. If you do not specify Run_limit, the default is 1.

By default, the maximum value that can be specified in Run_limit is 220 in the UNIX version and 190 in the Windows version. If you have set the limit on the number of requests that can be run simultaneously by using the SEt RUn_limit sub command, the upper limit is equal to that value.

If you specify STaywait, a request with a time specified (-a option of qsub) is made to wait in the pipe queue.

If you specify Check, the check function checks whether there is any destination to which to transfer a request before registering it in the pipe queue. Note that, in this case, the destination queue must be in the local machine. If no destination is found, the request is not queued, resulting in an error.

The JobCenter administrator privileges are required.

DElete Complex \$complex

The queue complex specified in \$complex is deleted.

The JobCenter administrator privileges are required.

DElete DESTination =\$destination \$queue

DElete DESTination =(\$destination[, \$destination]...) \$queue

The destination queue specified in \$destination is deleted from the request transfer destination queue list of the pipe queue specified in \$queue.

Note that, if there is any request being transferred from the specified queue to the destination queue to be deleted, its transfer continues.

If all the destination queues of the pipe queue are deleted in the way described above, the pipe queue enters a state in which it keeps requests queued without transferring them, as when it is stopped.

The JobCenter administrator privileges are required.

DElete DEVICE \$device

(This function is not supported in R12.7 and later.)

The device specified in \$device is deleted. Note that you cannot delete the device unless it is disabled (see the DIsable Device command).

The JobCenter administrator privileges are required.

DElete DEVICE =\$device \$queue

DElete DEVICE =(\$device [, \$device]...) \$queue

(This function is not supported in R12.7 and later.)

The device specified in \$device is deleted from the enabled JobCenter device list of the device queue specified in \$queue. The processing of any request running on the specified JobCenter device continues.

If all the entries are deleted from the enabled JobCenter device list of the device queue, it enters the same state as when it is stopped.

The JobCenter administrator privileges are required.

DElete Forms \$form_name [\$form_name]...

The form specified in \$form_name is deleted from the valid form list.

The JobCenter administrator privileges are required.

DElete Groups = \$group \$queue

DElete Groups =(\$group[, \$group]...) \$queue

The group specified in \$group is deleted from the access permission list of the queue specified in \$queue. You can specify a group in the following two ways.

■ \$group-name

■ [\$group-id]

For example, when \$group-name is "group1" and \$group-id is "2001", specify either "group1" or "[2001]".

The JobCenter administrator privileges are required.

DElete Managers \$manager [\$manager]...

The account specified in \$manager is deleted from the JobCenter administrator list.

You can specify \$manager in the following two formats.

■ \$local_account_name:m or \$local_account_name:o

■ [\$local_user_id]:m or [\$local_user_id]:o

For example, when \$ local_account_name is "user1" and \$local_user_id is "1001", specify either "user1:m" or "[1001]:m".

■ When :m is added to an account name or [user_id]

This means to delete the privileges of a JobCenter administrator who is permitted to use all the qmgr commands.

■ When :o is added to an account name or [user_id]

This means to delete the privileges of a JobCenter operator who is permitted to use some of the commands.

For details about privileges, see [Section 3.13.3, "Privileges"](#) earlier in this section.

In the case of UNIX, the nsumsmgr and root (super user) accounts cannot be deleted from the JobCenter administrator list.

In the case of Windows, JobCenter administrator accounts cannot be deleted from the JobCenter administrator list.

The JobCenter administrator privileges are required.

DElete Queue \$queue

The queue specified in \$queue is deleted.

In order for a queue to be deleted, that queue must have no requests queued in it and be disabled (see Disable Queue).

The default network queue (DefaultNetQue) cannot be deleted by using this sub command. (This default queue exists only in UNIX; it does not exist in Windows.)



Even if the queue to be deleted is specified as the destination of another pipe queue, the deletion done by this sub command is not reflected in the destination list of that pipe queue, thus leaving the destination in the list.

The JobCenter administrator privileges are required.

DElete Request \$request_id

The request specified in \$request_id is deleted. If a specified batch request is in the process of outputting a result file, that batch request becomes the parent and all the associated network requests are deleted. The result file that is originally supposed to be transferred by each of those deleted network requests is placed in the home directory of the request owner on the machine executing the request.

Note that you cannot specify a network request directly in this sub command.

The JobCenter operator privileges are required.

DElete User = \$user \$queue

DElete User =(\$user[, \$user]...) \$queue

The user specified in \$user is deleted from the access permission list of the queue specified in \$queue. You can specify a user in the following two formats.

■ \$user-name

■ [\$user-id]

For example, when \$user-name is "user1" and \$user-id is "1001", specify either "user1" or "[1001]".

The JobCenter administrator privileges are required.

Disable Device \$device

(This function is not supported in R12.7 and later.)

The device specified in \$device is disabled.

If all the entries in the enabled JobCenter device list of a device queue are disabled, that device queue enters the same state as when it is stopped.

The JobCenter operator privileges are required.

Disable Group = \$group

Disable Group =(\$group[, \$group]...)

The group specified in \$group is disabled for JobCenter.

This makes it impossible for the users who belong to that group to submit requests to any queue of JobCenter.

You can specify a group in the following two formats.

■ \$group-name

■ [\$group-id]

For example, when \$group-name is "group1" and \$group-id is "2001", specify either "group1" or "[2001]".

The JobCenter administrator privileges are required.

Disable Queue \$queue

The queue specified in \$queue is disabled.

Unlike with the STOp Queue sub command, new requests cannot be submitted to a disabled queue. The processing of the requests already running on the queue continues.

If the specified queue is already disabled, nothing happens.

The JobCenter operator privileges are required.

Disable User = \$user

Disable User =(\$user[, \$user]...)

The user specified in \$user is disabled for JobCenter.

This makes it impossible for that user to submit requests to any queue of JobCenter.

You can specify a user in the following two formats.

■ \$user-name

■ [\$user-id]

For example, when \$user-name is "user1" and \$user-id is "1001", specify either "user1" or "[1001]".

The JobCenter administrator privileges are required.

ENable Device \$device

(This function is not supported in R12.7 and later.)

The device specified in \$device is enabled.

The JobCenter operator privileges are required.

ENable Group = \$group

ENable Group =(\$group[, \$group]...)

The group specified in \$group is enabled for JobCenter.

This allows the users who belong to the group enabled for JobCenter to submit requests to the JobCenter queues.

You can specify a group in the following two formats.

■ \$group-name

■ [\$group-id]

For example, when \$group-name is "group1" and \$group-id is "2001", specify either "group1" or "[2001]".

The JobCenter administrator privileges are required.

ENable Queue \$queue

The queue specified in \$queue is enabled.

Once a queue is enabled, requests can be submitted to that queue.

If the specified queue is already enabled, nothing happens.

The JobCenter operator privileges are required.

ENable User = \$user

ENable User =(\$user[, \$user]...)

The user specified in \$user is enabled for JobCenter.

This allows the user enabled for JobCenter to submit requests to the JobCenter queues.

You can specify a user in the following two formats.

■ \$user-name

■ [\$user-id]

For example, when \$user-name is "user1" and \$user-id is "1001", specify either "user1" or "[1001]".

The JobCenter administrator privileges are required.

EXit

The qmgr command is terminated.

HElp [\$command]

The help information can be obtained.

If you execute the HElp command with no argument, the available commands are listed.

If you execute the HElp command with the \$command argument, the information about the specified command is displayed.

In \$command, you can specify a substring as long as it is unique.

If you specify the \$command argument in more detail, more detailed information is displayed.

HOId Request \$request_id

The request specified in \$request_id is held.

A request held by this sub command cannot be released by the qrls(1) command. To release the request, use RELEase Request.

If you shut down JobCenter with a request held, that status is inherited the next time you start up.

The qrls(1) command is supported only in the UNIX version.

The JobCenter operator privileges are required.

Lock Local_daemon

The text segment and data segment of the JobCenter NQS daemon are locked in memory. See the PROCLOCK option of plock(2).

To unlock the segments, you need to execute Unlock Local_daemon described later.

The JobCenter operator privileges are required.

MODify Request DATA_limit = (\$value) \$request_id

The per-process data segment size limit of the request specified in \$request_id is changed to the value specified in \$value. See [Section 3.13.5, "Resource Limits"](#) earlier in this section.

Note that you cannot make this change if the specified request is already running.

In the case of a batch queue, you can change the limit only when the system supports the per-process data segment size limit function.

The JobCenter operator privileges are required.

MODify Request MAIL_End \$mode \$request_id

The mode in which mail is transmitted at the end of the execution of the request specified in \$request_id is changed to the mode specified in \$mode.

Note that you cannot make this change if the specified request is already running.

Here is how you specify \$mode.

■on

Transmit mail

■off

Do not transmit mail

The JobCenter operator privileges are required.

MODify Request MAIL_Start \$mode \$request_id

The mode in which mail is transmitted at the start of the execution of the request specified in \$request_id is changed to the mode specified in \$mode.

Note that you cannot make this change if the specified request is already running.

Here is how you specify \$mode.

■on

Transmit mail

■off

Do not transmit mail

The JobCenter operator privileges are required.

MODify Request MAIL_User \$user \$request_id

The user to whom mail related to the request specified in \$request_id is transmitted is changed to the user specified in \$user.

Note that you cannot make this change if the specified request is already running.

The JobCenter operator privileges are required.

MODify Request Nice_value \$value \$request_id

The nice limit value of the request specified in \$request_id is changed to the value specified in \$value. See [Section 3.13.5, "Resource Limits"](#) earlier in this section.

Note that you cannot make this change if the specified request is already running.

The JobCenter operator privileges are required.

MODify Request PPCOresize_limit = (\$value) \$request_id

The per-process core file size limit of the request specified in \$request_id is changed to the value specified in \$value. See [Section 3.13.5, "Resource Limits"](#) earlier in this section.

Note that you cannot make this change if the specified request is already running.

In the case of a batch queue, you can change the limit only when the system supports the per-process core file size limit function.

The JobCenter operator privileges are required.

MODify Request PPCPu_limit = (\$value) \$request_id

The per-process CPU time limit of the request specified in \$request_id is changed to the value specified in \$value. See [Section 3.13.5, "Resource Limits"](#) earlier in this section.

Note that you cannot make this change if the specified request is already running.

In the case of a batch queue, you can change the limit only when the system supports the per-process CPU time limit function.

The JobCenter operator privileges are required.

MODify Request PPMemory_limit = (\$value) \$request_id

The per-process memory size limit of the request specified in \$request_id is changed to the value specified in \$value. See [Section 3.13.5, "Resource Limits"](#) earlier in this section.

Note that you cannot make this change if the specified request is already running.

You can change the limit only when the system supports the per-process memory size limit function.

The JobCenter operator privileges are required.

MODify Request PPPermfile_limit = (\$value) \$request_id

The per-process permanent file size limit of the request specified in \$request_id is changed to the value specified in \$value. See [Section 3.13.5, "Resource Limits"](#) earlier in this section.

Note that you cannot make this change if the specified request is already running.

You can change the limit only when the system supports the per-process permanent file size limit function.

The JobCenter operator privileges are required.

MODify Request PRLority \$priority \$request_id

The request priority of the request specified in \$request_id is changed to the value specified in \$priority.

By changing the request priority, you can change the order in which requests are processed in a queue.

Note that you cannot make this change if the specified request is already running.

The JobCenter operator privileges are required.

MODify Request Restart \$mode \$request_id

The mode regarding the restart permission of the request specified in \$request_id is changed to the mode specified in \$mode.

Note that you cannot make this change if the specified request is already running.

Here is how you specify \$mode.

■on

Restart permitted

■off

Restart not permitted

The JobCenter operator privileges are required.

MODify Request SHell = (\$shellname) \$request_id

The shell that executes the request specified in \$request_id is changed to the shell specified in \$shellname.

Note that you cannot make this change if the specified request is already running.

In the case of UNIX, specify the absolute path to a shell program that can be used in the OS, such as "/bin/sh" or "/bin/csh". (Note that normal operation is not guaranteed if a user program is specified.)

Windows does not support execution program specification other than the specification of the absolute path to cmd.exe.

The JobCenter operator privileges are required.

MODify Request STAck_limit = (\$value) \$request_id

The stack segment size limit of the request specified in \$request_id is changed to the value specified in \$value. See [Section 3.13.5, "Resource Limits"](#) earlier in this section.

Note that you cannot make this change if the specified request is already running.

The JobCenter operator privileges are required.

MODify Request STDERR \$filename \$request_id

The standard error output result file of the request specified in \$request_id is changed to the file specified in \$filename.

Note that you cannot make this change if the specified request is already running.

The JobCenter operator privileges are required.

MODify Request STDERR_Spool \$mode \$request_id

The transfer mode of the standard error output result file of the request specified in \$request_id is changed to the mode specified in \$mode.

Note that you cannot make this change if the specified request is already running.

Here is how you specify \$mode.

■on

Spool mode

■off

Non-spool mode

The JobCenter operator privileges are required.

MODify Request STDOUT \$filename \$request_id

The standard output result file of the request specified in \$request_id is changed to the file specified in \$filename.

Note that you cannot make this change if the specified request is already running.

The JobCenter operator privileges are required.

MODify Request STDOUT_Spool \$mode \$request_id

The transfer mode of the standard output result file of the request specified in \$request_id is changed to the mode specified in \$mode.

Note that you cannot make this change if the specified request is already running.

Here is how you specify \$mode.

■on

Spool mode

■off

Non-spool mode

The JobCenter operator privileges are required.

MOVE Queue \$queue1 \$queue2

All the requests currently queued in the queue specified in \$queue1 are moved to the queue specified in \$queue2.

A running request cannot be moved.

The JobCenter operator privileges are required.

MOVE Request \$request_id [\$request_id ...] \$queue

The request specified in \$request_id is moved to the queue specified in \$queue.

A running request cannot be moved.

When you specify multiple \$request_id parameters in this sub command, delimit them with a space.

The JobCenter operator privileges are required.

Purge Queue \$queue

All the requests queued in the queue specified in \$queue are deleted.

Note that running requests are not be deleted.

The JobCenter operator privileges are required.

RELease Request \$request_id

The held request specified in \$request_id is released.

If you specify a request other than a held one, a warning is issued. However, the status of the request is not affected.

The JobCenter operator privileges are required.

REMOve Queue =(\$queue[, \$queue]...) \$complex

The queue specified in \$queue is removed from the queue complex specified in \$complex.

The JobCenter administrator privileges are required.

RESEt Log_file

The JobCenter log file is initialized.

The existing log file is retained with .bak added to its log file name unless NO_Backup is specified in the SET LOG_file sub command.

The JobCenter administrator privileges are required.

RESEt USERTOKEN

The user token cached when a job was initially executed is cleared.

After the existing user token is cleared, another one is obtained and cached the next time a job is initially executed.

This sub command is supported only in the Windows version.

The JobCenter administrator privileges are required.

REStart Request \$request_id

The running request specified in \$request_id is stopped and registered in the queue again.

Note that a request not permitted to restart is not registered again.

Normally, the request is registered at the end of other waiting requests. If its request priority is higher than those of the other requests, the request is registered at the corresponding preferential position.

When the execution of the request is restarted, it is executed again from the beginning of the script.

The JobCenter operator privileges are required.

RESUme Request \$request_id

The execution of the request specified in \$request_id is resumed.

Only a suspended request can be specified.

The JobCenter operator privileges are required.

SEt ALI_queue Intra_queue_scheduling_type \$type

The scheduling type of the requests in all the batch queues in the system is set to the type specified in \$type.

Here is how you specify \$type.

■type0

Between requests using the current scheduling type and having the same request priority, the execution order is determined by the submission order.

■type1

As for requests having the same request priority, the requests of the individual users are executed in order, regardless of the submission order. The number of requests that a user can execute continuously is predetermined. If the user submits more requests than that, he or she waits until other users' requests are executed.

This prevents the requests of only one particular user from being executed. The number of requests that a user can execute continuously is set in the set continuous_scheduling_number sub command.

The JobCenter administrator privileges are required.

SEt CHeck \$queue

The pre-check function is set for the pipe queue specified in \$queue.

The JobCenter administrator privileges are required.

SEt COMplex Group_run_limit = \$run_limit \$complex

The number of requests that can be run simultaneously for a group in the queue complex specified in \$complex is set.

This sub command is supported only in the UNIX version.

If you set a limit on a queue complex by using this sub command, requests of the users of a group are not run simultaneously in that queue complex when the total number of their requests exceeds the number specified in \$run_limit.

If an attempt is made to start a request beyond the number specified in \$run_limit, that request is queued.

In \$run_limit, you can specify a value between 1 and the upper limit. The default upper limit is 220. If you have set the limit on the number of requests that can be run simultaneously by using the SET RUN_limit sub command, the upper limit is equal to that value.

The default value of \$run_limit is 1.

The JobCenter operator privileges are required.

SEt COMplex Run_limit = \$run_limit \$complex

The limit on the number of requests that can be run simultaneously in the JobCenter queue complex specified in \$complex is set.

In \$run_limit, specify the maximum number of requests that are allowed to be run simultaneously in the queue complex.

In \$run_limit, you can specify a value between 1 and the upper limit. By default, the upper limit is 220 in the UNIX version and 190 in the Windows version. If you have set the limit on the number of requests that can be run simultaneously by using the SET RUN_limit sub command, the upper limit is equal to that value.

The JobCenter operator privileges are required.

SEt COMplex User_run_limit = \$run_limit \$complex

The number of requests that can be run simultaneously by a user in the queue complex specified in \$complex is set.

This sub command is supported only in the UNIX version.

If you set a limit on a queue complex by using this sub command, requests of a user are not run simultaneously in that queue complex when the number of the requests exceeds the number specified in \$run_limit.

If an attempt is made to start a request beyond the number specified in \$run_limit, that request is queued.

In \$run_limit, you can specify a value between 1 and the upper limit. The default upper limit is 220. If you have set the limit on the number of requests that can be run simultaneously by using the SET RUN_limit sub command, the upper limit is equal to that value.

The default value of \$run_limit is 1.

The JobCenter operator privileges are required.

SEt CONTinuous_scheduling_number \$number \$queue

The number of requests that a user can execute continuously in the queue specified in \$queue is specified in \$number.

In \$number, you can specify a value between 0 and 50. The default is 0 (unlimited).

This function takes effect only for a batch queue.

The JobCenter administrator privileges are required. The function is not available in the Windows version.

SEt CORefile_limit = (\$limit) \$queue

The per-process core file limit is set for the batch queue specified in \$queue.

This value is compared with the per-process core file size limit set in a submitted request. If the value is larger than that set in the queue, the registration of that request is rejected.

If the per-process core file limit is not set for a request, the value set in the queue is used as the limit of that request.

Requests that are already registered in the queue are not affected by a newly set value.

For the syntax of \$limit, see [Section 3.13.5, "Resource Limits"](#) earlier in this section.

This sub command is valid only when the system supports the setting of the per-process core file limit.

The JobCenter administrator privileges are required.

SEt CUStomjob Batch_queue \$queue

The custom queue attribute is set for the batch queue specified in \$queue. The custom queue attribute needs to be set for a batch queue that executes extended custom jobs.

The administrator privileges are required.



If an extended custom job component is submitted to a batch queue that is not assigned the custom queue attribute, the execution of the custom job ends with an error.

A regular custom job component or unit job component cannot be executed in a batch queue assigned the custom queue attribute. Submitting any such component to a custom queue places that custom queue in the [STOPPED] status.

SEt DAta_limit = (\$limit) \$queue

The per-process data segment size limit is set for the batch queue specified in \$queue.

This value is compared with the per-process data segment size limit set in a submitted request. If the value is larger than that set in the queue, the registration of that request is rejected.

If the per-process data segment size limit is not set for a request, the value set in the queue is used as the limit of that request.

Requests that are already registered in the queue are not affected by a newly set value.

For the syntax of \$limit, see [Section 3.13.5, "Resource Limits"](#) earlier in this section.

This sub command is valid only when the system supports the setting of the per-process data segment size limit.

The JobCenter administrator privileges are required.

SEt DEBug \$level

The debug level is set. In \$level, you can specify one of the following values.

■ 0

No debugging

■ 1

Minimum debug level

■ 2 or higher

Maximum debug level

The JobCenter administrator privileges are required.

SEt DEFault Batch_request Priority \$priority

The default request priority that determines the start order of a batch request is changed. If you do not change the default request priority, the default is 31.

This is the default value to be used when the user omits to specify a request priority in the qsub(1) command.

The qsub(1) command is supported only in the UNIX version.

The lowest priority of \$priority is 0, and the highest is 63.

The JobCenter administrator privileges are required.

SEt DEFault Batch_request Queue \$queue

The queue specified in \$queue is set as the default batch queue. This is the default value to be used when the user omits to specify a submission queue name in the qsub(1) command.

The default is NONE.

The qsub(1) command is supported only in the UNIX version.

The JobCenter administrator privileges are required.

SEt DEFault DESTination_retry Time \$retry_time_in_seconds

The retry time per request transfer is defined in \$retry_time_in_seconds.

In a single request transfer, JobCenter continues to retry establishing a connection with the request destination until the number of seconds defined as the retry time elapses.

If the retry time expires, JobCenter considers that this transfer has failed and waits for a time specified in Destination_retry Wait. Then, it attempts to transfer the request again.

In \$retry_time_in_seconds, specify a value between 0 and 100000000. The default is 16 (seconds).

The JobCenter administrator privileges are required.

SEt DEFault DESTination_retry Wait \$retry_interval_in_seconds

The time to wait before retrying a request transfer is defined.

If the request destination fails to receive data, JobCenter waits for the defined wait time and then retries the transfer.

In \$retry_interval_in_seconds, specify a value between 1 and 10000000. The default is 300 (seconds).

The JobCenter administrator privileges are required.

SEt DEFault DEVice_request Priority \$priority

(This function is not supported in R12.7 and later.)

The default request priority of a device request is set. This is the value to be used when the user omits to specify a request priority in the qpr(1) command.

The qpr(1) command is supported only in the UNIX version.

The JobCenter administrator privileges are required.

SEt DEFault Expire Time \$expire_time_in_seconds

The number of seconds is specified as the time to keep the status information of a request in the NQS database after the execution of that request.

In \$expire_time_in_seconds, you can specify a value between 0 and 100000000. The default is 259200 (seconds).

The JobCenter administrator privileges are required.

SEt DEFault Network_retry TIME \$retry_time_in_seconds

The maximum retry interval is defined for establishing a connection for a purpose other than transferring a request or returning a network request as an execution result.

JobCenter continues to retry establishing a connection while increasing the retry interval from 1 to 2, 2 to 4, and so on until the interval is equal to the defined value.

In \$retry_time_in_seconds, you can specify a value between 0 and 100000000. The default is 16 (seconds).

The JobCenter administrator privileges are required.

SEt DEFault Network_retry TIME_Out \$time_out_time_in_seconds

The time is set that is allowed to elapse before a socket stream is considered to be disconnected when the network connection destination does not respond.

In \$time_out_time_in_seconds, you can specify a value between 0 and 100000. The default is 300 (seconds).

The JobCenter administrator privileges are required.

SEt DEFault Network_retry Wait \$retry_interval_in_seconds

The wait time is defined for a connection for a purpose other than transferring a request or returning a network request as an execution result. If a network failure occurs, JobCenter waits for the defined wait time and then retries the connection.

In \$retry_interval_in_seconds, you can specify a value between 0 and 10000000. If you specify 0, JobCenter does not retry. The default is 0 (seconds).

The JobCenter administrator privileges are required.

SEt DEFault PRINT_Request Forms \$form-name

(This function is not supported in R12.7 and later.)

The form specified in \$form-name is set as the default output form. This is the form to be used when the user omits to specify the form parameter in the qpr(1) command.

The default is (None).

The qpr(1) command is supported only in the UNIX version.

The JobCenter administrator privileges are required.

SEt DEFault PRINT_Request Queue \$queue

(This function is not supported in R12.7 and later.)

The queue specified in \$queue is set as the default output queue. This is the queue to be used when the user omits to specify the queue parameter in the qpr(1) command.

The default is NONE.

The qpr(1) command is supported only in the UNIX version.

The JobCenter administrator privileges are required.

SEt DEFault STAGe_retry Time \$retry_time_in_seconds

The number of seconds is set as the time during which a network request is allowed to exist.

In \$retry_time_in_seconds, you can specify a value between 0 and 100000000. The default is 259200 (seconds).

A network request continues to retry from the moment its parent batch request enters the result file output state (EXITING) until the time specified in this sub command elapses.

If the existence time expires, the network request no longer retries even when it fails to output a result file for a reason for which a retry is allowed. In that case, the result file is saved in the home directory of the request execution user on the execution machine (UNIX) or in %InstallDirectory%\spool\work (Windows) and the user is notified by mail.

The JobCenter administrator privileges are required.

SEt DESTination = \$destination \$queue

SEt DESTination = (\$destination[, \$destination]...) \$queue

One or more destination queues specified in \$destination are associated with the pipe queue specified in \$queue.

While there is no limit to the number of \$destination parameters you can specify, the byte length of a line of the sub command is limited. To add destination queues beyond that limit, use the ADD DESTination sub command.

The JobCenter administrator privileges are required.

SEt DEVICE = \$device \$queue

SEt DEVICE =(\$device [, \$device]...) \$queue

(This function is not supported in R12.7 and later.)

One or more devices specified in \$device are associated with the queue specified in \$queue.

The JobCenter administrator privileges are required.

SEt DEFault STAGe_retry Wait \$retry_interval_in_seconds

The number of seconds is set as the retry interval for a network request.

In `$retry_interval_in_seconds`, you can specify a value between 0 and 10000000. The default is 300 (seconds).

If a network request fails to output a result file for a reason for which a retry is allowed, it retries after waiting for the time specified in this sub command. In the meantime, the request is in the retry wait (WAITING) state. The network queue to which this request has been submitted does not start the scheduling of a new network request until the next retry begins.

The JobCenter administrator privileges are required.

```
SEt DEVICE_server = ($server) $device
```

(This function is not supported in R12.7 and later.)

The device specified in `$device` is associated with the server specified in `$server`.

In `$server`, specify the absolute path name of the server program followed by the necessary arguments.

The JobCenter administrator privileges are required.

```
SEt ERRor_log [Size=($logfile_size)] [Backup=$backup_number]
```

The settings of the NQS-related error log are changed.

■ `$logfile_size`

Specify the file size of the error log. The specifiable value range is 65536 (64kb) to 4194304 (4mb). The default value is 1048576 (1mb).

If you omit the unit, the byte is assumed. You can specify kb or mb as the unit.

Example of specifying 4 MB: `size=(4mb)`

■ `$backup_number`

Specify the number of backup generations of the error log. The specifiable value range is 1 to 10. The default value is 10.

The JobCenter administrator privileges are required.

```
SEt EVenT_spool Size =($size)
```

The maximum spooling size for an internal event that waits using NQSwatch is set to the value specified in `$size`.

This sub command is supported only in the UNIX version.

A spooling file of an event is created for each entry of that event.

The size of any event notification file created before this setting is made is not affected by this sub command. Also, the size of an event notification file that is actually created is extended up to the system-dependent page size unit.

In `$size`, you can specify a value between 0 and 10485760. The default is 2097152 (bytes). For information about how to specify the size, see [Section 3.13.5, "Resource Limits"](#) earlier in this section.

The JobCenter administrator privileges are required.

SEt EVent_spool Time =(\$time)

The maximum spooling time for an internal event that waits using NQSwatch is set to the time specified in \$time.

The wait of any event notification file created before this setting is made is not affected by it. If the event reception process does not receive an event within the specified time, all the events registered in the corresponding event notification file are discarded and the file is deleted.

Specify \$time in one of the following formats. For details about how to specify the time, see [Section 3.13.5, "Resource Limits"](#) earlier in this section.

■HH:MM:SS.MS

■HH:MM:SS

■MM:SS.MS

■MM:SS

■SS.MS

■SS

■.MS

Note that the format of .MS (specification of milliseconds) is meaningless in this sub command.

In \$time, you can specify a value between 0 and 4294967295 in seconds. The default is 3600 (seconds).

The JobCenter administrator privileges are required.

SEt Forms \$form-name [\$form-name]...

The form specified in \$form-name is set.

If you execute this command, the previously set form list is deleted and a new list is set. To add a form to the form list, use the ADd Forms sub command.

The JobCenter administrator privileges are required.

SEt Forms = \$form-name \$device

(This function is not supported in R12.7 and later.)

The form specified in \$form-name is set in the device specified in \$device.

The JobCenter operator privileges are required.

SEt FRee_destination Pipe_queue \$queue

The pipe queue specified in \$queue is set to be used as a free destination pipe queue.

When submitting a new request, the user can specify a desired destination queue as an option for submission. The free destination pipe queue reads the option written in a request and transfers the request to its destination queue.

If the request cannot be transferred to its destination or no destination is specified in the request using an option, the destination specified in this free destination pipe queue is used.

The destination option specified in the request becomes invalid when the request is transferred to another queue. Note that the pipe queue to which the request is submitted cannot be specified as the destination.

The JobCenter administrator privileges are required.

SEt GLobal Batch_request_limit \$limit

The upper limit on the number of requests that can be run simultaneously in the batch queues of the entire system is set.

Even when the number of requests running in each individual batch queue is below the number of requests that can be run simultaneously, no more batch request is run if the number of batch requests running in the entire system reaches the value set in this sub command.

If you set a value in \$limit that is larger than the total number of requests running simultaneously in all the batch queues in the system, the setting of this sub command becomes meaningless.

The JobCenter administrator privileges are required.

■For 13.1 and earlier

In \$limit, you can specify a value between 1 and 1000 in the UNIX version or 1 and 100 in the Windows version. The default is 100.

■For 13.2 and later

In \$limit, you can specify a value between 1 and 1000. The default is 1000.

SEt GLobal GROUP_Run_limit \$run_limit

The upper limit on the number of requests that can be run simultaneously by a group in the system is set.

This sub command is supported only in the UNIX version.

No more requests than the number set in this sub command can be run by a group. If an attempt is made to start a request beyond the set number, that request is queued.

In \$run_limit, you can specify a value between 0 and 32767. The default is 0 (unlimited).

The JobCenter operator privileges are required.

SEt GLobal GROUP_Submit_limit \$limit

The upper limit on the number of requests that can be submitted by a group in the system is set.

This sub command is supported only in the UNIX version.

The upper limit on the number of requests that can be submitted is the upper limit on the total number of those requests in the running, queued, suspending, waiting, holding, and exiting states and those requests having only a restart file left after obtaining a check point.

No more requests than the number set in this sub command can be submitted by a group. If a request is submitted beyond the specified number, an error occurs and a message is displayed.

In \$limit, you can specify a value between 0 and 32767. The default is 0 (unlimited).

The JobCenter administrator privileges are required.

SEt GLobal Network_request_limit \$limit

The upper limit on the number of network requests that can be run in the entire system is set.

This sub command is supported only in the UNIX version.

Even when the number of requests running in each individual network queue is below the number of requests that can be run simultaneously, no more request is run simultaneously if the total number of requests running in all the network queues reaches the value set in this sub command.

The JobCenter administrator privileges are required.

■For 13.1 and earlier

In \$limit, you can specify a value between 1 and 1000. The default is 50.

■For 13.2 and later

In \$limit, you can specify a value between 1 and 1000. The default is 1000.

If you set a value in \$limit that is larger than the total number of requests running simultaneously in all the network queues, the setting of this sub command becomes meaningless.

SEt GLobal Pipe_request_limit \$limit

The upper limit on the number of requests that can be transferred simultaneously by the pipe queues of the entire system is set.

Even when the number of requests being transferred by each individual pipe queue is below the number of requests that can be transferred simultaneously, no more request is transferred simultaneously if the total number of requests running in all the pipe queues reaches the value set in this sub command.

The JobCenter administrator privileges are required.

■For 13.1 and earlier

In this parameter, you can specify a value between 1 and 1000 in the UNIX version. The default is 100. In the Windows version, you can specify a value between 1 and 50. The default is 50.

■For 13.2 and later

In this parameter, you can specify a value between 1 and 1000. The default is 1000.

If you set a value in \$limit that is larger than the total number of requests running simultaneously in all the pipe queues in the system, the setting of this sub command becomes meaningless.

SEt GLobal USER_Run_limit \$run_limit

The upper limit on the number of requests that can be run simultaneously by a user in the system is set.

This sub command is supported only in the UNIX version.

No more requests than the number set in this sub command can be run by a user. If an attempt is made to start a request beyond the set number, that request is queued.

In \$run_limit, you can specify a value between 0 and 32767. The default is 0 (unlimited).

The JobCenter operator privileges are required.

SEt GLobal USER_Submit_limit \$limit

The upper limit on the number of requests that can be submitted by a user in the system is set.

This sub command is supported only in the UNIX version.

The upper limit on the number of requests that can be submitted is the upper limit on the total number of those requests in the running, queued, suspending, waiting, holding, and exiting states and those requests having only a restart file left after obtaining a check point.

No more requests than the number set in this sub command can be submitted by a user. If a request is submitted beyond the specified number, an error occurs and a message is displayed.

In \$limit, you can specify a value between 0 and 32767. The default is 0 (unlimited).

The JobCenter administrator privileges are required.

SEt GRoup Runlimit = \$limit \$queue

The number of requests that can be run simultaneously for each group of a queue specified in \$queue is set.

As for users who belong to the same group, no more request is run simultaneously if the total number of requests that they are running in the specified queue reaches the value set in this command.

In \$limit, you can specify a value between 0 and 50. The default is 0 (unlimited).

The JobCenter administrator privileges are required.

SEt INTER_queue_scheduling_type \$type

The inter-queue request scheduling type is set to the one specified in \$type. This sub command takes effect only for batch queues.

Here is how you specify \$type.

■ type0

Between queues using the current scheduling type and having the same queue priority, the execution order is determined by the submission order of the requests to be scheduled by those queues.

■ type1

As for queues having the same queue priority, the requests of the individual queues are executed in order, regardless of the submission order. This prevents the requests submitted to only one particular queue from being executed.

The default is type0.

The JobCenter administrator privileges are required.

SEt INTRa_queue_scheduling_type \$type \$queue

The request scheduling type in the queue specified in \$queue is set to the one specified in \$type. This sub command takes effect only for batch queues.

Here is how you specify \$type.

■ type0

Between requests using the current scheduling type and having the same request priority, the execution order is determined by the submission order.

■ type1

As for requests having the same request priority, the requests of the individual users are executed in order, regardless of the submission order.

The number of requests that a user can execute continuously is predetermined. If the user submits more requests than that, he or she waits until other users' requests are executed. This prevents the requests of only one particular user from being executed.

The number of requests that a user can execute continuously is set in the SET CONTinuous_scheduling_number sub command.

The default is type0.

The JobCenter administrator privileges are required.

SEt LIfetime \$life_time

The lifetime of a pipe queue request in the system is set in hours. (The lifetime cannot be set in minutes or seconds.)

Once its lifetime expires, a pipe queue request is deleted and an error is returned to the submission source.

In \$life_time, you can specify a value between 0 and 100000. The default is 0 (unlimited).

The JobCenter administrator privileges are required.

SEt LOAd_balance Batch_queue \$queue [keep_request_limit = \$num] [Deliver_wait = \$time]

The batch queue specified in \$queue is set to be used as a load balancing batch queue.

In \$num, specify the maximum number of requests that this batch queue can store.

In \$time, specify the request arrival wait time applicable when the demand delivery method is used for load balancing.

In \$num, you can specify a value between 0 and 6000 in the UNIX version. In the Windows version, you can specify a value between 0 and 2500. The default is 0.

In \$time, you can specify a value between 0 and 9999. The default is 30 (seconds).

Note that 0 does not mean unlimited in any of these parameters.

The JobCenter administrator privileges are required.

SEt LOAd_balance Pipe_queue \$queue [Reserve_run_limit = \$num Destination_retry_wait = \$time]

The pipe queue specified in \$queue is set to be used as a load balancing pipe queue.

In \$num, specify the number of requests to be stored for simultaneous transfer for demand delivery. This value must be less than the number of requests to be transferred simultaneously (run_limit - 1) that is set in the pipe queue. If this condition is not met, the requests are not transferred.

In \$time, specify the time a request waits in the WAIT state in the pipe queue during a load balancing scheduling wait. If the request remains in the WAIT state and is not transferred to the batch queue before the specified time expires, the transfer is retried.

In \$num, you can specify a value between 0 and the upper limit. By default, the upper limit is 220 in the UNIX version and 190 in the Windows version. If you have set the limit on the number of

requests that can be run simultaneously by using the SEt RUn_limit sub command, the upper limit is equal to that value.

The default value of \$num is 0.

In \$time, you can specify a value between 0 and 32400. The default is 3600 (seconds).

Note that 0 does not mean unlimited in any of these parameters.

The setting of this sub command cannot be used together with some of the other attributes of the pipe queue.

The JobCenter administrator privileges are required.



When setting Reserve_run_limit or Destination_retry_wait, be sure to specify the other parameter as well.

If you specify only one of these parameters, the sub command is set again, assuming that the default value is specified in the other parameter you omit.

SEt LOG_file \$file_name [Size = (\$logfile_size) [NO_Backup]]

The absolute path name of the log file for the JobCenter messages is specified.

In \$logfile_size, set the upper limit of the log file size. If the file size reaches the upper limit, rotation occurs and the current log file is saved in the backup file. If you do not set the upper limit, the size is not limited.

If you specify NO_Backup as well, no backup file is created.

In \$logfile_size, you can specify a value between 1 and 99999999. See [Section 3.13.5, "Resource Limits"](#) earlier in this section. As an internally set value, make sure that the size is within the range of 1024 to 2147483647 bytes.

To disable the setting of the size or backup, execute this command again without Size or NO_Backup.

The JobCenter administrator privileges are required.

SEt MACHine_group = ()

SEt MACHine_group = (\$scheduler_machine[\$machine]...)

The machine group is set or unset. To set the machine group, specify one or more machine name at the end of the command. The machine specified at the beginning of the machine group is considered the scheduler machine for that group. (While there is no limit on the number of machines, it is generally recommended to specify eight machines or so.)

To unset the machine group, execute this command again without specifying any group member.



Note that JobCenter does not operate normally unless at least your own machine is set in the machine group. Therefore, set the machine group immediately with your own machine specified.

The JobCenter administrator privileges are required.

SEt MAIL \$user_id

The sender of the mails transmitted by the JobCenter system is set.

The JobCenter administrator privileges are required.

SEt MANager \$manager [\$manager]...

The account specified in \$manager is set in the JobCenter administrator list.

You can specify \$manager in the following two formats.

■ \$local_account_name:m or \$local_account_name:o

■ [\$local_user_id]:m or [\$local_user_id]:o

For example, when \$ local_account_name is "user1" and \$local_user_id is "1001", specify either "user1:m" or "[1001]:m".

■ When :m is added to an account name or [user_id]

This means to set the privileges of a JobCenter administrator who is permitted to use all the qmgr commands.

■ When :o is added to an account name or [user_id]

This means to set the privileges of a JobCenter operator who is permitted to use some of the commands.

For details about privileges, see [Section 3.13.3, "Privileges"](#) earlier in this section.

In the case of UNIX, the nsumsmgr and root (super user) accounts are registered with the JobCenter administrator privileges in advance. In the case of Windows JobCenter administrators are registered with the JobCenter administrator privileges in advance.

The JobCenter administrator privileges are required.

SEt MAPping_mode \$mode The JobCenter network mapping mode is set. The method to determine the job execution user on the local host for a request transferred from a remote host user differs depending on the mapping mode.

In \$mode, specify a character string (type1, type2, or type3) or a number (0, 1, or 2).

■ 0 or TYPE1

The request is mapped to the user having the same name as the remote host user.

■ 1 or TYPE2

The request is mapped to the user set in nmapmgr, regardless of the user name.

■ 2 or TYPE3

The request is mapped to the user who is set in nmapmgr or whose user name matches.

The JobCenter administrator privileges are required.

SEt MAXimum Copies \$copies

The value of \$copies is set as the maximum number of copies to be printed. The default is 2.

The JobCenter administrator privileges are required.

SEt MAXimum IDc_connection \$number

The maximum number of simultaneous IDC connections for inter-host communication is set.

This sub command is supported only in the UNIX version.

This is the upper limit on the number of connections that the scheduler machine maintains with other machines in order to reduce the overhead resulting from the communication between machines of the machine group.

This IDC communication is also accomplished when the demand delivery load balancing function is used. Note, however, that in this case, if the set value is smaller than the number of hosts across which jobs are distributed, connections are established and aborted frequently, possibly leading to a reduction in the overall job throughput.

In \$number, you can specify a value between 1 and 1024. The default is 32.

The JobCenter administrator privileges are required.

SEt MAXimum Open_retries \$retries

(This function is not supported in R12.7 and later.)

The number specified in \$retries is set as the maximum number of retries that can be made when a device fails to be opened. The default is 2 retries.

The JobCenter administrator privileges are required.

SEt MAXimum Print_size \$size

The number specified in \$size is set as the maximum number of bytes of a file printed by JobCenter.

In \$size, you can specify a value between 1 and 100000000. The default is 1000000 (bytes).

The JobCenter administrator privileges are required.

SEt MAXimum Request_priority \$limit

The request priority values that can be specified in the -p option of the qsub(1) command are limited to 0 to the value specified in \$limit. If a priority beyond the limit is specified, the submission of the request is rejected.

You need to specify a value that is equal to or higher than the default request priority that is set using the SEt DEFault Batch_request Priority sub command.

The lowest priority of \$limit is 0, and the highest is 63. The default is 0 (unlimited).

The qsub(1) command is supported only in the UNIX version.

The JobCenter administrator privileges are required.

SEt NETWORK Client = (\$client)

The name of the server program used to transfer the request execution result is set.

If any server is not specified in the Create Network_queue sub command, the program specified in this sub command is used.

In \$client, specify the absolute path name of the server program and the parameters with the necessary arguments added.

The JobCenter administrator privileges are required.



Be careful because the name of this sub command looks like that of the SET NETWORK_client sub command.

SEt NETWORK_client = (\$client) \$queue

The server program name of the network queue specified in \$queue is changed to the one specified in \$client.

In \$client, specify the absolute path name of the server program of the network queue.

The JobCenter administrator privileges are required.



Be careful because the name of this sub command looks like that of the SET NETWORK Client command.

SEt NIce_limit = \$nice_value \$queue

A nice limit value (\$nice_value) is set for the batch queue specified in \$queue.

Requests that are already registered in the queue are not affected by a newly set value.

This value is compared with the nice value set in a submitted request. If the value is smaller than \$nice_value set in the batch queue (the priority is higher), the registration of that request is rejected.

If the nice value is not set in a request, \$nice_value set in the queue is used as the nice value of that request.

In \$nice_value, specify a positive or negative integer. See [Section 3.13.5, "Resource Limits"](#) earlier in this section.

The JobCenter administrator privileges are required.

SEt NO_Access \$queue

Access restrictions are enabled for each individual user and group of the queue specified in \$queue.

See also the Add User and ADd Group sub commands.

The JobCenter administrator privileges are required.

Set NO_Check \$queue

The pre-check function is disabled for the pipe queue specified in \$queue.

See also the Create Pipe_queue and SEt CHeck sub commands.

The JobCenter administrator privileges are required.

SEt NO_CUStomjob Batch_queue \$queue

The custom queue attribute of the batch queue specified in \$queue is disabled.

See also the SEt CUStomjob Batch_queue sub command.

The administrator privileges are required.

SEt NO_Default Batch_request Queue

The default batch request queue is disabled.

See also the SEt DEFault Batch_request Queue sub command.

The JobCenter administrator privileges are required.

SEt NO_Default PRINT_Request Forms

The default printer request form is disabled.

See also the SEt DEFault PRINT_Request Forms sub command.

The JobCenter administrator privileges are required.

SEt NO_Default Print_request Queue

The default printer request queue is disabled.

See also the SEt DEFault PRINT_Request Queue sub command.

The JobCenter administrator privileges are required.

SEt NO_Free_destination Pipe_queue \$queue

The attribute of the pipe queue specified in \$queue as a free destination pipe queue is disabled.

See also the SEt FRee_destination Pipe_queue sub command.

The JobCenter operator privileges are required.

SEt NO_Load_balance Batch_queue \$queue

The load balancing batch queue specified in \$queue is disabled.

See also the SEt LOAd_balance Batch_queue sub command.

The JobCenter administrator privileges are required.

SEt NO_Load_balance Pipe_queue \$queue

The load balancing pipe queue specified in \$queue is disabled.

See also the SEt LOAd_balance Pipe_queue sub command.

The JobCenter administrator privileges are required.

Set NO_Staywait \$queue

The staywait function is disabled for the pipe queue specified in \$queue.

See also the Create Pipe_queue and SEt STAYwait sub commands.

The JobCenter administrator privileges are required.

Set NO_Transparent Pipe_queue \$queue

The transparent pipe queue specified in \$queue is disabled.

See also the SEt Transparent Pipe_queue sub command.

The JobCenter administrator privileges are required.

SEt Open_wait \$interval

(This function is not supported in R12.7 and later.)

The interval (number of seconds) for which JobCenter waits if it fails to open a device is specified.

In \$interval, you can specify a value between 1 and 1024. The default is 5 (seconds).

The JobCenter administrator privileges are required.

SEt PER_Process Cpu_limit = (\$limit) \$queue

The per-process CPU time limit is set for the batch queue specified in \$queue.

This sub command is valid only when the system supports the setting of the per-process CPU time limit.

This value is compared with the per-process CPU time limit set in a submitted request. If the value is larger than that set in the queue, the registration of that request is rejected. If the per-process CPU time limit is not set for a request, the value set in the queue is used as the limit of that request.

Requests that are already registered in the queue are not affected by a newly set value.

For the syntax of \$limit, see [Section 3.13.5, "Resource Limits"](#) earlier in this section.

The JobCenter administrator privileges are required.

SEt PER_Process Memory_limit = (\$limit) \$queue

The per-process memory size limit is set for the batch queue specified in \$queue.

This sub command is valid only when the system supports the setting of the per-process memory size limit.

This value is compared with the per-process memory size limit set in a submitted request. If the value is larger than that set in the queue, the registration of that request is rejected. If the per-process memory size limit is not set for a request, the value set in the queue is used as the limit of that request.

Requests that are already registered in the queue are not affected by a newly set value.

For the syntax of \$limit, see [Section 3.13.5, "Resource Limits"](#) earlier in this section.

The JobCenter administrator privileges are required.

SEt PER_Process Permfile_limit = (\$limit) \$queue

The per-process permanent file size limit is set for the batch queue specified in \$queue.

This sub command is valid only when the system supports the setting of the per-process permanent file size limit.

This value is compared with the per-process permanent file size limit set in a submitted request. If the value is larger than that set in the queue, the registration of that request is rejected. If the per-process permanent file size limit is not set for a request, the value set in the queue is used as the limit of that request.

Requests that are already registered in the queue are not affected by a newly set value.

For the syntax of \$limit, see [Section 3.13.5, "Resource Limits"](#) earlier in this section.

The JobCenter administrator privileges are required.

SEt PIpe_client = (\$client) \$queue

The pipe queue specified in \$queue is associated with the transfer program specified in \$client.

In \$client, specify the absolute path name of the transfer program and the parameters with the necessary arguments added.

The JobCenter administrator privileges are required.

SEt PRiority = \$priority \$queue

The queue priority is set for the queue specified in \$queue.

The lowest priority of \$priority is 0, and the highest is 63.

The JobCenter administrator privileges are required.

SEt Queue Reboot_mode = \$val \$queue

The queue reboot mode is specified for the queue specified in \$queue.

Specify \$val in one of the following formats.

■RESTART

■STOP

■PURGE

■MIGRATION_STOP

■MIGRATION_PURGE

The JobCenter administrator privileges are required.



Since ERP and BI jobs are submitted without using a queue, they are not affected by the setting of the queue reboot mode (reboot_mode).

If you want to stop the execution of a job by setting the reboot mode to "STOP", place a dummy unit job immediately before each ERP and BI job.

SEt RUn_limit = \$run_limit \$queue

The limit on the number of requests that can be run simultaneously in the batch or pipe queue specified in \$queue is changed to the number specified in \$run_limit.

In \$run_limit, you can specify a value between 1 and the upper limit. The upper limit is the total value of GLocal Batch_request_limit, GLocal Pipe_request_limit, and GLocal Network_request_limit.

By default, the upper limit is 200 in the UNIX version or 190 in the Windows version.

The JobCenter operator privileges are required.

SEt SHell_strategy Fixed = (\$shell)

The shell program selection strategy for batch request execution is fixed to the shell program specified in \$shell.

In the case of UNIX, specify the absolute path to a shell program that can be used in the OS, such as "/bin/sh" or "/bin/csh". (Note that normal operation is not guaranteed if a user program is specified.)

This specification is the default for Windows, which does not support execution program specification other than the specification of the absolute path to cmd.exe.

The JobCenter administrator privileges are required.

SEt SHell_strategy FRee

The shell program selection strategy for batch request execution is set to FREE.

When a batch request is executed, the login shell of the request user is started first. Next, based on the content of the batch request, the login shell selects an appropriate shell, which in turn executes the batch request.

Thus, the batch request is executed as in interactive processing.

For UNIX, this specification is the default. This specification is not interpreted by Windows. The program is fixed to cmd.exe, and the operation does not change.

The JobCenter administrator privileges are required.

SEt SHell_strategy Login

In the case of UNIX, the shell that executes batch requests is set to the login shell of the user. As the login shell, the shell specified in the /etc/passwd file (see passwd(4)) is selected.

This specification is not interpreted by Windows. The program is fixed to cmd.exe, and the operation does not change.

The JobCenter administrator privileges are required.

SEt STACK_limit = (\$limit) \$queue

The stack segment size limit is set for the batch queue specified in \$queue.

This sub command is valid only when the system supports it.

This value is compared with the stack segment size limit set in a submitted request. If the value is larger than that set in the queue, the registration of that request is rejected. If the stack segment size limit is not set for a request, the value set in the queue is used as the stack segment size limit of that request.

Requests that are already registered in the queue are not affected by a newly set value.

For the syntax of \$limit, see [Section 3.13.5, "Resource Limits"](#) earlier in this section.

The JobCenter administrator privileges are required.

SEt STAYwait \$queue

The staywait function is set for the pipe queue specified in \$queue.

See also the Create Pipe_queue and SEt NO_STAYwait sub commands.

The JobCenter administrator privileges are required.

SEt Transparent Pipe_queue \$queue

The pipe queue specified in \$queue is set to be used as a transparent pipe queue.

The setting of this sub command cannot be used together with some of the other attributes of the pipe queue such as the staywait function mentioned above.

See also the Set NO_Transparent Pipe_queue sub command.

The JobCenter administrator privileges are required.

SEt UNrestricted_access \$queue

The access restrictions of the queue specified in \$queue are disabled, making the queue available to all users.

See also the SEt NO_Access sub command.

The JobCenter administrator privileges are required.

SEt USer Runlimit = \$limit \$queue

The number of requests that can be run simultaneously by a user of the queue specified in \$queue is set.

The number of requests that a user can run simultaneously in the specified queue is limited to the value specified in \$limit. If the total number of requests that are running in the specified queue reaches the value set in this command, no more request of the user is run simultaneously.

In \$limit, you can specify a value between 0 and 50. The default is 0 (unlimited).

The JobCenter administrator privileges are required.

SHOw All

A summary of information about devices, forms, limits, administrators, parameters, and queues is shown.

For details, see the commands described below.

SHOw Complex [\$complex-name]

The status of all queue complexes is shown.

If you specify \$complex-name, only the status of that particular queue complex is shown.

SHOw Device [\$device]

(This function is not supported in R12.7 and later.)

The status of all JobCenter devices is shown.

If you specify \$device, only the information about that particular device is shown.

SHOw Forms

The list of valid forms is shown.

SHOw Limits_supported

The list of JobCenter resource limits supported by the system is shown.

SHOw LOng Complex [\$complex-name]

The status of all queue complexes is shown in an extended format.

If you specify \$complex-name, only the status of that particular queue complex is shown.

SHOw LOng Queue [\$queue [\$user]]

The status of all JobCenter queues is shown in an extended format.

If you specify \$queue, only the status of that particular queue is shown.

If you specify \$user, only the information about the requests belonging to that particular user is shown.



In R13.2 and later, the request information of queues cannot be shown when JobCenter is stopped.

SHOw MACHine_group

The set machine group is shown.

SHOw MANagers

The list of JobCenter administrators assigned the administrator or operator privileges is shown.

SHOw Parameters

The list of JobCenter parameters related to the entire system is shown.

SHOw Queue [\$queue [\$user]]

The status of all queues is shown.

If you specify \$queue, only the status of that particular queue is shown.

If you specify \$user, only the status of those requests in the queue that have been submitted by that particular user is shown.



In R13.2 and later, the request information of queues cannot be shown when JobCenter is stopped.

SHUtdown [\$seconds]

JobCenter is shut down.

First, the SIGTERM signal is sent to each process of the currently running requests. After the lapse of the actual time specified in \$seconds, the SIGKILL signal is sent to the remaining processes of each request.

In \$seconds, you can specify a value between 0 and 600. The default is 60 (seconds).

The JobCenter operator privileges are required.



The official command to shut down JobCenter is nqsstop, not this sub command. This sub command stops the resident NQS process and some of the resident processes that depend on it. Do not use this sub command except in an emergency such as when the entire system is malfunctioning.

STArt Queue \$queue

The queue specified in \$queue is started.

If you start the queue specified in \$queue by using this sub command, the request processing is resumed when that queue is enabled.

If the specified queue has already been started, nothing happens.

The JobCenter operator privileges are required.

STArt All Queue

All queues in the system are started.

If you start all queues by using this sub command, the request processing is resumed for enabled queues.

For queues that have already been started, nothing happens.

The JobCenter operator privileges are required.

STOp Queue \$queue

The queue specified in \$queue is stopped.

If you stop the queue specified in \$queue by using this sub command, the processing is suppressed without starting the execution of the requests in that queue. The processing of the requests already running on the queue continues.

Unlike with the Disable Queue sub command, you can submit new requests to the stopped queue. However, these requests are only queued, and the processing is suppressed without starting the execution.

The JobCenter operator privileges are required.

STOp All Queue

All queues in the system are stopped.

The processing is suppressed without starting the execution of the requests in all queues. The processing of the already running requests continues.

Unlike with the Disable All Queue sub command, you can submit new requests to the stopped queues. However, these requests are only queued, and the processing is suppressed without starting the execution.

The JobCenter operator privileges are required.

SUSpend Request \$request_id

The execution of the request specified in \$request_id is suspended.

The request you specify in this sub command must be in the RUNNING state. If you suspend a request by using this command, you cannot resume its execution by using the qrsm user command of JobCenter. To resume the execution, use the RESUme Request sub command.

The qrsm command is supported only in the UNIX version.

The JobCenter operator privileges are required.

Unlock Local_daemon

The JobCenter NQS daemon process locked in memory is unlocked.

See the UNLOCK option of plock(2).

The JobCenter operator privileges are required.

3.14. jdj_download (Downloading Definition Information)

■ UNIX version

```
/usr/lib/nqs/gui/bin/jdj_download [def] [-h $hostname[:$port]] [-e [$execuser]] [-u $user | -a] [-p $password] [-t $target] [-g[j]$jnwgroup | -j $jnw] [-gs $schgroup | -s $sch] [-gd $customgroup | -d $customname] [-c $wcalname] [-l $triggername] [-r $rel_target] [-o $filename | -o $dirname] [-w $second] [-f]
/usr/lib/nqs/gui/bin/jdj_download conf [-h $hostname[:$port]] [-u $user] [-p $password] [-o $filename] [-w $second]
```

■ Windows version

```
%InstallDirectory%\bin\jdj_download [def] [-h $hostname[:$port]] [-e [$execuser]] [-u $user | -a] [-p $password] [-t $target] [-g[j]$jnwgroup | -j $jnw] [-gs $schgroup | -s $sch] [-gd $customgroup | -d $customname] [-c $wcalname] [-l $triggername] [-r $rel_target] [-o $filename | -o $dirname] [-w $second] [-f]
%InstallDirectory%\bin\jdj_download conf [-h $hostname[:$port]] [-u $user] [-p $password] [-o $filename] [-w $second]
```

3.14.1. Function Description

The `jdj_download` command downloads definition information or configuration information from JobCenter MG/SV and creates a JPF file. From the command line, you can execute the functions equivalent to those used for the JobCenter CL/Win download operation. You can specify a sub command after the command name. If you do not specify anything in the sub command or specify "def", the command downloads definition information. If you specify "conf" in the sub command, the command downloads configuration information.

3.14.2. Options

`-h $hostname[:$port]`

In `$hostname`, specify the host (site) name or IP address of the JobCenter MG/SV from which the information is to be downloaded.

In `$port`, you can specify the connected port number of the JobCenter MG/SV (TCP port number of `jccombase`) from which the information is downloaded.



■ Note on port number specification

Specify a port number only when you execute this command in an environment that meets both of the following conditions.

- The `jdj_download` command provided with Definition Helper is used when neither JobCenter MG/SV nor JobCenter CL/Win is installed.
- The connected port number of the JobCenter MG/SV (TCP port number of `jccombase`) from which the information is to be downloaded has been changed to other than the default value (611).



You can also specify an IP address in `$hostname`. Here is how you specify an IP address.

■ When not specifying a port number

Specify an IP address in the format of `<IP-address>`.

■ When specifying a port number

In the case of an IPv4 address, specify it in the format of <IPv4-address:port-number>.

In the case of an IPv6 address, specify it in the format of <[IPv6-address]:port-number>.

If you do not specify this option, information is downloaded from the command execution machine.

-u \$user | -a

Specify the user whose information is to be downloaded.

■ -u \$user

The command is targeted at the user specified in \$user.

■ -a

The command is targeted at all users.



If you specify -a, the command behaves in the same way as when jscf is specified in the -r option.

If you do not specify this option, the command logs in as the command execution user.

-e \$execuser

Specify the login destination user.

If you do not specify this option, the command behaves as follows.

■ When a user is specified in -u

The command logs in as the user specified in -u.

■ Other than the above

The command logs in using the name of the user who is executing it.

-p \$password

Specify the password of the login user.

If you do not specify this option, the password prompt appears.

-t \$target

Select the type of definition information to be downloaded. To select the type, specify one of the following characters.

j: Job network

s: Schedule

c: Calendar

d: Custom job definition

l: Trigger

If you do not specify this option, "j" is assumed.

-g[j] \$jnwgroup | -j \$jnw

Specify the job network group or job network to be downloaded.

■ **-g[j] \$jnwgroup**

The command is targeted at the group specified in \$jnwgroup.

To specify the hierarchy of the group, use "." as the delimiter. To specify a group path, start with the root level (example: .daily_job.job_bat01).

Both -g and -gj cause the command to behave the same way. -g is provided for backward compatibility.

■ **-j \$jnw**

The command is targeted at the job network specified in \$jnw.

If you do not specify this option when j is specified in the -t option, the definition information of all job networks is downloaded.



You cannot specify this option together with the -a option.

-r \$rel_target

Specify whether to download the related sub job network, calendar branch reference schedule, calendar, and monitoring target text log as well.

j: The related sub job network is downloaded as well.

s: The related calendar branch reference schedule is downloaded as well.

c: The calendar selected in the related calendar branch reference schedule is downloaded as well.

f: All monitoring target text log data is downloaded as well.

m: The related monitor target text log data is downloaded as well.

-gs \$schgroup | -s \$sch

Specify the schedule group or schedule to be downloaded.

■ **-gs \$schgroup**

The command is targeted at the group specified in \$schgroup.

To specify the hierarchy of the group, use "." as the delimiter. To specify a group path, start with the root level (example: .daily_sch.sch01).

■ **-s \$sch**

The command is targeted at the schedule specified in \$sch.

If you do not specify this option when s is specified in the -t option, the definition information of all schedules is downloaded.



You cannot specify this option together with the -a option.

Be sure to specify 's' in the -t option when specifying this option.

-l \$triggername

The trigger specified in \$triggername is downloaded.

If you do not specify this option when l is specified in the -t option, the definition information of all launch triggers is downloaded.



You cannot specify this option together with the -a option.

Be sure to specify 'l' in the -t option when specifying this option.

-gd \$customgroup | -d \$customname

Specify the custom job template group or custom job template to be downloaded.

■ -gd \$customgroup

The command is targeted at the group specified in \$customgroup.

To specify the hierarchy of the group, use "." as the delimiter. To specify a group path, start with the root level (example: .User.NewGroup1).

■ -d \$customname

The command is targeted at the custom job template specified in \$customname.

If you do not specify this option when d is specified in the -t option, the definition information of all custom job templates is downloaded.



You cannot specify this option together with the -a option.

Be sure to specify 'd' in the -t option when specifying this option.

-c \$wcalname

The calendar specified in \$wcalname is downloaded.

If you do not specify this option when c is specified in the -t option, the definition information of all calendars is downloaded.



Be sure to specify 'c' in the -t option when specifying this option.

-o \$filename | -o \$dirname

Specify the name of the JPF file to which to output the downloaded definition information. If you have specified the download of information of all users, specify a directory name. The JPF file of each user is created under that directory.

If you do not specify this option, the information is output to a file named "jc_def_<execution-date-and-time-expressed-in-YYYYMMDDhhmmss-format>.jpf". If you have downloaded the information of all users, a directory named "jc_def_<execution-date-and-time-expressed-in-YYYYMMDDhhmmss-format>" is created.



If you have downloaded the information of all users, you cannot specify a directory name that contains .jpf. If you specify a directory name that contains .jpf, a directory is created with the .jpf part removed from its name.

-w \$second

Specify the timeout time (seconds) for Comagnet communication. As the timeout value (seconds), enter an integer between 30 and 86400.

If you do not specify this option, the timeout value is assumed to be 600 seconds (10 minutes) by default.

-f

The command downloads the target definition even if it is being edited on CL/Win.

If you do not specify this option, an attempt to download the target definition fails if it is being edited on CL/Win.

3.14.3. Return Values

Return value	Description
0	Normal end
1	Abnormal end

3.14.4. Main Messages

Message	Description
Invalid host format error.	The format of the connection destination host name or port number is invalid. Check that it is specified in the format of <site-name>:<port-number>.
jccombase port number should be from 1 to 65535.	Specify the connection destination port number in the range of 1 to 65535.
invalid timeout. [30 - 86400]	Specify the timeout time (seconds) for Comagnet communication in the range of 30 to 86400.
Authentication failure. Check username or password.	The user name or password is invalid. Check whether the login user information is valid.
connection failure. hostname=(\$hostname) port=(\$port)	The specified host cannot be connected. Check the host to connect to.
Invalid first group path. (". " required)	To specify groups, append "." at the beginning as the root level.
no such a group path.	The specified group cannot be found.

3.14.5. Notes

- When specifying a host (site) name, make sure that the network environment of the OS is configured so that the name can be resolved.
- Depending on the amount of definition information to be downloaded, an error may occur due to a memory shortage because the command attempts to secure more memory space than available on the JobCenter MG/SV from which the information is to be downloaded. See details about how to estimate

the amount of memory required for downloads in <Environment Guide>Section 19.7, "Estimating the Memory Amount Used for Uploads and Downloads".

3.15. jd_h_upload (Uploading Definition Information)

■ UNIX version

```
/usr/lib/nqs/gui/bin/jd_h_upload [-h $hostname[:$port]] [-u $user] [-p $password] [-c] [-r $rulefile]
[-f] [-i] [-w $second] {$jpf_file|-a $jpf_dir}
```

■ Windows version

```
%InstallDirectory%\bin\jd_h_upload [-h $hostname[:$port]] [-u $user] [-p $password] [-c] [-r
$rulefile] [-f] [-i] [-w $second] {$jpf_file|-a $jpf_dir}
```

3.15.1. Function Description

The `jd_h_upload` command uploads definition information to JobCenter MG/SV. From the command line, you can execute the functions equivalent to those used for the JobCenter CL/Win upload operation.

3.15.2. Options

`-h $hostname[:$port]`

In `$hostname`, specify the host (site) name or IP address of the JobCenter MG/SV for the upload destination.

In `$port`, you can specify the connected port number of the JobCenter MG/SV (TCP port number of `jccombase`) for the upload destination .



■ Note on port number specification

Specify a port number only when you execute this command in an environment that meets both of the following conditions.

- The `jd_h_upload` command provided with Definition Helper is used when neither JobCenter MG/SV nor JobCenter CL/Win is installed.
- The connected port number of the JobCenter MG/SV (TCP port number of `jccombase`) for the upload destination has been changed to other than the default value (611).



You can also specify an IP address in `$hostname`. Here is how you specify an IP address.

■ When not specifying a port number

Specify an IP address in the format of `<IP-address>`.

■ When specifying a port number

In the case of an IPv4 address, specify it in the format of `<IPv4-address:port-number>`.

In the case of an IPv6 address, specify it in the format of `<[IPv6-address]:port-number>`.

If you do not specify this option, information is uploaded to the command execution machine.

`-u $user`

Specify the login destination user of the JobCenter MG/SV to be connected.

If you do not specify this option, the command logs in as the command execution user.

`-p $password`

Specify the password of the login user.

If you do not specify this option, the password prompt appears.

`-w $second`

Specify the timeout time (seconds) for Comagnet communication. As the timeout value (seconds), enter an integer between 30 and 86400.

If you do not specify this option, the timeout value is assumed to be 600 seconds (10 minutes) by default.

`-c`

The command runs in check mode. In check mode, the command only performs a dependency check on the definition information and does not update the definition information in the server.

`-r $rulefile`

Conversion is performed according to the host name conversion format written in the rule file.

`-f`

The queue information check is omitted.

`-i`

The extended icon data is included in the upload target. To upload an extended custom job template, set this option.

`$jpf_file`

Specify the definition information (JPF file) to be uploaded. Be sure to specify either this parameter or the directory parameter of the `-a` option.

`-a $jpf_dir`

All the JPF files in the specified directory (`$jpf_dir`) are uploaded. Specify the directory created when you downloaded the information of all users.

3.15.2.1. Rule File

Write host name conversion definitions as follows.

```
conversion-source-host-name-1 conversion-destination-host-name-1
conversion-source-host-name-2 conversion-destination-host-name-2
.
.
```

3.15.3. Return Values

Return value	Description
--------------	-------------

0	Normal end
1	Abnormal end

3.15.4. Main Messages

Message	Description
Invalid host format error.	The format of the connection destination host name or port number is invalid. Check that it is specified in the format of <site-name>:<port-number>.
jccombase port number should be from 1 to 65535.	Specify the connection destination port number in the range of 1 to 65535.
invalid timeout. [30 - 86400]	Specify the timeout time (seconds) for Comagnet communication in the range of 30 to 86400.
Authentication failure. Check username or password.	The user name or password is invalid. Check whether the login user information is valid.
connection failure. hostname=(\$hostname) port=(\$port)	The specified host cannot be connected. Check the host to connect to.
cannot access "filename": No such file	The specified JPF file cannot be found.
Dependency Error	The information cannot be uploaded because of an invalid parameter dependency. For example, a dependency error occurs if a calendar associated with a schedule cannot be found in the definition information to be uploaded or the definition information in JobCenter MG/SV. See details in <Basic Guide>Section 12.4, "Dependency Errors".

3.15.5. Notes

- When specifying a host (site) name, make sure that the network environment of the OS is configured so that the name can be resolved.
- Depending on the amount of definition information to be uploaded, an error may occur due to a memory shortage because the command attempts to secure more memory space than available on the JobCenter MG/SV to which the information is to be uploaded. See details about how to estimate the amount of memory required for uploads in <Environment Guide>Section 19.7, "Estimating the Memory Amount Used for Uploads and Downloads".
- A communication timeout may occur when a huge JPF file is uploaded. Note that, even if a timeout occurs, the update of the definition continues on the server side. You cannot edit the definition during the update process.

3.16. jdh_trkget (Getting Tracker Information)

■ UNIX version

```
/usr/lib/nqs/gui/bin/jdh_trkget [-u $user] [-t arc] [-o $output] $YYYYMMDD-YYYYMMDD

/usr/lib/nqs/gui/bin/jdh_trkget [-u $user] -t act [-n $tracker-name] [-i $tracker-id] [-o $output]
[$YYYYMMDD-YYYYMMDD]

/usr/lib/nqs/gui/bin/jdh_trkget [-u $user] -t all [-n $tracker-name] [-i $tracker-id] [-o $output]
$YYYYMMDD-YYYYMMDD
```

■ Windows version

```
%InstallDirectory%\bin\jdh_trkget [-u $user] [-t arc] [-o $output] $YYYYMMDD-YYYYMMDD

%InstallDirectory%\bin\jdh_trkget [-u $user] -t act [-n $tracker-name] [-i $tracker-id] [-o $output]
[$YYYYMMDD-YYYYMMDD]

%InstallDirectory%\bin\jdh_trkget [-u $user] -t all [-n $tracker-name] [-i $tracker-id] [-o $output]
$YYYYMMDD-YYYYMMDD
```

3.16.1. Function Description

The `jdh_trkget` command collects the tracker information that matches the specified options and creates a JPF file.

Tracker information refers to unarchived tracker information and tracker archive files.

Execute this command as the root user (a user with the Administrator privileges in the case of Windows).

3.16.2. Options

`-u $user`

Specify the user whose tracker information you want to get.

If you do not specify this option, the command gets the tracker information of all users.

`-t $target`

Specify the target tracker information. You can specify the following character strings.

■ arc

The command gets only tracker archive files.

■ act

The command gets only unarchived tracker information.

■ all

The command gets all tracker information (unarchived tracker information and tracker archive files).

If you do not specify this option, the command gets only tracker archive files.

`-n $tracker-name`

Specify the target tracker name.

If you do not specify this option, the target information is not limited based on a tracker name.



This option is valid only when the command gets only unarchived tracker information. You cannot specify this option when -t arc is specified.

-i \$tracker-id

Specify the target tracker ID. The specified tracker ID is not affected by the target period.

If you do not specify this option, the command does not get tracker information based on a tracker ID.

(Example) NewJnw.20121221095106



This option is valid only when the command gets only unarchived tracker information. You cannot specify this option when -t arc is specified.

-o \$output

Specify the output file name. If you omit this option, the file is output with the default name (trkget_YYYYMMDDhhmmss.jpj).

\$YYYYMMDD-YYYYMMDD

Specify the period of the tracker archive files you want to get. To specify the period, enter the start date and end date, each as an eight-digit number (YYYYMMDD[local-time]) and hyphenated.

(Example) 20110702-20110912

The command gets a tracker archive file even when its period only partially overlaps the specified period.

You can omit this option only when -t act is specified.

3.16.3. Return Values

Return value	Description
0	Normal end
1	Abnormal end

3.16.4. Main Messages

Message	Description
get trkdata(archive) user(user-count).	This is the information that the command displays before getting the tracker information. It indicates that the command gets only tracker archive files, and the number of target users is shown in user-count.
get trkdata(active) user(user-count) tracker(tracker-count).	This is the information that the command displays before getting the tracker information. It indicates that the command gets only unarchived tracker information, with the number of target users shown in user-count and the number of target unarchived trackers shown in tracker-count.
get trkdata(all) user(user-count) tracker[active] (tracker-count).	This is the information that the command displays before getting the tracker information.

Message	Description
	It indicates that the command gets all tracker information, with the number of target users shown in user-count and the number of target unarchived trackers shown in tracker-count.
Only root user can execute this command.	Execute this command as the root user (a user with the Administrator privileges in the case of Windows).
Too few arguments.	The number of command arguments is insufficient. Specify the argument that defines the period for getting tracker archive files (YYYYMMDD-YYYYMMDD).
no trkdata in YYYYMMDD-YYYYMMDD	No tracker archive file is found in the specified period (YYYYMMDD-YYYYMMDD).
Invalid time format(YYYYMMDD-YYYYMMDD)	The specified time format is invalid. Use the YYYYMMDD-YYYYMMDD format.
Invalid time format(start>end)	The specified end date of the target period is earlier than the start date. Make sure that the end date is later than the start date.
Invalid target(target)	The specified target is invalid. Specify a target that can be specified in -t.
No such a user (user=username)	The specified user cannot be found. Check the specified user name and whether the specified user can log in to CL/Win.
failed to get tracker[active] user=(user-count) tracker(tracker-count).	<p>This is the information that the command displays after getting the target information.</p> <p>The command failed to get some of the unarchived tracker information. The number of users whose information was not obtained is shown in user-count, and the number of unarchived trackers that was not obtained is shown in tracker-count.</p> <p>Check the collected information by using the jpf_stat command. If the necessary tracker information cannot be found, execute this command again.</p>
failed to get tracker[archive] user=(user-count) tracker(tracker-count).	<p>This is the information that the command displays after getting the target information.</p> <p>The command failed to get some of the tracker archive files. The number of users whose information was not obtained is shown in user-count, and the number of tracker archives that was not obtained is shown in tracker-count.</p> <p>Check the collected information by using the jpf_stat command. If the necessary tracker information cannot be found, execute this command again.</p>

3.16.5. Notes

- When you execute the command for a cluster site, you need to set the environment variables NQS_SITE and NQS_SITEDB in advance.

For details, see the cautions given in [Chapter 1, 「Command List」](#) .

- About environment variable LANG

In the case of the UNIX version, make sure that environment variable LANG used when the command is executed matches the setup language of JobCenter.

In the case of the Windows version, there is no particular need to pay attention because the setup language specified during installation is used instead of environment variable LANG.

3.17. jpf_stat (Checking JPF File Information)

■UNIX version

```
/usr/lib/nqs/gui/bin/jpf_stat [-v | -h] $jpf_file
/usr/lib/nqs/gui/bin/jpf_stat -h -a $jpf_folder
```

■Windows version

```
%InstallDirectory%\bin\jpf_stat [-v | -h] $jpf_file
%InstallDirectory%\bin\jpf_stat -h -a $jpf_folder
```

3.17.1. Function Description

The `jpf_stat` command displays JPF file information. The output items differ depending on the type of JPF file.

3.17.2. Options

`-v`

Detailed information is displayed.

`-h`

The host names set in the parameters for the definitions in the JPF file to be converted are listed below.

`-a`

Specify this option when the JPF files under `$jpf_folder` are the target of this command.



Be sure to specify the `-h` option when specifying this option.

`$jpf_file`

Specify the JPF file.

This parameter must be specified when any option other than the `-a` option is specified.

`$jpf_folder`

Specify the directory where the JPF file is saved.

This parameter must be specified when the `-a` option is specified.

3.17.3. Return Values

Return value	Description
0	Normal end
1	Abnormal end

3.17.4. Execution Result Items (Common)

Message	Description
Format type	Version of the JPF file
Created JobCenter version	Version of JobCenter MG/SV used to create the JPF file
Creation time	Creation date of the JPF file
Created platform	Type of OS used to create the JPF file

Example of the command execution result (in the case when showing the details of a JPF file of JobCenter definition information)



Part of the displayed information is modified. Text enclosed in `/* */` is a comment about the content. `...` indicates that the rest of the information is omitted.

```

./jpf_stat.exe -v jc_def.jpj
=== JPF file status ===
Format type: Definition 7.0
Created JobCenter version: 15.1.0
Creation time: 2017:05:08 14:32:28
Created platform: Windows

=== Each type of information ===

*** JobNetworks(3) *** /* "3" indicates the number of definitions in the JPG file */
Daily task1 (Update) /* The definition names and the operation selected at the time of the upload
*/
Monthly task1 (Update)
Monthly task2 (Update)
*** Schedules(3) ***
Daily schedule (Update)
Monthly schedule (Update)
Special operation (Update)
*** Triggers(TRG)(2) ***
Error log monitoring on application A (Update)
Warning log monitoring on application B (Update)
*** Triggers(TARGET_TEXTLOG)(3) ***
Log of application A (Update)
Log of application B (Update)
Log of application C (Update)
*** Calendars(1) ***
Corporate calendar (Update)
*** CustomJob Templates(4) ***
ProcessB (Update)
ProcessC (Update)
ProcessE (Update)
EndProcess (Update)
    
```

Example of the command execution result (when showing a list of host names to be converted in the JPF file of the JobCenter definition information)

```

./jpf_stat.exe -h jc_def.jpj
=== JPF file hostlist ===
    
```

```
hostname1,hostname2,hostname3
```

Example of the command execution result (in the case of a JPF file of JobCenter configuration information)

```
./jpf_stat -v jc_conf.jpf

=== JPF file status ===
Format type: Configuration 3.0
Created JobCenter version: 13.1
Creation time: 2012:07:03 12:29:57
Created platform: Linux

=== Each type of information ===

** Configuration Status **

Backup mode: Local

*** Host infomation *** /* Information of a host*/
hostname: aaa /* Host name */
mid: 123 /* Machine name */

*** Queues *** /* Statuses of queues */
guilb_def (type=pipe, enable=true, running=true) /* Queue name and its status */
guinw (type=pipe, enable=true, running=true)
guitp_1 (type=pipe, enable=true, running=true)
guibs_1 (type=batch, enable=true, running=true)

*** Machine List *** /* List of machines */
aaa
bbb

*** Maching groups *** /* Machine groups */
aaa

*** Spool configuration files *** /* Configuration files on the local or cluster site */
jobmsg.conf
umsconf.f
trkcolor.f
...

*** Configuration backup users *** /* The name of the user that makes a backup of the configuration */
nsumsmgr

*** Machine backup files *** /* Setting files of the environment where JobCenter MG/SV is installed */
comagent.sh
destconf.f
jnwcaster.sh
...

*** Backup OS infomation *** /* OS information that is backed up */
xxxxxx
```

```

*** OS Account Infomation *** /* Information of users obtained when the backup was made */
root (uid=0)
bin (uid=1)
...

```

Example of the command execution result (in the case of a JPF file of JobCenter execution result information (trackers))

```

./jpf_stat trkget.jpf
=== JPF file status ===
Format type: Tracker 3.0
Created JobCenter version: 13.1.2
Creation time: 2012:12:21 19:46:28
Created platform: Linux

=== Each type of information ===

** Tracker Status **

===== User (npxjc) =====
-- Tracker [archive] --
  trkarc20121019-20121019
  trkarc20121113-20121113
  trkarc20121119-20121119
  trkarc20121219-20121219
  trkarc20121220-20121220
  trkarc20121221-20121221
-- Tracker [active] --
count : 3

===== User (nsumsmgr) =====
-- Tracker [archive] --
  trkarc20121221-20121221
-- Tracker [active] --
count : 1

===== User (root) =====
-- Tracker [archive] --
  trkarc20121221-20121221
-- Tracker [active] --
count : 1

```

3.18. jc_backup (Backing Up Configuration Information)

■ UNIX version

```
/usr/lib/nqs/gui/bin/jc_backup conf [-c $clusterdb] [-o $output]
```

■ Windows version

```
%InstallDirectory%\bin\jc_backup conf [-c $clusterdb] [-o $output]
```

3.18.1. Function Description

The `jc_backup` command backs up JobCenter configuration information. Execute this command as the root user in the UNIX version or as the JobCenter administrator user in the Windows version. The command can be used regardless of whether JobCenter is running or stopped. See details in <Environment Guide>Chapter 16, 「Backing Up and Restoring JobCenter Environment」 .

3.18.2. Options

`-c $clusterdb`

When backing up cluster configuration information, specify the cluster database path of JobCenter. If you do not specify this option, the command backs up local configuration information.

`-o $output`

Specify the output file name. If you do not specify this option, the command outputs information using one of the following file names.

Local configuration information: `jc_conf_local_YYYYMMDDhhmmss.jpj`

Cluster configuration information: `jc_conf_cluster_YYYYMMDDhhmmss.jpj`

3.18.3. Return Values

Return value	Description
0	Normal end
1	Abnormal end

3.18.4. Notes

- The backup and restoration functions assume that the configuration information that has been backed up is restored to the original server (or a host that is physically different but has the same name on the same platform). These functions do not support such operations as distributing configuration information to another machine as a template to apply the information after partially modifying it.

3.19. jc_restore (Restoring Configuration Information)

■UNIX version

```
/usr/lib/nqs/gui/bin/jc_restore conf [-c $clusterdb] $jpf_file
```

■Windows version

```
%InstallDirectory%\bin\jc_restore conf [-c $clusterdb] $jpf_file
```

3.19.1. Function Description

The `jc_restore` command restores configuration information. Execute this command as the root user in the UNIX version or as the JobCenter administrator user in the Windows version. This command can be used when JobCenter is stopped. See details in <Environment Guide>Chapter 16, 「Backing Up and Restoring JobCenter Environment」 .

3.19.2. Options

```
-c $clusterdb
```

When restoring cluster configuration information, specify the cluster database path of JobCenter. If you do not specify this option, the command restores local configuration information.

```
jpf_file
```

Specify the backup file of configuration information (the JPF file created by the `jc_backup` command). This parameter is required.

3.19.3. Return Values

Return value	Description
0	Normal end
1	Abnormal end

3.19.4. Notes

- The backup and restoration functions assume that the configuration information that has been backed up is restored to the original server (or a host that is physically different but has the same name on the same platform). These functions do not support such operations as distributing configuration information to another machine as a template to apply the information after partially modifying it.

3.20. jdh_conv (Performing Mutual Conversion between JDF and JPF Files)

■ UNIX version

```
/usr/lib/nqs/gui/bin/jdh_conv [-w $version] [-c] [-o $output] [$JDFFile ...]
/usr/lib/nqs/gui/bin/jdh_conv -x [-o $output] [$JPFFile]
```



■ Windows version


```
%InstallDirectory%\bin\jdh_conv [-w $version] [-c] [-o $output] [$JDFFile ...]
%InstallDirectory%\bin\jdh_conv -x [-o $output] [$JPFFile]
```

3.20.1. Function Description

The `jdh_conv` command creates a JPF file of definition information from a JDF file. If you specify the `-x` option, the command converts a JPF file to the JDF format.

3.20.2. Options

Option	Description
-w \$version	<p>Specify the version of the JPF file to be created in the format of "x.x". Currently, you can specify "1.0", "2.0", "3.0", "4.0", "5.0", "6.0", and "7.0". If you omit this option, <code>jdh_conv</code> creates the file of the latest version it supports.</p> <div style="border: 1px solid red; padding: 5px;">  <ul style="list-style-type: none"> ■ Compatibility has not been checked for Versions 12.8.2 to 12.10. ■ If you specify 7.0 or any earlier version, the created JPF file does not include the trigger definition and monitoring target text log. </div>
-c	<p>The command only checks the JDF file without creating a JPF file.</p> <div style="border: 1px solid red; padding: 5px;">  <p>In Versions 12.8.2 to 12.10, this check is partial.</p> </div>
-o \$output	<p>Specify the output destination.</p> <p>If you do not specify this option, the information is output to a file named "jc_def_<execution-date-and-time-expressed-in-YYYYMMDDhhmmss-format>.jpf". If you specify the <code>-x</code> option, the output destination is "jpf_YYYYMMDDHHMMSS" by default.</p>
\$JDFFile ...	<p>Specify the JDF file you want to convert to the JPF format. There is no need to specify all the types (customjob_template, jnw, sched, trg, and wkcal). You can specify only the types of definitions that you created. It is also possible to specify only the operation file (operation.jdf).</p> <p>(Example 1) <code>\$jdh_conv customjob_template.jdf jnw.jdf sched.jdf trg.jdf wkcal.jdf operation.jdf</code></p> <p>(Example 2) <code>\$jdh_conv jnw.jdf operation.jdf</code></p> <p>(Example 3) <code>\$jdh_conv operation.jdf</code></p>

Option	Description
	 <p>When performing conversion using the <code>jdh_conv</code> command, be sure to specify the operation file that defines the operation (<code>operation.jdf</code>).</p>
-x	<p>JPF files are converted to the JDF format. The configuration of the result is as follows.</p> <pre data-bbox="421 443 1465 763"> jpf/ -- def/ -- customjob_template.jdf //A customjob_template type JDF -- jnw.jdf //A jnw type JDF -- sched.jdf //A sched type JDF -- trg.jdf //A trg type JDF `-- wkcal.jdf //A wkcal type JDF `-- operation/ `-- operation.jdf //An operation type JDF </pre> <p>A JDF file does not exist if it has no definition.</p>
\$JPFFile	Specify the JPF file you want to convert to the JDF format.

3.20.3. Return Values

Return value	Description
0	Normal end
1	Abnormal end

3.20.4. Main Messages

Message	Description
JPF file "jc_def_20111028162232.jpf" conversion successfully.	A JPF file has been created successfully from a JDF file. The name of the created file is "jc_def_20111028162232.jpf".
Output JDFs to "jpf_20111028162342/jpf".	A JPF file has been converted successfully to a JDF file. The output destination is "jpf_20111028162342/jpf".
Convert check successfully.	The JDF check was successful.
ERROR: test.jdf:2: Invalid line format. "param="	This is the general error message displayed if the JDF format of a line is invalid. Check the corresponding line of the indicated file to see whether it complies with the JDF format. In this case, the message means that the format of the second line of the test.jdf file is invalid and that the content of the second line is "param=".
ERROR: test.jdf: No body section.	No body section exists in the JDF file. Specify a body section because every JDF file requires it. Specifically, the "%%%" separator is necessary.
ERROR: test.jdf:1: No "version" key in header section.	No version key exists in the header section. Specify a version key in the header section of a JDF file.
ERROR: test.jdf:1: No "type" key in header section.	No type key exists in the header section. Specify a type key in the header section of a JDF file.

Message	Description
ERROR: test.jdf:4: No "id" key	A key required for an object does not exist. Specify the required key. In this case, the message means that the id key required for the object near the fourth line of the test.jdf file does not exist.
ERROR: test.jdf:5: Unknown object type(soso)	An undefined key is specified. In this case, a nonexistent key "soso" is specified.
ERROR: test.jdf:4: pre_parts (1) not found for obj 3	Dependency error for a JNW flow. In this case, the dependency error means that the component whose id is 1 does not exist when it is specified as the preceding component for the component whose id is 3.
ERROR: test.jdf:9: Duplicate object id (3)	The component id is duplicate. In this case, the message means that id of 9 is duplicate on the ninth line of the test.jdf file.
WARNING: test.jdf: JDF of Launch Trigger was ignored. This JDF format is available in version 7.0 or later.	Since the JDF version of 7.0 or earlier is specified, the JPF file does not include the launch trigger definition and monitoring target text log.
WARNING: test.jdf: Operation of Launch Trigger was ignored. This type is available in version 7.0 or later.	Since the JDF version of 7.0 or earlier is specified, the operation of uploading the launch trigger definition and monitoring target text log was ignored.

3.20.5. Notes

- `jdh_conv` bundled with 12.8.2 to 12.10 is a function currently not open to general users and is subject to the restrictions described below. Before using this command, make sure you understand these restrictions.
 - The JDF format error check is not sufficient. Before using a JDF file, check that it is written correctly in the JDF format.
 - No JPF version compatibility check is implemented. When creating a JDF file, check its compatibility thoroughly.
- If you upload a JPF file created from an invalid JDF file may result in an unexpected error.

3.21. sleep (sleep Command for Windows)

■ Windows version

```
%InstallDirectory%\bin\utils\sleep.exe [$second]
```

3.21.1. Function Description

This command provides a function equivalent to the sleep command for Windows. The operation pauses for the specified number of seconds.

3.21.2. Options

\$second

Specify the number of seconds (integer) for which the operation is to pause. Specify a value between 0 and 2147483647.

3.21.3. Return Values

Return value	Description
0	Normal end
1	Abnormal end

3.21.4. Notes

■ To specify the number of seconds, enter an integer. (Entering a negative value causes an error.)

Do not specify two or more arguments. (This causes an error as well.)

3.22. spoolconv (Migrating User Definition Information of R12.10.x or Earlier)

■ UNIX version

```
/usr/lib/nqs/gui/bin/spoolconv [-c $clusterdb]
```

■ Windows version

(None)

3.22.1. Function Description

The spoolconv command migrates user definition information created in an environment of R12.10.x or earlier to R13.1. In the UNIX version, you need to migrate user definition information by using this command, while the information is automatically migrated in the Windows version.

3.22.2. Options

```
-c $ clusterdb
```

When migrating the definition information of a cluster site, specify the cluster database path of JobCenter. If you do not specify this option, the command migrates the definition information of the local site. To specify this option, you need to set environment variable NQS_SITE.

3.22.3. Return Values

Return value	Description
0	Normal end
1	Abnormal end

3.22.4. Notes

- When executing this command, use the root privileges.

3.23. jc_iedata_conv (Converting Export Data to a JPF File)

■ UNIX version

```
/usr/lib/nqs/gui/bin/jc_iedata_conv [-o $output] [-r $rulefile] $exportdata_dir
```

■ Windows version

```
%InstallDirectory%\bin\jc_iedata_conv.exe [-o $output] [-r $rulefile] $exportdata_dir
```

3.23.1. Function Description

The `jc_iedata_conv` command converts export data to the JPF format. You can convert the host name by specifying a rule file at the time of conversion.

An example of executing the command of the UNIX version is shown below.

```
$ /usr/lib/nqs/gui/bin/jc_iedata_conv -r rulefile -o /tmp/jpf /usr/spool/nqs/users/nsumsmgr/
exportfile/JNW_jcserver_201304161001001
convert successfully.(output=/tmp/jpf)
```

3.23.2. Options

`-o $output`

Specify the directory to which to output the conversion result. If you omit this option, the output destination directory is the one named `jc_YYYYMMDDhhmmss` under the current directory where the command was executed.

YYYYMMDDhhmmss indicates the date and time the command was executed.

`-r $rulefile`

Specify the name of the rule file for host name conversion. If you omit this option, host name conversion is not performed.

Host name conversion is performed for the following parameters.

- Host name part of the default job network submission queue
- Host name part of the unit job, WOBS job, and custom job submission queues
- Host name of the transmission destination of event transmission parts
- Host name of the transmission source of the normal end and abnormal end conditions of event reception parts

`$exportdata_dir`

Specify the directory of the export data to be converted.

3.23.2.1. Rule File

Write host name conversion definitions, each delimited by a single-byte space, as follows.

```
conversion-source-host-name-1 conversion-destination-host-name-1
conversion-source-host-name-2 conversion-destination-host-name-2
.
```



- The maximum number of bytes for the conversion destination host name is 63.
- In a host name, you cannot use tabs, single-byte kana characters, double-byte characters, and the characters shown below. Also, a host name must not begin with a number.

! " # \$ % & ' () * , / : ; < = + > ? @ [\] ^ ` { | } ~

3.23.3. Return Values

Return value	Description
0	Normal end
1	Abnormal end

3.23.4. Main Messages

Message	Description
Too many arguments.	There are too many arguments. Check whether the number of specified arguments is appropriate.
Too few arguments.	There are too few arguments. Check whether the number of specified arguments is appropriate.
No such data_dir path "JNW_jcserver_20130415091900".	The specified directory cannot be found. Check whether the path to the directory in which you have saved the export data is correct.
Invalid export data dir "JNW_jcserver_20130415091900".	The export data in the specified directory is invalid. Check that the following two files exist under the specified directory. <ul style="list-style-type: none"> ■ Export data file Example: JNW_jcserver_20130415091900.zip ■ Attribute definition file Example: JNW_jcserver_20130415091900.param
Invalid rule format(line=1)	The specified rule file is invalid. Check whether the host name conversion definition on the first line of the rule file is specified properly.
Invalid character(source machine name).	The conversion source host name in the specified rule file is invalid. Check the conversion source host name in the specified rule file to see whether it contains any prohibited character.
Invalid character(destination machine name).	The conversion destination host name in the specified rule file is invalid. Check the conversion destination host name in the specified rule file to see whether it contains any prohibited character.
No such rulefile path "rulefile".	The specified rule file cannot be found. Check whether the rule file path is correct.
Cannot convert ExportData.(DataLang=<export-data-character-code>, TermLang=<terminal-character-code>)	Conversion has been canceled because unsupported export data is specified.

Message	Description
	This message is for the Windows version only. For details about whether conversion is supported in the Windows version, see "Notes".
convert (output=jc_20130415202745) successfully.	Export data has been converted successfully to a JPF file. A JPF file of each user is created under the directory jc_20130415202745.

3.23.5. Notes

- If export data has any of the problems described below, it can be converted to a JPF file but uploading of the JPF file fails. When migrating definition information, export all necessary definitions including those that have dependencies.

The job network definition referenced as a sub job network part does not exist in the export data.

The schedule definition referenced as a calendar branch part does not exist in the export data.

The schedule definition used to start a schedule does not exist in the export data.

The working day calendar definition referenced from a schedule does not exist in the export data.

- The conversion result of calendar information is saved to the users mentioned below. When migrating definition information, specify the directory to which to output the conversion result as the upload target directory and upload the result to all users.

In the case of the UNIX version, the result is saved in the JPF file of nsumsmgr.

In the case of the Windows version, the result is saved in the JPF file of the JobCenter administrator user.

- Support of export data conversion in the Windows version

In the case of the Windows version, conversion may be canceled depending on the character code of the export data to be converted.

Export data conversion is supported as shown in the table below.

✓: Conversion supported, N/A: Conversion not supported

Table 3.14. Support of conversion by the jc_iedata_conv command for Windows

Character code of export data	Japanese environment (SJIS)	OS Chinese OS environment (GB18030)	English OS environment
ASCII(C)	✓	✓	✓
SJIS	✓	N/A	N/A
GB18030	N/A	✓	N/A
EUC	N/A	N/A	N/A
UTF-8	N/A	N/A	N/A

3.24. jc_perf (Analyzing the Execution Status)

■ UNIX version

```
/usr/lib/nqs/gui/bin/jc_perf config [-s $server] [$key=$val]
/usr/lib/nqs/gui/bin/jc_perf stat [-q] $data_file
/usr/lib/nqs/gui/bin/jc_perf view { greq $qtype $qname | rtrk | rjob[-a $jobtype] } [-s $server] [-f{csv | rd}] [-v{0 | 1 | 2}] [-n] [-d $date] [-p $period] [-i $interval] [-e $extdir]
```

■ Windows version

```
%InstallDirectory%\bin\jc_perf config [-s $server] [$key=$val]
%InstallDirectory%\bin\jc_perf stat [-q] $data_file
%InstallDirectory%\bin\jc_perf view { greq $qtype $qname | rtrk | rjob[-a $jobtype] } [-s $server] [-f{csv | rd}] [-v{0 | 1 | 2}] [-n] [-d $date] [-p $period] [-i $interval] [-e $extdir]
```

3.24.1. Function Overview

The `jc_perf` command collects and analyzes data about the execution status of JobCenter.

This command allows you to toggle between enabling and disabling data collection, check settings, and analyze data. The following three types of data can be collected.

■ Tracker execution status (rtrk data)

When the execution of a job network starts, the data of its flow and parts is copied to memory. After the execution is complete, the data is archived and deleted from memory. This means that the more job networks are executed simultaneously, the more memory JobCenter consumes.

Normally, the amounts of memory used during the peak time zones and at specific times are estimated through sizing in advance. Actually, however, the originally estimated amount of memory may be exceeded as a result of definitions added after the start of operation or for some other reason. Resource statistics such as the amount of memory used and CPU usage can be monitored by the standard function of the OS or other means. To find how many job networks were running and when they were running, however, you need to check their execution results (trackers).

Collecting this data makes it possible to check the execution status of these trackers.

■ Number of jobs ready to be submitted in a flow (rjob data)

When the execution of a job network starts and its flow reaches a certain job part, that component is submitted. If there are multiple job parts ready to be submitted (the execution of the previous component is completed), they are not processed simultaneously in its true sense. Rather, these job parts are submitted sequentially, beginning with the one that first becomes ready to be submitted.

Therefore, if many job parts need to be submitted simultaneously because the flow reaches a branch that requires a high degree of parallelism or because a large number of job networks are running at the same time, a subsequent job component may not start promptly (its status does not change from WAIT to SUBMIT) although the execution of the previous part is completed.

The number of jobs that can be submitted per unit time is determined by the specifications of the server to a certain extent. If more jobs than that arise at the same time, a delay occurs. Of course, such a delay between one part and another in the flow can result from a shortage of resources such as memory, CPU, etc. When there are sufficient resources, however, it may occur for the reason mentioned above.

By collecting this data, you can check the number of jobs ready to be submitted, which may cause a delay like the one described above.

■ Accumulation status of requests queued in NQS (qreq data)

When the flow reaches a unit job part, that component is submitted and queued in the specified queue. It changes its status several times in the queue and, when the number of currently running requests is below the number of requests that can be run simultaneously in the queue, its execution is started. If SUBMIT is displayed as the status of the target job in the flow and its execution does not start promptly (its status does not change to RUN), it can be considered that the queue to which the job has been submitted has more jobs running than the number of requests that can be run simultaneously, thus keeping the job waiting for its turn.

Collecting this data makes it possible to check the accumulation status of a queue.

3.24.2. Function Description

The `jc_perf` command has three command modes. This section describes how to use each of these command modes and their options.

3.24.2.1. Checking and Changing Settings (config)

■ Overview

If you specify `config` in the first argument, you can check and change the settings of the data collection function.

■ Options

`-s $server`

Specify a remote server. If you do not specify this option, the local server is assumed.

If you want to check and change the settings of a remote server, specify this option. If the target is the local server you are currently logged in, do not specify it.

If both the local site and cluster site reside in the local server, they are distinguished using environment variable `NQS_SITE`.

`$key=$val`

The value of the specified parameter (`$key`) is changed to the specified value (`$val`). The available parameter names and values are shown below.

Parameter name (\$key)	Specifiable value (\$val)	Default value	Description
<code>qreq</code>	on or off	off	Specify whether to collect the qreq data.
<code>rtrk</code>	on or off	off	Specify whether to collect the rtrk data.
<code>rjob</code>	on or off	off	Specify whether to collect the rjob data.
<code>file_size</code>	64 to 1024	128	Specify the maximum size of a single file to which to save data (data file) in MB.
<code>file_lifetime</code>	1 to 365	90	Specify the number of days to retain a data file.

If you specify `$key=$val`, the settings are changed. Otherwise, you only check the settings.



In order to check and change the settings in config mode, the target JobCenter needs to be running.

■ Output example

If you specify the config mode, data similar to the following is output to the standard output.

```
server: jobmanager.nec.com
collect(qreq): off
collect(rtrk): off
collect(rjob): off
file_size: 128 MB
file_lifetime: 90 days
```

Each output line corresponds to one of the parameters specified in \$key.

■ About the data file

The collected data is saved in one of the following files.

Destination directory

UNIX	Local site	/usr/spool/nqs/perf/<YYYYMMDD>
	Cluster site	<cluster-database-path>/perf/<YYYYMMDD>
Windows	Local site	%InstallDirectory\spool\perf\%<YYYYMMDD>
	Cluster site	<cluster-database-path>\perf\<YYYYMMDD>

File name

qreq data	qreq_<n>.dat
rtrk data	rtrk_<n>.dat
rjob data	rjob_<n>.dat

<YYYYMMDD> in the destination directory name represents a date. Directories are managed on a daily basis. Data files are created under this directory. <n> in the file name is an integer equal to or larger than 1. If the size of a data file exceeds the value specified in the file_size parameter within a day, one or more additional data files are created with the number incrementing to 2, 3, and so on.

Data files are stored in platform-independent binary format. To reference and analyze the data requires the use of the view mode. For details about the view mode, see [Section 3.24.2.3, "Viewing and Analyzing Data \(view\)"](#).



To copy data file to another server, be sure to copy all the files of the dated directory.

3.24.2.2. Checking a Data File (stat)

■ Overview

If you specify stat in the first argument, you can check the property information of a saved data file.

■ Options

```
-q
```

The queue names contained in the data file are listed. This option is available only when qreq data is specified in \$data_file.

To analyze data in view mode requires a queue type and a queue name. If they are unknown, check them in advance by using this -q option.

\$data_file

Specify the data file whose property information you want to check. This parameter is required.

■ Output example

If you execute the command in stat mode, data similar to the following is output to the standard output. (This is an example of data to be output when the -q option is specified.)

```

=== Data file status ===
Server: jobmanager.nec.com
JobCenter version: 13.2
Platform: Linux
Data type: qreq
Data version: 100
Data size: 44
Number of data: 1637
Creation time: 2013/09/20 01:03:12
File date: 20130920
File id: 1

=== Queue list ===
--- Pipe queue ---
guilb_def
guinw
guitp_1
pipe1
pipe2
--- Batch queue ---
batch1
batch2
guibs_1
--- Network queue ---
DefaultNetQue
    
```

The output items for Data file status are as follows.

Server	Name of the server that collects data
JobCenter version	Version of JobCenter
Platform	Type of the platform. It must be HPUX, AIX, Linux, or Windows.
Data type	Type of the data. It must be qreq, rtrk, or rjob.
Data version	Version of the data
Data size	Size of a data unit (bytes)
Number of data	Number of data units contained in this data file
Creation time	Time when this data file was created
File date	Date of the data file. It must match <YYYYMMDD> of the destination directory.
File id	File ID of the data file. It must match <n> of the data file name.

Under Queue list, the queues contained in the target data file are listed.

3.24.2.3. Viewing and Analyzing Data (view)

■ Overview

If you specify view in the first argument, you can analyze the collected data.

In the second argument, you need to specify one of the identifiers shown below to indicate the data type.

qreq	<p>The qreq data is analyzed.</p> <p>If you select qreq, you need to specify the type of queue to be analyzed (\$qtype) and its queue name (\$qname) after that.</p> <p>In \$qtype, specify one of the following.</p> <ul style="list-style-type: none"> ▪ pipe: Pipe queue ▪ batch: Batch queue ▪ network: Network queue <p>In \$qname, specify a queue name. The queue name format is <queue-name>@<host-name>. But, here, specify only the queue name without the part after the @ mark.</p> <p>You can also check the list of queues contained in a data file in stat mode. For information about how to do this, see Section 3.24.2.1, "Checking and Changing Settings (config)".</p>
rtrk	<p>The rtrk data is analyzed.</p>
rjob	<p>The rjob data is analyzed.</p> <p>By default, only unit jobs are analyzed. By using -a \$jobtype, however, you can specify an additional job type.</p> <p>In \$jobtype, you can specify the following.</p> <ul style="list-style-type: none"> ▪ sap: SAP job components (ERP job, BI job, and PC job)

■ Other options

-s \$server

Specify a remote server. If you do not specify this option, the local server is assumed.

If you want to analyze the data of a remote server, specify this option. If the target is the local server you are currently logged in, do not specify it.

If both the local site and cluster site reside in the local server, they are distinguished using environment variable NQS_SITE.



You cannot specify this option together with the -e option.

-f

Specify the output format of the analysis results. Select one of the following.

- rd: Results are output in readable format.
- csv: Results are output in CSV format.

If you do not specify this option, results are output in readable format.

For information about the output format of each output level, see [Section 3.24.2.4, "Output Formats for the View Mode"](#).

-v

Specify the output level of the analysis results. Select one of the following.

- 0: IN and OUT counts of the target data
- 1: Time average value for each status
- 2: All information to be output when 0 and 1 are specified and maximum and minimum values per unit time

If you do not specify this option, 0 is assumed.

For information about the output format of each output level, see [Section 3.24.2.4, "Output Formats for the View Mode"](#).



If you specify level 2, the amount of information to be output is too large to be displayed in readable format. When specifying level 2, select the CSV format in the -f option.

-n

Specify this option if you do not want the title line (header line) to be included when analysis results are output.

-d \$date

Specify the date of the data you want to analyze in the format of YYYYMMDD.

For example, if you want to analyze the data of September 20, 2013, specify 20130920.

If you do not specify this option, the data of the present day is analyzed.

-p \$period

Specify the period of the data you want to analyze in the format of hhmm-hhmm. The value range is 0000 and 2400.

For example, if you want to analyze the data from 12:00 to 18:00, specify 1200-1800. In this case, the data recorded from 12:00:00.000 to 17:59:59.999 is analyzed. (The data recorded at 18:00:00.000 is not included.)



When you analyze the data of the present day, the data after the current time has yet to be output and, therefore, the period of the output data may be shorter than that specified in \$period.

If you do not specify this option, all the data of the specified date is analyzed.

-e \$extdir

Specify this option if you want to analyze data copied from another server, instead of the data in the target server.

In \$extdir, specify the directory one level higher than the dated directory (YYYYMMDD). For information about the directory to store data files, see "[About the data file](#)".



You cannot specify this option together with the -s option.

3.24.2.4. Output Formats for the View Mode

This section describes the output formats for the view mode.

The output examples given below all assume that 60 minutes is specified in the -i option (interval).

■ qreq, level 0

Readable format

DATE	IN	OUT
2013/09/20 00:00	48	48
2013/09/20 01:00	31	31
2013/09/20 02:00	0	0
2013/09/20 03:00	11	10
2013/09/20 04:00	8	9
2013/09/20 05:00	0	0
...		

CSV format

DATE, IN, OUT
2013/09/20 00:00,48,48
2013/09/20 01:00,31,31
2013/09/20 02:00,0,0
2013/09/20 03:00,11,10
2013/09/20 04:00,8,9
2013/09/20 05:00,0,0
...

Interpret the IN and OUT counts as follows.

IN	Number of requests enqueued (put into the queue) during the time zone
OUT	Number of requests dequeued (put out of the queue) during the time zone

■ qreq, level 1

Since a request takes different statuses depending on the type of queue, the output format also varies.

Each value shows the average time the requests remained in that particular status. Under TOTAL, the total of these values is shown.

Therefore, if the value of any status is over 1.0, it means that one or more requests remained in that status on average during the corresponding time zone.

Readable format (pipe queue)

DATE	TOTAL	DEPART	ROUTE	QUEUED	WAIT	HOLD	ARRIVE
2013/09/20 00:00	0.1	0.0	0.1	0.0	0.0	0.0	0.0
2013/09/20 01:00	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2013/09/20 02:00	0.1	0.0	0.1	0.0	0.0	0.0	0.0
2013/09/20 03:00	0.8	0.0	0.8	0.0	0.0	0.0	0.0
2013/09/20 04:00	0.1	0.0	0.1	0.0	0.0	0.0	0.0
2013/09/20 05:00	0.0	0.0	0.0	0.0	0.0	0.0	0.0
...							

CSV format (pipe queue)

```
DATE,TOTAL,DEPART,ROUTE,QUEUED,WAIT,HOLD,ARRIVE
2013/09/20 00:00,0.1,0.0,0.1,0.0,0.0,0.0,0.0
2013/09/20 01:00,0.0,0.0,0.0,0.0,0.0,0.0,0.0
2013/09/20 02:00,0.1,0.0,0.1,0.0,0.0,0.0,0.0
2013/09/20 03:00,0.8,0.0,0.8,0.0,0.0,0.0,0.0
2013/09/20 04:00,0.1,0.0,0.1,0.0,0.0,0.0,0.0
2013/09/20 05:00,0.0,0.0,0.0,0.0,0.0,0.0,0.0
...
```

The pipe queue is primarily intended to transfer requests to another queue and, while a transfer is in progress, the status is either ROUTE or DEPART. Therefore, if any of these values is large, it means that there are many requests that took long to be transferred, which requires attention.

Readable format (batch queue)

DATE	TOTAL	EXIT	RUN	SUSPND	STAGE	QUEUED	WAIT	HOLD	ARRIVE
2013/09/20 00:00	0.2	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.1
2013/09/20 01:00	20.4	0.0	8.0	0.0	0.0	12.3	0.0	0.0	0.1
2013/09/20 02:00	20.0	0.0	8.0	0.0	0.0	11.8	0.0	0.0	0.2
2013/09/20 03:00	17.9	0.1	8.0	0.0	0.0	9.8	0.0	0.0	0.0
2013/09/20 04:00	5.1	0.0	4.9	0.0	0.0	0.1	0.0	0.0	0.1
2013/09/20 05:00	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
...									



Since the display width is adjusted for space considerations, it differs from the actual output width.

CSV format (batch queue)

```
DATE,TOTAL,EXIT,RUN,SUSPEND,STAGE,QUEUED,WAIT,HOLD,ARRIVE
```

```

2013/09/20 00:00,0.2,0.0,0.0,0.0,0.0,0.1,0.0,0.0,0.1
2013/09/20 01:00,20.4,0.0,8.0,0.0,0.0,2.3,0.0,0.0,0.1
2013/09/20 02:00,20.0,0.0,8.0,0.0,0.0,1.8,0.0,0.0,0.2
2013/09/20 03:00,17.9,0.1,8.0,0.0,0.0,0.8,0.0,0.0,0.0
2013/09/20 04:00,5.1,0.0,4.9,0.0,0.0,0.1,0.0,0.0,0.1
2013/09/20 05:00,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
...
    
```

The purpose of the batch queue is to execute queued requests (jobs). While a request is running, its status is RUN. For a request to take the RUN status, the number of currently running requests needs to be below the maximum number of requests that can be run simultaneously. Otherwise, the request must wait in the QUEUED status. Therefore, a large QUEUED value means that the target queue is clogged with requests, which requires attention.

For instance, suppose that the maximum number of requests that can be run simultaneously in the target queue in the above output example is 8. From the fact that the RUN values for 1:00 to 3:00 are 8.0, it is evident that the number of requests close to the upper limit remained in the RUN status throughout this period. It can also be seen that the queue had many requests waiting in the QUEUED status during this period.

Readable format (network queue)

DATE	TOTAL	RUN	QUEUED	WAIT	HOLD	ARRIVE
2013/09/20 00:00	0.0	0.0	0.0	0.0	0.0	0.0
2013/09/20 01:00	0.3	0.3	0.0	0.0	0.0	0.0
2013/09/20 02:00	0.3	0.3	0.0	0.0	0.0	0.0
2013/09/20 03:00	0.2	0.2	0.0	0.0	0.0	0.0
2013/09/20 04:00	0.1	0.1	0.0	0.0	0.0	0.0
2013/09/20 05:00	0.0	0.0	0.0	0.0	0.0	0.0
...						

CSV format (network queue)

```

DATE,TOTAL,RUN,QUEUED,WAIT,HOLD,ARRIVE
2013/09/20 00:00,0.0,0.0,0.0,0.0,0.0,0.0
2013/09/20 01:00,0.3,0.3,0.0,0.0,0.0,0.0
2013/09/20 02:00,0.3,0.3,0.0,0.0,0.0,0.0
2013/09/20 03:00,0.2,0.2,0.0,0.0,0.0,0.0
2013/09/20 04:00,0.1,0.1,0.0,0.0,0.0,0.0
2013/09/20 05:00,0.0,0.0,0.0,0.0,0.0,0.0
...
    
```

The purpose of the network queue is to return the results of executed requests (jobs) to the transfer source. While a result is being returned, the status of the request is RUN. If the maximum number of requests that can be run simultaneously is exceeded, requests wait in the QUEUED status.



The network queue is the type of queue supported only in the UNIX version; it is not available in the Windows version.

■ qreq, level 2

In addition to the information of levels 0 and 1, the minimum value (MIN) and maximum value (MAX) of each time zone are output. Specify this level if you want to check more detailed information.

Here, only the output formats are shown.

CSV format (pipe queue)

```
DATE,IN,OUT,TOTAL(AVE),TOTAL(MIN),TOTAL(MAX),DEPART(AVE),DEPART(MIN),
DEPART(MAX),ROUTE(AVE),ROUTE(MIN),ROUTE(MAX),QUEUED(AVE),QUEUED(MIN),
QUEUED(MAX),WAIT(AVE),WAIT(MIN),WAIT(MAX),HOLD(AVE),HOLD(MIN),HOLD(MAX),
ARRIVE(AVE),ARRIVE(MIN),ARRIVE(MAX)
2013/09/20 00:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 01:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 02:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 03:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 04:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 05:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
...
```

CSV format (batch queue)

```
DATE,IN,OUT,TOTAL(AVE),TOTAL(MIN),TOTAL(MAX),EXIT(AVE),EXIT(MIN),EXIT(MAX),
RUN(AVE),RUN(MIN),RUN(MAX),SUSPEND(AVE),SUSPEND(MIN),SUSPEND(MAX),STAGE(AVE),
STAGE(MIN),STAGE(MAX),QUEUED(AVE),QUEUED(MIN),QUEUED(MAX),WAIT(AVE),WAIT(MIN),
WAIT(MAX),HOLD(AVE),HOLD(MIN),HOLD(MAX),ARRIVE(AVE),ARRIVE(MIN),ARRIVE(MAX)
2013/09/20 00:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 01:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 02:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 03:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 04:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 05:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
...
```

CSV format (network queue)

```
DATE,IN,OUT,TOTAL(AVE),TOTAL(MIN),TOTAL(MAX),RUN(AVE),RUN(MIN),RUN(MAX),QUEUED(AVE),
QUEUED(MIN),QUEUED(MAX),WAIT(AVE),WAIT(MIN),WAIT(MAX),HOLD(AVE),HOLD(MIN),HOLD(MAX),
ARRIVE(AVE),ARRIVE(MIN),ARRIVE(MAX)
2013/09/20 00:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 01:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 02:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 03:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 04:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 05:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0,0.0,0,0
...
```



While the title is broken into multiple lines for space considerations, it is actually output on a single line.

■ rtrk, level 0

Readable format

DATE	IN	OUT
2013/09/20 00:00	10	8
2013/09/20 01:00	12	11
2013/09/20 02:00	14	15
2013/09/20 03:00	14	14
2013/09/20 04:00	10	10
2013/09/20 05:00	10	11
...		

CSV format

DATE,IN,OUT
2013/09/20 00:00,10,8
2013/09/20 01:00,12,11
2013/09/20 02:00,14,15
2013/09/20 03:00,14,14
2013/09/20 04:00,10,10
2013/09/20 05:00,10,11
...

Interpret the IN and OUT counts as follows.

IN	<p>Number of trackers read into memory during the time zone A tracker is read into memory in one of the following cases.</p> <ul style="list-style-type: none"> ■ A waiting tracker reaches the expected start time. ■ An operation is performed for a waiting tracker, placing it in the PREPARE status. ■ A job network is submitted immediately.
OUT	<p>Number of trackers freed from memory during the time zone A tracker is freed from memory in one of the following cases.</p> <ul style="list-style-type: none"> ■ An unarchived tracker is archived. ■ An unarchived tracker is deleted.

■ rtrk, level 1

Readable format

DATE	TOTAL	PREPARE	RUNCTRL	RUN	RERUNWAI
2013/09/20 00:00	0.0	0.0	0.0	0.0	0.0
2013/09/20 01:00	6.5	0.2	0.0	2.3	4.0
2013/09/20 02:00	19.8	1.1	4.8	9.1	4.8
2013/09/20 03:00	17.0	0.0	0.0	5.8	11.2
2013/09/20 04:00	9.0	0.0	0.0	3.9	5.1
2013/09/20 05:00	0.0	0.0	0.0	0.0	0.0
...					

CSV format

DATE, TOTAL, PREPARE, RUNCTRL, RUN, RERUNWAI
2013/09/20 00:00,0.0,0.0,0.0,0.0,0.0
2013/09/20 01:00,6.5,0.2,0.0,2.3,4.0
2013/09/20 02:00,19.8,1.1,4.8,9.1,4.8
2013/09/20 03:00,17.0,0.0,0.0,5.8,11.2
2013/09/20 04:00,9.0,0.0,0.0,3.9,5.1
2013/09/20 05:00,0.0,0.0,0.0,0.0,0.0
...

Interpret the statuses as follows.

PREPARE	Time average of the number of trackers in the PREPARE status
RUNCTRL	Time average of the number of trackers waiting in the PREPARE status due to the limit on the number of trackers that can be started simultaneously in the job network parameter
RUN	Time average of the number of running trackers
RERUNWAI	Time average of the number of trackers that have been executed but not archived yet

■ rtrk, level 2

In addition to the information of levels 0 and 1, the minimum value (MIN) and maximum value (MAX) of each time zone are output. Specify this level if you want to check more detailed information.

Here, only the output formats are shown.

CSV format

```
DATE,IN,OUT,TOTAL(AVE),TOTAL(MIN),TOTAL(MAX),PREPARE(AVE),PREPARE(MIN),
PREPARE(MAX),RUNCTRL(AVE),RUNCTRL(MIN),RUNCTRL(MAX),RUN(AVE),RUN(MIN),
RUN(MAX),RERUNWAI(AVE),RERUNWAI(MIN),RERUNWAI(MAX)
2013/09/20 00:00,0,0,0.0,0,0,0.0,0,0,0,0,0,0,0.0,0,0,0,0
2013/09/20 01:00,0,0,0.0,0,0,0.0,0,0,0,0,0,0,0.0,0,0,0,0
2013/09/20 02:00,0,0,0.0,0,0,0.0,0,0,0,0,0,0,0.0,0,0,0,0
2013/09/20 03:00,0,0,0.0,0,0,0.0,0,0,0,0,0,0,0.0,0,0,0,0
2013/09/20 04:00,0,0,0.0,0,0,0.0,0,0,0,0,0,0,0.0,0,0,0,0
2013/09/20 05:00,0,0,0.0,0,0,0.0,0,0,0,0,0,0,0.0,0,0,0,0
...
```



While the title is broken into multiple lines for space considerations, it is actually output on a single line.

■ rjob, level 0

Readable format

DATE	UNIT IN	UNIT OUT
2013/09/20 00:00	121	121
2013/09/20 01:00	374	374
2013/09/20 02:00	811	811
2013/09/20 03:00	374	374
2013/09/20 04:00	374	374
2013/09/20 05:00	370	370
...		

CSV format

DATE,UJ_IN,UJ_OUT
2013/09/20 00:00,121,121
2013/09/20 01:00,374,374
2013/09/20 02:00,811,811
2013/09/20 03:00,374,374
2013/09/20 04:00,374,374
2013/09/20 05:00,370,370
...

Interpret the IN and OUT counts as follows.

IN	Number of job parts that became ready to be submitted (that were reached by the flow) during the time zone
OUT	Number of job parts whose submission was completed during the time zone Note that, even if a job component is recorded in OUT, its execution has not necessarily been started. Whether a job is actually executed depends on the available space of the submission destination queue. You can check the condition of the queue by looking at the qreq data.



Shown above is an example of data that is shown when the -a option is not specified and only unit jobs are output. If the -a option is specified, the field of the specified job part is also output.

■ rjob, level 1

Readable format

DATE	UNIT TOTAL	UNIT WAIT	UNIT RUN
2013/09/20 00:00	0.0	0.1	0.0
2013/09/20 01:00	4.5	4.5	0.0
2013/09/20 02:00	12.0	12.0	0.0
2013/09/20 03:00	10.0	10.0	0.0
2013/09/20 04:00	46.2	46.1	0.1
2013/09/20 05:00	0.0	0.0	0.0
...			

CSV format

DATE,UJ_TOTAL,UJ_WAIT,UJ_RUN
2013/09/20 00:00,0.0,0.1,0.0
2013/09/20 01:00,4.5,4.5,0.0
2013/09/20 02:00,12.0,12.0,0.0
2013/09/20 03:00,10.0,10.0,0.0
2013/09/20 04:00,46.2,46.1,0.1
2013/09/20 05:00,0.0,0.0,0.0
...

WAIT and RUN indicate the internal processing statuses of JobCenter for those job parts that became ready to be submitted. Under TOTAL, the total of these values is shown.

WAIT	Time average of the number of job parts that became ready to be submitted but whose submission did not start yet
RUN	Time average of the number of job parts that became ready to be submitted and whose submission already started

When the flow reaches a job part, it is added to the IN count of the rjob data, and its initial status is WAIT. Then, when the submission actually starts, the status changes to RUN. Upon completion of the submission, the job component is added to the OUT count of the rjob data.

In the level 1 data, the time a job component remains in the RUN status, namely the time required for submission, is almost instantaneous. Actually, a job component is counted as the WAIT status in most cases. Therefore, pay attention only to the TOTAL values.

A large TOTAL value means that jobs ready to be submitted are accumulating without being processed. In that case, a subsequent job component does not start promptly (its status does not change to SUBMIT) although the execution of the previous component is completed. If this situation frequently occurs and the delay time being experienced is unacceptable for the operation, the scheduling setting needs to be changed to prevent jobs from concentrating.



Shown above is an example of data that is shown when the -a option is not specified and only unit jobs are output. If the -a option is specified, the field of the specified job component is also output.

■ rjob, level 2

In addition to the information of levels 0 and 1, the minimum value (MIN) and maximum value (MAX) of each time zone are output. Specify this level if you want to check more detailed information.

Here, only the output formats are shown.

CSV format

```
DATE,UJ_IN,UJ_OUT,UJ_TOTAL(AVE),UJ_TOTAL(MIN),UJ_TOTAL(MAX),UJ_WAIT(AVE),
UJ_WAIT(MIN),UJ_WAIT(MAX),UJ_RUN(AVE),UJ_RUN(MIN),UJ_RUN(MAX)
2013/09/20 00:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 01:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 02:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 03:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 04:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0
2013/09/20 05:00,0,0,0.0,0,0,0.0,0,0,0.0,0,0
...
```



While the title is broken into multiple lines for space considerations, it is actually output on a single line.



Shown above is an example of data that is shown when the -a option is not specified and only unit jobs are output. If the -a option is specified, the field of the specified job component is also output.

3.24.3. Return Values

Return value	Description
0	Normal end
1	Abnormal end

3.24.4. Main Messages

Message	Description
Only manager user can execute this command.	The command was executed by a user other than the JobCenter administrator user. Execute the command as the JobCenter administrator user.
Too many arguments.	There are too many arguments. Check whether the number of specified arguments is appropriate.
Too few arguments.	There are too few arguments. Check whether the number of specified arguments is appropriate.
Data file or data not found.	The data file does not exist or contain the data to be analyzed.
Invalid command mode	The command mode is invalid. You need to specify config, stat, or view.
Invalid data type	The specified data type is invalid. You need to specify qreq, rtrk, or rjob.

3.24.5. Notes

- You need to execute this command as the JobCenter administrator user.
- If you enable the data collection function by using this command, the data about the execution status is accumulated. Pay attention to the retention period of the data and the remaining free disk space.

3.25. trfdelete (Deleting Tracking Files)

■ UNIX version

```
/usr/lib/nqs/trfdelete
```

■ Windows version

```
%InstallDirectory%\bin\trfdelete
```

3.25.1. Function Description

The `trfdelete` command is used to delete tracking files when you upgrade the version of JobCenter or perform troubleshooting. Execute this command as the root user in the UNIX version or as the JobCenter administrator user in the Windows version. When using the command, make sure that JobCenter is stopped.

3.25.2. Return Values

Return value	Description
0	Normal end
1	Abnormal end

3.25.3. Notes

- When you execute the command for a cluster site, you need to set the environment variables `NQS_SITE` and `NQS_SITEDB` in advance.
- Do not use this command unless you are instructed to delete tracking files while upgrading the version of JobCenter or performing troubleshooting.
- After restoring the NQS-related data obtained when upgrading the version of the active system of the cluster site of the UNIX version, you do not need to execute the `trfdelete` command for the cluster site if the directories `O0000` to `O0029` and `S0000` to `S0029` do not exist under the directory shown below.

UNIX version	<cluster-database-path>/lock/tracking
--------------	---------------------------------------

3.26. JCSession (Running a Specified Command from a Unit Job as an Active Session)

■ Windows version

```
%InstallDirectory%\bin\utils\JCSession.exe [-i $sessionid] $command [$arguments...]
```

3.26.1. Function Description

The JCSession command runs a specified command from a unit job as an active session.

This command allows you to execute GUI applications such as NotePad and Excel macro-embedded jobs normally in the environment of Windows Server 2008 or later.

To use this command, you need to specify the LSA queue as the unit job submission queue.

3.26.2. Options

`-i $sessionid`

This is the ID of the session for running the specified command.

An error occurs if the specified session ID is not active.

If you omit this option, the value of environment variable JC_JOBRUNSESSION is assumed.



If neither this option nor environment variable JC_JOBRUNSESSION is set, an active session is automatically selected. To explicitly specify a particular session for running the job, be sure to specify this option or environment variable JC_JOBRUNSESSION.

`$command $arguments...`

Specify the command to run and its arguments.



Specify the value for `$command` of up to 259 bytes.

3.26.3. Environment Variables

This command uses the environment variables shown below.

Environment variable name	Description
JC_JOBRUNSESSION	This is the ID of the session for running the specified command. This value is used if the <code>-i</code> option is omitted. An error occurs if the specified session ID is not active.
JC_USE_USERENV	Set how to inherit the environment variables of the specified command. The following values are available. ■ JobCenter The environment variables of JobCenter are used. ■ User

The environment variables for the session user that runs the specified command are used.

■ Both

Both the environment variables of JobCenter and the environment variables for the session user that runs the specified command are used.

This is the default to be assumed when this environment variable is not set.



If the same environment variables are defined for both JobCenter and the session user, those of the session user take precedence.

3.26.4. Return Values

The return values of the JCSession command are as follows.

Return value	Description
0	Normal end
1	Abnormal end



When the JCSession command is executed successfully, the command specified in \$command is run. Therefore, the return value obtained upon the end of the command is that of the command specified in \$command.

3.26.5. How to Use This Command

This section describes how to use the JCSession command, by using an example in which NotePad is started from an Administrator user session.

1. Check the Administrator user session ID.

Check the ID of the session for which you want to display NotePad on the screen.

Log on to the job execution server. Start the Task Manager by pressing down the Ctrl, Shift, and Esc keys at the same time, and display the [User] tab. The values in the ID column are session IDs. Here, you can see that the session ID of the Administrator user is "1".

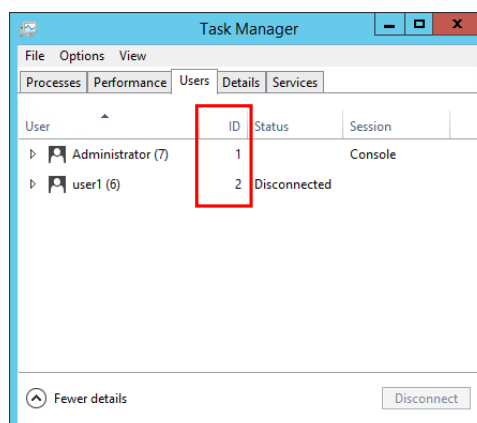


Figure 3.1. Checking the session ID



- The session ID changes every time the user logs on to or logs out from Windows. Therefore, you need to keep the target user logged on.
- If you do not see the ID column, take the following steps.
 - For Windows Server 2008
 - a. Display the [User] tab of the Task Manager.
 - b. In the menu bar, click [View] -> [Select Columns].
 - c. Check the check box for ID.
 - For Windows Server 2012
 - a. Display the [User] tab of the Task Manager.
 - b. Right-click the title of the column, and select ID.

2. Create a job network.

Create a job network. For information about how to create one, see in <Basic Guide>Section 3.1, "Creating a Job Network".

3. Place a unit job.

Place a unit job in the flow of the job network you created in step 2. See details about how to place a unit job in <Basic Guide>Section 4.1, "Placing a Part Object".

4. Set the script of the unit job.

Open the script of the unit job, and write the command shown below. Here, specify "1" in the -i option, which is the session ID you checked in step 1.

```
%InstallDirectory%\bin\utils\JCSession.exe -i 1 notepad.exe
```

5. Set the LSA queue as the submission queue of the unit job.

Open the window for setting the unit job parameters. Click the [Browse] button in [Submission Queue name] of the [Submit Parameters] tab.

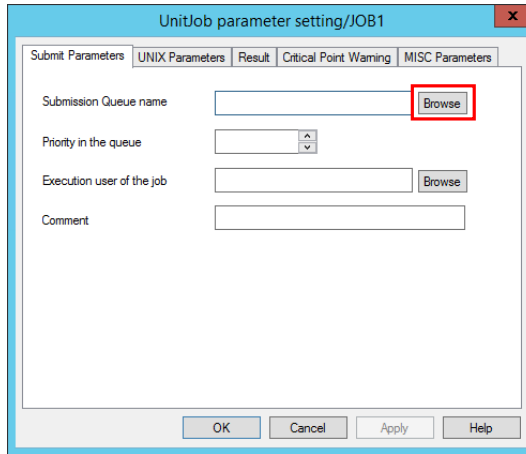


Figure 3.2. Unit job parameter setting window

The window for setting the submission destination queue appears. Select the LSA queue, and click the [OK] button.

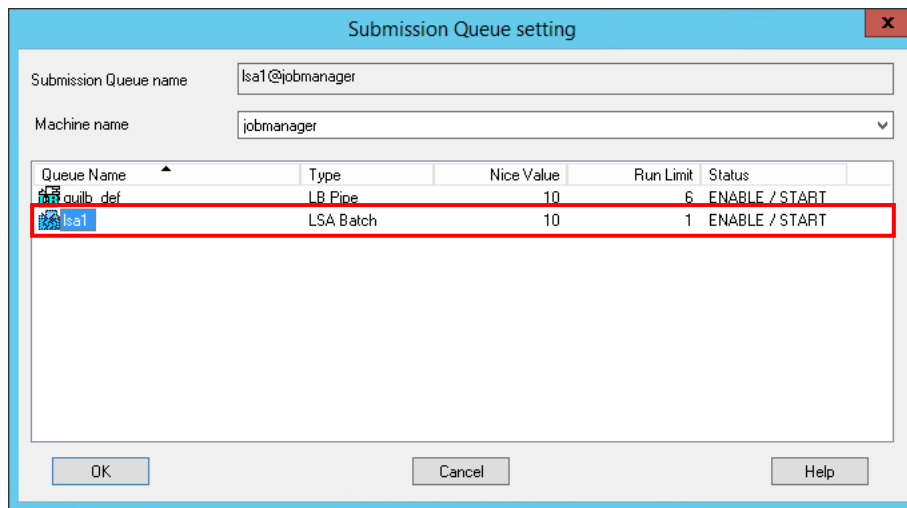


Figure 3.3. Submission Queue setting window

In the unit job parameter setting window, click the [OK] button to finish the setting.

6. Execute the job network.

Execute the created job network. See details about how to execute a job network in <Basic Guide>Section 3.2, "Executing a Job Network".

NotePad starts for the session of the Administrator user. While NotePad is running, the unit job remains in the RUN status. When you end NotePad, the unit job ends as well.

3.26.6. Notes

- You cannot execute this command directly from the command prompt.
- If JobCenter has been started by cjcpw and the status is Running (cjcpw), you cannot use this command. Start it as a service so that the status will be Running (service) .

See details about how to start JobCenter in <Environment Guide>Section 12.3.2, "Starting and stopping a site".

- If you forcibly stop (or skip) a running unit job in which this command is written, that unit job itself is stopped (or skipped). Note, however, that the application started via JCSession continues to run.
- To execute the JCSession command (\$command) so that the user account control is displayed when the user account control is enabled, it is required to execute the command via "cmd.exe /c".

```
%InstallDirectory%\bin\utils\JCSession.exe cmd.exe /c $command
```

If the command is not executed via "cmd.exe /c", an error occurs and the following message is displayed in the standard error output.

```
CreateProcessAsUser ERROR, ec=740.
```

3.27. setuserpwd (Updating the User Password)

■ Windows version

```
%InstallDirectory%\bin\setuserpwd -c
%InstallDirectory%\bin\setuserpwd $user $password
```

3.27.1. Function Description

The setuserpwd command updates the password of a specified user in the Windows version of JobCenter MG/SV.

In the Windows version of JobCenter MG/SV, when a job is executed with the privileges of a specified user, OS logon authentication is required in the background. It is therefore necessary to register the OS password in JobCenter MG/SV in advance. This is done when a user is registered in JobCenter. If the OS password is changed, however, the new password needs to be registered.

The password can be registered by updating the current one using this command as well as during the server environment setup. See details about how to update the password during the server environment setup in <Environment Guide>Section 12.4, "Setting Up a User".

3.27.2. Options

-c

The registration status of the password is displayed in the standard output.

\$user

Specify the name of the user whose password is to be updated.

\$password

Specify the user password to be updated.

If the password contains any of the characters shown below, each of those characters needs to be escaped using the corresponding escape characters shown below.

Character that needs to be escaped	Escape character	Command execution example
^	^	setuserpwd.exe user1 te^^st^<123^>45
&		
<		
>		
"	\	setuserpwd.exe user1 test\"123\"45

3.27.3. Output Result

When -c is specified, the command outputs the following to the standard output.

UserName	HostName	PasswordStatus
-----	-----	-----
user1	testdomain.local	OK

user2	testdomain.local	NG
testuser	jobmanager	OK
Administrator	testdomain.local	OK

Item	Description
UserName	User name
HostName	The name of the domain to which the target user belongs is displayed. In the case of a local user, the local host name is displayed.
PasswordStatus	The password registration status of the target user is displayed. OK: A correct password is registered. NG: The password is incorrect. NOTSET: No password is registered.

Other main messages are shown below.

Message	Description
Updating JobCenter administrator password... OK	The password of the JobCenter administrator user has been updated successfully.
Set user password OK!	The password of a general user has been updated successfully.
The password is incorrect.(ERRORCODE=xxx)	Authentication by the entered password has failed. (Error code = xxx)
The account does not belong to JobCenter group.	The target user does not belong to a JobCenter group.
Error! The parameter is incorrect.	The parameter is incorrect.

3.27.4. Return Values

■ When -c is specified

Return value	Description
0	The PasswordStatus of all users is OK.
1	There is one or more users whose PasswordStatus is NG.
2	The command ended abnormally.

■ When a password is updated

Return value	Description
0	Normal end
2	Abnormal end

3.28. paramctrl (Registering a Parameter in an Encrypted Database)

■ UNIX version

```
/usr/lib/nqs/gui/tools/paramctrl list
/usr/lib/nqs/gui/tools/paramctrl set [-f] $key $value
/usr/lib/nqs/gui/tools/paramctrl del [-f] $key
```

■ Windows version

```
%InstallDirectory%\tools\paramctrl.bat list
%InstallDirectory%\tools\paramctrl.bat set [-f] $key $value
%InstallDirectory%\tools\paramctrl.bat del [-f] $key
```

3.28.1. Function Description

- This command provides the function to encrypt a parameter to be used in an extended custom job and store it to a JobCenter database.
- The set encrypted parameter can be specified in the format of \$key\$ as a custom parameter of the extended custom job. When you use a secret parameter such as a password, this command allows you to encrypt it before saving it.
- The parameter is decrypted in the custom parameters when the job is executed. Therefore, the script review function shows it as \$key\$.

3.28.2. Sub Commands

Sub command	Function
list	Lists the set keys.
set	Registers the combination of a key (\$key) and a value (\$value) in the encrypted database. The alphanumeric characters in \$key are case sensitive.
del	Deletes the key (\$key) registered in the encrypted database. The alphanumeric characters in \$key are case sensitive.

3.28.3. Options

-f

If you specify this option, the command overwrites (when set is specified) or deletes (when del is specified) the existing key without confirmation.

If you do not specify this option, the following confirmation message appears.

■ When set is specified

```
Entry of "$key" already exists. Do you want to overwrite it? (y/n)
```

■ When del is specified

Do you want to delete key ? [\$key] (y/n)

To continue, type y. If you type n, the processing is canceled.

3.28.4. Return Values

Return value	Description
0	Normal end
1	Abnormal end

3.28.5. Notes

- When you execute the command for a cluster site, you need to set the environment variables NQS_SITE and NQS_SITEDB in advance.

3.29. jcrs (HTTP Daemon Dedicated for JobCenter MG/SV)

■UNIX version

```
/usr/lib/nqs/gui/tools/jcrs
/usr/lib/nqs/gui/tools/jcrs -e
/usr/lib/nqs/gui/tools/jcrs -v
```

■Windows version

```
%InstallDirectory%\tools\jcrs.bat
%InstallDirectory%\tools\jcrs.bat -e
%InstallDirectory%\tools\jcrs.bat -v
```

3.29.1. Function Description

This command starts or stops jcrs, the HTTP daemon dedicated for JobCenter MG/SV. By using the API of jcrs, you can check the connection between MG and SV from the browser.

See details about how to use jcrs in <Environment Guide>Chapter 17, 「Using jcrs」 .

3.29.2. Options

```
-e
jcrs is stopped.
-v
The version information is displayed.
```

3.29.3. Configuration File

This file allows you to configure how jcrs behaves. Set this file if you want to change the default behavior.

The configuration file is stored in the following directory.

UNIX version	/usr/lib/nqs/rc/jcrs.conf
Windows version	%InstallDirectory%\etc\jcrs.conf

When setting the file using a text editor, use the format shown below.

```
parameter=value
```

The parameters you can set are as follows.

Parameter	Description	Range	Default value
LOGDIR	Specify the absolute path to the directory that stores the access log and error log.	Absolute path	<p>■ for UNIX :</p> <p>/usr/spool/nqs/log</p> <p>■ for Windows:</p> <p>%InstallDirectory%\spool\log\error</p>

Parameter	Description	Range	Default value
PORT	Specify a port number.	1 to 65535	50080
IPADDRESS	Specify an IP address.	IPv4 address	0.0.0.0

A sample configuration file is shown below.

```
LOGDIR=/usr/spool/nqs/log
PORT=50080
IPADDRESS=0.0.0.0
```



- Before editing the configuration file, stop jcrs. If you edit the configuration file while jcrs is running, it may become impossible to stop jcrs normally.
- If the configuration file fails to be read, the default value is assumed for each parameter.
- If a parameter is not set or an invalid parameter is specified, the default value is assumed for that parameter.
- To set LOGDIR, create the target directory in advance. If you specify a nonexistent directory or a directory in which files cannot be created, the default value is assumed.

3.29.4. Access Log

An access log is recorded for received HTTP requests.

The access log is stored in the following directory.

UNIX	/usr/spool/nqs/log/jcrs_access.log
Windows	%InstallDirectory%\spool\log\error\jcrs_access.log

The directory storing the log can be changed in the configuration file. For details about the configuration file, see [Section 3.29.3, "Configuration File"](#).

The format of the access log is shown below.

```
Source IP address [YYYY/MM/DD hh:mm:ss] HTTP method URL Response status
```

■ Source IP address

Source IP address of the HTTP request

■ [YYYY/MM/DD hh:mm:ss]

Reception date and time of the HTTP request

■ HTTP method

Name of the HTTP method

■ URL

URL of the request

■ Response status

Response code returned to the client

3.29.5. Error Log

This log records the information necessary to identify the cause of an error.

The error log is stored in the following directory.

UNIX	/usr/spool/nqs/log/jcres_error.log
Windows	%InstallDirectory%\spool\log\error\jcres_error.log

The directory storing the log can be changed in the configuration file. For details about the configuration file, see [Section 3.29.3, "Configuration File"](#).

Note that the format of the error log is not disclosed.

3.29.6. API

This is the API that can be used for jcres. Issue an HTTP request in the format shown below.

```
http://<IP-address-of-jcres>:<port-number-of-jcres>/<URL-of-API>
```



The result is returned in application/octet-stream format. Depending on the browser in use, you may need to download the relevant file and open it with a text editor.

3.29.6.1. ping

■ Function description

A connection check is performed for jcres.

When jcres is running, the character string "pong" is returned.

■ URL

```
/ping
```

■ Parameters

None

■ HTTP method

GET

3.29.6.2. jc_ping

■ Function description

The jc_ping command performs a connection check from the source site to the destination site.

As the source site, you can specify the local site or cluster site of the machine on which jcres is running. You cannot specify a site of another machine. To specify a site of another machine as the source site, issue the HTTP request of this API to jcres running on the machine in which the destination site resides.

When the jc_ping command is executed successfully, the result is returned in the readable format of the command.

For information about the jc_ping command, see [Section 6.3, "jc_ping \(Checking Communication with a Specified Site\)"](#).

■ URL

- When the source site is the local site

```
/jc_ping?dhost=<destination-site>&af=<address-family>&ctw=<timeout-time>
```

- When the source site is the cluster site

```
/sites/<source-site >/jc_ping?dhost=<destination-site>&af=<address-family>&ctw=<timeout-time>
```

■ Parameters

- Source site

Source site name.

Specify this when the source site is the cluster site. Make sure it is an official site name registered in JobCenter.

- dhost=<destination-site>

Destination site name. This corresponds to \$destination_host of the jc_ping command.

Make sure it is an official site name registered in JobCenter.

- af=<address-family>

Address family used for the connection check. This corresponds to the -p option of the jc_ping command.

For an IPv4 address, specify "ipv4". For an IPv6 address, specify "ipv6". If you omit this option, the connection check is performed using an IPv4 address.

- ctw=<timeout-time>

Timeout time (seconds) for the connection check for each daemon. This corresponds to the -w option of the jc_ping command.

Specify an integer between 0 and 600. If you omit this option, 30 seconds is assumed.

■ HTTP method

GET

3.29.7. Return Values

Return value	Description
0	Normal end
1	Abnormal end

3.29.8. Main Messages

Message	Description
error: Only root user can execute this command.	In the UNIX version, execute the command as the root user.
error: Only JobCenter admin can execute this command.	In the Windows version, execute the command as the JobCenter administrator user.
failed to bind. (Detailed error information)	Binding failed.

Message	Description
failed to stop server. (Detailed error information)	Stopping jgres failed.

3.29.9. Notes

- Execute this command as the root user in the UNIX version or as the JobCenter administrator user in the Windows version.
- IPv6 addresses are not supported.
- Do not run multiple jgres commands at the same time.

3.30. jnwqctrl (Displaying or Deleting a Job Network Submission Reservation List Managed by the Trigger)

■ UNIX version

- Displaying a job network submission reservation list

```
/usr/lib/nqs/gui/tools/jnwqctrl show -a
/usr/lib/nqs/gui/tools/jnwqctrl show -u $user [-t $trg ] [-l $line [-r]]
/usr/lib/nqs/gui/tools/jnwqctrl show -h
```

- Deleting a job network submission reservation list

```
/usr/lib/nqs/gui/tools/jnwqctrl delete -a
/usr/lib/nqs/gui/tools/jnwqctrl delete -u $user [-t $trg ]
/usr/lib/nqs/gui/tools/jnwqctrl delete -h
```

■ Windows version

- Displaying a job network submission reservation list

```
%InstallDirectory%\tools\jnwqctrl show -a
%InstallDirectory%\tools\jnwqctrl show -u $user [-t $trg ] [-l $line [-r]]
%InstallDirectory%\tools\jnwqctrl show -h
```

- Deleting a job network submission reservation list

```
%InstallDirectory%\tools\jnwqctrl delete -a
%InstallDirectory%\tools\jnwqctrl delete -u $user [-t $trg ]
%InstallDirectory%\tools\jnwqctrl delete -h
```

3.30.1. Function Description

This command provides the function to display and delete a job network submission reservation list managed by the trigger function.

The job network submission reservation list manages the submission data of job networks submitted from the trigger function.

When many job networks are submitted from the trigger function (as in a message rush), this command allows you to check the status and cancel the submission of job networks.

3.30.2. Sub Commands

Sub command	Function
show	The submission reservation list managed by the trigger function is displayed.
delete	Part or all of the job network submission data in the submission reservation list managed by the trigger function is deleted.

3.30.3. Options

```
-a, --all
```

The command is targeted at all the job network submission data in the job network submission reservation list.

-u, --user

Of the job network submission data in the job network submission reservation list, the command is targeted at the job network submission data that matches the user name specified in \$user.



You cannot specify this option together with the -a option.

-t, --target

Of the job network submission data in the job network submission reservation list, the command is targeted at the job network submission data that matches the trigger name specified in \$trg.



You cannot specify this option together with the -a option.

-l, --line

The command is targeted at as many job network submission data records as the number specified in \$line, beginning with the oldest one that was added to the job network submission reservation list.

If you specify the -u or -t option, this option applies after the operation target is narrowed down by the -u or -t option.

In \$line, you can specify up to 1000 records.

-r, --reverse

If you specify this option when the -l option is specified, the order of the operation target is changed to descending order of submission reservation (beginning with the one whose submission reservation was made most recently).

-h, --help

The help message of the command is displayed.

3.30.4. Return Values

Return value	Description
0	Normal end
1	Abnormal end

3.30.5. Example of the Execution Result

■ Displaying a job network submission reservation list

- Output format of text log monitoring
 - The output format is as follows.

```
username, trigger name, JNWname, environment variable1, environment variable2,...
```


Environment variables corresponding to environment-variable-1, environment-variable-2, etc. are output in the order shown below.

See details about these environment variables in <Basic Guide>Section 7.2.1.4, "Setting the startup condition".

- NSJNW_TRG_LOG_MATCH_LINE
- NSJNW_TRG_LOG_MATCH_0
- NSJNW_TRG_LOG_MATCH_1
- NSJNW_TRG_LOG_MATCH_2
- NSJNW_TRG_LOG_MATCH_3
- NSJNW_TRG_LOG_MATCH_4
- NSJNW_TRG_LOG_MATCH_5
- NSJNW_TRG_LOG_MATCH_6
- NSJNW_TRG_LOG_MATCH_7
- NSJNW_TRG_LOG_MATCH_8
- NSJNW_TRG_LOG_MATCH_9



When an environment variable is not set, a null character string is displayed.

- An output example is shown below.

```
testuser, log monitoring1, JNW with an error ,2017/03/28 10:00:00 (error) error
message.,2017/03/28 10:00:00,error,,,,,,,,
testuser, log monitoring2, JNW with a warning,2017/03/28 10:00:00 (warn) warning
message.,2017/03/28 10:00:00,warn,,,,,,,,
```

■Deleting a job network submission reservation list

After showing the deletion conditions corresponding to the specified options, the command confirms whether you want to delete the list.



Part of the displayed information is modified. Text enclosed in `/* */` is a comment about the content.

```
Remove the records matching the following conditions:
User = testuser          /* This line is displayed only when -u option is specified */
Target = trgname        /* This line is displayed only when -t option is specified */
All records              /* This line is displayed only when -a option is specified */

remove?(y/n)
```

If you type `y` in response to the deletion confirmation message, the progress of the processing and the result summary are displayed.

The result summary shows the number of job network submission data records deleted for each user and each trigger definition.

```
process: 1000 records, delete: 1000 records
process: 2000 records, delete: 2000 records
process: 3000 records, delete: 3000 records
process: 3967 records, delete: 3967 records
----- delete result -----
user=testuser1
  trg=trgname10, deleted=2910
  trg=trgname11, deleted=792
user=testuser2
  trg=trgname20, deleted=265
```

3.30.6. Notes

- Execute this command as the root user in the UNIX version or as the JobCenter administrator user in the Windows version.
- This command is available only when JobCenter is running.
- When you execute the command for a cluster site, you need to set the environment variable `NQS_SITE` in advance.

3.31. jc_usermgr (Managing the JobCenter users)

■ Windows version


- Adding a user to the JobCenter group

```
%InstallDirectory%\bin\jc_usermgr.exe add -u [%domainname%]\$username -p $password
```

3.31.1. Description of functions

This command provides the function to manage the JobCenter users.

3.31.2. Subcommands

Subcommand	Function
add	<p>A user is added to the JobCenter group.</p> <div style="border: 1px solid blue; padding: 5px;">  <p>To use a user as a JobCenter user, it is required to log into JobCenter or create the user environment by the init subcommand of the jc_usercfg command.</p> <p>For details of the jc_usercfg command, refer to Section 3.32, "jc_usercfg (Creating the user environment, and browsing and changing the user environment setting)".</p> </div>

3.31.3. Options

`-u [%domainname%]\$username`

Specify the user name. For the domain user, specify the domain name as well.



Restrictions on the user name and domain name are described below.

1. Specify both the user name and domain name within 15 bytes.
2. Spaces, tabs, and the following characters cannot be used as the user name.

```
!"#$%&'()*+,-./:;<=>?@[\\]^_{|}~
```

3. Spaces, tabs, and the following characters cannot be used as the domain name.

```
\/:*?"<>|
```

4. When the domain controller is used as the server that executes the command, it is not required to specify the domain name. If the domain name is specified, it will be ignored.

5. The user name and domain name are not case sensitive.

`-p $password`

Specify the password of the user specified by the `-u` option.

The following list shows the characters to be escaped and the characters used for escaping.

Characters to be escaped	Characters used for escaping	Example of command execution
^	^	setuserpwd.exe user1 te^^st^<123^>45

Characters to be escaped	Characters used for escaping	Example of command execution
&		
<		
>		
"	\	setuserpwd.exe user1 test\"123\"45

3.31.4. Return Values

Return Value	Description
0	Normal end
1	Abnormal end

3.31.5. Main Messages

Message	Description
Successfully added <user_name> to the JobCenter group.	A user has been added to the JobCenter group successfully.
WARN: The specified user has already existed in the JobCenter group.	The specified user already exists in the JobCenter group.
ERROR: Too few arguments.	The number of arguments is insufficient.
ERROR: Too many arguments.	The number of arguments is too large.
ERROR: Invalid subcommand (subcommand).	An invalid subcommand is specified.
ERROR: Please input user name.	Enter the user name.
ERROR: Please input password.	Enter the password.
ERROR: User name can't more than 15 bytes.	A value exceeding 15 bytes cannot be specified in the user name.
ERROR: User name can't use space, tab and the characters as shown below. The forbidden characters: !"#%&'()*.,/;<=>?@[\\]^_{ }~	The user name contains a prohibited character.
ERROR: Username is invalid.	The user name is invalid.
ERROR: User does not exist in OS.	The specified user does not exist in OS.
ERROR: Please execute jc_usermgr.exe with the user in Administrators group.	Execute the command with the user having the Administrators account.
ERROR: Failed to get JobCenter Administrator.	Acquisition of the JobCenter administrator user has failed.

3.31.6. Notes

- Execute this command with the user account having the administrator privileges in the OS.
- Select "Run as administrator" from the right-click menu of cmd.exe for starting up, and execute this command from the command line.
- If a user with the name described in "Invalid user name" in <Release Notes>Section 5.2.1, "Notes" is added to the JobCenter group, JobCenter may not work correctly.

- If a domain user is not set up but a local user is set up as the JobCenter administrator, normal operation after adding the domain user to the JobCenter group is not ensured.
- A local user and domain user with the same account name or a domain user and user of another domain with the same account name cannot be added to the JobCenter group at the same time. If users with the same account name are added to the JobCenter group at the same time, JobCenter may not work correctly.

3.32. jc_usercfg (Creating the user environment, and browsing and changing the user environment setting)

■ UNIX version

- Creating the user environment

```
/usr/lib/nqs/gui/bin/jc_usercfg init [-u $username]
```

- Referring to the user environment setting

```
/usr/lib/nqs/gui/bin/jc_usercfg getcfg [-u $username]
```

- Changing the user environment setting

```
/usr/lib/nqs/gui/bin/jc_usercfg setcfg [-u $username] $paramname $paramvalue
```

■ Windows version

- Creating the user environment

```
%InstallDirectory%\bin\jc_usercfg.exe init [-u $username]
```

- Referring to the user environment setting

```
%InstallDirectory%\bin\jc_usercfg.exe getcfg [-u $username]
```

- Changing the user environment setting

```
%InstallDirectory%\bin\jc_usercfg.exe setcfg [-u $username] $paramname $paramvalue
```

3.32.1. Description of functions

This command provides the function to create the user environment at the initial login and the function to refer to and change the user environment settings.

3.32.2. Subcommands

Subcommand	Function
init	Creates the user environment for the initial connection to CL/Win from the target user so that the user can use JobCenter. If the user name is omitted, the command is used for the user who executes the command.
getcfg	Refers to the user environment setting of the target user. If the user name is omitted, the command is used for the user who executes the command.
setcfg	Changes the user environment settings of the target user. If the user name is omitted, the command is used for the user who executes the command.

3.32.3. Options

```
-u $username
```

Specifies the target user name.

Only the JobCenter administrator user can use this option.



There are restrictions on the user name that can be used for JobCenter. Check the details in <Release Notes>Chapter 5, 「Notes and Restrictions」 .

3.32.4. Output Result

The output result of the getcfg subcommand is described below.

```

=====basic config=====
def_queue                guilb_def
error_action             continue
overtime_action          continue
erp_auto_start           on
flow_zoom                100

=====tracker config=====
trk_filter               off
trk_status_filter        waiting,running,terminated
show_skipped_subjnw(branch skipped) off
skipped_waitjnw_status(branch skipped) hide
show_skipped_waitjnw(branch skipped) on
show_arch_trk            on
latest_trk_mode          off
latest_trk_target        all
latest_trk_period        10(day)
disp_date_format         mid(mm/dd hh:mm)
disp_start_time          00:00
hierarchical_mode        auto
trk_disp_period          1(day)
trk_show_hierarchy       on
trk_hierarchical_level   all
trk_display_mode         tree
auto_refresh             on
refresh_interval         30(second)
refresh_after_ctrl       on

=====archive config=====
enable_trk_archive       on
arch_file_name           trkarc
retention_interval       1(day)
disable_cross_month      on
auto_del_arch_file       on
retention_file_num       90
arch_stdout              on
arch_stderr              on
arch_log                 on
arch_env                 on
arch_timing_mode         relative
relative_time            10(minute)
absolute_time            00:00
include_errstop_trk     off

```

3.32.5. Parameter List

The following list shows the parameters that can be referred to and changed in the user environment setting.

Parameter	Description
def_queue	<p>You can set the default value of the queue to which each job that belongs to the job network is submitted.</p> <p>Set the value in the format of <Queue name>@<Machine name>.</p> <p>Restrictions on <Queue name> and <Machine name> are described below.</p> <p>■ Queue name</p> <ul style="list-style-type: none"> ▪ Specify the queue name of up to 15 bytes. ▪ Only alphanumeric characters and underscores can be specified. Do not use a numeric character at the head. <p>■ Machine name</p> <ul style="list-style-type: none"> ▪ Specify the machine name of up to 63 bytes. ▪ A name consisting only of numeric values cannot be used. ▪ Spaces, tabs, linefeed characters, and the following characters are unavailable. <p style="text-align: center;">! " # \$ % & ' () * , / : ; < = + > ? @ [\] ^ ` { } ~</p> <p>"guilb_def" of the own machine is set as the default value when creating the user environment.</p>
error_action	<p>You can set the default operation of the job network to be performed in case of an error.</p> <p>■ stop</p> <p>Stops the job network in case of an error.</p> <p>■ continue</p> <p>Prevents the job network from stopping in case of an error.</p> <p>■ abort</p> <p>Discontinues execution of the job network in case of an error.</p> <p>"continue" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
overtime_action	<p>You can set the operation to be performed if execution of the job network or unit job exceeds the expected exit time.</p> <p>■ continue</p> <p>Continues execution of the job network or unit job.</p>

Parameter	Description
	<p>■ stop</p> <p>Stops execution of the job network or unit job.</p> <p>■ skip</p> <p>Skips execution of the job network or unit job.</p> <p>"continue" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
erp_auto_start	<p>You can set whether the ERP job automatically starts.</p> <p>■ on</p> <p>JobCenter automatically executes the job.</p> <p>■ off</p> <p>Executes the job registration processing for the SAP ERP system when the target ERP job is submitted. However, the ERP job does not start, and the job with [SUBMIT] status waits for the process to be executed.</p> <p>"on" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
flow_zoom	<p>You can set the default value of the display magnification of the job network flow.</p> <p>The range for the display magnification is 10 to 100.</p> <p>"100" is set as the default value when creating the user environment.</p>
trk_filter	<p>You can set whether to use the tracker display condition settings.</p> <p>■ on</p> <p>Uses the display condition settings.</p> <p>■ off</p> <p>Does not use the display condition settings.</p> <p>"off" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
trk_status_filter	<p>You can set the conditions of the tracker status to be displayed as the tracker display condition.</p> <p>Select the condition from those described below. To select multiple conditions, separate them with commas (,).</p> <p>(Example) To select "waiting", "running", and "terminated", specify as follows: "waiting,running,terminated"</p>

Parameter	Description
	<p>■ waiting Displays only the tracker which has not started.</p> <p>■ running Displays only the tracker which has already started.</p> <p>■ terminated Displays only the tracker which has already exited.</p> <p>■ error Displays only the tracker in which an error has occurred.</p> <p>■ warning Displays only the tracker from which a warning is issued.</p> <p>■ stop Displays only the tracker that has stopped due to an error.</p> <p>■ skip Displays only the tracker being skipped.</p> <p>■ hold Displays only the tracker being held.</p> <p>■ wait Displays only the tracker that is waiting.</p> <p>"waiting,running,terminated" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
show_skipped_subjnw	<p>You can set whether to display the sub JNW whose branch has already been skipped as the tracker display condition.</p> <p>■ on Displays the sub JNW whose status is SKIPPED ON BRANCH.</p> <p>■ off Does not display the sub JNW whose status is SKIPPED ON BRANCH.</p> <p>"off" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
skipped_waitjnw_status	<p>You can set whether to judge the actual processing branch (calendar branch) as the tracker display condition for the waiting tracker.</p>

Parameter	Description
	<p>By setting "skipped_waitjnw_status" to "show" and then setting "show_skipped_waitjnw" to "off", the sub JNW not executed by the calendar branch for the waiting tracker is judged and displayed as SKIPPED ON BRANCH.</p> <p>■ show</p> <p>Judges the sub JNW not executed by the calendar branch for the waiting tracker.</p> <p>■ hide</p> <p>Does not judge the sub JNW not executed by the calendar branch for the waiting tracker.</p> <p>"hide" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
show_skipped_waitjnw	<p>You can set whether to display the sub JNW not executed by the calendar branch for the waiting tracker when "skipped_waitjnw_status" is set to "show".</p> <p>■ on</p> <p>Displays the sub JNW not executed by the calendar branch for the waiting tracker.</p> <p>■ off</p> <p>Does not display the sub JNW not executed by the calendar branch for the waiting tracker.</p> <p>"on" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
show_arch_trk	<p>You can set whether to display an archived tracker.</p> <p>■ on</p> <p>Displays an archived tracker.</p> <p>■ off</p> <p>Does not display an archived tracker.</p> <p>"on" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
latest_trk_mode	<p>You can set whether to display the tracker list screen in all tracker display mode or the latest tracker display mode.</p> <p>■ on</p> <p>Displays the screen in the latest tracker display mode.</p> <p>■ off</p>

Parameter	Description
	<p>Displays the screen in all tracker display mode.</p> <p>"off" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
latest_trk_target	<p>You can set the conditions for the tracker displayed in the latest tracker display mode when "latest_trk_mode" is set to "on".</p> <ul style="list-style-type: none"> ■all <p>Searches the latest tracker from all the previous trackers, and displays only the latest tracker.</p> ■period <p>Searches the trackers within the period specified in "latest_trk_period", and displays only the latest tracker.</p> <p>"all" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
latest_trk_period	<p>You can set the period to search the tracker when "latest_trk_target" is set to "period".</p> <p>The range for the period is 1 to 9999.</p> <p>"10" is set as the default value when creating the user environment.</p>
disp_date_format	<p>You can set the default value of the display format of the expected and actual start/exit time of the tracker list to be displayed in text mode.</p> <ul style="list-style-type: none"> ■short <p>Displays the hour and minute in "hh:mm" format.</p> ■mid <p>Displays the month, day, hour, and minute in "MM/DD hh:mm" format.</p> ■long <p>Displays the year, month, day, hour, minute, and second in "YYYY/MM/DD hh:mm:ss" format.</p> <p>"mid" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
disp_start_time	<p>You can set the first time to start displaying the tracker list.</p> <p>You can set the time with the integer value within the range from 0 to 23 (hour).</p> <p>"0" is set as the default value when creating the user environment.</p>
hierarchical_mode	<p>You can set the default value to display the tracker layer of the subjob network automatically.</p>

Parameter	Description
	<p>■select</p> <p>Holds the selected layer.</p> <p>■auto</p> <p>Displays the layer automatically.</p> <p>■non_auto</p> <p>Does not display the layer automatically.</p> <p>"auto" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
trk_disp_period	<p>You can set the default value of the tracker list display period.</p> <p>You can set the display period within the range from 1 to 59 minutes, 1 to 23 hours, 1 to 7 days, 1 to 5 weeks, and 1 to 3 months. Specify the period with "min" for minutes, "h" for hours, "d" for days, "w" for weeks, and "m" for months.</p> <p>(Example) "12h" for 12 hours, "1m" for 1 month</p> <p>"1 (day)" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
trk_show_hierarchy	<p>You can set whether to display the layer of the tracker display format.</p> <p>■on</p> <p>Displays the layer from the root job network at all times.</p> <p>■off</p> <p>Displays only the tracker that matches the specified conditions.</p> <p>"on" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
trk_hierarchical_level	<p>You can set the layer level of the job network to be displayed in tracker display mode.</p> <p>You can set the layer level to "all" and a numeric value from 1 to 40.</p> <p>■all</p> <p>Display all the job networks.</p> <p>■1 to 40</p> <p>Displays the job networks up to the specified layer.</p> <p>"all" is set as the default value when creating the user environment.</p>

Parameter	Description
<p>trk_display_mode</p>	<p>The value is not case sensitive in the setcfg subcommand.</p> <p>You can set the default value of the tracker display mode.</p> <ul style="list-style-type: none"> ■ tree Displays the layers of the trackers in a tree structure. ■ time Rearranges the trackers in ascending order of the expected start time, and displays them. ■ sort Rearranges the trackers in ascending order of the expected start time for each machine or related machine that belongs to the machine group, and displays them. <p>"tree" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
<p>auto_refresh</p>	<p>You can set whether to update the windows related to the tracker or request automatically. The windows are updated at the intervals specified in "refresh_interval" when this value is set to "on".</p> <ul style="list-style-type: none"> ■ on Updates the display at fixed intervals. ■ off Does not update the display automatically. <p>"on" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
<p>refresh_interval</p>	<p>You can set the intervals at which the display is updated when "auto_refresh" is set to "on".</p> <p>You can set the interval within the range from 10 to 3600 seconds or from 1 to 60 minutes. Add "s" to set the value in a second unit, or add "min" to set the value in a minute unit.</p> <p>(Example) "40s" for 40 seconds, "30min" for 30 minutes</p> <p>"30 (second)" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
<p>refresh_after_ctrl</p>	<p>You can set whether to update all the trackers on the tracker list window after operating the tracker.</p> <ul style="list-style-type: none"> ■ on Updates all the trackers on the tracker list window after operating the tracker.

Parameter	Description
	<p>■off</p> <p>Does not update all the trackers on the tracker list window after operating the tracker.</p> <p>"on" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
enable_trk_archive	<p>You can set whether to save the tracker information in the tracker archive file.</p> <p>■on</p> <p>Saves the tracker information.</p> <p>■off</p> <p>Does not save the tracker information. (The tracker is deleted after the tracker is archived.)</p> <p>"on" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
arch_file_name	<p>You can set the name of the tracker archive file where the tracker information is saved.</p> <p>The tracker archive file name has the following restrictions.</p> <p>■Use of two-byte characters and the following characters is prohibited.</p> <p>"+", " ", "#", ":", ";", "*", "?", " ", "<", ">", " "</p> <p>■Specify the file name of up to 192 bytes.</p> <p>■Please note that the full path name that leads to the archive file must be shorter than the path length for each OS.</p> <p>The length of the full path name that leads to the archive file is described below.</p> <p>length of JobCenter installation directory or site database path + length of the archive file name + 53 bytes</p> <p>"trkarc" is set as the default value when creating the user environment.</p>
retention_interval	<p>You can set the intervals at which the archive file is saved.</p> <p>You can set the saving interval within the range from 1 to 365 days, 1 to 53 weeks, or 1 to 12 months. Specify the value with "d" for days, "w" for weeks, and "m" for months.</p> <p>(Example) "10d" for 10 days, "1m" for 1 month</p> <p>It is required to set "retention_interval" and "retention_file_num" so that the sum of these values will not exceed one year.</p>

Parameter	Description
	<p>"1 (day)" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
disable_cross_month	<p>You can set whether the archive file saving interval extends to the following month when the unit of the saving interval is set to "w" (week).</p> <p>■on</p> <p>Does not extend to the following month.</p> <p>■off</p> <p>Extends to the following month.</p> <p>"on" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
auto_del_arch_file	<p>You can set whether to delete the archived file automatically when the saving period has expired.</p> <p>■on</p> <p>Deletes the file automatically.</p> <p>■off</p> <p>Does not delete the file automatically.</p> <p>"on" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
retention_file_num	<p>You can set the number of expired archived files to be saved.</p> <p>The range for the number of files is 0 to 90.</p> <p>"90" is set as the default value when creating the user environment.</p> <p>It is required to set "retention_interval" and "retention_file_num" so that the sum of these values does not exceed one year.</p>
arch_stdout	<p>You can set whether to save the contents of the standard output for each job in the tracker archive file.</p> <p>■on</p> <p>Saves the contents.</p> <p>■off</p> <p>Does not save the contents.</p> <p>"on" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
arch_stderr	<p>You can set whether to save the contents of the standard error output for each job in the tracker archive file.</p>

Parameter	Description
	<p>■on</p> <p>Saves the contents.</p> <p>■off</p> <p>Does not save the contents.</p> <p>"on" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
arch_log	<p>You can set whether to save the log of the job network execution in the tracker archive file.</p> <p>■on</p> <p>Saves the log.</p> <p>■off</p> <p>Does not save the log.</p> <p>"on" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
arch_env	<p>You can set whether to save the environment variable of the job network execution in the tracker archive file.</p> <p>■on</p> <p>Saves the environment variable.</p> <p>■off</p> <p>Does not save the environment variable.</p> <p>"on" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
arch_timing_mode	<p>You can set the timing of deleting or archiving the tracker.</p> <p>■relative</p> <p>Deletes or archives the tracker when the time specified as the waiting time has elapsed after completion of job network execution.</p> <p>■absolute</p> <p>Deletes or archives the tracker at the next specified time after completion of job network execution.</p> <p>"relative" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>

Parameter	Description
relative_time	<p>You can set the waiting time interval from completion of job network execution to the deletion or archiving of the tracker when "arch_timing_mode" is set to "relative".</p> <p>You can set the waiting time within the range from 0 to 2678400 seconds, 0 to 44640 minutes, 0 to 744 hours, or 0 to 31 days. Add "s" to specify the value in a second unit, "min" in a minute unit, "h" in a hour unit, or "d" in a day unit.</p> <p>(Example) "5min" for 5 minutes, "1d" for 1 day</p> <p>"10 (minute)" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
absolute_time	<p>You can set the time to delete or archive the tracker when "arch_timing_mode" is set to "absolute".</p> <p>The range for the time is 00:00 to 23:59.</p> <p>"00:00" is set as the default value when creating the user environment.</p>
include_errstop_trk	<p>You can set whether to delete or archive a tracker that has stopped due to an error according to the timing of deleting/archiving a tracker.</p> <ul style="list-style-type: none"> ■ on <p>Deletes or archives a tracker that has stopped due to an error according to the timing of deleting/archiving a tracker.</p> ■ off <p>Does not delete or archive a tracker that has stopped due to an error according to the timing of deleting/archiving a tracker.</p> <p>"off" is set as the default value when creating the user environment.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>

3.32.6. Return Values

Return Value	Description
0	Normal end
1	Abnormal end

3.32.7. Main Messages

- Messages shared by commands

Message	Description
ERROR: Too few arguments.	The number of arguments is insufficient.
ERROR: Invalid subcommand(\$subcommand).	An invalid subcommand is specified.
ERROR: It is only JobCenter manager that can use '-u' option.	A user other than the JobCenter administrator user cannot use the -u option.

Message	Description
ERROR: User name can't more than 15 bytes.	The length of a user name exceeds the limit. The user name cannot exceed 15 bytes.
ERROR: Invalid character.	The user name contains an invalid character.
ERROR: Invalid site.	There is a problem with the site name specified in the environment variable NQS_SITE.

- Message specific to the init subcommand

Message	Description
Successfully initialized \$username.	Creation of the user environment has succeeded.
WARN: User is already initialized.	The user environment of the target user has already been created.
ERROR: User does not exist (user=\$username).	The target user does not exist.

- Message specific to the getcfg subcommand

Message	Description
ERROR: You are not initialized user.	The user environment of the user who executed the command has not been created yet.
ERROR: User does not exist (user=\$username).	The target user does not exist.

- Message specific to the setcfg subcommand

Message	Description
ERROR: You are not initialized user.	The user environment of the user who executed the command has not been created yet.
ERROR: User does not exist (user=\$username).	The target user does not exist.
ERROR: Set userconfig failure("No "\$paramname" key.")	The specified parameter name is incorrect.
ERROR: Set userconfig failure("Invalid parameter value.")	The specified parameter value is invalid.
ERROR: Get lock failure.	Another user is editing the user environment setting of the target user, and setting of the user environment has failed.

3.32.8. Notes

■ When executing the command for the cluster site, it is required to set the environment variable NQS_SITE in advance.

For details, refer to the notes in [Chapter 1, 「Command List」](#) .

■ For the getcfg and setcfg subcommands, both the execution user of the command itself and the target user must be the user who created the user environment. Create the user environment by logging into JobCenter and executing the init subcommand.

■ For the Windows version, the user environment for the JobCenter administrator user is not created during the installation. Therefore, it is required to log into JobCenter and create the user environment by the init subcommand before using the getcfg and setcfg subcommands.

■ Check the details of the parameters that can be set by the setcfg subcommand in <Environment Guide>Section 6.2, "Changing User Environment Settings".

- Please note that you cannot check whether the specified queue name exists when setting "def_queue" in the setcfg subcommand.

3.33. jc_syscfg (Browsing and changing the system environment setting)

■ UNIX version

- Displaying the system environment setting

```
/usr/lib/nqs/gui/bin/jc_syscfg getcfg [ -p $paramname | -c $category ]
```

- Changing the system environment setting

```
/usr/lib/nqs/gui/bin/jc_syscfg setcfg $paramname $paramvalue
```

■ Windows version

- Displaying the system environment setting

```
%InstallDirectory%\bin\jc_syscfg.exe getcfg [ -p $paramname | -c $category ]
```

- Changing the system environment setting

```
%InstallDirectory%\bin\jc_syscfg.exe setcfg $paramname $paramvalue
```

3.33.1. Description of functions

This command provides the function to display and change the system environment setting.

3.33.2. Subcommands

Subcommand	Function
getcfg	Refers to the system environment setting.
setcfg	Changes the system environment setting specified by \$paramname to the value specified by \$paramvalue.

3.33.3. Options

-p \$paramname	Specify the parameter name of the system environment setting to be displayed. If this option is omitted, all the system environment settings are the target. For the parameters that can be specified, refer to Section 3.33.5, "Parameter List" .
-c \$category	Specify the category of the system environment setting to be displayed. If this option is omitted, all the system environment settings are the target. For the categories that can be specified, refer to Section 3.33.5, "Parameter List" .

3.33.4. Output Result

The output result of the getcfg subcommand is described below.

```
=====time config=====
time.customized      off
time.mode            all
time.shift           00:00
time.zone            off
```

```

=====op&exec log=====
op.apply                off
op.period               1(year)
op.baknum               0
op.size                 1(mb)
exec.apply              off
exec.period             1(year)
exec.baknum             0
exec.size               1(mb)

=====smtp server=====
smtp.mode               each
smtp.hostname
smtp.port               25
smtp.auth               off
smtp.auth.cram.md5     off
smtp.auth.login        off
smtp.auth.plain        off
smtp.userid
smtp.password
    
```

3.33.5. Parameter List

The following list shows the parameters and categories that can be displayed and changed in the system environment setting.

Category	Parameter	Description
time	time.customized	<p>Indicates whether to use the corrected time.</p> <ul style="list-style-type: none"> ■ on Uses the corrected time. ■ off Does not use the corrected time. <p>The value is not case sensitive in the setcfg subcommand.</p>
	time.mode	<p>Indicates how to set the corrected time.</p> <ul style="list-style-type: none"> ■ all Applies the corrected time to the entire site. ■ each You can set the corrected time for each job network. <p>When the time.mode parameter is set to "each", the time.shift parameter value changes to "00:00".</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
	time.shift	<p>Indicates the corrected time when the corrected time is applied to the entire site.</p> <p>The corrected time can be set when the time.mode parameter is set to "all" within the range from -23:59 to 23:59 as the relative time setting starting from 00:00.</p>

Category	Parameter	Description
	time.zone	<p>Indicates whether to use the time zone setting function in the calendar.</p> <p>■on</p> <p>Uses the time zone setting function in the calendar.</p> <p>■off</p> <p>Does not use the time zone setting function in the calendar.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
log	op.apply	<p>Indicates whether to output the operation log.</p> <p>■on</p> <p>Outputs the operation log.</p> <p>■off</p> <p>Does not output the operation log.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
	op.period	<p>Indicates the operation log retention period.</p> <p>The retention period can be specified within the range from 1 to 365 days or from 1 to 5 years. Add "d" to set the value in a day unit, or add "y" to set the value in a year unit.</p> <p>(Example) "30d" for 30 days, "5y" for 5 years</p> <p>Displays output of the getcfg subcommand as "1 (year)" when the retention period is set to "365d."</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
	op.baknum	<p>Indicates the number of backup files of the operation log.</p> <p>You can specify the number of backup files within the range from 0 to 9.</p>
	op.size	<p>Indicates the maximum size per file of the operation log.</p> <p>The maximum size per file can be specified within the range from 1 to 1024KB or from 1 to 1024MB. Add "kb" to set the value in a KB unit, or add "mb" to set the value in a MB unit.</p> <p>(Example) "100kb" for 100KB, "512mb" for 512MB</p> <p>Displays output of the getcfg subcommand as "1 (mb)" when the maximum size per file is set to "1024kb."</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
	exec.apply	<p>Indicates whether to output the execution log.</p> <p>■on</p> <p>Outputs the execution log.</p>

Category	Parameter	Description
		<p>■off</p> <p>Does not output the execution log.</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
	exec.period	<p>Indicates the execution log saving period.</p> <p>The saving period can be specified within the range from 1 to 365 days or from 1 to 5 years. Add "d" to set the value in a day unit, or add "y" to set the value in a year unit.</p> <p>(Example) "30d" for 30 days, "5y" for 5 years</p> <p>Displays output of the getcfg subcommand as "1 (year)" when the saving period is set to "365d."</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
	exec.baknum	<p>Indicates the number of backup files of the execution log.</p> <p>You can specify the number of backup files within the range from 0 to 9.</p>
	exec.size	<p>Indicates the maximum size per file of the execution log.</p> <p>The maximum size per file can be specified within the range from 1 to 1024KB or from 1 to 1024MB. Add "kb" to set the value in a KB unit, or add "mb" to set the value in a MB unit.</p> <p>(Example) "100kb" for 100KB, "512mb" for 512MB</p> <p>Displays output of the getcfg subcommand as "1 (mb)" when the maximum size per file is set to "1024kb."</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
smtp	smtp.mode	<p>Indicates how to specify the SMTP server.</p> <p>■all</p> <p>Applies the SMTP server to the entire site.</p> <p>■each</p> <p>You can set the SMTP server for each job network.</p> <p>It is required to set the smtp.host name and smtp.port parameters when the smtp.mode parameter is set to "all".</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
	smtp.hostname	<p>Indicates the SMTP server name.</p> <p>Set the host name or IP address of the SMTP server to set the SMTP server name. Specify the SMTP server name of up to 256 bytes. Spaces, tabs, one-byte Katakana, two-byte characters, !, ", #, \$, %, &, ', (,), *, ,(comma), /, ;, <, =, >, ?, @, [, \,], ^, `, {, , }, and ~ cannot be used.</p>
	smtp.port	<p>Indicates the SMTP server port number.</p>

Category	Parameter	Description
		You can specify the port number from 1 to 65535.
	smtp.auth	<p>Indicates whether to use the SMTP user authentication function.</p> <p>■ on</p> <p>Uses the SMTP user authentication function.</p> <p>■ off</p> <p>Does not use the SMTP user authentication function.</p> <p>It is required to set the smtp.userid and smtp.password parameters and to set any of the smtp.auth.cram.md5 parameter, the smtp.auth.login parameter, or the smtp.auth.plain parameter to "on" when setting the smtp.auth parameter to "on".</p> <p>The value is not case sensitive in the setcfg subcommand.</p>
	smtp.auth.cram.md5	<p>Indicates whether to use CRAM-MD5 as the user authentication method of the SMTP user authentication function.</p> <p>■ on</p> <p>Uses CRAM-MD5 as the user authentication method.</p> <p>■ off</p> <p>Does not use CRAM-MD5 as the user authentication method.</p> <p>It is required to set either the smtp.auth.login parameter or the smtp.auth.plain parameter to "on" when setting the smtp.auth.cram.md5 parameter to "off" while the smtp.auth parameter is "on".</p> <p>The value is not case sensitive in the setcfg subcommand.</p> <p>When more than one user authentication method has been specified, authentication to the SMTP server is executed in the order of CRAM-MD5, LOGIN, and PLAIN.</p>
	smtp.auth.login	<p>Indicates whether to use LOGIN as the user authentication method of the SMTP user authentication function.</p> <p>■ on</p> <p>Uses LOGIN as the user authentication method.</p> <p>■ off</p> <p>Does not use LOGIN as the user authentication method.</p> <p>It is required to set either the smtp.auth.cram.md5 parameter or the smtp.auth.plain parameter to "on" when setting the smtp.auth.login parameter to "off" while the smtp.auth parameter is "on".</p> <p>The value is not case sensitive in the setcfg subcommand.</p>

Category	Parameter	Description
		When more than one user authentication method has been specified, authentication to the SMTP server is executed in the order of CRAM-MD5, LOGIN, and PLAIN.
	smtp.auth.plain	<p>Indicates whether to use PLAIN as the user authentication method of the SMTP user authentication function.</p> <p>■ on</p> <p>Uses PLAIN as the user authentication method.</p> <p>■ off</p> <p>Does not use PLAIN as the user authentication method.</p> <p>It is required to set either the smtp.auth.cram.md5 parameter or the smtp.auth.login parameter to "on" when setting the smtp.auth.plain parameter to "off" while the smtp.auth parameter is "on".</p> <p>The value is not case sensitive in the setcfg subcommand.</p> <p>When more than one user authentication method has been specified, authentication to the SMTP server is executed in the order of CRAM-MD5, LOGIN, and PLAIN.</p>
	smtp.userid	<p>Indicates the user ID used for the SMTP user authentication function.</p> <p>Specify the user ID of up to 255 bytes. Tabs, one-byte Katakana, two-byte characters, special characters, ", #, and ' cannot be used.</p>
	smtp.password	<p>Displays the password corresponding to the user ID used for the SMTP user authentication function. The masked value is displayed if the password is enabled.</p> <p>Specify the password of up to 64 bytes.</p>

3.33.6. Return Values

Return Value	Description
0	Normal end
1	Abnormal end

3.33.7. Main Messages

■ Messages common to the commands

Message	Description
ERROR: Too few arguments.	The number of command arguments is insufficient.
ERROR: Too many arguments.	The number of command arguments is excessive.
ERROR: Invalid subcommand (subcommand).	An invalid subcommand is specified.
ERROR: This user is not registered.	The user who is executing the command does not belong to the JobCenter group.
ERROR: User does not exist.	The user environment of the user who is executing the command has not been created yet.

Message	Description
ERROR: Invalid site.	The site specified by the environment variable NQS_SITE is invalid.

■ Message specific to the getcfg subcommand

Message	Description
ERROR: The specified parameter not found.	The specified parameter does not exist.
ERROR: The specified category not found.	The specified category does not exist.
ERROR: Paramname can't more than 40 bytes.	A value exceeding 40 bytes cannot be specified in the parameter name.
ERROR: Get sysconfig failure ("Error message")	Acquisition of the system environment settings has failed.

■ Message specific to the setcfg subcommand

Message	Description
Successfully set system config.	Setting of the system environment has succeeded.
ERROR: Set system config failure ("No "Parameter name" key.")	The specified parameter does not exist, and setting of the system environment has failed.
ERROR: Set system config failure("Invalid parameter value.")	The specified parameter value is invalid, and setting of the system environment has failed.
ERROR: Set system config failure("The required paramaters aren't proper.")	The value of the parameter that has dependency is invalid, and setting of the system environment has failed.
ERROR: Set system config failure("Can't set the time.shift paramater when the time.mode is "each".")	The time.mode parameter is set to "each", and setting of the time.shift parameter has failed.
ERROR: Set system config failure ("Error message")	Setting of the system environment has failed.
ERROR: Only manager can execute this command.	Only the JobCenter administrator user can execute this subcommand.
ERROR: Get lock failure.	The system environment setting is being edited, and setting has failed.

3.33.8. Notes

- When executing the command for the cluster site, it is required to set the environment variable NQS_SITE in advance.

For details, refer to the notes in [Chapter 1, 「Command List」](#) .

- The user who logged into JobCenter and created the user environment with the init subcommand of the jc_usercfg command needs to execute this command.

For details of the jc_usercfg command, refer to [Section 3.32, "jc_usercfg \(Creating the user environment, and browsing and changing the user environment setting\)"](#).

- A JobCenter administrator user needs to execute the setcfg subcommand.

3.34. jc_permcfg (Browsing and Changing the Permission Settings)

■ UNIX version

```
/usr/lib/nqs/gui/bin/jc_permcfg $subcommand $options...
```

■ Windows version

```
%InstallDirectory%\bin\jc_permcfg.exe $subcommand $options...
```

3.34.1. Description of functions

This command provides the function to display and change the permission setting.

The subcommands of this command are listed below.

Subcommand	Function
getperm	Outputs the permission information of the authority group.
creategrp	Newly creates an authority group.
delgrp	Deletes an authority group.
copygrp	Copies an authority group.
setperm	Sets the access authority of an authority group.
assignuser	Changes the authority group to which the user belongs.
addaccuser	Adds permissions for the authority group to access other users.
delaccuser	Deletes permissions to access other users from an authority group.

The following sections describe how to use each subcommand and option.

■ Default authority groups

This command inputs and outputs the default authority group with the following name. (The group name is displayed on the CL/Win window with an asterisk "*" at the tail. In this command, the group name is input and output as the character string that starts with "@".)

Authority group name	Name that is displayed and input for this command
JobCenter administrator	@JobCenterManager
Job network developer	@JnwDeveloper
Job network operator	@JnwOperator
Execution surveillant	@ExecSurveillant
General user	@NormalUser



Restrictions on the authority group name are listed below.

- The maximum number of bytes of the authority group name is 40 bytes.
- Spaces, tabs, special characters, and the following characters cannot be used.
! " # \$ % ' () * , . / : ; < = > ? @ [\] ^ ` { | } ~
- Except for specifying the default authority group, at marks "@" cannot be used for the authority group name.

3.34.1.1. Displaying the authority group permission information (getperm)

■ Command line interface

```
jc_permcfg getperm [-g $groupname]
```

■ Overview

Outputs the permission information of the authority group.

■ Options

```
-g $groupname
```

Specify the name of the authority group whose permission information you want to output.

When this option is omitted, the permission information of all authority groups is displayed.

■ Output format

Describes the output format of the permission information of an authority group.

```
<Authorization_group_name>
  Member
  <user_list>
  Authority
  <access_permission_list>
  Permitted User
  <permitted_user_list>
```

Section	Description																														
Member	Displays the list of names of the users who belong to the authority group, separated by comma.																														
Authority	<p>Displays whether the access privilege exists in the format of "Privilege indicator string <on/off>".</p> <p>The privilege indicator strings and corresponding access privileges are listed below.</p> <table border="1"> <thead> <tr> <th>Privilege string</th> <th>indicator</th> <th>Access privilege</th> </tr> </thead> <tbody> <tr> <td>own.jnw.read</td> <td></td> <td>Listing and referring to the job networks of the user him/herself</td> </tr> <tr> <td>own.jnw.write</td> <td></td> <td>Creating/changing/deleting the job networks of the user him/herself</td> </tr> <tr> <td>own.jnw.execute</td> <td></td> <td>Executing the job networks of the user him/herself</td> </tr> <tr> <td>own.trk.read</td> <td></td> <td>Listing and displaying the trackers of the user him/herself</td> </tr> <tr> <td>own.trk.operation</td> <td></td> <td>Operating the trackers of the user him/herself</td> </tr> <tr> <td>other.jnw.read</td> <td></td> <td>Listing and displaying the job networks of other users</td> </tr> <tr> <td>other.jnw.write</td> <td></td> <td>Creating/changing/deleting the job networks of other users</td> </tr> <tr> <td>other.jnw.execute</td> <td></td> <td>Executing the job networks of other users</td> </tr> <tr> <td>other.trk.read</td> <td></td> <td>Listing and displaying the trackers of other users</td> </tr> </tbody> </table>	Privilege string	indicator	Access privilege	own.jnw.read		Listing and referring to the job networks of the user him/herself	own.jnw.write		Creating/changing/deleting the job networks of the user him/herself	own.jnw.execute		Executing the job networks of the user him/herself	own.trk.read		Listing and displaying the trackers of the user him/herself	own.trk.operation		Operating the trackers of the user him/herself	other.jnw.read		Listing and displaying the job networks of other users	other.jnw.write		Creating/changing/deleting the job networks of other users	other.jnw.execute		Executing the job networks of other users	other.trk.read		Listing and displaying the trackers of other users
Privilege string	indicator	Access privilege																													
own.jnw.read		Listing and referring to the job networks of the user him/herself																													
own.jnw.write		Creating/changing/deleting the job networks of the user him/herself																													
own.jnw.execute		Executing the job networks of the user him/herself																													
own.trk.read		Listing and displaying the trackers of the user him/herself																													
own.trk.operation		Operating the trackers of the user him/herself																													
other.jnw.read		Listing and displaying the job networks of other users																													
other.jnw.write		Creating/changing/deleting the job networks of other users																													
other.jnw.execute		Executing the job networks of other users																													
other.trk.read		Listing and displaying the trackers of other users																													

Section	Description	
	Privilege string	indicator Access privilege
	other.trk.operation	Operating the trackers of other users
Permitted User	Displays the list of names of the users for which access is permitted from the authority group in a comma-separated manner.	

■ Example of execution

An example of execution in the UNIX version is shown below.

- Outputting the information of the specified authority group

```
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g <a user group that monitors other users>
<user group that monitors other users>
Member
  jnwmonitor
Authority
  own.jnw.read      off
  own.jnw.write     off
  own.jnw.execute   off
  own.trk.read      off
  own.trk.operation off
  other.jnw.read    on
  other.jnw.write   on
  other.jnw.execute on
  other.trk.read    on
  other.trk.operation on
Permitted User
  jnwmonitor,nsumsmgr,root
```

- Outputting the information of all the authority groups

```
# /usr/lib/nqs/gui/bin/jc_permcfg getperm
@JobCenterManager
Member
  nsumsmgr,root
Authority
  own.jnw.read      on
  own.jnw.write     on
  own.jnw.execute   on
  own.trk.read      on
  own.trk.operation on
  other.jnw.read    on
  other.jnw.write   on
  other.jnw.execute on
  other.trk.read    on
  other.trk.operation on
Permitted User
  jnwmonitor,nsumsmgr,root

@JnwDeveloper
Member

Authority
  own.jnw.read      on
```

```

own.jnw.write      on
own.jnw.execute    on
own.trk.read       on
own.trk.operation  on
other.jnw.read     on
other.jnw.write    on
other.jnw.execute  on
other.trk.read     on
other.trk.operation on
Permitted User
  jnwmonitor,nsumsmgr,root

@JnwOperator
Member

Authority
  own.jnw.read      on
  own.jnw.write     off
  own.jnw.execute   on
  own.trk.read      on
  own.trk.operation on
  other.jnw.read    on
  other.jnw.write   off
  other.jnw.execute on
  other.trk.read    on
  other.trk.operation on
Permitted User
  jnwmonitor,nsumsmgr,root

@ExecSurveillant
Member

Authority
  own.jnw.read      off
  own.jnw.write     off
  own.jnw.execute   off
  own.trk.read      on
  own.trk.operation off
  other.jnw.read    off
  other.jnw.write   off
  other.jnw.execute off
  other.trk.read    on
  other.trk.operation off
Permitted User
  jnwmonitor,nsumsmgr,root

<a user group that monitors other users>
Member
  jnwmonitor
Authority
  own.jnw.read      off
  own.jnw.write     off
  own.jnw.execute   off
  own.trk.read      off
  own.trk.operation off
  other.jnw.read    on

```

```

other.jnw.write      on
other.jnw.execute    on
other.trk.read       on
other.trk.operation  on
Permitted User
jnwmonitor,nsumsmgr,root

@NormalUser
Member

Authority
own.jnw.read         on
own.jnw.write        on
own.jnw.execute      on
own.trk.read         on
own.trk.operation    on
other.jnw.read       off
other.jnw.write      off
other.jnw.execute    off
other.trk.read       off
other.trk.operation  off
Permitted User

```

■ Main Messages

Refer to [Section 3.34.3, “Main messages shared by commands”](#).

3.34.1.2. Creating a authority group (creategrp)

■ Command line interface

```
jc_permcfg creategrp $groupname
```

■ Overview

Newly creates an authority group.

All the privileges of the authority group created by this subcommand are disabled. Set the privileges according to [Section 3.34.1.5, “Changing the access permissions of the authority group \(setperm\)”](#) and [Section 3.34.1.7, “Adding permissions for the authority group to access other users \(addaccuser\)”](#).

■ Options

```
$groupname
```

Specify the name of the authority group to be newly created.



The same name as an existing authority group cannot be specified.

■ Example of execution

An example of execution in the UNIX version is shown below.

```
# /usr/lib/nqs/gui/bin/jc_permcfg creategrp <a new authority group>
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g <a new authority group>
<a new authority group>
```



```
Member

Authority
  own.jnw.read      off
  own.jnw.write     off
  own.jnw.execute   off
  own.trk.read      off
  own.trk.operation off
  other.jnw.read    off
  other.jnw.write   off
  other.jnw.execute off
  other.trk.read    off
  other.trk.operation off
Permitted User
```

■ Main Messages

Describes the message specific to this subcommand. For other messages, refer to [Section 3.34.3, "Main messages shared by commands"](#).

Message	Description
Successfully set the permission.	Creation of an authority group has succeeded.
ERROR: The specified group already exists.	An authority group of the specified name already exists.

3.34.1.3. Deleting the authority group (delgrp)

■ Command line interface

```
jc_permcfg delgrp $groupname
```


■ Overview

Deletes the authority group.

■ Options

```
$groupname

Specify the name of the authority group to be deleted.
```


The default authority group cannot be deleted.

■ Example of execution

An example of execution in the UNIX version is shown below.

```
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g <a user group that monitors other users>
<a user group that monitors other users>
Member
  jnwmonitor
Authority
  own.jnw.read      off
  own.jnw.write     off
  own.jnw.execute   off
  own.trk.read      off
```

```

own.trk.operation    off
other.jnw.read      on
other.jnw.write     on
other.jnw.execute   on
other.trk.read      on
other.trk.operation on
Permitted User
  jnwmonitor,nsumsmgr,root
# /usr/lib/nqs/gui/bin/jc_permcfg delgrp <a user group that monitors other users>
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g <a user group that monitors other users>
ERROR: The specified group could not be found.

```

■ Main Messages

Describes the message specific to this subcommand. For other messages, refer to [Section 3.34.3, "Main messages shared by commands"](#).

Message	Description
Successfully set the permission.	Deletion of the authority group has succeeded.
ERROR: Can't specify the group name to be a built-in group.	The default authority group cannot be deleted.

3.34.1.4. Copying the authority group (copygrp)

■ Command line interface

```
jc_permcfg copygrp $srcgroup $dstgroup
```

■ Overview

Creates an authority group that has the same contents as an existing authority group.

■ Options

\$srcgroup

Specify the name of the authority group to be the copy source.

\$dstgroup

Specify the name of the authority group to be the copy destination.



The JobCenter administrator group "@JobCenterManager" cannot be specified in \$srcgroup.
The same name as that of an existing authority group cannot be specified in \$dstgroup.

■ Example of execution

An example of execution in the UNIX version is shown below.

```

# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g <a user group that monitors other users>
<a user group that monitors other users>
Member
  jnwmonitor
Authority
  own.jnw.read      off
  own.jnw.write     off

```

```

own.jnw.execute    off
own.trk.read       off
own.trk.operation  off
other.jnw.read     on
other.jnw.write    on
other.jnw.execute  on
other.trk.read     on
other.trk.operation on
Permitted User
  jnwmonitor,nsumsmgr,root
# /usr/lib/nqs/gui/bin/jc_permcfg copygrp <a user group that monitors other users> <a copy group>
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g <a copy group>
<a copy group>
Member

Authority
  own.jnw.read      off
  own.jnw.write     off
  own.jnw.execute   off
  own.trk.read      off
  own.trk.operation off
  other.jnw.read    on
  other.jnw.write   on
  other.jnw.execute on
  other.trk.read    on
  other.trk.operation on
Permitted User
  jnwmonitor,nsumsmgr,root
    
```

■ Main Messages

Describes the message specific to this subcommand. For other messages, refer to [Section 3.34.3, “Main messages shared by commands”](#).

Message	Description
Successfully set the permission.	Copying of the authority group has succeeded.
ERROR: The specified srcgroup could not be found.	An authority group of the name specified as the copy source does not exist.
ERROR: The specified dstgroup already exists.	An authority group of the name specified as the copy destination already exists.
ERROR: Can't copy from JobCenter manager group.	The JobCenter administrator group cannot be copied.

3.34.1.5. Changing the access permissions of the authority group (setperm)

■ Command line interface

```
jc_permcfg setperm $groupname $mode
```

■ Overview

Changes the access permissions of the authority group.

■ Options

```
$groupname
```

Specify the name of the authority group whose access permissions you want to change.

\$mode

Specify the access permissions with character strings. For how to specify the permissions, refer to "[How to specify the access permissions](#)".



The access permissions of the default authority group cannot be changed.

■ How to specify the access permissions


Specify the permission indicator string of the access permissions to be assigned to the target authority group. To specify more than one permission indicator string, separate them with commas (,). The permission indicator strings that can be specified are listed below.


Permission indicator string	Access permission name
own.jnw.read	Listing and displaying the job network of the user him/herself
own.jnw.write	Creating/changing/deleting the job network of the user him/herself
own.jnw.execute	Executing the job networks of the user him/herself
own.trk.read	Listing and displaying the trackers of the user him/herself
own.trk.operation	Operating the trackers of the user him/herself
other.jnw.read	Listing and displaying the job networks of other users
other.jnw.write	Creating/changing/deleting the job networks of other users
other.jnw.execute	Executing the job networks of other users
other.trk.read	Listing and displaying the trackers of other users
other.trk.operation	Operating the trackers of other users

Some access permissions have dependency on other permissions from which the former are derived. The latter serve as the source of permissions and are required to set the former. When setting access permissions by using this command, the source access permissions are automatically set even if they are not specified explicitly.

Permission indicator string	Permission indicator string of the access permission (given at the same time) that serves as the source of the permission shown in the left
own.jnw.write	own.jnw.read
own.jnw.execute	own.trk.read own.trk.operation
own.trk.operation	own.trk.read
own.jnw.write	own.jnw.read
other.jnw.execute	other.trk.read other.trk.operation
other.trk.operation	other.trk.read

Use of "all" in the permission indicator string assigns more than one access permissions at the same time.

Permission indicator string	Permission indicator string of the access permission to be given
all	own.jnw.read own.jnw.write own.jnw.execute own.trk.read own.trk.operation other.jnw.read other.jnw.write other.jnw.execute other.trk.read other.trk.operation
own.all	own.jnw.read own.jnw.write own.jnw.execute own.trk.read own.trk.operation
own.jnw.all	own.jnw.read own.jnw.write own.jnw.execute own.trk.read own.trk.operation <div data-bbox="703 1417 1476 1547" style="border: 1px solid blue; padding: 5px;">  The access permissions relevant to own.trk are also given at the same time due to the dependencies. </div>
own.trk.all	own.trk.read own.trk.operation
other.all	other.jnw.read other.jnw.write other.jnw.execute other.trk.read other.trk.operation
other.jnw.all	other.jnw.read

Permission indicator string	Permission indicator string of the access permission to be given
	other.jnw.write other.jnw.execute other.trk.read other.trk.operation <div style="border: 1px solid blue; padding: 5px; margin-top: 10px;">  The access permissions relevant to other.trk are also given at the same time due to the dependencies. </div>
other.trk.all	other.trk.read other.trk.operation

If a null character " " is specified in the permission indicator string, all the access permissions of the authority group become invalid.

■ Example of execution

An example of execution in the UNIX version is shown below.

```
# /usr/lib/nqs/gui/bin/jc_permcfg creategrp <a new authority group>
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg setperm <a new authority group> own.all,other.trk.all
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g <a new authority group>
<a new authority group>
Member

Authority
  own.jnw.read      on
  own.jnw.write     on
  own.jnw.execute   on
  own.trk.read      on
  own.trk.operation on
  other.jnw.read    off
  other.jnw.write   off
  other.jnw.execute off
  other.trk.read    on
  other.trk.operation on
Permitted User
  jnwmonitor,nsumsmgr,root
# /usr/lib/nqs/gui/bin/jc_permcfg setperm <a new authority group> ""
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g <a new authority group>
<a new authority group>
Member

Authority
  own.jnw.read      off
  own.jnw.write     off
  own.jnw.execute   off
```

```

own.trk.read      off
own.trk.operation off
other.jnw.read    off
other.jnw.write   off
other.jnw.execute off
other.trk.read    off
other.trk.operation off
Permitted User
    
```

■ Main Messages

Describes the message specific to this subcommand. For other messages, refer to [Section 3.34.3, "Main messages shared by commands"](#).

Message	Description
Successfully set the permission.	Change of the access permission has succeeded.
ERROR: Invaield permission mode.	An invalid permission indicator string has been specified.
WARN: The permission mode has no change.	There is no difference between the access permissions before and after changing.
ERROR: Can't specify the group name to be a built-in group.	The access permission of the default authority group cannot be changed.

3.34.1.6. Changing the authority group to which the user belongs (assignuser)

■ Command line interface

```

jc_permcfg assignuser $groupname $user1 [$user2 $user3 ... $userN]
    
```

■ Overview

Changes the authority group to which the user belongs.


■ Options

```

$groupname
Specify the name of the authority group to which the user belongs.
    
```

```

$user1 [ $user2 $user3 ... $userN ]
Specify one or more names of users who belong to the authority group specified in $groupname.
Up to 100 users can be specified at the same time.
    
```



The authority group to which the JobCenter administrator belongs cannot be changed.
A general user cannot belong to the JobCenter administrator group.

■ Example of execution

An example of execution in the UNIX version is shown below.

```

# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g @NormalUser
@NormalUser
Member
    
```

```

    jnwmonitor,user01,user02
  Authority
    own.jnw.read      on
    own.jnw.write     on
    own.jnw.execute   on
    own.trk.read      on
    own.trk.operation on
    other.jnw.read    off
    other.jnw.write   off
    other.jnw.execute off
    other.trk.read    off
    other.trk.operation off
  Permitted User

# /usr/lib/nqs/gui/bin/jc_permcfg assignuser <a user group that monitors other users> jnwmonitor
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g <a user group that monitors other users>
<a user group that monitors other users>
  Member
    jnwmonitor
  Authority
    own.jnw.read      off
    own.jnw.write     off
    own.jnw.execute   off
    own.trk.read      off
    own.trk.operation off
    other.jnw.read    on
    other.jnw.write   on
    other.jnw.execute on
    other.trk.read    on
    other.trk.operation on
  Permitted User
    jnwmonitor,nsumsmgr,root,user01,user02
# /usr/lib/nqs/gui/bin/jc_permcfg assignuser @JnwDeveloper user01 user02
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g @JnwDeveloper
@JnwDeveloper
  Member
    user01,user02
  Authority
    own.jnw.read      on
    own.jnw.write     on
    own.jnw.execute   on
    own.trk.read      on
    own.trk.operation on
    other.jnw.read    on
    other.jnw.write   on
    other.jnw.execute on
    other.trk.read    on
    other.trk.operation on
  Permitted User
    jnwmonitor,nsumsmgr,root,user01,user02

```

■ Main Messages

Describes the message specific to this subcommand. For other messages, refer to [Section 3.34.3, "Main messages shared by commands"](#).

Message	Description
Successfully set the permission.	Change of the authority group to which the user belongs has succeeded.
ERROR: Can't change the JobCenter Manager group's user.	The JobCenter administrator group cannot be specified as the authority group to which the user belongs.
WARN: The user already exists(username).	The user already belongs to the specified authority group.
ERROR: Can't modify the JobCenter manager(username) to group(groupname).	The authority group to which the JobCenter administrator user belongs cannot be changed.

3.34.1.7. Adding permissions for the authority group to access other users

(addaccuser)

■ Command line interface

```
jc_permcfg addaccuser $groupname $user1 [$user2 $user3 ... $userN]
jc_permcfg addaccuser $groupname -a
```

■ Overview

This command adds permissions for the authority group to access other users.

■ Options

\$groupname

Specify the name of the authority group to which to add access permissions.

\$user1 [\$user2 \$user3 ... \$userN]

Specify one or more names of users of the authority group specified in \$groupname to which to give access permissions.

Up to 100 users can be specified at the same time.

-a

Adds all the users of the authority group specified in \$groupname as the users to which to give access permissions.



The access-permitted users in the JobCenter administrator group cannot be changed.

The access-permitted users cannot be changed for a privilege group that does not have any privileges for other users. Therefore, it is required that the target authority group should have any access permissions to other users in advance.

■ Example of execution

An example of execution in the UNIX version is shown below.

- Adding access permission of a specified user

```
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g <a user group that monitors other users>
<a user group that monitors other users>
```

```

Member
  jnwmonitor
Authority
  own.jnw.read      off
  own.jnw.write     off
  own.jnw.execute   off
  own.trk.read      off
  own.trk.operation off
  other.jnw.read    on
  other.jnw.write   on
  other.jnw.execute on
  other.trk.read    on
  other.trk.operation on
Permitted User

# /usr/lib/nqs/gui/bin/jc_permcfg addaccuser <a user group that monitors other users> nsumsmgr
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g <a user group that monitors other users>
<a user group that monitors other users>
Member
  jnwmonitor
Authority
  own.jnw.read      off
  own.jnw.write     off
  own.jnw.execute   off
  own.trk.read      off
  own.trk.operation off
  other.jnw.read    on
  other.jnw.write   on
  other.jnw.execute on
  other.trk.read    on
  other.trk.operation on
Permitted User
  nsumsmgr

```

- Adding access permissions of all the users

```

# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g <a user group that monitors other users>
<a user group that monitors other users>
Member
  jnwmonitor
Authority
  own.jnw.read      off
  own.jnw.write     off
  own.jnw.execute   off
  own.trk.read      off
  own.trk.operation off
  other.jnw.read    on
  other.jnw.write   on
  other.jnw.execute on
  other.trk.read    on
  other.trk.operation on
Permitted User
  nsumsmgr
# /usr/lib/nqs/gui/bin/jc_permcfg addaccuser <a user group that monitors other users> -a
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g <a user group that monitors other users>

```

```
<a user group that monitors other users>
Member
  jnwmonitor
Authority
  own.jnw.read      off
  own.jnw.write     off
  own.jnw.execute   off
  own.trk.read      off
  own.trk.operation off
  other.jnw.read    on
  other.jnw.write   on
  other.jnw.execute on
  other.trk.read    on
  other.trk.operation on
Permitted User
  jnwmonitor,nsumsmgr,root,user01,user02
```

■ Main Messages

Describes the message specific to this subcommand. For other messages, refer to [Section 3.34.3, "Main messages shared by commands"](#).

Message	Description
Successfully set the permission.	Addition of access permission has succeeded.
WARN: Duplicate username(username).	The same user name is specified in the argument more than once.
WARN: The user already exists(username).	Access by a specified user has already been permitted.
ERROR: No permission.	The authority group does not have the access permission to other users, and access permission cannot be added.

3.34.1.8. Removing access permissions to other users from the authority group (delaccuser)

■ Command line interface

```
jc_permcfg delaccuser $groupname $user1 [$user2 $user3 ... $userN]
jc_permcfg delaccuser $groupname -a
```

■ Overview

This command removes permissions for the authority group to access other users.

■ Options

\$groupname

Specify the name of the authority group from which to remove access permissions.

\$user1 [\$user2 \$user3 ... \$userN]

Specify one or more names of users of the authority group specified in \$groupname from which to remove access permissions.

Up to 100 users can be specified at the same time.

-a

Removes all the users from the access-permitted users in the authority group specified in \$groupname.



The access-permitted users in the JobCenter administrator group cannot be changed.

The access-permitted users cannot be changed for a authority group that does not have any rights to other users. Therefore, it is required that the target authority group should have any access permissions to other users in advance.

■ Example of execution

An example of execution in the UNIX version is shown below.

- Deleting access permission of a specified user

```
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g <a user group that monitors other users>
<a user group that monitors other users>
Member
  jnwmonitor
Authority
  own.jnw.read      off
  own.jnw.write     off
  own.jnw.execute   off
  own.trk.read      off
  own.trk.operation off
  other.jnw.read    on
  other.jnw.write   on
  other.jnw.execute on
  other.trk.read    on
  other.trk.operation on
Permitted User
  jnwmonitor,nsumsmgr,root,user01,user02
# /usr/lib/nqs/gui/bin/jc_permcfg delaccuser <a user group that monitors other users> user01
user02
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g <a user group that monitors other users>
<a user group that monitors other users>
Member
  jnwmonitor
Authority
  own.jnw.read      off
  own.jnw.write     off
  own.jnw.execute   off
  own.trk.read      off
  own.trk.operation off
  other.jnw.read    on
  other.jnw.write   on
  other.jnw.execute on
  other.trk.read    on
  other.trk.operation on
Permitted User
  jnwmonitor,nsumsmgr,root
```

- Deleting access permissions of all the users

```
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g <a user group that monitors other users>
<a user group that monitors other users>
  Member
    jnwmonitor
  Authority
    own.jnw.read      off
    own.jnw.write     off
    own.jnw.execute   off
    own.trk.read      off
    own.trk.operation off
    other.jnw.read    on
    other.jnw.write   on
    other.jnw.execute on
    other.trk.read    on
    other.trk.operation on
  Permitted User
    jnwmonitor,nsumsmgr,root
# /usr/lib/nqs/gui/bin/jc_permcfg delaccuser <a user group that monitors other users> -a
Successfully set the permission.
# /usr/lib/nqs/gui/bin/jc_permcfg getperm -g <a user group that monitors other users>
<a user group that monitors other users>
  Member
    jnwmonitor
  Authority
    own.jnw.read      off
    own.jnw.write     off
    own.jnw.execute   off
    own.trk.read      off
    own.trk.operation off
    other.jnw.read    on
    other.jnw.write   on
    other.jnw.execute on
    other.trk.read    on
    other.trk.operation on
  Permitted User
```

■ Main Messages

Describes the message specific to this subcommand. For other messages, refer to [Section 3.34.3, "Main messages shared by commands"](#).

Message	Description
Successfully set the permission.	Deletion of access permission has succeeded.
WARN: Duplicate username(username).	The same user name is specified in the argument more than once.
WARN: The user could not be found(username).	The specified user does not exist among the access-permitted users.
ERROR: No permission.	The authority group does not have the access permissions to other users, and access permissions cannot be deleted.

3.34.2. Return Values

Return Value	Description
0	Normal end
1	Abnormal end

3.34.3. Main messages shared by commands

Message	Description
ERROR: Too few arguments.	The number of command arguments is insufficient.
ERROR: Invalid subcommand(subcommand).	An invalid subcommand is specified.
ERROR: The user name could not be found(username).	The specified user does not exist.
ERROR: The specified group could not be found.	The specified authority group does not exist.
ERROR: Group name can't more than 40 bytes.	The specified authority group name exceeds 40 bytes.
ERROR: Group name has invalid character.	The authority group name contains an invalid character.
ERROR: Invalid group name.	The authority group name contains a null character.
ERROR: Invalid site.	The value of the environment variable NQS_SITE is invalid.
ERROR: Only manager can execute this command.	A general user cannot execute this command.

3.34.4. Notes

- When executing the command for the cluster site, it is required to set the environment variable NQS_SITE in advance.

For details, refer to the precautions in [Chapter 1, 「Command List」](#) .

- When specifying the authority group in the Windows-version JobCenter, the value is not case sensitive.

3.34.5. Restrictions

Permission is not granted to a user who has not logged into the target JobCenter MG from CL/Win. Therefore, the target user needs to log into JobCenter MG in advance to set the permissions. The user environment is created on MG at this first login, and the permissions for the target user can be set.

The user environment is also created by executing the init subcommand of jc_usercfg without logging in from CL/Win. For details, refer to [Section 3.32, “jc_usercfg \(Creating the user environment, and browsing and changing the user environment setting\)”](#).

4 Commands for a Cluster Environment

These commands provide the functions to create a cluster site, rebuild a database, start, monitor, and stop a daemon process, etc.

4.1. cjcmsite (Creating a Site)

■ UNIX version

```
/usr/lib/nqs/cluster/cjcmsite $site-name $nqs-host-id $db-directory
```

■ Windows version

```
%InstallDirectory%\bin\cluster\cjcmsite $site-name $nqs-host-id $db-directory
```

4.1.1. Function Description

- The cjcmsite command creates a cluster site. A site is a JobCenter execution environment created for each individual package.
- In \$site-name, specify a site name. A site name must be a host name (the network's official full domain name) that corresponds to the relocatable IP address to be used. An alias cannot be specified.
- In \$nqs-host-id, specify the machine ID to be used in JobCenter.
A machine ID must be a number that is unique in the system.
- In \$db-directory, specify the name of the database directory to be used in JobCenter.
Specify a directory that resides in the shared disk of the package. The JobCenter database is created in the specified directory.
- Execute this command as the root user for the UNIX version or as the JobCenter administrator account for the Windows version.

4.1.2. Options

None

4.1.3. Return Values

Return value	Description
0	Normal end (the message "cjcmsite:complete." is displayed).
1	Abnormal end

4.1.4. Notes

- A cluster site is bound by identifying its relocatable IP address. Depending on the router settings, therefore, you may be unable to connect to the cluster site. The same applies when you specify a local site.
- If you fail to create a site because you specify an invalid site name in cjcmsite or for some other reason, delete the content under \$db_directory and then create a site again using cjcmsite.
- Make sure that the database path of the site you are adding is not identical to that of any other site.

4.2. cjcpw (Starting, Monitoring, or Stopping a Daemon Process)

■ UNIX version

For a cluster environment

```
/usr/lib/nqs/cluster/cjcpw [-v] [-c] $site-name $db-directory
/usr/lib/nqs/cluster/cjcpw -stop $site-name
```

For a local environment

```
/usr/lib/nqs/cluster/cjcpw [-v] [-c] -local
/usr/lib/nqs/cluster/cjcpw -stop -local
```

■ Windows version

For a cluster environment

```
%InstallDirectory%\bin\cluster\cjcpw [ [-service [-f] ] | [ [-u JobCenter-administrator-account] [-c] ] ] $site-name $db-directory
%InstallDirectory%\bin\cluster\cjcpw -stop $site-name
```

For a local environment

```
%InstallDirectory%\bin\cluster\cjcpw [ [-service [-f] ] | [ [-u JobCenter-administrator-account] [-c] ] ] -local
%InstallDirectory%\bin\cluster\cjcpw -stop -local
```

4.2.1. Function Description

■ For a cluster environment

- The command of the first syntax starts the cluster site specified in \$site-name. After the cluster site is started, the command monitors to see whether the resident processes are running (when the -c option is not specified). If the necessary resident process ends for some reason, then the command attempts to stop the other resident processes and the cluster site.
- The command of the second syntax stops the cluster site. It stops the resident processes of the cluster site one by one and then stops itself.
- In the UNIX environment, if the symbolic link and other data necessary to start the cluster site cannot be found, this command automatically creates them.
- The command of the second syntax stops the cluster site specified in \$site-name. It stops the resident processes of the cluster site one by one and then stops itself. Normally, write this command in the cluster package stop script of cluster management software so that it is executed from that script.
- In \$site-name, specify a site name.
- In \$db-directory, specify the name of the database directory to be used in JobCenter.

■ For a local environment

- The command of the first syntax starts a local site. After the local site is started, the command monitors to see whether the resident processes of the local site are running. If the necessary resident process ends for some reason, then the command attempts to stop the other resident processes.

If you start a local site by using `cjcpw` of the Windows version, `jcservice`, which is a JobCenter service, does not start and `cjcpw` becomes the top process.

- The command of the second syntax stops the local site. It stops the resident processes of the local site one by one and then stops itself.

4.2.2. Options

`-c`

`cjcpw` does not monitor daemon processes. The command starts the necessary daemon processes, checks that they become resident, and then terminates itself.

`-v`

The operating status is displayed in the standard error output.

`-stop`

The site is stopped.



The `cjcpw` command that waits for the JobCenter resident processes to end is the one that you executed without the `-c` option when starting the site and that is now resident. The `cjcpw` command that you executed with this option does not wait for the JobCenter resident processes to end.

`-u`

Specify the JobCenter administrator account.

`-service [-f]`

The site is started as a Windows service using a local system account.

When you start the site as a Windows service for the first time, the following message appears asking whether you want to register it as a Windows service.

Would you like to register this site as a Windows Service?[y/n](default:n)

If the `-f` option is specified, you are not asked whether you want to register the site as a Windows service.

4.2.3. Maintenance Mode

If `maintenance=ON` is written in the `daemon.conf` file, the `ngsstart` command started by `cjcpw` does not start daemons or other processes. In this case, `cjcpw` does not detect any daemon start error.

The administrator can perform maintenance on the site without starting daemons.

Likewise, if daemons are stopped by `ngsstop`, `cjcpw` does not detect an error, either. In this case, restarting daemons with `ngsstart` enables the process monitoring by `cjcpw` again.

4.2.4. Related Files

/usr/lib/nqs/nqsstart
/usr/lib/nqs/nqsstop
/usr/lib/nqs/rc/daemon.conf
/usr/spool/nqs/daemon.conf

4.2.5. Return Values

Return value	Description
0	Normal end
1	Abnormal end

4.2.6. Notes

- Starting cjcpcw without the -c option leaves this command in the running state, in which case you will not get a prompt. Therefore, if you want to stop a cluster site or local site, execute cjcpcw -stop from another terminal. Note that this cjcpcw -stop command executes its processing asynchronously with cjcpcw, which is left in the running state and was used to start the site.
- The cjcpcw process originally started without the -c option may take long to wait until the entire cluster site is stopped, because it also performs the processing of the active job requests in the NQS queue, such as stopping the processes and saving the status.

4.3. cjcls (Listing the Sites Running on the Machine That Executed the Command)

■ UNIX version

```
/usr/lib/nqs/cluster/cjcls [-a|-p]
```

■ Windows version

```
%InstallDirectory%\bin\cluster\cjcls [-a|-l|-p]
```

4.3.1. Function Description

- An example of the execution result of the UNIX version command is shown and described below.

[Example of the execution result of the UNIX version]

SITE-NAME	DAEMON-PID	DB-PATH(LINK)
testsite	18616	/mnt/jobdb (/usr/spool/nqs/0AC0120A)
othersite	18672	/mnt/other-jobdb (/usr/spool/nqs/0AC0120B)

The SITE-NAME column lists site names, and the DAEMON-PID column lists nqsdaemon process IDs (SHUT is displayed when the process is not running). The DB-PATH(LINK) column lists combinations of a database directory and the symbolic link to that database directory (path in parentheses), each on two lines.

This command can be executed by any user.

- An example of the execution result of the Windows version command is shown and described below.

[Example of the execution result of the Windows version]

SITE-NAME	PID	DB-PATH	STATUS
testsite	4284	D:\testsite	RUN
othersite	4356	E:\othersite	RUN

The SITE-NAME column lists site names, and the PID column lists cjcpw process IDs (SHUT is displayed when the process is not running). The DB-PATH column lists database directories, and the STATUS column shows the status of each site. Note that, when you display the local site information using the relevant option, the process ID of jcservice is shown as the local site PID, if the local site is running as a service.

The correspondence between the status shown in the STATUS column and the site status is as follows.

Status	Site status
RUN	The site is running.
RUN(SERV)	The local site is running as a service.
NW	The site is running in unmonitored mode (the site has been started with the -c option specified in cjcpw).

SHUT	The site is stopped.
BROKEN	One or more of the resident processes necessary for the site are not running (they may be being started).
DISCON	The database cannot be accessed.

This command can be executed by any user.

4.3.2. Options

-a

The process monitoring status is displayed, including the local site. If you specify this option for the UNIX version, the CJCPW column is added to the execution result in which the monitoring status (ON/OFF) is shown.

-l

This option is valid only for the Windows version.

The process monitoring status is displayed in long format, including the local site. In long format, all items are displayed, including those that are too long to display on a single line.

-p

The running site names, process names, and PIDs are displayed, including those of the local site. In the Windows version, both -p and -P output the same result.

[Example of the execution result of -p (UNIX version)]

SITE-NAME	PROCESS-NAME	PID
(localsite)	jcdbs	6341
	logdaemon	6351
	nqsdaemon	6352
	netdaemon	6354
	jnwcaster	6358
	sclaunchd	6364
	jnwengine	6362
	jnwlauncher	6372
	comagent	6371
clustersite	jcdbs	6438
	logdaemon	6448
	nqsdaemon	6449
	netdaemon	6451
	jnwcaster	6455
	sclaunchd	6461
	jnwengine	6459
	jnwlauncher	6472
	comagent	6468

[Example of the execution result of -p (Windows version)]

SITE-NAME	PROCESS-NAME	PID
-----------	--------------	-----

(local)	cjc	1412
	comagent	3104
	jcdb	1960
	jnwengine	3052
	jnwlauncher	3062
	logdaemon	2444
	netdaemon	2840
	nqsdaemon	2404
	sclaunchd	3092
ClusterSite	cjc	6684
	comagent	724
	jcdb	936
	jnwengine	1444
	jnwlauncher	4062
	logdaemon	5400
	netdaemon	4960
	nqsdaemon	5788
	sclaunchd	5924



In the execution result of `-p` of the Windows version, the process name "cjc" is managed in Windows using the image name (process name) of "jcservice (when started as a service)" or "cjcpw (when started by the cjcpw command)", and the PID associated with that image name is displayed.

4.3.3. Return Values

Return value	Description
0	Normal end
1	Abnormal end

4.3.4. Notes

- If you create an unnecessary site by using the `cjcmksite` command, `cjcls` displays that unnecessary site. In the UNIX version, you can resolve this situation by deleting the symbolic link shown in the `DB-PATH(LINK)` column.
- This command does not output any information if you have never executed the cluster site on the node machine.

4.4. nqspath (Displaying a Database Path Name)

■ UNIX version

```
/usr/lib/nqs/nqspath [$site-name]
```

■ Windows version

(None)

4.4.1. Function Description

- The nqspath command displays the database path name (standard spool path name) used by a specified site. (Note that linefeed codes are not added.)
- In \$site-name, specify a site name. If you specify a nonexistent site name, the command ends abnormally.
- If you omit \$site-name, the value of environment variable NQS_SITE is used. If environment variable NQS_SITE is not set, the command displays the database path name of the local site.
- This command can be executed by any user.

4.4.2. Options

None

4.4.3. Return Values

Return value	Description
0	Normal end
1	Abnormal end

4.5. nqsportkpr (Binding the Ports for JobCenter-related Services Temporarily)

■ UNIX version

```
/usr/lib/nqs/nqsportkpr [-d] [-s service1,service2...] [-i IP1,IP2...] [$hostname]
```

■ Windows version

(None)

4.5.1. Function Description

- The nqsportkpr command temporarily binds the ports for JobCenter-related services specified in service1, service2, etc. to the IP addresses specified in IP1, IP2, etc. or the IP address acquired from the site specified in \$hostname, thus occupying these bound ports to prevent other processes from using them. As JobCenter-related service ports, use those written in /etc/services (default nqs: 607/tcp jccombase: 611/tcp).

(Execution example)

Local site	/usr/lib/nqs/nqsportkpr -s nqs,jccombase localhost
Cluster site	/usr/lib/nqs/nqsportkpr -i 192.168.32.34 testsite1
Unbinding	/usr/lib/nqs/nqsportkpr -d testsite1

- The ports bound by nqsportkpr are automatically unbound when the JobCenter daemon (netdaemon) is started.
- This command can be executed only by the root user.

4.5.2. Options

-d

This option unbinds the port that is bound to a specified address by nqsportkpr.

-s service1,service2...

This option binds the ports for specified services. You can specify up to six services. If you omit this option, the command binds the ports for the nqs and jccombase services.

-i IP1,IP2...

This option binds the ports to specified IP addresses. You can specify up to six IP addresses. If you omit this option, the command binds the ports to the IP address acquired from the site specified in \$hostname.

\$hostname

This option binds the ports to a specified site. In the case of a local site, specify localhost or omit this option. In the case of a cluster site, specify the cluster site name.



- If you specify only the -i option, the command binds the ports to the local site using the IP addresses specified in the -i option.

- If you specify only the `$hostname` option, the command binds the ports to the site specified in `$hostname` using the IP address acquired from that site.
- If you specify both the `-i` and `$hostname` options, the command binds the ports to the site specified in `$hostname` using the IP addresses specified in the `-i` option.
- If you omit both the `-i` and `$hostname` options, the command binds the ports to the `INADDR_ANY` address of the local site.

4.5.3. Return Values

Return value	Description
0	Normal end
1	Abnormal end

4.5.4. Notes

- When you install JobCenter, it is set up to automatically execute this command as the rc script (`nqs.pre`) of the OS. Note that this automatic setup covers only the local site.
- If you want to occupy service ports using this command before starting a cluster site, you need to execute the command immediately after the relocatable IP addresses become effective in the script of the cluster package.

5 Commands for SAP ERP

These commands are used when you use ERP Option that enables linkage from JobCenter to a SAP ERP system.

5.1. sapclient (Establishing Linkage with SAP ERP)

■ UNIX version

```
/usr/lib/nqs/sap/sapclient $command [parameter...]
```

■ Windows version

```
%InstallDirectory%\bin\sapclient $command [parameter...]
```

5.1.1. Function Description

- The sapclient command is used when jnwengine executes and monitors ERP job parts. It is also used when the user logs in to ERP. It is used to set the audit level for the XMI log of SAP ERP as well.

In \$command, specify the sub command to be used directly by the user. The user directly uses the following sub commands.

■ Sub commands

conntest destination

Performs a connection test for SAP ERP.

In destination, specify the destination parameter set name defined in configuration file destconf.f.

Set the necessary parameters in destconf.f and saprfc.ini before executing this command.

auditlevel level

Sets the audit level.

In level, specify a value between 0 and 3.

5.1.2. Options

None

5.1.3. Return Values

If contest ends successfully, "connected successful" is displayed.

If an error occurs, an indication of the error is displayed.

5.1.4. Related Files

/usr/lib/nqs/sap/destconf.f

/usr/spool/nqs/gui/<user-name>/sapopt.f

/usr/spool/nqs/saprfc.ini



In the case of a cluster environment, replace /usr/spool with <cluster-database-path>.

5.2. sapcmd (Generating SAP ERP Connection Parameters)

■ UNIX version

```
/usr/lib/nqs/sap/sapcmd -ms  
/usr/lib/nqs/sap/sapcmd -mp -U $sap_user -P $password
```

■ Windows version

```
%InstallDirectory%\bin\sapcmd -ms  
%InstallDirectory%\bin\sapcmd -mp -U $sap_user -P $password
```

5.2.1. Function Description

- The command of the first syntax creates a sample file of the connection parameter file.
- The command of the second syntax generates an encrypted SAP user password to be written in the parameter file.

5.2.2. Options

```
-ms
```

A sample file of the connection parameter file is created.

```
-mp -U $sap_user -P $password
```

An encrypted SAP user password to be written in the parameter file is generated.

The format of the generated password is as follows.

\$sap_user: User name

\$password: Password

5.2.3. Return Values

The command of the second syntax displays the encrypted SAP user password.

5.2.4. Related Files

destconf.f.sapmle ... Sample of connopt.f

6 Configuration Check Commands

Problems in the network settings and user data directory settings may affect the startup or operation of JobCenter. For example, if a mistake is made or a necessary setting is missing when the network settings of the system are changed or the data directories are moved, JobCenter may fail to operate normally.

The configuration check commands check the current JobCenter execution environment for problems after the environment has been changed.

6.1. jc_check (Checking JobCenter Configuration)

■ UNIX version

```
/usr/lib/nqs/check/jc_check [-v] [-l [-b] [-u $user] [$server]]
```

■ Windows version

```
%InstallDirectory%\bin\check\jc_check\jc_check [-v] [-l [-b] [-u $user] [$server]]
```

6.1.1. Function Description

- The jc_check command checks the JobCenter configuration.
- It examines the network setting status, user data directory status, etc. to see whether the environment is appropriate for JobCenter to operate.
- In \$server, specify a host name. The command performs checks on the specified host name and outputs the results. This option is valid only when -l is specified.

6.1.2. Options

-v

The command is executed in verbose mode.

-l

The command performs all checks on all users and all host names (site names) and outputs the results.

-b

This option is valid only when -l is specified.

The command omits to check job network definition files and to collect detailed queue information. Only the list of queue information is collected.

When the "-l" option is specified, the jc_check command checks the definition files of all users. Therefore, when there are many definition files, it takes a very long time for the command to complete its processing.

If you want to reduce this processing time, specify this option.

-u \$user

This option is valid only when -l is specified.

The command performs the data directory check on the data directory of the user specified in \$user and outputs the results. (By default, the command checks the data directories of all the users registered in JobCenter.)

This does not affect the network setting check.

6.1.3. Notes

- When you execute this command, make sure that environment variables NQS_SITE and NQS_SITEDB are not set, even if you are using a cluster site.

If you execute this command when NQS_SITE and NQS_SITEDB are set, it is unable to collect information normally.

- When executing the jc_check command, use the root privileges in the UNIX version or the Administrator privileges in the Windows version.

When executed, the command checks the items below and displays the results as [OK] or [NG].

- Result of the resolv.def format check (function available only in the Windows version)
- Network setting status (name resolution status of the host name or site name and the IP address)
- Setting status of the data directory of each JobCenter user (directory and file owner name check and key parameter setting status check)
- Machine group registration status of each host (site)
- List of queues registered in each host (site) and detailed information
- Values set in the global parameters related to the entire NQS of each host (site)

When the result is [NG], the cause is displayed as well. (The items to be checked and output slightly differ between the Windows version and UNIX version.)

Note that, if there are extremely many users or job networks, the command may take time to complete.

- When the -l option is specified, an extremely large amount of information may be output to the terminal depending on the environment. In that case, redirect the output results to a file to view them.
- Specify the -b option only when there is an extremely large amount of user definition information and it takes very long to process all the information. If you specify the -b option, the command omits to collect the user definition information and detailed queue information.

Note that the command still collects the information necessary for the primary analysis of a JobCenter system or process failure such as network settings and environment settings. Basically, it is desirable to collect all information. Where possible, therefore, collect information without specifying the -b option.

- When multiple host names (site names) are defined for a single machine in a cluster environment, the network setting check is not affected even if \$server is set.

6.1.4. Example of the Execution Result (Windows Version)

An example of the execution result of the Windows version is shown below.

This example shows that there are problems with the two points indicated below.

[Example of the execution result]

```
#### jc_check command output ####
#### 2013/01/07 16:27:30 REV. *** #####
Host:hostA
-----
resolv.def's format:          [OK]

resolv.def:
The resolv.def file is empty.
```

```

Windows (OS):
hostA.example.com      (192.168.1.144) [OK]

Windows (Wrapper):
hostA.example.com      (192.168.1.144) [OK]

-----
Check local site:
hostA.example.com      hostA.example.com[OK]
-----
Users(ALL)
-----
nsagent                (1001)    PASSWORD:[NG](NO PASSWORD)Note 1
Administrator          (0)      PASSWORD:[OK]
                       PRIVILEGE:[NG]Note 2
                       SeBatchLogonRight

localsite:
  nsagent(1001):
    JNW:[--]
    SCH:[--]
  Administrator(0):
    JNW:[--]
    SCH:[--]

-----
NQS_site=hostA.example.com
-----
(The result of the "depend_tool" command will be shown here.)
-----
NQS_site=hostA.example.com
-----
NMAP LIST
-----
(The result of the "nmapmgr show state" command will be shown here.)
-----
MACHINE GROUP
-----
(The result of the "qmgr show machine_group" command will be shown here.)
-----
QUEUES AND REQUESTS
-----
(The result of the "qmgr show long q" command will be shown here.)
-----
PARAMETERS
-----
(The result of the "qmgr show parameter" command will be shown here.)
-----
#### OK=4, NG=2 ####
#### normal end of list ####

```



- Note 1 The password of a "nsagent" user is not set or invalid (CL/Win is not connected or the password has expired).
- Note 2 The rights (SeBatchLogonRight) of the "Administrator" user (JobCenter administrator indicated by uid=0) is insufficient.

6.2. depend_tool (Checking JobCenter Definition Validity)

■ UNIX version

```
/usr/lib/nqs/check/depend_tool [-r] [-u $user]
```

■ Windows version

```
%InstallDirectory%\bin\check\depend_tool\depend_tool [-r] [-u $user]
```

6.2.1. Function Description

- The depend_tool command checks whether the definition files managed by JobCenter are consistent with the data of the internal database that stores the related information.
- It examines job network definitions, schedule definitions, startup trigger definitions, monitoring target text log, calendar definitions, and custom job definitions to see whether the environment is appropriate for JobCenter to operate.
- When executing this command, use the root privileges in the UNIX version or the administrator privileges in the Windows version.

6.2.2. Options

-u \$user

The command performs the user definition data check on the user specified in \$user and outputs the results. (By default, the command searches the data directories of all the users registered in JobCenter.)

-r

The command is executed in definition repair mode.

Definition dependencies and the data consistency between definition files and the database are checked. If there is any data inconsistency, you can choose to repair the data.

Use this option when you are instructed to do so in troubleshooting.

6.2.3. Main Messages

■ Messages for the definition check

Message	Description
Dependencies are valid	Invalid dependencies have not been detected.
Missing subjnw ["JNW_NAME"]	The definition file of the job network "JNW_NAME" that is referenced as a sub job network part does not exist or is not registered in the database.
Missing calbranch ["SCH_NAME"]	The definition file of the schedule "SCH_NAME" that is referenced as a calendar branch part does not exist or is not registered in the database.
Missing schedule ["SCH_NAME"]	The definition file of the schedule "SCH_NAME" used to start the schedule does not exist or is not registered in the database.

Message	Description
Missing calendar ["CAL_NAME"]	The definition file of the operating day calendar "CAL_NAME" that is referenced from the schedule does not exist or is not registered in the database.
Missing jobnetwork ["JNW_NAME"]	The definition file of the job network "JNW_NAME" that is submitted by the trigger does not exist or is not registered in the database.
Missing target_textlog ["TARGET_TEXTLOG_NAME"]	The definition file of the monitoring target text log "TARGET_TEXTLOG_NAME" that is referenced as the monitoring target of the trigger does not exist or is not registered in the database.
Invalid entries are registered in DB(subjnw). ["JNW_NAME"]	The definition file of the job network "JNW_NAME" that is referenced as a sub job network part does not exist but is registered in the database.
Invalid entries are registered in DB(calbranch). ["SCH_NAME"]	The definition file of the schedule "SCH_NAME" that is referenced as a calendar branch part does not exist but is registered in the database.
Invalid entries are registered in DB(schedule). ["SCH_NAME"]	The definition file of the schedule "SCH_NAME" used to start the schedule does not exist but is registered in the database.
Invalid entries are registered in DB(calendar). ["CAL_NAME"]	The definition file of the operating day calendar "CAL_NAME" that is referenced from the schedule does not exist but is registered in the database.
Invalid entries are registered in DB(jobnetwork). ["JNW_NAME"]	The definition file of the job network "JNW_NAME" that is submitted by the trigger does not exist but is registered in the database.
Invalid entries are registered in DB(target_textlog). ["TARGET_TEXTLOG_NAME"]	The definition file of the monitoring target text log "TARGET_TEXTLOG_NAME" that is referenced as the monitoring target of the trigger does not exist but is registered in the database.
Doesn't exist or not registered in DB.	The definition file does not exist or is not registered in the database.
Invalid entries are registered in DB.	A definition whose definition file does not exist is registered in the database.

■ Messages for the definition repair

Message	Description
DB was successfully repaired (\$OBJ_TYPE["OBJ_NAME"])	<p>The data in the database was successfully repaired.</p> <p>\$OBJ_TYPE shows the type of the repaired definition, and "OBJ_NAME" shows the name of the repaired definition.</p> <p>In \$OBJ_TYPE, one of the following definitions is displayed.</p> <ul style="list-style-type: none"> ▪ JNW: Job network

Message	Description
	<ul style="list-style-type: none"> ■ SCHED: Schedule ■ TRG: Trigger ■ TARGET_TEXTLOG: Monitoring target text log ■ WKCAL: Operating day calendar ■ CUSTOMJOB: Custom job
failed to repair DB (\$OBJ_TYPE["OBJ_NAME"])	<p>The data in the database failed to be repaired.</p> <p>\$OBJ_TYPE shows the type of the repaired definition, and "OBJ_NAME" shows the name of the repaired definition.</p> <p>In \$OBJ_TYPE, one of the following definitions is displayed.</p> <ul style="list-style-type: none"> ■ JNW: Job network ■ SCHED: Schedule ■ TRG: Trigger ■ TARGET_TEXTLOG: Monitoring target text log ■ WKCAL: Operating day calendar ■ CUSTOMJOB: Custom job
Created empty definition (\$OBJ_TYPE["OBJ_NAME"])	<p>In order to repair an existing definition having a dependency on another definition, a definition to depended on by the existing definition was newly created.</p> <p>\$OBJ_TYPE shows the type of the newly created definition, and "OBJ_NAME" shows the name of the newly created definition.</p> <p>In \$OBJ_TYPE, one of the following definitions is displayed.</p> <ul style="list-style-type: none"> ■ JNW: Job network ■ SCHED: Schedule ■ TARGET_TEXTLOG: Monitoring target text log ■ WKCAL: Operating day calendar ■ CUSTOMJOB: Custom job

6.2.4. Notes

- When a definition was newly created to repair an existing definition having a dependency on the new definition, it is necessary to set the newly created definition again.
- When you use this command for a cluster site, make sure that environment variables NQS_SITE is set.

6.3. jc_ping (Checking Communication with a Specified Site)

■ UNIX version

```
/usr/lib/nqs/check/jc_ping [-h] [-f{rd | json}] [-p{ipv4 | ipv6}] [-w $timeout] $destination_host
```

■ Windows version

```
%InstallDirectory%\bin\check\jc_ping [-h] [-f{rd | json}] [-p{ipv4 | ipv6}] [-w $timeout] $destination_host
```

6.3.1. Function Description

- The `jc_ping` command checks communication with the `nqs`, `jccombase`, and `jcevent` ports of the destination site specified in `$destination_host`, assuming that the local site or cluster site to be the source site.
- The source site is the one set in environment variable `NQS_SITE`. (If environment variable `NQS_SITE` is not set, the local site is assumed.)

6.3.2. Options

`-h`

Help is displayed. If you specify this option when any other options are specified, the other options are ignored.

`-f`

Specify the output format of the execution results. Select one of the following.

- `rd`: Results are output in a readable format.
- `json`: Results are output in JSON format.

If you do not specify this option, results are output in a readable format.

For information about these output formats, see [Section 6.3.5, "Output Formats \(Readable Format/JSON Format\)"](#).

`-p`

Specify the protocol to be used for the communication check. Select one of the following.

- `ipv4`: The communication check is done using an IPv4 address.
- `ipv6`: The communication check is done using an IPv6 address.

If you do not specify this option, the communication check is done using an IPv4 address.

`-w $timeout`

Specify the timeout time (seconds) for the connection with a port. As the timeout time (seconds), enter an integer between 0 and 600.

If you do not specify this option, the timeout time (seconds) is assumed to be 30 seconds by default.

\$destination_host

Specify the name of the destination site.

6.3.3. Return Values

Return value	Description
0	Normal end
1	Abnormal end

The command returns the value that indicates the abnormal end only when the communication check cannot be continued.

6.3.4. Notes

- This command checks the communication from the source site to the destination site. To check the communication between MG and SV in both directions, you need to execute the command in both sites respectively.
- In case you are running the local site in site mode in the UNIX environment, you need to have environment variable NQS_SITE set when executing this command, even if the source site is the local site.
- In the Windows version, if and only if an error occurs during Winsock initialization, the JSON format is used as the output format, regardless of the output format specified in the option.
- The destination site must be added to the machine list of the source site.
- The source site must be added to the machine list of the destination site.
- When executing this command, use the JobCenter administrator account or an account belonging to the JobCenter group in the Windows version. When executing the command using an account other than the JobCenter administrator account or an account belonging to the JobCenter group, right-click cmd.exe, choose "Run as administrator" from the resulting menu, and run the command at the displayed command prompt.

6.3.5. Output Formats (Readable Format/JSON Format)

6.3.5.1. Readable Format

The readable format is as follows.

```

results :
destination_host = Destination site name
destination_ip   = Destination site IP address
source_host      = Source site name
source_ip        = Source site IP address

connect :
nqs              : [Communication check result of the nqs port]
  Message for the communication check of the nqs port
jccombase        : [Communication check result of the jccombase port]
  Message for the communication check of the jccombase port
jcevent          : [Communication check result of the jcevent port]
  Message for the communication check of the jcevent port
    
```

Communication check result message

Each line of the readable format is described below.

Table 6.1. Readable output format

Name	Description
Destination site name	The name of the destination site is displayed.
Destination site IP address	The IP address of the destination site is displayed.
Source site name	The name of the source site is displayed.
Source site IP address	The IP address of the source site is displayed. If multiple IP addresses are acquired by resolving the source site name, each displayed IP address is separated by a comma. The IP address used by this command is prefixed with "*" and always displayed at the beginning.
Communication check result of the nqs port	This line shows OK when the communication check result of the nqs port is normal or NG when the result is an error.
Message for the communication check of the nqs port	In the case of an error, an error message is displayed. (The output message may be multiple lines long.) When wildcard is set to on and the communication check result of the nqs port is normal, "wildcard=on" is displayed. When wildcard is set to off and the communication check result of the nqs port is normal, no message is displayed.
Communication check result of the jccombase port	This line shows OK when the communication check result of the jccombase port is normal or NG when the result is an error.
Message for the communication check of the jccombase port	In the case of an error, an error message is displayed. (The output message may be multiple lines long.) When the result is normal, no message is displayed.
Communication check result of the jcevent port	This line shows OK when the communication check result of the jcevent port is normal or NG when the result is an error.
Message for the communication check of the jcevent port	In the case of an error, an error message is displayed. (The output message may be multiple lines long.) When the result is normal, no message is displayed.
Communication check result message	In the case of an error, an error message is displayed. (The output message may be multiple lines long.) When all communication checks end successfully, "\$destination_host is alive" is displayed.

[Example of the output when all communication checks end successfully]

```
results :
destination_host = hostB
```

```

destination_ip = 192.168.0.2
source_host    = hostA
source_ip      = 192.168.0.1

connect :
nqs          : [OK]
jccombase    : [OK]
jcevent      : [OK]

hostB is alive.

```

[Example of the output when the communication check of a port ends with an error]

```

results :
destination_host = hostB
destination_ip   = 192.168.0.2
source_host      = hostA
source_ip        = 192.168.0.1

connect :
nqs             : [NG]
                NQS net daemon is not present at transaction peer. Retry later.
jccombase       : [NG]
                20388:cl.c(98): cl_open_timeout:jcsock_connect_challenge() failure("No connected socket.
                errno=111")
jcevent         : [NG]
                No connected socket. errno=111

Communication Failure.

```

6.3.5.2. JSON Format

The JSON format is as follows.

```

{
  "status_code":0,
  "result_message":"hostB is alive."
  "results":{
    "source_host":"hostA",
    "source_ip":"192.168.0.1",
    "destination_host":"hostB",
    "destination_ip":"192.168.0.2",
    "connect":{
      "nqs":{
        "status_code":0,
        "result_message":""
      },
      "jccombase":{
        "status_code":0,
        "result_message":""
      },
      "jcevent":{
        "status_code":0,
        "result_message":""
      }
    }
  }
}

```

Each parameter of the JSON format is described below.

Table 6.2. Parameter names of the JSON format

Parameter name	Description
status_code	The check result is set. When the communication check result is normal, 0 is set. Otherwise, 1 is set.
result_message	The check result message is set. In the case of an error, an error message is set. When all communication checks end successfully, "\$destination_host is alive." is set. When wildcard is set to on and the communication check result of the nqs port is normal, "wildcard=on" is set only for result_message of nqs.
results	The result of the communication check is set. This becomes empty if an error occurs before the communication check.
source_host	The name of the source site is set.
source_ip	The IP address of the source site is set. If multiple IP addresses are acquired by resolving the source site name, each set IP address is separated by a comma. The IP address used by this command is prefixed with "*" and always set at the beginning.
destination_host	The name of the destination site is set.
destination_ip	The IP address of the destination site is set.
connect	The communication check result of each port is set.
nqs	The communication check result of the nqs port is set.
jccombase	The communication check result of the jccombase port is set.
jcevent	The communication check result of the jcevent port is set.

6.3.6. Main Messages

6.3.6.1. Main Messages for Communication Check Results

Message	Description
\$destination_host is alive.	The communication checks of all ports ended successfully.
Communication Failure.	The communication checks of one or more ports failed. See the error message in the communication check result for the failed port.
Usage: jc_ping [-f rd json] [-p ipv4 ipv6] [-w \$timeout] \$destination_host	The argument is invalid. Check whether the argument is valid.
Invalid format. [rd json]	A value other than rd or json is specified in the -f option. Set a valid value in the -f option.
Invalid address family. [ipv4 ipv6]	A value other than ipv4 or ipv6 is specified in the -p option. Set a valid value in the -p option.
Invalid timeout. [0 - 600]	A value outside the range of 0 to 600 is specified in the -w option. Set a valid value in the -w option.

Message	Description
Too many arguments.	There are too many arguments. Check whether the number of specified arguments is appropriate.
Too few arguments.	There are too few arguments. Check whether the number of specified arguments is appropriate.
Failed to get source hostname.	Getting the source site name failed. Check whether the machine name or the site name set in environment variable NQS_SITE is valid.
Failed to get source ipaddress.	Getting the IP address of the source site name failed. Check the name resolution of the site name and the IP address.
Failed to get destination ipaddress.	Getting the IP address of the destination site name failed. Check the name resolution of the site name and the IP address.
Failed to create process.(errno=error-number)	You may not have used the JobCenter administrator account or an account belonging to the JobCenter group when executing the command. You may not have executed the command by right-clicking cmd.exe, choosing "Run as administrator" from the resulting menu, and then running the command from the displayed command prompt.

6.3.6.2. Main Messages for the Communication Check of the nqs Port

Message	Description
wildcard=on	The communication check ended successfully with NQSDAEMON_OPT set to -x and wildcard set to ON in daemon.conf of the source site.
Unable to chdir() to the NQS root directory. (errno=error-number)	Moving to the data directory of the NQS failed. Check whether the source site name is invalid and whether the JobCenter site database of the source site has been created normally.
failed to getaddrinfo: node=\$hostname service=\$servicename (error-number-information).	Name resolution or getting the port number failed. Resolve the error corresponding to the error number of the OS.
failed to jc_getaddrinfo: node=\$hostname service=\$servicename ai_family=\$family(error-number-information)	Name resolution or getting the port number failed. Resolve the error corresponding to the error number of the OS.
failed to NSgetaddrinfo: parameter error (service=\$servicename errno=error-number)	Getting the port number failed. Resolve the error corresponding to the error number of the OS.
Unable to get Machine ID.(host=\$hostname, ret=return-value-of-machine-ID-acquisition-function)	Getting the machine ID of the corresponding site name failed. Check whether the corresponding site name is registered in the machine list of the source site.
Machine-id conflict between client and peer at transaction peer. Seek staff support.	The machine ID of the destination site registered in the machine list of the source site does not match the machine ID of the destination site. Check whether the machine IDs match.
Client machine-id unknown at transaction peer. Seek staff support.	The source site is not registered in the machine list of the destination site. Check whether the source site is registered in the machine list of the destination site.

Message	Description
Connect(2) or read(2) timeout at local host. Retry later.	The connect or read processing for the destination site ended due to a timeout. Check whether the destination site name is valid, whether JobCenter is running in the destination site, and whether the network is valid.
NQS net daemon is not present at transaction peer. Retry later.	The NQS port of the destination site cannot be connected. Check whether JobCenter is running in the destination site and whether the network is valid.
Local network database error at local host. Seek staff support.(detailed-error-message)	Getting the IP address of the source site failed. Check the name resolution of the site name and the IP address.
No network port available at local host. Retry later.(detailed-error-message)	There is no available network port. Re-executing the command may solve the problem.

6.3.6.3. Main Messages for the Communication Check of the jccombase Port

Message	Description
get_jccombase_port() failure("error-number-information"(error-number))	Getting the port number failed. Resolve the error corresponding to the error number of the OS.
jcsock_getaddrinfo() failure("failed to getaddrinfo: node=\$hostname service=\$servicename (error-number-information).")	Name resolution or getting the port number failed. Resolve the error corresponding to the error number of the OS.
NSgetaddrinfo() failure(failed to NSgetaddrinfo: parameter error (service=\$servicename errno=error-number)) servername=\$servicename	Name resolution or getting the port number failed. Resolve the error corresponding to the error number of the OS.
jcsock_connect_challenge() failure("connect timedout.(timeout = <\$timeout>sec)")	The connect processing for the destination site ended due to a timeout. Check whether the destination site name is valid, whether JobCenter is running in the destination site, and whether the network is valid.
skt_ConnectChallenge() failure(connect timedout.(timeout = <\$timeout>sec))	The connect processing for the destination site ended due to a timeout. Check whether the destination site name is valid, whether JobCenter is running in the destination site, and whether the network is valid.
jcsock_connect_challenge() failure("connect error. errno=error-number")	The jccombase port of the destination site cannot be connected. Check whether JobCenter is running in the destination site and whether the network is valid, or solve the error corresponding to the error number of the OS.
skt_ConnectChallenge() failure(connect error. errno=error-number)	The jccombase port of the destination site cannot be connected. Check whether JobCenter is running in the destination site and whether the network is valid, or solve the error corresponding to the error number of the OS.
skt_Read:read() connection close	The connection was closed while waiting for a response. The version of JobCenter in the destination site may be R12.4 or earlier.

6.3.6.4. Main Messages for the Communication Check of the jcevent Port

Message	Description
failed to getaddrinfo: node=\$hostname service=\$servicename (error-number-information).	Name resolution or getting the port number failed. Resolve the error corresponding to the error number of the OS.
failed to jc_getaddrinfo: node=\$hostname service=\$servicename ai_family=\$family(error-number-information)	Name resolution or getting the port number failed. Resolve the error corresponding to the error number of the OS.
failed to NSgetaddrinfo: parameter error (service=\$servicename errno=error-number)	Getting the port number failed. Resolve the error corresponding to the error number of the OS.
connect timed out.(timeout = <\$timeout>sec)	The connect processing for the destination site ended due to a timeout. Check whether the destination site name is valid, whether JobCenter is running in the destination site, and whether the network is valid.
connect error. errno=error-number	The jcevent port of the destination site cannot be connected. Check whether JobCenter is running in the destination site and whether the network is valid, or solve the error corresponding to the error number of the OS.
No connected socket. errno=error-number	The jcevent port of the destination site cannot be connected. Check whether JobCenter is running in the destination site and whether the network is valid, or solve the error corresponding to the error number of the OS.

7 Commands for Collecting Information

When a failure occurs in JobCenter, these commands collect all the primary information necessary to identify its cause.

7.1. jc_getinfo (Collecting All Primary Information Necessary to Identify the Cause of a Failure in JobCenter)

■ UNIX version

```
/usr/lib/nqs/check/jc_getinfo [-b] [-d $output] [-e]
```

■ Windows version

```
%InstallDirectory%\bin\check\jc_getinfo [-b | -p] [-d $output] [-m {0 | 1 | 2 | 3 | 4}] [-e] [-w]
```

7.1.1. Function Description

- If you execute the `jc_getinfo` command when a failure has occurred in JobCenter, the information necessary to identify its cause (configuration information and log information) is automatically collected.
- The collected information is saved with the default file name in the directory specified in the `-d` option. (If you do not specify the `-d` option, the information is saved in the current directory.)
- In the UNIX version, "jcdata_<MMDDhhmm>_<hostname>.tar.Z" (other than Linux) or "jcdata_<MMDDhhmm>_<hostname>.tar.gz" (Linux) is created under the directory mentioned above.
- In the Windows version, the collected information is saved with the default file name under the "jcdata" directory created under the directory mentioned above according to the directory structure.

7.1.2. Options

`-b`

The command omits to check job network definition files and to collect the list of JobCenter user definition information files and detailed queue information. Only the list of queue information is collected.

The `jc_getinfo` command checks the definition files of all users. Therefore, when there are many definition files, it takes a very long time for the command to complete its processing.

If you want to reduce this processing time, specify this option.

`-p`

This option is valid only for the Windows version.

Only the user specified as the JobCenter administrator checks the user passwords.

Use this option only if instructions are issued in case of a failure.

`-d $output`

Specify the location to save the collected information.

`-m {0 | 1 | 2 | 3 | 4}`

This option is valid only for the Windows version.

It is the event mode option that lets you specify the type of Windows OS event log to be collected.

If you do not specify this option, the command collects the system event log and application event log.

- 0 Collect all event logs
- 1 Collect the system event log only
- 2 Collect the application event log only
- 3 Collect the security event log only
- 4 Collect the system event log and application event log

-e

Specifying this option omits the acquisition of the database that manages the metadata of the definition information or tracker execution status.

Use this option only if instructions are issued in case of a failure.

-w

This option is valid only for the Windows version.

The command omits to collect the Windows error reports (Watson log). Do not use this option unless instructed to do so by the support center.

7.1.3. Notes

- When you execute this command, make sure that environment variables NQS_SITE and NQS_SITEDB are not set, even if you are using a cluster site.

If you execute this command when NQS_SITE and NQS_SITEDB are set, it is unable to collect information normally.

- When executing this command, use the root account in the UNIX version or the JobCenter administrator account in the Windows version.

- When executing this command in an environment of Windows Server 2008 or later, right-click cmd.exe, choose "Run as administrator" from the resulting menu, and run the command from the command line.

- Specify the -b option only when there is an extremely large amount of user definition information and it takes very long to process all the information. If you specify the -b option, the command omits to collect the user definition information and detailed queue information.

Note that the command still collects the information necessary for the primary analysis of a system or process failure in JobCenter, such as network settings and environment settings. Basically, it is desirable to collect all information. Where possible, therefore, collect information without specifying the -b option.

7.1.4. Example of the Execution Result (Windows Version)

An example of the execution result of the Windows version is shown in [Figure 7.1, "Example of the jc_getinfo command execution result of the Windows version"](#).

In this example, the "jcdata" directory was created under the execution directory because the -d option was not specified.

The information to be collected, as well as the numbers and types of directories and files to be created, may change according to the operation status of JobCenter.

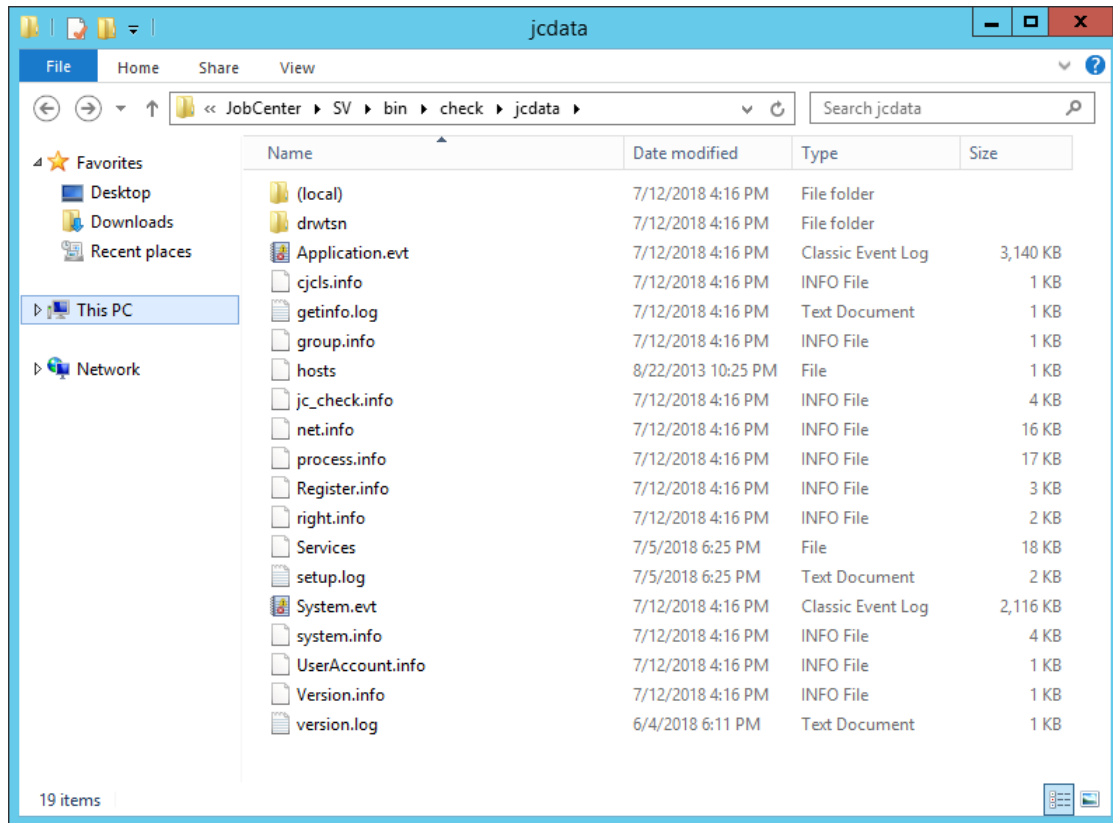


Figure 7.1. Example of the jc_getinfo command execution result of the Windows version

7.2. clweb_getinfo (Collecting All Primary Information Necessary to Identify the Cause of a CL/Web Server Failure)

■ UNIX version

```
%InstallDirectory%/script/clweb_getinfo/clweb_getinfo.sh [-d $output]
```

■ Windows version

```
%InstallDirectory%\script\clweb_getinfo\clweb_getinfo [-d $output]
```

7.2.1. Function Description

- If you execute the `clweb_getinfo` command when a failure has occurred in a CL/Web server, the information necessary to identify its cause (configuration information and log information) is automatically collected.
- The collected information is saved with the default file name in the directory specified in the `-d` option. (If you do not specify the `-d` option, the information is saved in the current directory.)
- "`clwebinfo_<YYYYMMDDhhmmss>.zip`" is created under the directory mentioned above.

7.2.2. Options

```
-d $output
```

Specify the location to save the collected information.

7.2.3. Notes

- When executing this command, use the root account in the UNIX version or the JobCenter administrator account in the Windows version.

8 Command for the Analysis Helper Function

This command is used when you use the JobCenter Analysis Helper function.

8.1. jdh_trkupdate (Updating the Content of the Local Data Folder of Analysis Helper)

```
%InstallDirectory%\bin\jdh_trkupdate [-f] [-u $user] $jpf_file
```

8.1.1. Function Description

■ The `jdh_trkupdate` command extracts the necessary data from the JPF file collected by the `jdh_trkget` command and converts the data to a format available to Analysis Helper. The converted data is saved in the data folder under the installation folder of Analysis Helper.

■ In `$jpf_file`, specify the JPF file collected by the `jdh_trkget` command.

8.1.2. Options

`-f`

The command carries out the update regardless of the content of the data folder. If you omit this option, the command checks the converted data in the data folder and processes only the data that needs to be updated.

`-u $user`

The command updates only the data of the user specified in `$user`. If you omit this option, the command updates the data of all users.

8.1.3. Return Values

Return Values	Description
0	Normal end
1	Abnormal end. An error message is output to the standard error output.

8.1.4. Main Messages

Message	Description
All tracker data update successfully!	The update of the data has ended successfully.
Not necessary to be updated.	It is not necessary to update the data.

8.1.5. Example of the Execution Result

An example of the execution result of this command is shown below.

```
D:\NECJCPkg-jah>bin\jdh_trkupdate trkget_sample.jpf
extracting...
checking...
(1/14) user=root file=trkarc20110730-20110730 updating...
(2/14) user=root file=trkarc20110731-20110731 updating...
(3/14) user=root file=trkarc20110801-20110801 updating...
(4/14) user=root file=trkarc20110802-20110802 updating...
(5/14) user=root file=trkarc20110803-20110803 updating...
(6/14) user=root file=trkarc20110804-20110804 updating...
(7/14) user=root file=trkarc20110805-20110805 updating...
(8/14) user=user1 file=trkarc20110730-20110730 updating...
(9/14) user=user1 file=trkarc20110731-20110731 updating...
```

```
(10/14) user=user1 file=trkarc20110801-20110801 updating...
(11/14) user=user1 file=trkarc20110802-20110802 updating...
(12/14) user=user1 file=trkarc20110803-20110803 updating...
(13/14) user=user1 file=trkarc20110804-20110804 updating...
(14/14) user=user1 file=trkarc20110805-20110805 updating...
All tracker data update successfully!
```

8.1.6. Notes

- The JPF file contains compressed tracker archive files, and this command first decompresses these files. Therefore, the command temporarily consumes as much local computer disk space as the size of the archive files.
- If you execute this command while Analysis Helper is importing tracker data from an Excel file, the command ends with an error without updating the data folder, so as to keep the data consistent. When executing this command, make sure that tracker data is not being imported from an Excel file.

