

**IBM® Z Workload Scheduler**  
**Quick Reference**  
**Version 10.1 (Revised March 2025)**

## Note

Before using this information and the product it supports, read the information in [Notices on page lxxviii](#).

This edition applies to version 10, release 1, modification level 0 of IBM® Z Workload Scheduler (program number 5698-T09) and to all subsequent releases and modifications until otherwise indicated in new editions.

# Contents

List of Figures.....	v	Error codes.....	65
About this publication.....	vi	Operation status codes.....	69
Support information.....	vi	Extended status codes.....	70
<b>Chapter 1. Using the ISPF dialogs.....</b>	<b>7</b>	Job log retrieval status codes.....	72
Specifying list criteria.....	7	Operation reason codes.....	72
Using generic search arguments.....	8	<b>Chapter 8. Free-day rule.....</b>	<b>74</b>
Sorting list output.....	8	<b>Chapter 9. Defining day to day workload .....</b>	<b>75</b>
Locating data strings in list output.....	8	Job related.....	75
Graphically displaying lists.....	9	NOERROR syntax for JCL return code	
PF key assignment.....	9	management.....	75
<b>Chapter 2. TSO commands.....</b>	<b>10</b>	USRREC syntax for end-to-end scheduling.....	75
BACKUP.....	10	Modeling related for end-to-end scheduling.....	75
BULKDISC.....	13	CPUREC.....	76
JSUACT.....	15	DOMREC.....	77
OPINFO.....	17	Notices.....	lxxviii
OPSTAT.....	21	Index.....	82
SRSTAT.....	26		
WSSTAT.....	32		
<b>Chapter 3. Commands to control the scheduler.....</b>	<b>37</b>		
Starting the scheduler.....	37		
Stopping the scheduler.....	37		
Canceling the scheduler.....	38		
Modifying the scheduler.....	38		
Modifying the data store.....	50		
<b>Chapter 4. Automatic recovery.....</b>	<b>54</b>		
RECOVER statement.....	55		
<b>Chapter 5. JCL directives.....</b>	<b>57</b>		
NOP directive.....	57		
SCAN directive.....	57		
SEARCH directive.....	57		
TABLE directive.....	57		
SETFORM directive.....	57		
Usage notes.....	57		
SETVAR directive.....	58		
Usage notes.....	58		
BEGIN and END directives.....	59		
FETCH directive.....	60		
COMP keyword on BEGIN and FETCH directives.....	60		
Usage notes.....	60		
<b>Chapter 6. Supplied JCL variables.....</b>	<b>61</b>		
Occurrence-related JCL variables.....	61		
Operation-related JCL variables.....	62		
Date-related JCL variables.....	63		
Dynamic-format supplied variables.....	64		
<b>Chapter 7. Codes.....</b>	<b>65</b>		

# List of Figures

Figure 1: EQQSOPFP - Selecting operations..... 8

## About this publication

*IBM Z Workload Scheduler: Quick Reference* is a handy guide to the more commonly used dialogs, commands, and directives of IBM Z Workload Scheduler.

The term *scheduler*, when used in this publication, refers to IBM Z Workload Scheduler. The term DB2®, when used in this publication, refers to DATABASE 2 and DB2 Universal Database™.

The following sent

## Support information

IBM provides several ways for you to obtain support when you encounter a problem.

If you have a problem with your IBM software, you want to resolve it quickly. IBM provides the following ways for you to obtain the support you need:

- Searching knowledge bases: You can search across a large collection of known problems and workarounds, Technotes, and other information.
- Obtaining fixes: You can locate the latest fixes that are already available for your product.
- Contacting IBM Software Support: If you still cannot solve your problem, and you need to work with someone from IBM, you can use a variety of ways to contact IBM Software Support.

For more information about these three ways of resolving problems, see the appendix about support information in *IBM Workload Scheduler: Troubleshooting Guide*.

# Chapter 1. Using the ISPF dialogs

**Table 1. Some primary commands for panels**

Command	Action
RETURN	Return to main menu. An end operation is run for each panel in the sequence leading back to the main menu (all changes on the individual panels are saved).
CANCEL	Return to the previous panel without making any changes.
RIGHT	Display the right part of the data. This is available only from panels that have the text <code>LEFT PART</code> in the panel title.
LEFT	Display the left part of the data. This is available only from panels that have the text <code>RIGHT PART</code> in the panel title.
HELP	Display help information.
SORT	Sort information in a list.
LOCATE <i>lparm</i>	Scroll to the field specified. If the field is not found, the list is displayed starting with the entry before which the specified field would have occurred. If the list is sorted by application name, <i>lparm</i> is the name of the application; if sorted by job name, <i>lparm</i> is a job name.
GRAPH	Display a network of dependencies.
GDDM®	Execute Graphical Data Display Manager (GDDM®) functions on a graphically displayed network.
ATTR	Set graphic attributes.

## Specifying list criteria

Limit the length of lists by using filter panels like the SELECTING OPERATIONS panel:

Figure 1. EQQSOPFP - Selecting operations

```

EQQSOPFP ----- SELECTING OPERATIONS -----
Command ==>>

Specify selection criteria below and press ENTER to create an operation list.

JOBNAME           => P*_____      WORK STATION NAME => ____
APPLICATION ID    => _____      OWNER ID           => _____
AUTHORITY GROUP   => _____      PRIORITY           => _
GROUP DEFINITION  => _____      STATUS             => _____
CLEAN UP TYPE     => _____      CLEAN UP RESULT    => __
OP.EXTENDED NAME  => _____
OP. SE NAME       => _____
Input arrival in format YY/MM/DD   HH.MM
FROM              => _____
TO                => _____
Additional Options:
FAST PATH         => Y                Valid only along with jobname
Set Y, N, or leave blank to select all:
MANUALLY HELD    => _      WAITING FOR SE    => _      STARTED ON WAIT WS => _
CRITICAL PATH    => _      COND RECOVERY JOB => _      RECOVERED BY COND => _
UNEXPECTED RC    => _      UNDEFINED COND   => _      SHADOW JOB         => _
STARTED AT STARTUP => N      LATE              => _
Set P, M, B, E, or leave blank to select all:
WAITING PEND.PRED. ==> _

```

You can use blanks, complete names, IDs, or search arguments in the input fields.

On some selection panels you can choose the *fastpath* option, so that IBM Z Workload Scheduler searches for matching job names on automatic workstations. If there is a matching job name, IBM Z Workload Scheduler includes all operations with that job name, whether on an automatic workstation or not.

## Using generic search arguments

Many of the input fields in the dialog accept generic search arguments. Use an asterisk (\*) to represent any character string or a null string. Use a percent sign (%) for any single character.

To select all application identifiers whose first three letters are PAY, enter this in the input field:

```
APPLICATION ID ==>> PAY*_____
```

To select all applications where P is the first letter and Y is the third letter, enter this:

```
APPLICATION ID ==>> P%Y*_____
```

## Sorting list output

In all list displays, enter the `SORT` command to display a panel where you can specify the order of the list items. The sort order you request remains in effect for that specific list type until changed.

If you sort on date fields, take into account the date format that has been specified, because this can affect the order.

## Locating data strings in list output

Enter `LOCATE` on any list display panel to find a data string in the list. The command also supports generic search strings. For example, you can enter `LOCATE ABC*` to find any item in the list beginning with ABC. Locate scrolls to the field specified.

If the list is sorted by application name, request `LOCATE applname`; similarly, if sorted by job name, request `LOCATE jobname`. If you need to issue a locate command on a list that is not sorted by the item you want to locate, change the order with the `SORT` command.

## Graphically displaying lists

If you have GDDM® installed and have a terminal capable of displaying graphics, you can also display lists of applications, occurrences, and operations graphically. Graphic displays contain the same information as edit or selection lists, but the format is different; you can see dependency connections that might be hard to see from a conventional list.

To see a graphically displayed list, enter `GRAPH` at the command prompt of a panel on which graph is an option.

## PF key assignment

The dialog maintains separate program function (PF) keys from your normal ISPF key assignments. Enter `KEYS` at the command prompt to display or change the current assignment.

You can define PF keys to run a command that you use regularly, for example, to display the ready list. To ensure that the command will be run correctly, regardless of the panel it is entered from, define the PF key like this:

```
PF5 ==> ;=4.1.cpu1
```

Where `;` is your ISPF command delimiter.

You can define unique PF key assignments for different panels. For example, if you regularly use the application description dialog, you can define PF keys for the `OPER` and `RUN` commands.

It is recommended that you do not alter the key assignments for PF1 (HELP), or PF12 (RETRIEVE). The retrieve PF key returns the command you last run to the command prompt. A stack of approximately 25 commands is maintained.

The PF KEY DEFINITIONS AND LABELS panel lets you assign labels to the PF key definitions. When you enter the `PFSHOW` command from a panel, the PF key labels are displayed. To remove the display, enter `PFSHOW OFF`.

## Chapter 2. TSO commands

This chapter describes the syntax for the IBM Z Workload Scheduler TSO commands, BACKUP, BULKDISC, JSUACT, OPINFO, OPSTAT, SRSTAT, and WSSTAT.

You can abbreviate keywords to their shortest unambiguous form. For example, you can shorten the AVAIL keyword to an 'A'. Before you invoke an IBM Z Workload Scheduler TSO command, you must allocate the EQQLIB data set to the TSO address space, either by adding DD statements to the logon procedure, or by using the ALLOC command after TSO logon. Error messages are sent directly to the terminal.

The TSO commands can be directed to a specific IBM Z Workload Scheduler subsystem (tracker). The tracker does not have to be active when you issue the command. An event is generated and queued in ECSA along with other job-tracking events.

### BACKUP

#### Purpose

The BACKUP command is used to initiate a backup of the current plan (CP) or the JCL repository (JS) on request. You can request the backup process by issuing the BACKUP command from your TSO session or from within a batch job. You can schedule current plan or JS backups by defining the job as an operation in the current plan.

The JS and current plan files are managed by the controller. A request generated by the BACKUP command is communicated to the controller as an event record processed by the event writer task of a tracker. The BACKUP command can be issued from any z/OS® system that runs a tracker from IBM® Z Workload Scheduler Release 2 or a later release. If you request a BACKUP on multiple systems at the same time, you will cause multiple file backups to occur when the events reach the controller.

A current plan backup is performed automatically under these circumstances:

- During normal shutdown of the controller.
- At the beginning and end of the daily planning process.
- When the number of new job-tracking records is greater than the value specified by the JTOPTS keyword BACKUP. But if the JTOPTS BACKUP keyword specifies NO, these regular backups of the current plan are not taken.

You can also request an immediate current plan backup at any time by issuing the BACKUP command for the current plan resource. You might do this to:

- Request a backup at a predefined time for disaster recovery purposes.
- Ensure regular backups are taken when IBM® Z Workload Scheduler activity is low.
- Take backups only at set times, in which case the JTOPTS BACKUP keyword is probably set to NO.

When the JTOPTS BACKUP keyword specifies a numeric value, a counter increases by one every time a new job-tracking record is written. The counter is reset to zero after every current plan backup.



## Parameters

### RESDS (CP | JS)

The RESDS keyword specifies which data set the backup will be performed on. If you specify CP as the keyword value, a current plan backup will be performed. If you specify JS as the value for this keyword, the JCL repository data set will be copied to the alternate JS file.

The RESDS keyword must be specified.

### SUBSYS (MSTR | *subsystem name* | OPCA)

The name of the tracker subsystem that the BACKUP command is directed to. The name can be up to four characters long. The first character must be alphabetic; the remaining characters must be alphanumeric. All lowercase characters are converted to uppercase. If the SUBSYS keyword is omitted the on-request backup command will be directed to a subsystem called OPCA.

If you specify **MSTR**, the BACKUP command is directed to all tracker subsystems on the z/OS® system that the BACKUP command was directed to.



**Attention:** This will cause multiple backups to occur if more than one tracker subsystem is active on the system.

*Subsystem name* is the name of the subsystem that the BACKUP command will be directed to. If the tracker and controller run in separate address spaces in your IBM® Z Workload Scheduler configuration, specify the name of the tracker subsystem in this parameter.

### TRACE (*level* | 0)

Event tracing indicator. When a positive number is specified, a trace entry is created for each event generated by the BACKUP command. The trace record is written to the message log file identified by ddname EQQMLOG. The record identifies the name of each receiving subsystem. The default value 0 will not generate trace records.

## Example

### Examples

These two examples demonstrate how you can use the BACKUP command in TSO, or in a batch job (using the batch program EQQEVPGM).

#### BACKUP

##### Example 1 - TSO command

```
ALLOC F(EQQMLIB) DA('OPC.MESSAGE.LIBRARY') SHR REUSE
BACKUP RESDS(CP) SUBSYS(OPCA)
```

##### Example 2 - Batch job

```
//CPBACKUP JOB (ACCOUNT), 'Backup CP', CLASS=A
//STEP1 EXEC PGM=EQQEVPGM
```

```
//STEPLIB DD DSN=OPC.LOAD.MODULE.LIBRARY,DISP=SHR
//EQQLIB DD DSN=OPC.MESSAGE.LIBRARY,DISP=SHR
//EQQMLOG DD SYSOUT=A
//SYSIN DD *
BACKUP RESDS(CP) SUBSYS(OPCA)
/*
```

In both of these examples, the current plan data set will be copied to the old current plan data set, on a subsystem called OPCA.

## BULKDISC

### Purpose

The BULKDISC command is used to initiate a bulk discovery request. You can request a bulk discovery by issuing the BULKDISC command from your TSO session or from within a batch job. You can run a complete job discovery manually or each time a planning activity is performed (create, extend or replan), by setting the MONOPTS runtime option for the controller.

A request generated by the BULKDISC command is communicated to the controller as an event record processed by the event writer task of a tracker. The BULKDISC command can be issued from any z/OS® system that runs a controller from IBM® Z Workload Scheduler Release 3, or later. The command initiates a bulk discovery. While the bulk discovery is in progress, the current plan backup resource is locked to prevent other IBM® Z Workload Scheduler tasks from updating the same file concurrently.

### Usage

You can invoke the BULKDISC command as a TSO command or by using a batch job that runs program EQQVPGM. If you invoke BULKDISC as a TSO command, you must allocate the EQQLIB data set to the address space of the TSO user, either by adding DD statements to the logon procedure, or by using the ALLOC command after TSO logon. In the TSO environment, error messages and trace records are sent directly to the terminal user. Messages are not delivered to indicate successful command completion.

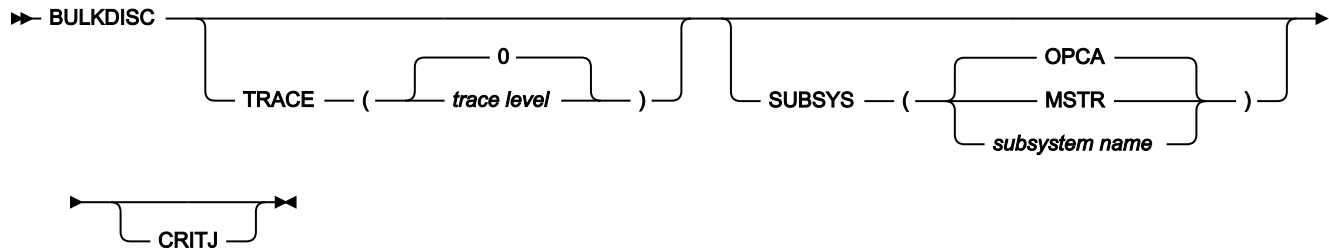
Use of the BULKDISC command can be restricted with the fixed resource code BUL. The authority of the requester is verified by the subsystem name identified in the command if an AUTHDEF statement is defined for that subsystem. When SUBSYS(MSTR) is specified, all subsystems defined on the z/OS® system to which the command is directed attempt to verify the authority of the requester before an event is generated. It is possible to be rejected by one subsystem and accepted by another.

You must be defined with update authority to the BUL resource to use the BULKDISC command. Resource codes are described in *Customization and Tuning*. Check with your administrator to confirm that you have this authority before you use BULKDISC.

If the subsystem to which you direct the command is not active when the command is issued, an event is generated and queued in CSA together with other job-tracking events. In this case, the authority of the requester is verified using the class name specified in the AUTHDEF statement when the subsystem was last started. If the subsystem has not been started

since a z/OS® IPL, no authority verification can be performed. For the bulk discovery to run successfully, the monitoring task must be active when the BULKDISC command is issued.

## Format



## Parameters

### CRITJ

If you work with IBM® Tivoli® Monitoring and specified the CRITICAL value in the MONPOL initialization statement, use this parameter to obtain all the available information about the critical jobs or the jobs in a critical path.

### SUBSYS (MSTR | *subsystem name* | OPCA)

The name of the controller that the BULKDISC command is directed to. The name can be up to four characters long. The first character must be alphabetic; the remaining characters must be alphanumeric. All lowercase characters are converted to uppercase. If the SUBSYS keyword is omitted, the on-request BULKDISC command is directed to a subsystem called OPCA.

If you specify **MSTR**, the BULKDISC command is directed to all controller subsystems on the z/OS® system that the BULKDISC command was directed to.



**Note:** Specifying **MSTR** causes multiple bulk discoveries to occur, if more than one controller subsystem is active on the system.

*Subsystem name* is the name of the subsystem that the BULKDISC command is directed to. If the tracker and controller run in separate address spaces in your IBM® Z Workload Scheduler configuration, specify the name of the tracker subsystem in this parameter.

### TRACE (*level* | 0)

Event tracing indicator. When a positive number is specified, a trace entry is created for each event generated by the BULKDISC command. The trace record is written to the message log file identified by ddname **EQQMLOG**. The record identifies the name of each receiving subsystem. The default value 0 does not generate trace records.

## Example

## Examples

These two examples demonstrate how you can use the BULKDISC command in TSO, or in a batch job (using the batch program EQQEVPGM).

### BULKDISC

#### Example 1 - TSO command

```
ALLOC F(EQQMLIB) DA('OPC.MESSAGE.LIBRARY') SHR REUSE

BULKDISC SUBSYS(OPCA)
```

#### Example 2 - Batch job

```
//BULKJOB JOB (ACCOUNT),'Bulkdisc',CLASS=A
//STEP1 EXEC PGM=EQQEVPGM
//STEPLIB DD DSN=OPC.LOAD.MODULE.LIBRARY,DISP=SHR
//EQQMLIB DD DSN=OPC.MESSAGE.LIBRARY,DISP=SHR
//EQQMLOG DD SYSOUT=A
//SYSIN DD *
BULKDISC SUBSYS(OPCA)
/*
```

## JSUACT

### Purpose

The JSUACT command is used to activate or inactivate the job submission function.

### Usage

You can invoke the JSUACT command as a TSO command or by using a batch job that executes program EQQEVPGM. If you invoke JSUACT as a TSO command, you must allocate the EQQMLIB data set to the TSO user address space, either by adding DD statements to the logon procedure, or by using the ALLOC command after TSO logon. In the TSO environment, error messages and trace records are sent directly to the terminal user. Messages are not delivered to indicate successful command execution.

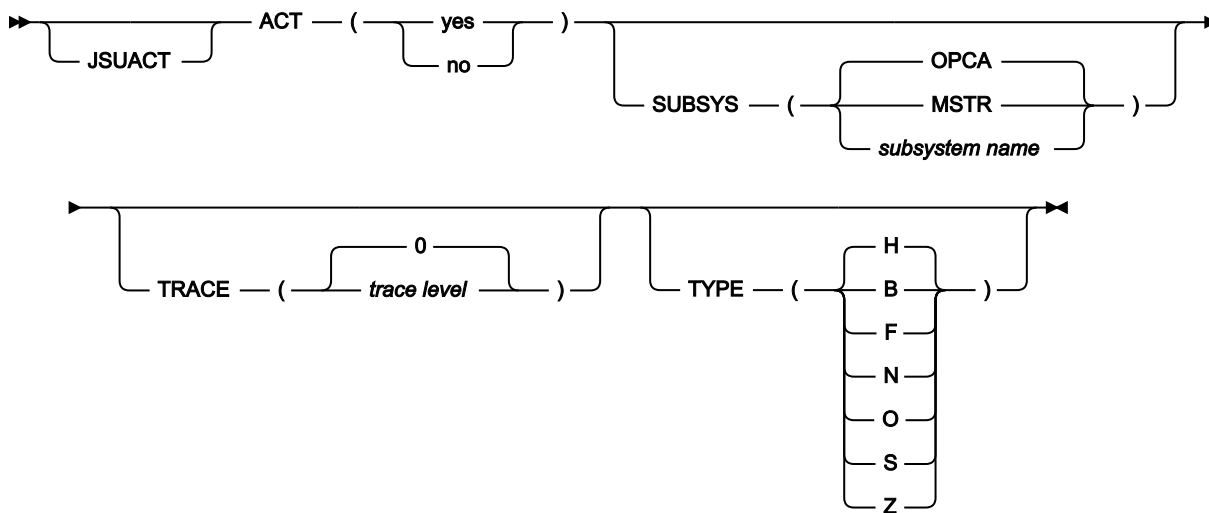
In any case, appropriate messages are issued in the controller log to indicate whether or not job submission has been activated.

Use of the JSUACT command can be restricted using the fixed resource JSUB. The authority of the requester is verified by the subsystem name identified in the command if an AUTHDEF statement is defined for that subsystem. When SUBSYS(MSTR) is specified, all subsystems defined on the z/OS® system to which the command is directed will attempt to verify the authorization. It is possible to be rejected by one subsystem and accepted by another.

You must be defined with update authority to the JSUB resource to be able to use the JSUACT command. Resource codes are described in *Customization and Tuning*. Check with your scheduler administrator to confirm that you have this authority before trying to use JSUACT.

The subsystem to which you direct the command does not have to be active when the command is issued. An event will be generated and queued in CSA along with other job-tracking events. If a subsystem is not active when the command is issued, the authorization of the requester is verified by using the class name specified in the AUTHDEF statement when the subsystem was started. If the subsystem has not been started since a z/OS® IPL, no authorization verification can be performed.

**Format**



**Parameters**

**ACT (Y | N)**

If you want to activate the job submission function specify Y, otherwise N.

**SUBSYS (MSTR | *subsystem name* | OPCA)**

The name of the tracker the JSUACT is directed to.

This parameter can be four characters in length. The first character must be alphabetic; the remaining characters must be alphanumeric. All lowercase characters are converted to uppercase.

If you specify **MSTR**, the JSUACT command is directed to all scheduler subsystems on the z/OS® system where the JSUACT command was issued.

**TRACE (*level* | 0)**

Event tracing indicator. When a nonzero positive number is specified, a trace entry is created for each event generated by the JSUACT command. The trace record is written to the message log file identified by ddname EQQMLOG. The record identifies the name of each receiving subsystem. The default value 0 will not generate trace records.

**TYPE (B | F | H | N | O | S | Z)**

Indicates whether the job submission must be activated or deactivated for:

- All the workload (B)
- Fault-tolerant workstations (F)
- z-Centric and host workstations (H)
- Fault-tolerant and host workstations (N)
- Fault-tolerant and z-centric workstations (O)
- host workstations (S)
- z-Centric workstations (Z)

### Example

### Examples

These two examples demonstrate how you can use the JSUACT command in TSO, or in a batch job (using the batch program EQQEVPGM).

#### JSUACT

##### Example 1 - TSO command

```
ALLOC F(EQQMLIB) DA('OPC.MESSAGE.LIBRARY') SHR REUSE

JSUACT ACT(YES) SUBSYS(OPCB)
```

##### Example 2 - Batch job

```
//JSUACT JOB (ACCOUNT),'Deactivate',CLASS=A
//STEP1 EXEC PGM=EQQEVPGM
//STEPLIB DD DSN=OPC.LOAD.MODULE.LIBRARY,DISP=SHR
//EQQMLIB DD DSN=OPC.MESSAGE.LIBRARY,DISP=SHR
//EQQMLOG DD SYSOUT=A
//SYSIN DD *
JSUACT ACT(N) SUBSYS(OPCB)
/*
```

## OPINFO

### Purpose

Use the OPINFO command to set the *user data* of an operation in the current plan. The user data can contain any data you require. In many installations, the field is used to record the problem record number for operations that have ended in error. OPINFO can be incorporated into your Information Management interface to enable immediate feedback of the problem report to IBM® Z Workload Scheduler.

### Usage

You can invoke OPINFO as a TSO command or by using a batch job which executes program EQQEVPGM. If you invoke OPINFO as a TSO command, allocate the EQQMLIB data set to the address space of the TSO user, either by adding DD statements to the logon procedure, or by using the ALLOC command after TSO logon. In the TSO environment, error

messages and trace records are sent directly to the terminal user. Messages are not delivered to indicate successful command execution.

You use the OPINFOSCOPE keyword of the JTOPTS to specify the scope of the command. If OPINFOSCOPE is set to (or defaults to) IP, IBM® Z Workload Scheduler considers only in-progress operations (where the current status is `R`, `A`, `*`, `S`, `I`, or `E`), and will not action OPINFO events for operations in any other status. If OPINFOSCOPE is set to ALL, IBM® Z Workload Scheduler also considers operations with `w` and `c` status.

With the OPINFO command, you specify the WSNAMES, ADID, IA, OPNUM, or JOBNAME parameters to identify the operation whose user data is to be updated. If the OPINFOSCOPE keyword is IP, which is the default, WSNAMES is a required parameter. If OPINFOSCOPE is ALL, the ADID or JOBNAME parameter is required. Also, for printer workstations, you can specify the CLASS and FORM parameters to identify the operation.

If you do not provide enough information to uniquely identify the operation, and IBM® Z Workload Scheduler finds more than one operation that matches your parameters, IBM® Z Workload Scheduler chooses the best operation. If OPINFOSCOPE is IP, IBM® Z Workload Scheduler uses this list, taking each item until it gets a single operation:

1. Priority 9
2. Earliest latest start time
3. Highest priority, if lower than 9
4. Earliest operation input arrival time, or occurrence input arrival time if the operation does not have input arrival specifically defined
5. Longest in Ready status

That is, if IBM® Z Workload Scheduler determines that there is more than one in-progress operation in the current plan, the operation with priority 9 is updated. If more than one operation specifies priority 9, or there are no priority 9 operations, the operation with the earliest latest start time is updated. The latest start is blank if the operation is added using the MCP panel. If the latest start time is equal, the operation with the highest priority is updated, and so on.

If OPINFOSCOPE is ALL, IBM® Z Workload Scheduler uses the same list as for OPINFOSCOPE(IP) to find a single operation. If no in-progress operation that matches your parameters is found, IBM® Z Workload Scheduler also searches operations with status `c` and `w` in the current plan. The operation with the earliest latest start time is selected.

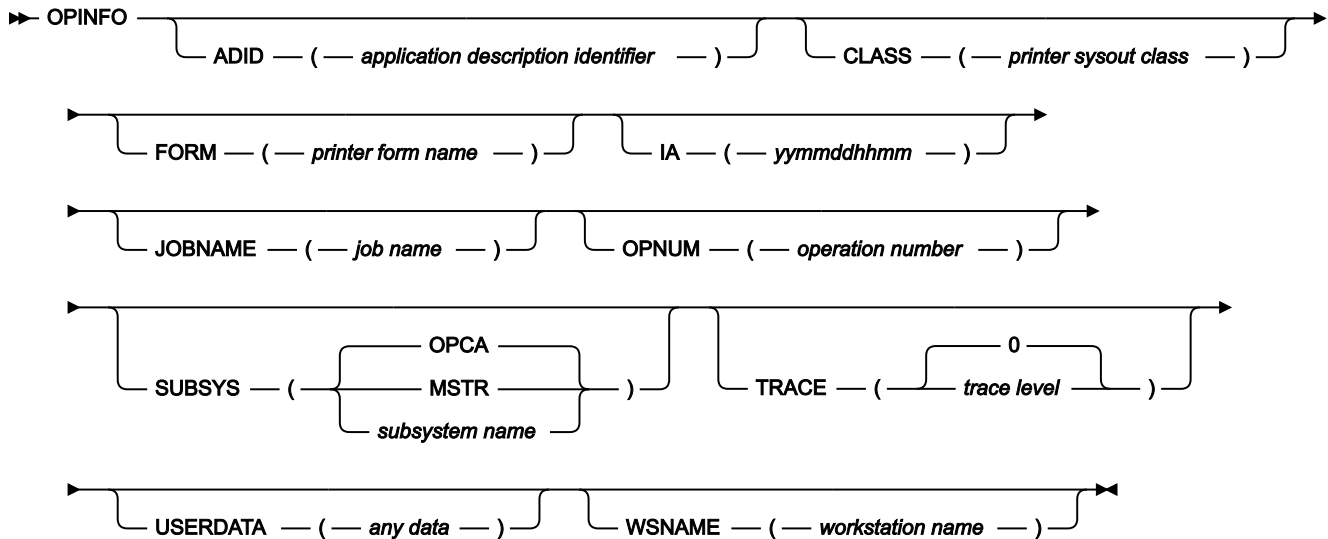
Use of the OPINFO command can be restricted with the fixed resource code CP. The authority of the requester is verified by the subsystem name identified in the command if an AUTHDEF statement is defined for that subsystem. When SUBSYS(MSTR) is specified, all tracker subsystems defined on the z/OS® system where the OPINFO command is issued will attempt to verify the authority of the requester before data will be passed. You might be rejected by one subsystem and accepted by another.

You need update authority to resource code CP to use this command. Resource codes are described in *Customization and Tuning*.

The subsystem to which you direct the command does not have to be active when the command is issued. An event will be generated and queued in CSA along with other job-tracking events. If the subsystem is not active when the command is issued, the authority of the requester is verified using the class name specified in the AUTHDEF statement when the

subsystem was last started. If the subsystem has not been started since a z/OS® IPL, no authority verification can be performed.

## Format



## Parameters

### ADID (*application description identifier*)

The application identifier of the operation whose user data is to be updated. If the OPINFOSCOPE keyword of the JTOPTS statement is ALL, ADID is a required parameter.

### CLASS (*printer sysout class*)

For a printer workstation, specifies the printer SYSOUT class of the operation whose user data is to be updated.

### FORM (*printer form name*)

For a printer workstation, specifies the printer FORM name of the operation whose user data is to be updated.

### IA (*yymmddhhmm*)

The input arrival date and time of the occurrence containing the operation whose user data is to be updated. You must specify this in the format *yymmddhhmm*.



**Note:** IBM® Z Workload Scheduler interprets the yy part as follows:

YY

Year

72 - 99

1972 - 1999



00 - 71

2000 - 2071

**JOBNAME** (*job name*)

The job name associated with the operation whose user data is to be updated. If the OPINFOSCOPE keyword of the JTOPTS statement is ALL, JOBNAME is a required parameter.

**OPNUM** (*operation number*)

The operation number of the operation whose user data is to be updated.

**SUBSYS** (**MSTR** | *subsystem name* | **OPCA**)

Name of the tracker subsystem that the OPINFO command is directed to. The name can be up to 4 characters long. The first character must be alphabetic; the remaining characters must be alphanumeric. All lowercase characters are converted to uppercase.

If you specify **MSTR**, the OPINFO command is directed to all tracker subsystems on the z/OS® system where the OPINFO command is issued.



**Note:** If the trackers and controller run on different subsystems in your configuration, specify the name of the tracker subsystem in this parameter.

**TRACE** (*level* | **0**)

Event tracing indicator. When a positive number is specified, a trace entry is created for each event generated by the OPINFO command. The trace record is written to the message log file identified by ddname EQQMLOG. The record identifies the name of each receiving subsystem. The default value 0 will not generate trace records.

**USERDATA** (*any data*)

You can use this 16-character parameter to pass information about an operation to the current plan in the *operation user data*. The USERDATA field cannot contain any blanks.

**WSNAME** (*workstation name*)

The name of the workstation for the operation whose user data is to be updated. If the OPINFOSCOPE keyword of the JTOPTS statement is IP, WSNAME is a required parameter.

**Example****Examples**

These two examples demonstrate how you can use the OPINFO command in TSO, or in a batch job (using the batch program EQQEVPGM).

**OPINFO****Example 1 - TSO command**

```
ALLOC F(EQQMLIB) DA('OPC.MESSAGE.LIBRARY') SHR REUSE
OPINFO W(BDEC) J(DNCD3000) A(ACLMSDLY) U(USER&DATA&HERE)
```

### Example 2 - Batch job

```
//OPINFOUS JOB (ACCOUNT),'Set completed',CLASS=A
//STEP1 EXEC PGM=EQQEVPGM
//STEPLIB DD DSN=OPC.LOAD.MODULE.LIBRARY,DISP=SHR
//EQQMLIB DD DSN=OPC.MESSAGE.LIBRARY,DISP=SHR
//EQQMLOG DD SYSOUT=A
//SYSIN DD *
OPINFO W(BDEC) J(DNCD3000) A(ACLMSDLY) U(PROBREC&01234567)
/*
```

In both of these examples, an operation for application ACLMSDLY at workstation BDEC will have the data fed back to it.

## OPSTAT

### Purpose

The OPSTAT command lets you set the status of an operation at any workstation, except workstations that have the nonreporting attribute. Events generated by OPSTAT are matched against operations on the ready list. Events received for operations in waiting (w), suppressed by condition (x), or complete (c) status are ignored. Jobs and started tasks that are running are always allowed to finish; for rules governing the changing of operation status, see the READY LIST panel description in Using the ready list .

You need update authority to resource code RL to use this command. Resource codes are described in *Customization and Tuning*.

The OPSTAT command gives you a portable method of using automatic event reporting (AER) facility. AER can help you coordinate many tasks that are not normally seen by IBM® Z Workload Scheduler. For example, you can use AER to trigger the start of an operation when a particular step in a job is complete, or as acknowledgment that a file has been received across the network.

The OPSTAT command also gives you the facility to automatically report the status of work executing in operating environments that do not support a tracker.

### Usage

You can invoke OPSTAT as a TSO command or by using a batch job that executes program EQQEVPGM. If you invoke OPSTAT as a TSO command, allocate the EQQMLIB data set to the address space of the TSO user, either by adding DD statements to the logon procedure, or by using the ALLOC command after TSO logon. In the TSO environment, error messages and trace records are sent directly to the terminal user. Messages are not delivered to indicate successful command execution.

OPSTAT is an alternative to the EQQUSIN subroutine for implementing automatic event reporting for general workstations.

With the OPSTAT command, you must specify the WSNAME parameter to identify the workstation at which the operation is changing status. IBM® Z Workload Scheduler then changes the status of the operation at that workstation to the status you have specified.

If there is more than one operation at the workstation, you can optionally specify the ADID, IA, OPNUM, or JOBNAME parameters to identify the particular operation whose status is to be changed. Also, for printer workstations, you can specify the CLASS and FORM parameters to identify an operation.

If you do not provide enough information to uniquely identify the operation and IBM® Z Workload Scheduler finds more than one operation which matches the criteria you specified, IBM® Z Workload Scheduler determines the most appropriate operation to update. IBM® Z Workload Scheduler chooses the most appropriate operation by investigating its characteristics in this order:

1. The operation has priority 9.
2. Earliest latest start time.
3. Priority 8-1.
4. Input arrival time specified for the operation, or the occurrence input arrival if the operation does not have input arrival specifically defined.

Therefore, if you define only the WSNAME parameter and IBM® Z Workload Scheduler determines that there is more than one operation in the current plan for that workstation in status R, A, \*, S, I, or E, the operation with priority 9 is updated. If more than one operation specifies priority 9, or there are no priority 9 operations, the operation with the earliest latest-start time is updated. If you add the operation using the MCP panel, the latest start time is blank. If latest start is equal, the operation with the highest priority is updated. If priority is equal, the operation which specifies the earliest input arrival time is updated. If input arrival is also equal, the update is performed on a first-in-first-out basis.

The parameter SORTIA can be specified among OPSTAT input parameters to automatically choose the oldest eligible operation in the ready list, in terms of occurrence input arrival time. SORTIA is processed and effectively used by the event manager when all the following conditions occur:

- Workstation, application name and operation number are all specified as input parameters for OPSTAT.
- Neither input arrival time nor token are specified as input parameters to OPSTAT.

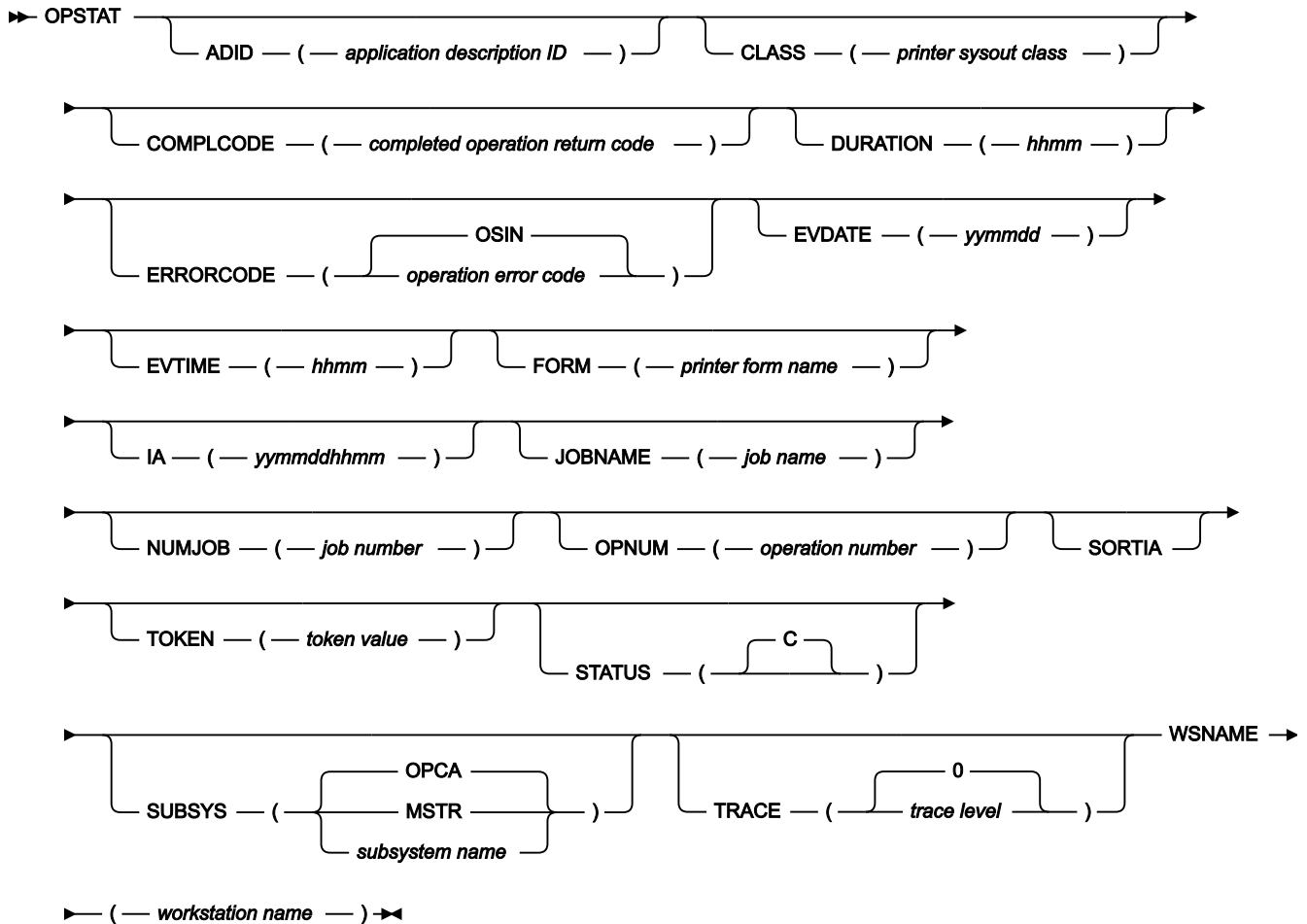
In any different situation the keyword SORTIA will be ignored.

Use of the OPSTAT command can be restricted with fixed resource code RL and the subresource RL.WSNAME. The authority of the requester is verified by the subsystem name identified in the command if an AUTHDEF statement is defined for that subsystem. When SUBSYS(MSTR) is specified, all tracker subsystems defined on the z/OS® system where the OPSTAT command is issued will attempt to verify the authority of the requester before an event is generated. You might be rejected by one subsystem and accepted by another.

The subsystem to which you direct the command does not have to be active when the command is issued. An event will be generated and queued in CSA along with other job-tracking events. If the subsystem is not active when the command is issued, the authority of the requester is verified using the class name specified in the AUTHDEF statement when the

subsystem was last started. If the subsystem has not been started since a z/OS® IPL, no authority verification can be performed.

## Format



## Parameters

### ADID (*application description identifier*)

The application identifier of the operation whose status you want to change.

### CLASS (*printer sysout class*)

For a printer workstation, specifies the printer SYSOUT class of the operation whose status you want to change.

### COMPCODE (*completed operation return code*)

If you are specifying STATUS(C) to set the operation status to *complete*, you can optionally specify a return code for the completed operation. It can be any 4 characters.

**DURATION (hhmm)**

If you are specifying STATUS(C) to set the operation status to *complete*, you can optionally specify a duration for the completed operation. You specify the duration in hours and minutes, in the format *hhmm*.

**ERRORCODE (operation error code | OSIN)**

If you are specifying STATUS(E) to set the operation status to *ended-in-error*, it is required that you specify an error code for the operation. The error code can be any 4 characters. The default value is OSIN.

**EVDATE (yymmdd)**

The date of this operation status event. You must specify the date in the format *yymmdd*.

See the comments under the following parameter, EVTIME.

**EVTIME (hhmm)**

The time of this operation status event. You must specify the time in the format *hhmm*.

You can use the EVDATE and EVTIME parameters if you want to indicate that the operation changed status at a time other than the current time. If you do not supply these parameters, the operation is considered to have changed status at the time IBM® Z Workload Scheduler processes the OPSTAT command.

**FORM (printer form name)**

For a printer workstation, specifies the printer FORM name of the operation whose status you want to change.

**IA (yymmddhhmm)**

The input arrival date and time of the occurrence containing the operation whose status you want to change. You must specify this in the format *yymmddhhmm*.



**Note:** IBM® Z Workload Scheduler interprets the `yy` part as follows:

**YY**

**Year**

**72 - 99**

1972 - 1999

**00 - 71**

2000 - 2071

**JOBNAME (job name)**

The job name associated with the operation whose status you want to change.

**NUMJOB (job number)**

Use this optional parameter to specify a job number for an operation on a user-defined computer automatic workstation. Specify a number from 0 to 999999. IBM® Z Workload Scheduler builds a job number in the format `USRnnnnnn` or `Unnnnnnnn`, depending on the value and padding the number with zeros on the left.

**OPNUM (operation number)**

The operation number of the operation whose status you want to change.

**SORTIA**

Makes the code to automatically choose the oldest eligible operation in the ready list, in terms of occurrence input arrival time. It applies when both the following conditions occur:

- Workstation, application name and operation number are all specified as input parameters for OPSTAT.
- Neither input arrival time nor token are specified as input parameters to OPSTAT.

In any different situation the keyword SORTIA will be ignored.

**TOKEN (token value)**

The token assigned for the operation whose status you want to change. A token is automatically assigned for operations started on workstations that specify a user-defined destination ID. The token can be used to uniquely identify the operation.

When TOKEN is used in conjunction with the ADID, IA, JOBNAME, or OPNUM parameters, all values must match the target operation. For example, if the token identifies the operation but OPNUM is also specified and does not match, the event will be rejected and message EQQE091E written to the controller message log.

Specify the token as a hexadecimal value in the format TOKEN(X'00ABCDEF').

**STATUS (C)**

Use the default value to set the new operation status as successfully completed at the workstation.

Changes to operation status using OPSTAT follow the same rules as status changes in the Ready List. In particular:

- You cannot change an operation status from W to C. This is because predecessor jobs might not be completed.
- You cannot change an operation status from X (suppressed by condition) or to X.
- To change the status of an operation to the previous logical status, specify STATUS(X). As from the previous point, do not use it to change the operation status to suppressed by condition.

For more information, refer to Using the ready list.

**SUBSYS (MSTR | *subsystem name* | OPCA)**

Name of the tracker subsystem that the OPSTAT command is directed to. The name can be up to 4 characters long. The first character must be alphabetic; the remaining characters must be alphanumeric. All lowercase characters are converted to uppercase.

If you specify **MSTR**, the OPSTAT command is directed to all tracker subsystems on the z/OS® system where the OPSTAT command is issued.



**Note:** If the trackers and controller run on different subsystems in your configuration, specify the name of the tracker subsystem in this parameter.

### TRACE (*level* | 0)

Event tracing indicator. When a positive number is specified, a trace entry is created for each event generated by the OPSTAT command. The trace record is written to the message log file identified by ddname EQQMLOG. The record identifies the name of each receiving subsystem. The default value 0 will not generate trace records.

### WSNAME (*workstation name*)

You must supply the name of the workstation that you are reporting the status of an operation for.

### Example

### Examples

These examples demonstrate how you can use the OPSTAT command in TSO, or in a batch job (using the batch program EQQVPGM).

### OPSTAT

#### Example 1 - TSO command

```
ALLOC F(EQQMLIB) DA('OPC.MESSAGE.LIBRARY') SHR REUSE
OPSTAT W(BDEC) ST(C) J(DNCD3000) A(ACLMSDLY)
```

#### Example 2 - Batch job

```
//OPSTATUS JOB (ACCOUNT),'Set completed',CLASS=A
//STEP1 EXEC PGM=EQQVPGM
//STEPLIB DD DSN=OPC.LOAD.MODULE.LIBRARY,DISP=SHR
//EQQMLIB DD DSN=OPC.MESSAGE.LIBRARY,DISP=SHR
//EQQMLOG DD SYSOUT=A
//SYSIN DD *
OPSTAT W(BDEC) ST(C) J(DNCD3000) A(ACLMSDLY)
/*
```

In both of these examples, an operation for application ACLMSDLY at workstation BDEC is reported as completed.

## SRSTAT

### Purpose

The SRSTAT command lets you change the overriding (global) availability, quantity, and deviation of a special resource. You can do this to prevent operations from allocating a particular resource, or to request the ETT function to add an application occurrence to the current plan.

## Usage

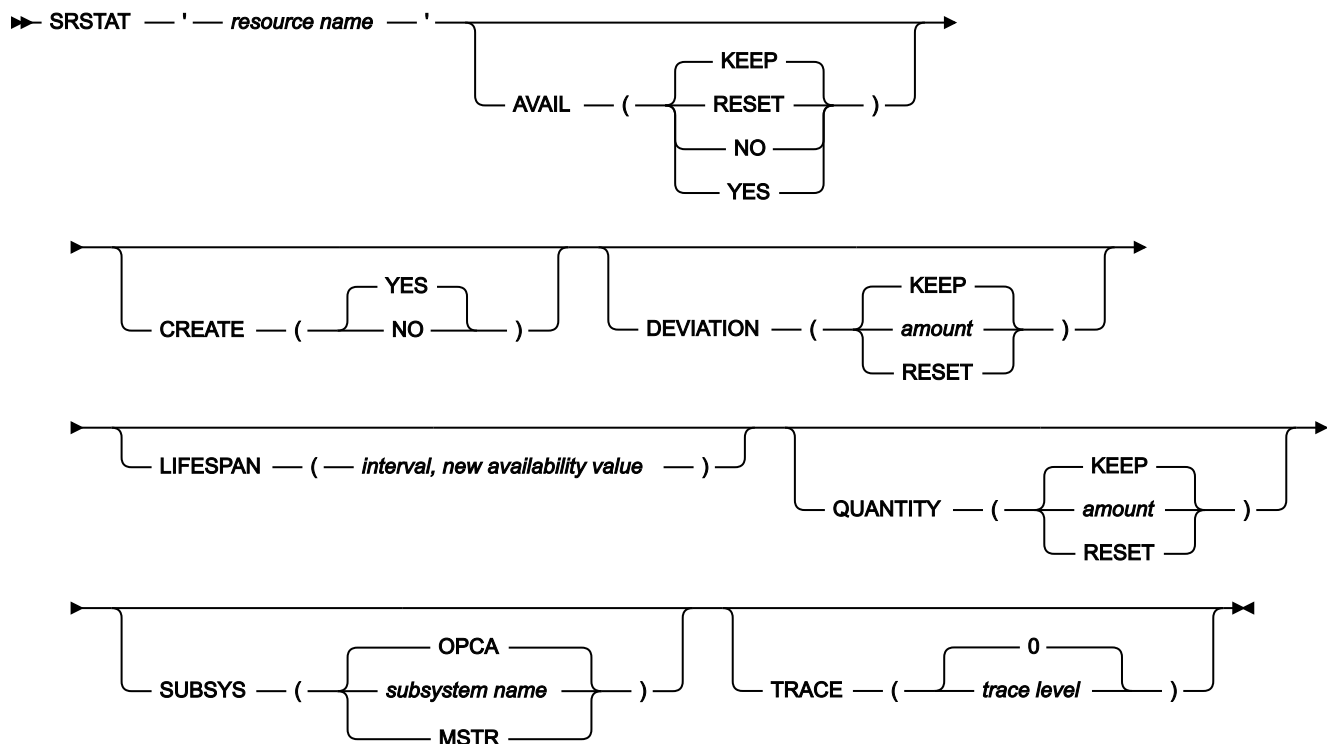
You can invoke SRSTAT as a TSO command or by using a batch job which executes program EQQEVPGM. If you invoke SRSTAT as a TSO command, you must allocate the EQQMLIB data set to the address space of the TSO user, either by adding DD statements to the logon procedure, or by using the ALLOC command after TSO logon. In the TSO environment, error messages and trace records are sent directly to the terminal user. Messages are not delivered to indicate successful command execution.

Use of the SRSTAT command can be restricted with fixed resource code SR and subresource SR.SRNAME. The authority of the requester is verified by the subsystem name identified in the command if an AUTHDEF statement is defined for that subsystem. When SUBSYS(MSTR) is specified, all tracker subsystems defined on the z/OS® system where the SRSTAT command is issued will attempt to verify the authority of the requester before an event is generated. It is possible to be rejected by one subsystem and accepted by another.

The subsystem to which you direct the command does not have to be active when the command is issued. An event will be generated and queued in CSA along with other job-tracking events. If the subsystem is not active when the command is issued, the authority of the requester is verified using the class name specified in the AUTHDEF statement when the subsystem was last started. If the subsystem has not been started since a z/OS® IPL, no authority verification can be performed.

See Creating special resources for information about special resources, how to connect them to workstations, and how to specify intervals.

## Format



## Parameters

### **'resource name'**

The name of the resource whose availability or quantity you want to change. This parameter must be contained within single quotation marks and can be up to 44 characters in length. All lowercase characters are converted to uppercase.

You must supply this parameter.

### **AVAIL (RESET | NO | YES | KEEP)**

YES indicates that the availability status of the resource should be set to YES. Operations requiring the resource can start, as long as there are no other conditions preventing them from starting. For example, if two operations both require a resource for exclusive (X) use, IBM® Z Workload Scheduler lets only one of them start.

NO indicates that the availability status of the resource should be set to NO. Any operation that requires the resource will be prevented from starting, regardless of whether it was specified for shared (S) or exclusive (X) use.

RESET sets the overriding availability to blank so that the interval or default value is used.

KEEP, the default, does not change the availability status.

When you set the availability with SRSTAT (or other interfaces such as the EQQUSIN subroutine or the MCP panel), the specified availability lasts over interval boundaries, even though the next interval can specify a different availability, and persists after a daily planning job. Specify RESET to restore the planned availability.

### **CREATE (NO| YES)**

NO indicates that the resource is not to be added to the current plan of the receiving IBM® Z Workload Scheduler subsystem, if it does not exist in the database. If the resource exists in the database, CREATE(NO) does not have any effect. You can specify CREATE(NO) if the resource is being used only as a means to generate an event for ETT—the event is generated even if the resource does not exist.

If YES is specified or defaulted, and the DYNAMICADD keyword of the RESOPTS initialization statement is set to YES or EVENT, IBM® Z Workload Scheduler adds the resource to the current plan of the receiving IBM® Z Workload Scheduler subsystem, if the resource is not in the database. It uses the following default values:

#### **Text**

Blank.

#### **Specres group ID**

Blank.

#### **Hiperbatch**

No.

**Used for**

Control.

**On error**

Blank. If an error occurs, IBM® Z Workload Scheduler uses the value specified in the operation details or, if that is also blank, the value of the ONERROR keyword of RESOPTS.

**On Complete**

Blank.

When the operation completes, the On Complete value is considered according to the following order:

1. The On Complete value set at operation definition level, if not blank.
2. The On Complete value set at special resource definition level, if not blank.
3. The ONCOMPLETE or DYNONCOMPLETE keyword value, respectively set for the not dynamically added resources or the dynamically added resources, in all the other cases.

**Max Usage Limit**

0.

**Max Usage Type**

Reset.

**Usage Counter**

0.

**Overriding availability, quantity, and deviation**

The value specified by SRSTAT, or blank.

**Default quantity**

1. The default quantity is automatically increased if contention occurs.

**Default availability**

Yes.

**Intervals**

No intervals are created.

**Workstations**

\* (all workstations can allocate the resource).

**DEVIATION (RESET | *amount* | KEEP)**

To make a temporary change to the quantity, you can specify a *deviation*, which is an amount to be added to (positive number) or subtracted from (negative number) the current quantity. A specified amount can be from -999 999 to +999 999. The default, KEEP, does not alter the deviation.



**Note:** The effect of deviation is cumulative. If you issue two SRSTAT commands with DEVIATION(-1), for example, this subtracts *two* from the deviation.

When the deviation is not zero, the value lasts over interval boundaries and persists after a daily planning job. Specify RESET or zero (0) to set the deviation to zero.

### **LIFESPAN (*interval*, *new availability value*)**

The interval of time, in minutes, after which the global availability of the special resource is changed according to the value specified as *new availability value*. When you specify this parameter, a pending LIFESPAN action (the one specified as *new availability value*) comes into effect for the resource. The controller runs that action as soon as the LIFESPAN interval of time expires.

Only one pending LIFESPAN action can exist for one resource. This means that issuing an SRSTAT command with LIFESPAN replaces any existing pending LIFESPAN action.

To cancel a pending LIFESPAN action, issue an SRSTAT command with an interval of 0. When the interval of time is 0, no LIFESPAN action is run.

*Interval* can be an integer from 0 to 99999. *New availability value* can be one of the following (you must always specify one, because no default is provided):

#### **YES**

The global availability is changed to Yes

#### **NO**

The global availability is changed to No

#### **RESET**

The global availability is changed to blank

### **QUANTITY (RESET | *amount* | KEEP)**

To change the overriding (global) quantity, specify the amount, from 1 to 999 999.

RESET sets the overriding quantity to blank so that the interval or default value is used. KEEP does not alter the quantity.

When you set the quantity with SRSTAT (or other interfaces such as the EQQUSIN subroutine or the MCP panel), the specified quantity lasts over interval boundaries, even though the next interval can specify a different quantity, and persists after a daily planning job. Specify RESET to restore the planned quantity.

### **SUBSYS (*subsystem name* | MSTR | OPCA)**

The name of the tracker subsystem that the SRSTAT command is directed to. This parameter can be up to 4 characters long. The first character must be alphabetic; the remaining characters must be alphanumeric. All lowercase characters are converted to uppercase.

If you specify **MSTR**, the SRSTAT command is directed to all tracker subsystems on the z/OS® system where the SRSTAT command is issued.



**Note:** If the trackers and controller in your configuration run on different subsystems, specify the name of the tracker subsystem in this parameter.

### TRACE (*level* | 0)

Event tracing indicator. When a positive number is specified, a trace entry is created for each event generated by the SRSTAT command. The trace record is written to the message log file identified by ddname EQQMLOG. The record identifies the name of each receiving subsystem. The default value 0 will not generate trace records.

### Example

### Examples

These examples demonstrate how you can use SRSTAT in TSO, or in a batch job (using the batch program EQQVPGM).

### SRSTAT

#### Example 1 - TSO command

```
ALLOC F(EQQMLIB) DA('OPC.MESSAGE.LIBRARY') SHR REUSE

SRSTAT 'IMS.DATA.BASE' SUBSYS(OPCB) AVAIL(YES)
```

#### Example 2 - Batch job

```
//CHSTATUS JOB (ACCOUNT),'Change DB status',CLASS=A
//STEP1 EXEC PGM=EQQVPGM
//STEPLIB DD DSN=OPC.LOAD.MODULE.LIBRARY,DISP=SHR
//EQQMLIB DD DSN=OPC.MESSAGE.LIBRARY,DISP=SHR
//EQQMLOG DD SYSOUT=A
//SYSIN DD *
SRSTAT 'IMS.DATA.BASE' SUBSYS(OPCB) AVAIL(YES)
/*
```

#### Example 3 - Reduce tape pool

```
ALLOC F(EQQMLIB) DA('OPC.MESSAGE.LIBRARY') SHR REUSE

SRSTAT 'TAPES' SUBSYS(OPCB) DEV(RESET) Q(6)
SRSTAT 'TAPES' SUBSYS(OPCB) DEV(-1)
SRSTAT 'TAPES' SUBSYS(OPCB) DEV(-1)
SRSTAT 'TAPES' SUBSYS(OPCB) DEV(0)
```

In examples 1 and 2, the availability status of the resource IMS™.DATA.BASE is changed to YES. In example 3, the number of tapes is set to 6, 5, 4, and then back to 6 (a deviation of 0 is a special value and means the same as reset).

# WSSTAT

## Purpose

The WSSTAT command lets you change the status of a workstation in the current plan. The status information is communicated to the controller to indicate a workstation as active, offline, or failed. When you use the WSSTAT command you can optionally define restart and routing options for the workload defined on the workstation when you are reporting a status of offline or failed.

## Usage

You can invoke WSSTAT as a TSO command or by using a batch job which executes program EQQEVPGM. If you invoke WSSTAT as a TSO command, you must allocate the EQQMLIB data set to the address space of the TSO user, either by adding DD statements to the logon procedure, or by using the ALLOC command after TSO logon. In the TSO environment, error messages and trace records are sent directly to the terminal user. Messages are not delivered to indicate successful command execution.

Use of the WSSTAT command can be restricted with fixed resource code RL and subresource RL.WSSTAT. The authority of the requester is verified by the subsystem name identified in the command if an AUTHDEF statement is defined for that subsystem. When SUBSYS(MSTR) is specified, all tracker subsystems defined on the z/OS® system where the WSSTAT command is issued will attempt to verify the authority of the requester before an event is generated. It is possible to be rejected by one subsystem and accepted by another.

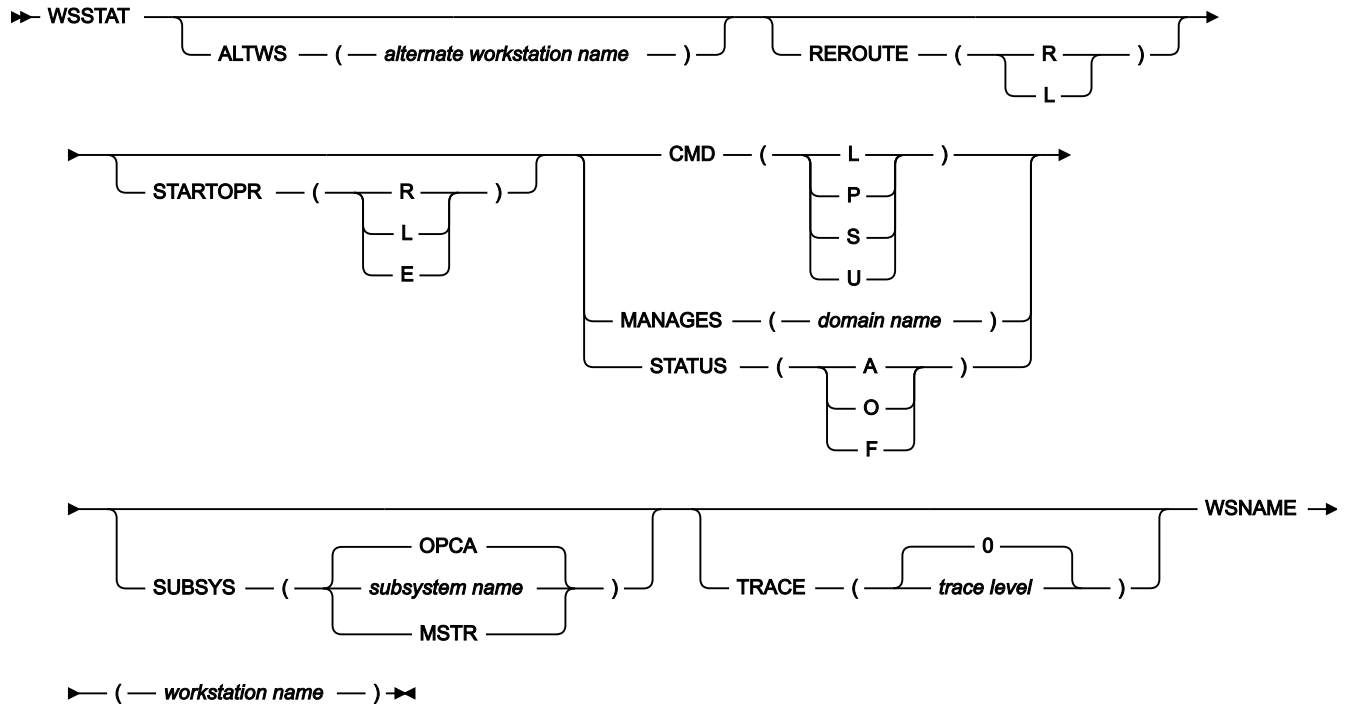
The subsystem to which you direct the command does not have to be active when the command is issued. An event will be generated and queued in CSA along with other job-tracking events. If the subsystem is not active when the command is issued, the authority of the requester is verified using the class name specified in the AUTHDEF statement when the subsystem was last started. If the subsystem has not been started since a z/OS® IPL, no authority verification can be performed.



**Note:** If the status of a workstation has been set to offline manually, using the panels, then you are not allowed to reset it to active using the WSSTAT command.

The status of a workstation is active and a WSSTAT is performed to change the status to offline. If the IBM® Z Workload Scheduler subsystem is stopped using a cancel command before a CP backup occurs, when the IBM® Z Workload Scheduler subsystem restarts the workstation status is active and not offline. The same behaviour happens when the status is varied from offline to active.

## Format



## Parameters

### ALTWS (*alternate workstation name*)

When the workstation status is set to offline or failed, you can specify the alternate workstation where reroutable operations should be started.

If this parameter is omitted the value defined for the current workstation open interval will be used. If the REROUTE parameter specifies L, or if the default specifies no rerouting, the value of ALTWS is ignored.

The parameter is optional.

### CMD (L | P | S | U)

Use this parameter to change the status of the fault-tolerant agent to one of the following statuses:

**L**

To link the workstation

**P**

To stop workstation

**S**

To start workstation

**U**

To unlink the workstation

You must specify either this or one of the alternative parameters.

### **MANAGES** (*domain name*)

Used in conjunction with the WSNAMES parameter, it instructs the workstation specified by WSNAMES to become the new domain manager of the domain specified by MANAGES. It cannot be specified in conjunction with STATUS or CMD. Example:

```
WSSTAT SUBSYS(TWST) WSNAMES(U001) MANAGES(UK-DM)
```

This command instructs the U001 workstation to become the new domain manager of the UK-DM domain. The command is sent through the TWST tracker subsystem.

The name of the domain, consisting of up to 16 characters, must start with a letter. It can contain alphanumeric characters, dashes and underscores. No other characters must be used because they can generate unpredictable situations.

### **REROUTE** (R | L)

This parameter is optional. When the workstation status is set to offline or failed, you can specify R for operations to be rerouted to the alternate workstation, or L for no rerouting (to leave the operations at the inactive workstation).

If this parameter is omitted, the value defined in either the WSOFFLINE or the WSFAILURE keyword on the JTOPTS initialization statement will be used as default.

### **STARTOPR** (R | E | L)

This parameter is optional. When the workstation status is set to offline or failed, you can specify what IBM® Z Workload Scheduler should do with operations that are currently in started status at the workstation, where:

#### **R**

Restart operations automatically on the alternate workstation.

#### **E**

Set all started operations to ended-in-error.

#### **L**

Leave the operations in started status.

If this parameter is omitted, the value defined in either the WSOFFLINE or the WSFAILURE keyword on the JTOPTS initialization statement is used as default.



#### **Note:**



1. If you select STARTOPR(E), a started job continues to run. IBM® Z Workload Scheduler never cancels jobs that have started.
2. For remote engine workstations, this parameter supports only the value L. If you specify any other value, it is forced to L.
3. If you set STARTOPR(R), a started job might be submitted again even if it is currently running, resulting in the same job being run twice.

**STATUS (A | O | F)**

The status you want to report for the workstation

**A**

Active

**O**

Offline

**F**

Failed

You must specify either this or one of the alternative parameters.

**SUBSYS (*subsystem name* | MSTR | OPCA)**

The name of the tracker subsystem that the WSSTAT command is directed to. This parameter can be up to 4 characters long. The first character must be alphabetic; the remaining characters must be alphanumeric. All lowercase characters are converted to uppercase.

If you specify **MSTR**, the WSSTAT command is directed to all tracker subsystems on the z/OS® system where the WSSTAT command is issued.



**Note:** If the tracker and controller in your configuration run on different subsystems, specify the name of the tracker subsystem in this parameter.

**TRACE (*level* | 0)**

Event tracing indicator. When a nonzero positive number is specified, a trace entry is created for each event generated by the WSSTAT command. The trace record is written to the message log file identified by dname EQQMLOG. The record identifies the name of each receiving subsystem. The default value 0 will not generate trace records.

**WSNAME (*workstation name*)**

The name of the workstation to be updated. This parameter is required.

The WSSTAT parameters are checked for validity and consistency. The validity checks are carried out in the tracker where the command is executed. The validity check processes parameter names, length and type of parameter values.



**Note:** If you try to set the status of a fault-tolerant workstation to Failed (an invalid status for this type of workstation), you do not receive an immediate message. Instead, you receive message EQQE112E in the controller message log, and the workstation status is not changed.

If the input is valid, a *workstation status event* is generated and communicated to the controller. The processing of the event includes a consistency check of the values specified in the parameters. The following consistency checks are made:

- The workstation name is checked for existence.
- The alternate workstation is checked for existence.
- If the value given in the STATUS parameter equals current status of the workstation, the command will be ignored.

### Example

### Examples

The following two examples demonstrate how you can use WSSTAT in TSO, or in a batch job (using the batch program EQQEVPGM).

#### WSSTAT

##### Example 1 - TSO command

```
ALLOC F(EQQMLIB) DA('OPC.MESSAGE.LIBRARY') SHR REUSE
WSSTAT SUBSYS(OPCB) WSNAME(AS4H) STATUS(O) START(R)
```

##### Example 2 - Batch job

```
//CHSTATUS JOB (ACCOUNT),'Change WS status',CLASS=A
//STEP1 EXEC PGM=EQQEVPGM
//STEPLIB DD DSN=OPC.LOAD.MODULE.LIBRARY,DISP=SHR
//EQQMLIB DD DSN=OPC.MESSAGE.LIBRARY,DISP=SHR
//EQQMLOG DD SYSOUT=A
//SYSIN DD *
WSSTAT SUBSYS(OPCB) WSNAME(AS4H) STATUS(A)
/*
```

In the first example the status of workstation AS4H is set to offline. Started operations will be restarted on the alternate workstation.

In the second example the status of workstation AS4H will be set to active status.

## Chapter 3. Commands to control the scheduler

You can start, stop, cancel, or modify the scheduler by using the following operator commands:

**S**  
START

**P**  
STOP

**C**  
CANCEL

**F**  
MODIFY

In addition, you can use the MODIFY (F) command to start and stop individual subtasks.

You can enter these commands from a multiple console support (MCS) console or from a program such as the spool display and search facility (SDSF). In both cases, the terminal or console operator must have the required authority to enter operator commands.

### Starting the scheduler

To start IBM® Z Workload Scheduler, enter this z/OS® operator command:

```
/S procname
```

where *procname* is the IBM® Z Workload Scheduler JCL procedure name.

If a started task with this name is already active, the second attempt to invoke it ends with an error message. If this happens, the started task in error cannot write an error message to the message log (DD name EQQMLOG) because the message log is already being used by the active started task. If IBM® Z Workload Scheduler is to run as a batch job, do not start it with an operator command. Instead, submit a batch job with the same name as IBM® Z Workload Scheduler subsystem. JES starts this job in the same manner as any ordinary job.



**Note:** Because IBM® Z Workload Scheduler uses JES exits, among other things, to track the progress of z/OS® jobs, it does not start before JES is active.

### Stopping the scheduler

To stop IBM® Z Workload Scheduler, enter the following z/OS® operator command:

```
/P procname
```

where *procname* is the IBM® Z Workload Scheduler JCL procedure name

If you are stopping a controller, the controller creates a backup copy of the current plan data set (if required) and ends all active functions.

When IBM® Z Workload Scheduler ends, it writes this message to the message log:

```
EQQZ006I NO ACTIVE PRODUCT SUBTASKS. THE SCHEDULER IS ENDING.
```

## Canceling the scheduler

If IBM® Z Workload Scheduler is still active 5 minutes after you enter the STOP operator command, you must cancel IBM® Z Workload Scheduler.

You might also need to cancel IBM® Z Workload Scheduler if the current plan is corrupt, because a normal shutdown causes a backup to the alternate file (for details about current plan recovery, see *Customization and Tuning*). There are two ways to do this. The first is to enter:

```
/C procname,DUMP
```

where *procname* is the IBM® Z Workload Scheduler JCL procedure name.

This causes IBM® Z Workload Scheduler to end with a dump on the SYSMDUMP file (if the DD name is in the started-task JCL). The second way is to enter:

```
/C procname
```

where *procname* is the IBM® Z Workload Scheduler JCL procedure name.

This causes IBM® Z Workload Scheduler to end without a dump.

If the STOP command is ineffective and you have no earlier documentation of the problem, cancel IBM® Z Workload Scheduler with a dump so that the error can be located.

## Modifying the scheduler

Use the MODIFY command to supply information to IBM® Z Workload Scheduler after it has started. The syntax of the MODIFY command is:

```
/F procname,modifyoption
```

where:

### ***procname***

The IBM® Z Workload Scheduler JCL procedure name.

### ***modifyoption***

Can be one of the following:

#### **S=*taskname***

Starts the specified subtask.

#### **P=*taskname***

Stops the specified subtask.

*taskname* can be one of the following:

**APPC**

APPC subtask.

**AR**

Automatic recovery subtask.

**A4**

APPC tracker router subtask.

**CPH**

Critical path handler subtask.

**DRT**

Data router subtask.

**EMGR**

Event manager subtask.

**ERDR**

All active event-reader subtasks.

**EWTR**

Event writer subtask.

**EXA**

External router subtask.

**FL**

Fetch job log task.

**GEN**

General service subtask.

**HTC**

HTTP Client subtask.

**HTS**

HTTP Server subtask.

**IP**

TCP/IP router subtask.

**JCC**

Job-completion-checker subtask.

**NMM**

Normal-mode-manager subtask. The normal mode manager must be restarted as soon as possible after it has stopped. Many functions of IBM® Z Workload Scheduler require an active NMM task to execute successfully.

**PENF53**

Deactivates the ENF mechanism implemented for the ENF 53 event. In this way, a dynamic time change for all the IBM Z Workload Scheduler subsystems cannot be performed. By default, the ENF 53 mechanism is automatically deactivated when the controller is stopped.

**PSU**

Pre-submit task.

**RODM**

RODM subtask.

**SENF53**

Activates the ENF mechanism implemented for the ENF 53 event, so that a dynamic time change for all the IBM Z Workload Scheduler subsystems can be performed. By default, the ENF 53 mechanism is automatically activated at controller startup.

**SUB**

Submit subtask.

**TWS**

End-to-end with fault tolerance capabilities task.

**VTAM®**

Network communication function (NCF) subtask.

**WSA**

Workstation analyzer subtask.

Only the tasks in IBM® Z Workload Scheduler subtask table can be activated by a MODIFY command. The subtask table is built when IBM® Z Workload Scheduler is started. This means that you can only start a task that has stopped earlier in the current session. If you attempt to start a started subtask or stop a stopped subtask, error message EQQZ049W is issued, and no action is taken.

**AWSCCLASS (*wsname*)**

Adds a workstation to the list of workstations for which you defined the class forcing by setting JTOPTS WSCLASS. For example, to add the workstation CPU1 issue the command:

```
/F procname,AWSCCLASS(CPU1)
```

**AWSSYSAFF (*wsname:system.destination*)**

Adds a workstation or a system and destination to the list of workstations where you forced the execution of z/OS jobs by setting `JTOPTS WSSYSAFF(wsname:system.destination)`.

For example to add the workstation CPU3 with the related destinations, issue the command:

```
/F procname,AWSSYSAFF(CPU3:*.*)
```

To add the destination D5 belonging to system SYSA to the workstation VIRT, issue the command:

```
/F procname,AWSSYSAFF(VIRT:SYSA.D5)
```

For detailed information about JTOPTS, see *Customization and Tuning*.

**BKSTATUS**

Issue this command on the primary controller or backup controller to collect detailed information about plans, JT events, and processes. By comparing the information that is returned in EQQMLOG, you can verify the alignment status between the controllers.

For example, on the primary controller, the following messages are shown in EQQMLOG:

```

EQQN133I PRIMARY CONTROLLER INFO:
EQQN133I CP INFO:
EQQN133I - CP RUN           : 00000004
EQQN133I - CP TOD           : CDDD39D2CA9AB661
EQQN133I LTP INFO:
EQQN133I - LT RUN           : 00000002
EQQN133I - LT TOD           : CDDD39D526710741
EQQN133I LATEST WRITTEN JT EVENT INFO:
EQQN133I - EVENT TYPE       : 29
EQQN133I - SEQUENCE NUMBER: 00000CD3
EQQN133I - TIME STAMP       : 141006/16522655
EQQN133I LATEST TRANSFER PLAN INFO:
EQQN133I - NCP JOB NAME     : ROZSENCP
EQQN133I - NCP JOB ID       : STC00814
EQQN133I - NCP JOB STATUS  : C
EQQN133I - LTP JOB NAME     : ROZSENLT
EQQN133I - LTP JOB ID       : STC00813
EQQN133I - LTP JOB STATUS  : C
EQQN133I - CP1 JOB NAME     :
EQQN133I - CP1 JOB ID       :
EQQN133I - CP1 JOB STATUS  :

```

On the backup controller, the following messages are shown in EQQMLOG:

```

BACKUP CONTROLLER INFO:
CP INFO:
- CP PLAN AVAILABLE         : Y
- CP PLAN DDNAME            : EQQCP1DS
- CP PLAN RUN NUMBER        : 00000004
- CP PLAN TOD KEY           : CDDD39D2CA9AB661
LTP INFO:
- LT PLAN AVAILABLE         : Y
- LT PLAN RUN NUMBER        : 00000002
- LT PLAN TOD KEY           : CDDD39D526710741
LATEST WRITTEN JT EVENT INFO:

```

```

- JT TYPE                : 29
- JT SEQNO               : 00000C15
- JT TIMESTAMP           : 141006/16264299
SYNCHRONIZATION INFO:
- SYNC IS IN PROGRESS   : N
- SYNC TYPE             :
- NCP STATUS            : restore not needed
- CP1 STATUS            : restore not needed
- CP2 STATUS            : restore not needed
- LTP STATUS            : restore not needed
LATEST RESTORE PLAN INFO:
- NCP JOB NAME          : ROZSENCP
- NCP JOB ID            : STC00805
- NCP JOB STATUS        : N
BKTAKEOVER INFO:
- POSTPONED             : N
-----

```

**BKTAKEOVER**

Orders a backup controller to take over the functions of the primary controller. Issue this command on the backup controller.

Specify the option `BKTAKEOVER, FORCE` to force the backup controller takeover, even when the connection between the controllers is apparently up and running. Specify the option `BKTAKEOVER, NOSUB` to deactivate the job-submit option, even if `JTOPTS JOBSUBMIT(YES)` is set.



**Note:** The backup controller takeover can occur if you have set the appropriate parameters in `BKPTOPTS` and `TRROPTS` initialization statements. For details about these statements, see *Customization and Tuning*.

**CDP=ON**

Activates the CDP logging. This command is effective provided that `OPCOPTS CDP_PATH=path_to_CDP_log_files` was already set at controller startup.

**CDP=OFF**

Deactivates the CDP logging.

**CPQSTA=ON**

Activates the `STATMSG(CPLOCK)` message.

**CPQSTA=OFF**

Deactivates the `STATMSG(CPLOCK)` message.

**DEPLOYCF**

Use this option to manually start the deployment process that refreshes the trackers event configuration file (member `EQQEVLS` of the `EQQJCLIB` data set) according to the current content of the controller `EQQEVLIB` data set.

The process can update the event configuration file for the trackers that are currently connected. At the end of the deployment process, each connected tracker monitors for the events that match the refreshed data, based on an in-storage copy of the configuration file.

If a tracker destination is referenced in the controller routing options (ROUTOPTS statement), but it is not currently connected, the tracker acquires the configuration file data when the connection is restored.

#### **DOPIADEP**

Displays the value set for OPIADEP in the JTOPTS statement.

#### **DSPDEST**

Lists the HTTP destinations that are currently used by the controller. The list is stored in the message log.

#### **DSPSMOOTHSUB**

Displays all the values that are set for the smooth submission feature in the JTOPTS and BATCHOPT statements.

#### **DSPPRODD**

Lists the DD names for data sets that cannot be discarded. The list can be obtained in one of the following ways:

- In the initial parameter statements: RCLOPTS DDPROT or DDPRMEM
- With the MODIFY command `/F procname,PROT(DD=member)`

#### **DSPPRODS**

Lists the data sets that cannot be discarded. The list can be obtained in one of the following ways:

- In the initial parameter statements: RCLOPTS DSNPROT or DSNPRMEM
- With the MODIFY command `/F procname,PROT(DS=member)`

#### **DSPSTA**

Displays, in message EQQZ095, the status of statistics messaging. The message indicates whether messaging is active for EVENTS, CPLOCKS, GENSERV, and WSATASK. It also gives the values currently set for EVELIM and STATIM. For details, refer to *Messages and Codes*.

#### **EVELIM=nnnn**

Sets the new value of the EVELIM parameter of the JTOPTS statement. Allowed values are 0 to 9999.

#### **DWSCLASS**

Displays the list of workstations for which you defined the class forcing by setting JTOPTS WSCCLASS.

**DWSSYSAFF**

Displays the list of workstations where you forced the execution of z/OS jobs by setting `JTOPTS WSSYSAFF(wsname:system.destination)`. For detailed information about the JTOPTS, see *Customization and Tuning*.

**EVESTA=ON**

Activates the STATMSG(EVENTS) message.

**EVESTA=OFF**

Deactivates the STATMSG(EVENTS) message.

**GENSTA=ON**

Activates the STATMSG(GENSERV) message.

**GENSTA=OFF**

Deactivates the STATMSG(GENSERV) message.

**JCLDBG=ON**

Activates the single JCL trace. For each job handled by WSA task information, such as the elapsed time in milliseconds needed to handle the job, retrieve the JCL, access the JS VSAM, or whatever else, will be shown.

This is a powerful trace and should be activated only for short periods of time to identify possible performance problems.

**JCLDBG=OFF**

Deactivates the single JCL trace.

**JESSYMBOL=ON**

Activates the JESSYMBOL parameter of the OPCOPTS statement.

**JESSYMBOL=OFF**

Deactivates the JESSYMBOL parameter of the OPCOPTS statement.

**LSTNOERR**

The controller lists the NOERROR table content. An example of the command output follows:

```

EQQZ024I Current NOERROR table content:
EQQZ024I !-----!-----!
EQQZ024I ! Statement                ! Member  !
EQQZ024I !-----!-----!
EQQN067I ! ABC123.*.*.0016.GE                ! STDCWSN !
EQQN067I ! ABC123.*.*.0012.NE                ! STDCWSN !
EQQN067I ! ABC123.*.*.0016.EQ                ! STDCWSN !
EQQN067I ! ABC123.*.*.0500.TO.0610        ! NOERR2  !
EQQN067I ! ABC123.*.*.0200.TO.0210        ! NOERR2  !
EQQN067I ! ABC123.*.*.0005.LT                ! NOERR   !
EQQZ024I !-----!-----!

```

The information returned in the `Member` column can help you locate members to be updated.

### **MAXSUB=nnnn**

Sets the new value of the MAXSUBJOBS parameter of the OPCOPTS statement. Allowed values are 0 to 9999.

### **MCPDSSTART**

Use this command to create and use an MCP data space. This overrides MCPDATASPACE(NO) in the JTOPTS statement.

### **MCPDSSTOP**

Use this command to stop using and delete the MCP data space. This overrides MCPDATASPACE(YES) in the JTOPTS statement.

### **NEWDSLST**

Use this command on the tracker side to rebuild the triggering selection table. The new table is read from member EQQEVLSLST (or EQQDSLST, if EQQEVLSLST does not exist) of the data set referenced by the EQQJCLIB DD name in the started task JCL for the tracker. The new table replaces the table in ECSA.



**Note:** If EQQJCLIB contains both EQQEVLSLST and EQQDSLST member, the resulting triggering selection table is the union of EQQEVLSLST and EQQDSLST. In this case, EQQEVLSLST data is processed first.

### **NOERROR**

You can use the commands NEWNOERR and NOERRMEM() only if NOERROR entries are coded under the LIST() parameter of the NOERROR statement.

For a description of the NOERROR initialization statement, see *Customization and Tuning*.

When you enter a NEWNOERR command, the program searches the controller PARMLIB member only for NOERROR statements. If none are found, an empty table is created, thus deleting the entries which were loaded when the controller started. Do not use the NEWNOERR command if the NOERROR list is coded under the NOERROR() parameter of the JTOPTS initialization statement. If you enter a NOERRMEM(*membername*) command, the current table is deleted and created again. The entries that had already been specified for *membername* are replaced with the entries found in the updated member of the EQQPARM library.

When using NOERROR, consider the following:

- Use this keyword only if you are sure you can stop the controller before updating the NOERROR data.
- Initialization statements cannot exceed 32 KB or 455 72-character lines. The available space for NOERROR entries in the initialization statement is more limited compared with using this keyword in the JTOPTS statement.
- Do not mix the three options for defining NOERROR entries:
  - JTOPTS NOERROR ()
  - NOERROR LIST () in the main parmlib member
  - NOERROR LIST () in separate parmlib members identified by the INCLUDE() initialization statement

The options for dynamically updating controller data are mutually exclusive and using them incorrectly might cause the deletion of the active table.

- If you must dynamically update the NOERROR data without stopping the controller, and the NOERROR entries are defined in a separate parmlib member, place all NOERROR entries in a single parmlib member. In this way, you can use only one member name in the NOERRMEM command.

### **NEWNOERR**

Orders a controller to rebuild the NOERROR table, in the case NOERROR statements have been modified in the parameter library member that contains the JTOPTS statement.

### **NOERRMEM(*member*)**

Orders a controller to rebuild the NOERROR table, in the case NOERROR statements have been modified in a parameter library member that was specified in an INCLUDE statement.

### **NOERRMEM(M1)**

Order a controller to delete all NOERROR codes defined by member M1, once you have previously changed M1 to contain only comments. The modified member can contain a different number of NOERROR codes than the original member.



**Note:** The scheduler opens the EQQPARM library when IBM® Z Workload Scheduler is started and parameter library members (residing in library extents), that have been created, cannot be accessed, after have been opened. To avoid this problem, the data sets that define the EQQPARM library should be allocated without any secondary extents.

### **PROT ([DD=*mem1*], [DS=*mem2*])**

Replaces the currently used list contents of the members (*mem1* and *mem2*) inside the PDS parameter library, if DD or DSN is protected. At least one of the two keywords DD or DS must be specified. The list of protected DD and DSN is dropped when *mem1* or *mem2* is blank. Also, the keywords DD and DS remove the list of protected DD names and DS names.

For example, if DDPROT is used in the parameter library at startup of the controller, the command drops this list and replaces it with the contents of *mem1*.

#### **QUELEN=nnnn**

Sets the new value of the QUEUELEN parameter of the JTOPTS statement. Allowed values are 0 to 9999, but a minimum value of 5 is forced.

#### **RFRDEST**

If you modify, add, or delete an HTTP or HTTPS destination in while IBM® Z Workload Scheduler is running, makes your changes immediately effective. This command does *not* update any changes you make to the PROXY parameter in ROUTOPTS.

RFRDEST manages up to a total of 100 new destinations, regardless if you add them at once or at different times. For detailed information about the destinations set by ROUTOPTS, see *Customization and Tuning*.

#### **RFRUSER**

If you modify the USRREC statement while IBM® Z Workload Scheduler is running, makes your changes immediately effective. For detailed information, see *Scheduling End-to-end with z-centric Capabilities*.

#### **RFRUX14T**

If you modify the criteria table pointed by the UX14IN DD name in the controller started task while the controller is running, makes your changes effective for the operations that become ready after modifying the table.

To make the changes effective also for the operations that were ready before you modified the table, you must replan the current plan.

For detailed information about the time-dependent-operation exit, see *Customization and Tuning*.

#### **RWSCLASS (wsname)**

Deletes a workstation from the list of workstations for which you defined the class forcing by setting JTOPTS WSCLASS. For example, to remove the workstation VIRT issue the command:

```
/F procname,RWSCLASS(VIRT)
```

#### **RWSSYSAFF (wsname[:system])**

Deletes a workstation or a single destination from the list of workstations where you forced the execution of z/OS jobs by setting JTOPTS WSSYSAFF(*wsname[:system.destination]*). The value *system* is optional, if you do not specify it the complete workstation definition is deleted.

For example, to delete the destination SYSB from the list of systems defined for the workstation VIRT, issue the following command:

```
/F procname,RWSSYSAFF(VIRT)
```

For detailed information about JTOPTS, see *Customization and Tuning*.

**SETSMOOTHCRITNET=ON**

Sets the SMOOTHCRITNET parameter in JTOPTS to YES, to prioritize the submission of operations belonging to the whole critical network.

**SETSMOOTHCRITNET=OFF**

Sets the SMOOTHCRITNET parameter in JTOPTS to NO.

**SETSMOOTHRATE=nnnn**

Changes the value for the SMOOTHSUBRATE parameter in the JTOPTS statement. Valid values are in the range from 0 to 99999. 0 means that the SMOOTHSUBRATE parameter is not set.

**SETSMOOTHSUB=ON**

Activates the smooth submission feature. To make it effective, run either a daily planning EXTEND or REPLAN.

**SETSMOOTHSUB=OFF**

Deactivates the smooth submission feature. To make it effective, run either a daily planning EXTEND or REPLAN.

**SKIPINC(*member name*)**

Replaces the currently used list of INCLUDEs that are to be left at the beginning of a JCL by the JCL tailoring process of Restart and Cleanup. *member name* can be:

- The name of a member inside the PDS parameter library
- Blank or absent

When a member name is specified, the current SKIPINCLUDE list is deleted and a new SKIPINCLUDE list is built by applying the syntax described for the RCLSKIP statement to the specified member. When no member name is specified, the current SKIPINCLUDE list is removed.

For details about RCLOPTS(SKIPINCLUDE) and the RCLSKIP statement, see *Customization and Tuning*.

**SOPIADEP (YES | NO | TIME)**

Sets a new value for the OPIADEP parameter of the JTOPTS statement.

**STATIM=nn**

Sets the new value of the STATIM parameter of the JTOPTS statement. Allowed values are 0 to 99.

**STATUS**

Returns a message on the system log with the status of IBM® Z Workload Scheduler subsystem. The status can be one of the following:

**FULLY\_OPERATIONAL**

Everything is active and is working properly.

**PARTIALLY\_OPERATIONAL**

The scheduler subsystem has limited functionality. For example, if a controller ER is stopped, the controller can still schedule jobs but cannot receive their statuses.

**NOT\_OPERATIONAL**

The major subsystem functionality is not available. For example, a controller is not able to execute a plan or to submit a job.

**STATUS, DD=*ddname***

Checks for the status of IBM® Z Workload Scheduler data set associated with the specified *ddname*. *ddname* can be a specific DD name, such as EQQWSDS, EQQCP1DS, or EQQLTDS, or it can assume the value ALL, CP, DB, LTP, or JTL. It returns the return code of the last I/O operation performed on that *ddname*. The status of the data set can be one of the following:

NORMAL  
WARNING  
SEVERE  
CRITICAL  
UNKNOWN

**STATUS, {OP\_COMP | OP\_ERR}, "*destination name*"**

Returns the number of completed operations (OP\_COMP) or the number of ended-in-error operations (OP\_ERR) for the specified tracker (*destination name*).

 **Note:**

1. If more than one workstation is defined for tracker *tracker name*, the number of completed or in-error operations is the sum of the operations on all the workstations defined on that tracker.
2. *destination name* is the destination name of a tracker, as specified in the ROUTOPTS initialization statement.

**STATUS, SUBTASK**

Lists all subtasks with their statuses. The status can be ACTIVE or INACTIVE.

**SWITCHMLOG**

If the MLOG switching feature is in use, forces the switch to the alternate data set (EQQMLOG or EQQMLOG2), regardless of the number of currently logged records, and starts counting from 0 again.

**TAKEOVER**

Orders a standby controller to take over the functions of the controller. This command is valid only when both systems are part of the same XCF group, and no controller is active. You can use this command only for IBM® Z Workload Scheduler address spaces where OPCHOST(STANDBY) is specified in the OPCOPTS statement.



**Note:** Takeover can occur automatically if you have set the TAKEOVER parameter in the XCFOPTS statement of a standby system. For details about the XCFOPTS statement, see *Customization and Tuning*.

**TRYNOERR(*member name*)**

Start a trial processing of the NOERROR statements contained in *member name*. The controller issues all the normal processing messages in EQMLOG. At the end of the trial processing, the controller issues message EQQN099I, leaving unchanged the NOERROR table.

*member name* is a member of the EQPPARM library.

**VSTRC=START**

Starts a trace on the message log of all VSAM I/O requests. In a busy scheduler system, you will need a large message-log data set, and the trace will affect the performance of IBM® Z Workload Scheduler system.

**VSTRC=STOP**

Stops a VSAM I/O request trace on the message log.

**WSASTA=ON**

Activates the STATMSG(WSATASK) message.

**WSASTA=OFF**

Deactivates the STATMSG(WSATASK) message.

After the STOP command is entered, the MODIFY command no longer functions, and issues the following message on SYSLOG:

```
IEE324I MODIFY REJECTED - TASK BUSY
```

## Modifying the data store

Use the MODIFY command to supply information to IBM® Z Workload Scheduler data store after it has started. The syntax of the MODIFY command is:

```
/F procname , modifyoption
```

where:

***procname***

The IBM® Z Workload Scheduler JCL procedure name.

***modifyoption***

Can be one of the following:

***S=taskname***

Start the specified data store subtask.

***P=taskname***

Stop the specified data store subtask.

*taskname* can be one of the following:

**ARRD**

Reader task.

**ARCU**

Cleanup task.

**ARCM**

Communication.

**ARDYWR**

Display number of active writers.

**ARDYTW**

Display WINTERVAL value.

**ARDYNY**

Display MAXSTOL value.

**ARDYNS**

Display MAXSYSL value.

**ARDYTU**

Display CINTERVAL value.

**ARDYNS**

Display MAXSYSL value.

**ARDYPM**

Display all initialization parameters values.

**ARSTKW O=[owner],K=[keyword]**

Display the statistical data for a specific owner or single keyword. The following describes the keyword and owner combinations:

**Table 2. Keyword and owner combinations**

Owner	Keyword	Meaning
JESQUEUE	COUNTJOB	Number of jobs still in the JES Queue control block
JESQUEUE	JOBINSQU	Number of jobs inserted in the JES Queue control block
JESQUEUE	DSIDINSQ	Number of DS IDs (sysout) inserted in the JES Queue control block
JESQUEUE	JOBRQUEU	Number of jobs re-queued: deleted from the JES Queue control block after the store
JESQUEUE	JOBDISCA	Number of jobs discarded: deleted from the JES Queue control block because of an error during the store in the database
DATAFILEnnnn	CNTDPAGE	Number of data pages for the data file specified
READER	JOBRQSDB	Number of jobs requested directly to the database
READER	JOBRQJES	Number of jobs requested to JES Queue because they are not stored in the database yet
COMMUNICATION	INPUTMSG	Number of input messages received
COMMUNICATION	OUTPTMSG	Number of output messages sent

**ARSTGN**

Display all the statistics for the data store.

**ARMDWR**

Modify number of active writers.

**ARMDTW=n**

Modify WINTERVAL value (seconds).

**ARMDNY=n**

Modify MAXSTOL value (number of lines).

**ARMDNS=n**

Modify MAXSYSL value (number of lines).

**ARMDTU=n**

Modify CINTERVAL value (seconds).

**ARMDNS=n**

Modify MAXSYSL value (number of lines).

**ARDGCM=on/off**

Activate or deactivate Communication task traces.

**ARDGWR=on/off**

Activate or deactivate Writer task traces.

**ARDGRD=on/off**

Activate or deactivate Reader task traces.

**ARDGJQ=on/off**

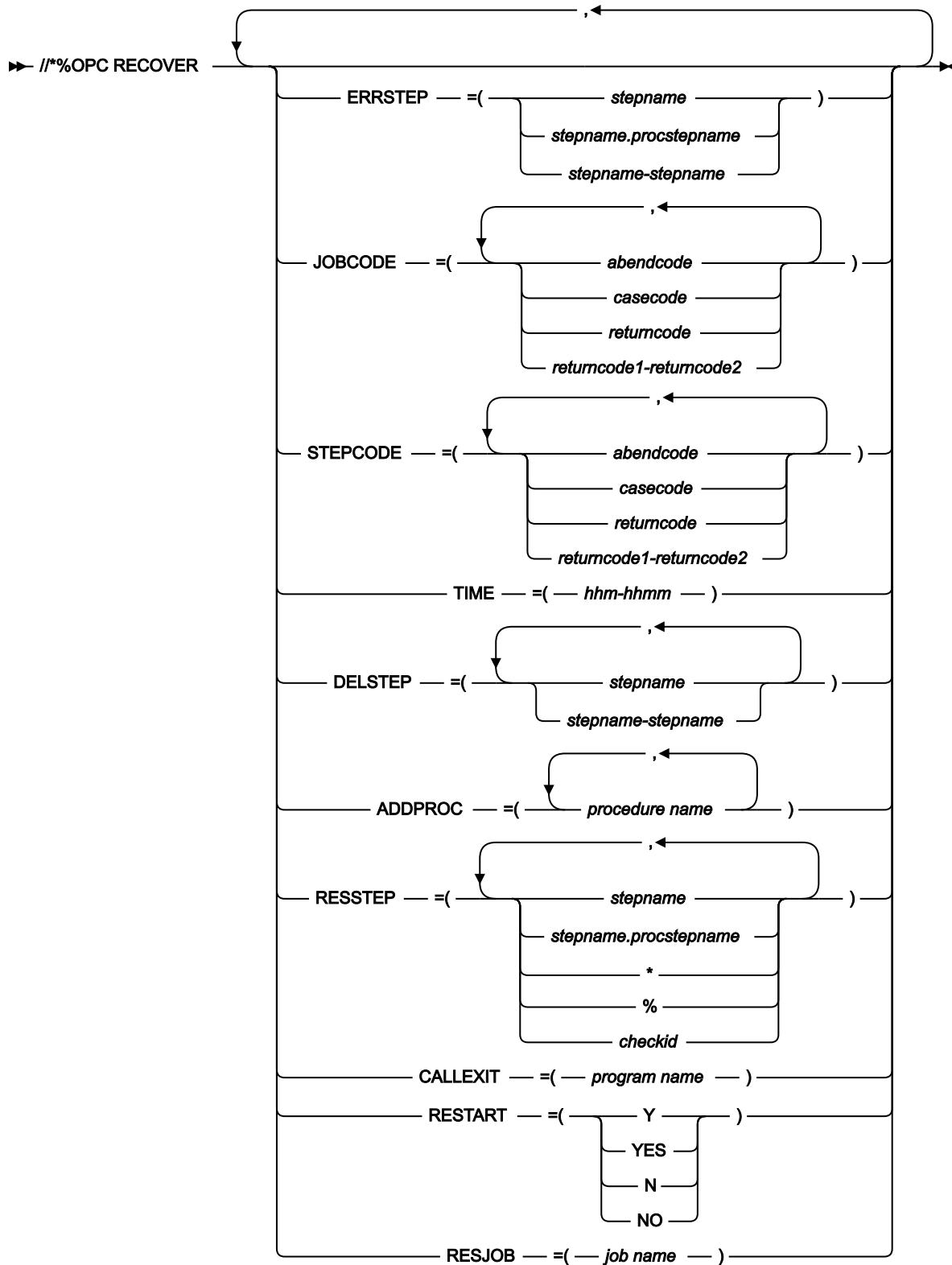
Activate or deactivate JES Queue task traces.

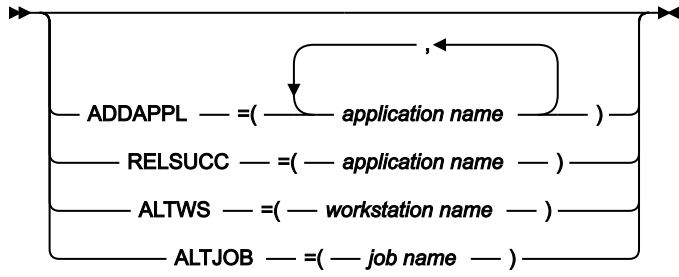
**ARDGDB=on/off**

Activate or deactivate Database task traces.

## Chapter 4. Automatic recovery

# RECOVER statement





You cannot have JCL variables on the RECOVER statement.

# Chapter 5. JCL directives

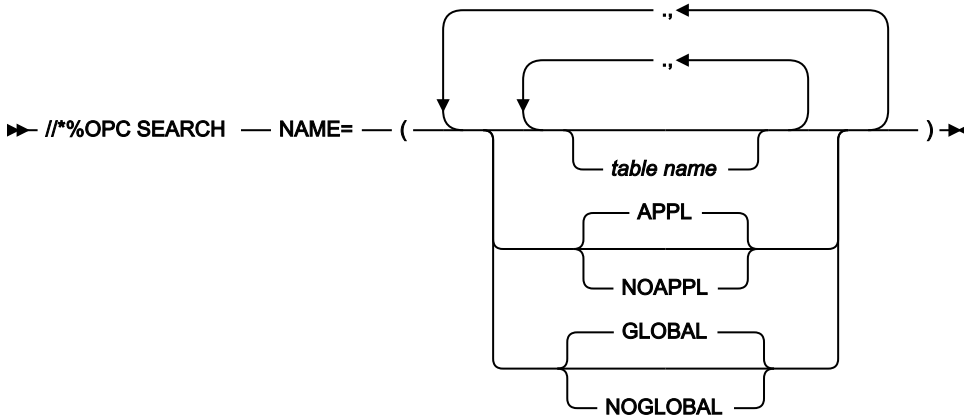
## NOP directive

▶▶ //\*%OPC NOP ▶▶

## SCAN directive

▶▶ //\*%OPC SCAN ▶▶

## SEARCH directive



## TABLE directive

▶▶ //\*%OPC TABLE — NAME= — ( — *table name* — ) ▶▶

## SETFORM directive

▶▶ //\*%OPC SETFORM — *dynamic-variable-name* — =( — *format* — ) ▶▶

## Usage notes

- Choose the variable from [Table 6: Dynamic-format date-related supplied variables on page 64](#).
- Compose the format from one or more of these keywords and any other characters except &, %, ?, =, ), and (

**CC**

Century (2 digits)

**YY**

Year (2 digits)

**MM**

Month

**DDD**

Day in year (Julian)

**DD**

Day in month

**HH**

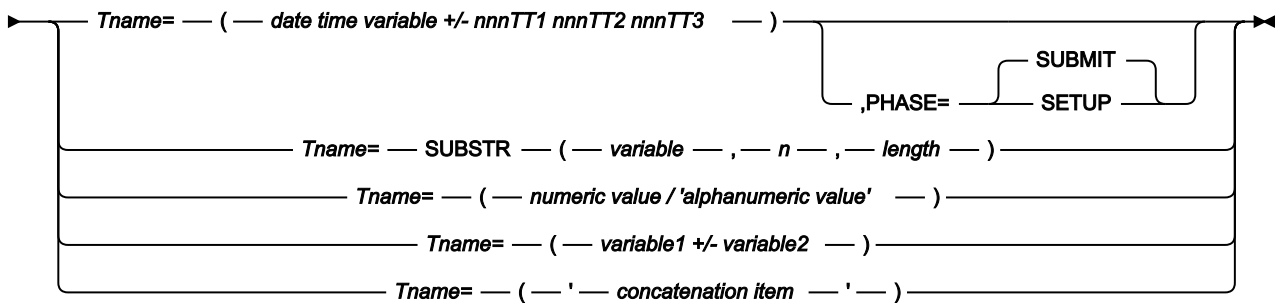
Hour (time variables only)

**MM**

Minutes (time variables only)

## SETVAR directive

▶▶ *!\**%OPC SETVAR →



## Usage notes

- Use parentheses around the expression for the temporary variable Tname.
- Begin the variable Tname with the letter T.
- *TT1* is the first possible type and can be one of the following values:

**WD**

Work days

**CD**

Calendar days

**WK**

Weeks

**MO**

Months

**YR**

Years

**HH**

Hours

**MM**

Minutes

**SS**

Seconds

- *TT2* and *TT3* are additional types allowed only for time-related variables. They can be one of the following values:

**HH**

Hours

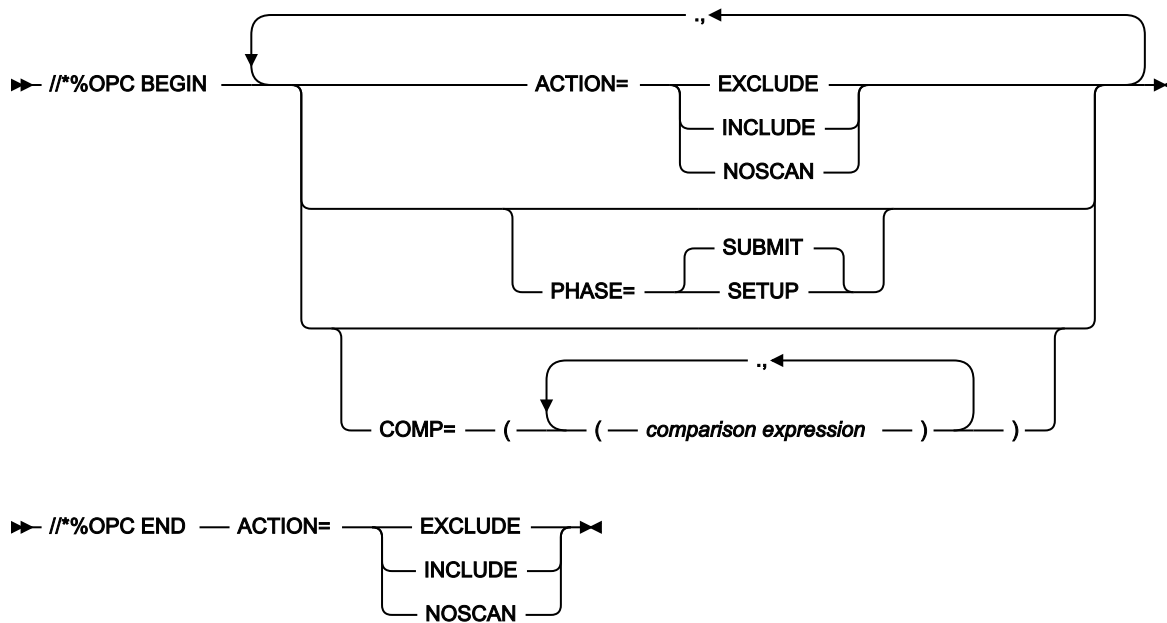
**MM**

Minutes

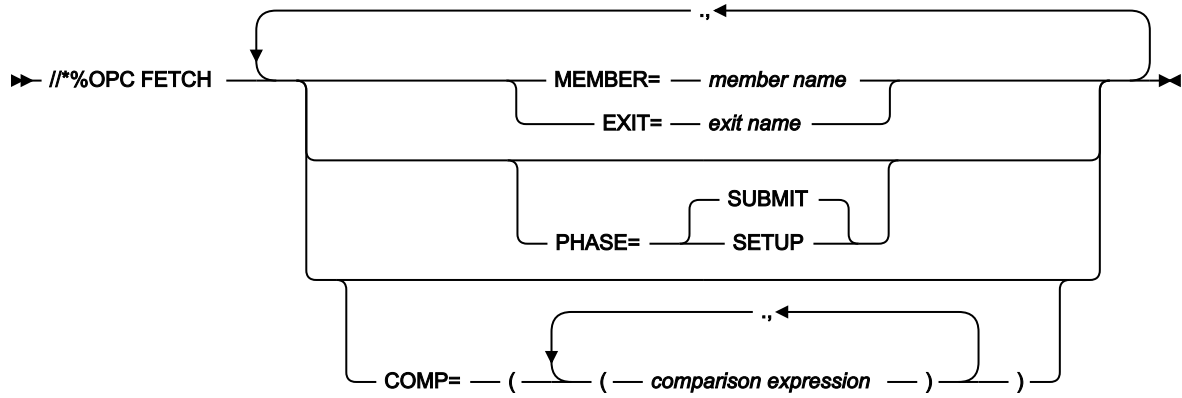
**SS**

Seconds

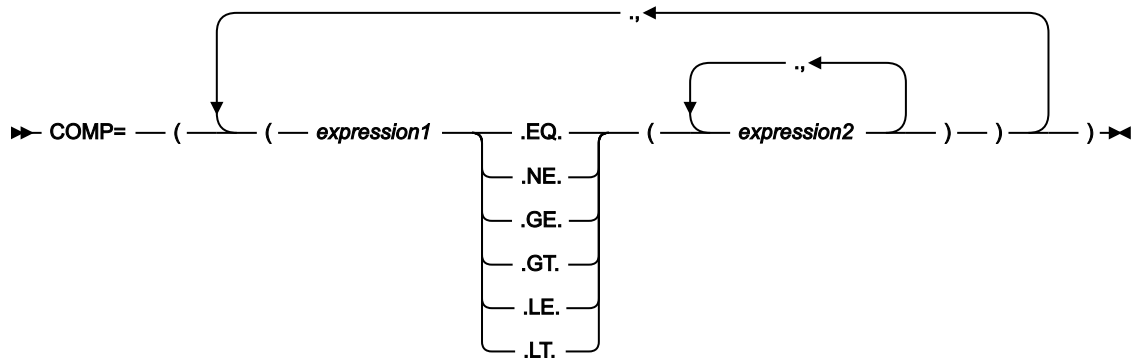
## BEGIN and END directives



## FETCH directive



## COMP keyword on BEGIN and FETCH directives



## Usage notes

Only & (ampersand) variables are valid in the `COMP` expressions, and embedded blanks are not allowed.

## Chapter 6. Supplied JCL variables



**Note:** IA = input arrival

### Occurrence-related JCL variables

**Table 3. Occurrence-related JCL variables**

Variable name	Length (bytes)	Description				
OADID	16	Application ID				
OADOWNER	16	Occurrence owner				
OAUGROUP	8	Authority group				
OCALID	16	Calendar name				
ODAY	1	Occurrence IA day of the week (1–7); 1 represents Monday				
ODD	2	Occurrence IA day of month, in DD format				
ODDD	3	Occurrence IA day of the year, in DDD format				
ODMY1	6	Occurrence IA date in DDMMYY format				
ODMY2	8	Occurrence IA date in DD/MM/YY format				
OFREEDAY	1	Denotes whether the occurrence IA date is a free day (F), or a workday (W)				
OHH	2	Occurrence IA hour in HH format				
OHHMM	4	Occurrence IA hour and minute in HHMM format				
OMM	2	Occurrence IA month in MM format				
OMMY	4	Occurrence IA month and year in MMY format				
OWW	2	Occurrence IA week of the year in WW format				
OWWD	3	Occurrence IA week, and day within week, in WWD format, where WW is the week number within the year, and D is the day within the week				
OWWLAST	1	A value, Y (yes) or N (no), that indicates whether the occurrence IA date is in the last week of the month				
OWWMONTH	1	A value between 1 and 6 that indicates the occurrence IA week-in-month, where each new week begins on a Monday. For example, consider these occurrence IA dates for the month of January in 1996:  <table border="0"> <tr> <td style="padding-right: 20px;"><b>Date</b></td> <td></td> </tr> <tr> <td style="padding-right: 20px;"><b>Value</b></td> <td></td> </tr> </table>	<b>Date</b>		<b>Value</b>	
<b>Date</b>						
<b>Value</b>						

**Table 3. Occurrence-related JCL variables (continued)**

Variable name	Length (bytes)	Description
		<b>Monday 1st</b> 1 <b>Monday 8th</b> 2 <b>Wednesday 31st</b> 5
OYMD	8	Occurrence IA date in YYYYMMDD format
OYM	6	Occurrence IA month within year in YYYYMM format
OYMD1	6	Occurrence IA date in YYMMDD format
OYMD2	8	Occurrence IA date in YY/MM/DD format
OYMD3	10	Occurrence IA date in YYYY/MM/DD format
OYY	2	Occurrence IA year in YY format
OYYDDD	5	Occurrence IA date as a Julian date in YYDDD format
OYYMM	4	Occurrence IA month within year in YYMM format
OYYYY	4	Occurrence IA year in YYYY format, for example, 1996

## Operation-related JCL variables

**Table 4. Operation-related JCL variables**

Variable name	Length (in bytes)	Description
OJOBNAME	8	Operation job name.
OLDAY	1	Operation latest start day (1-7); 1 represents Monday, 7 represents Sunday.
OLDD	2	Operation latest start day (day in the month).
OLHH	2	Operation latest start hour.
OLHHMM	4	Operation latest start in hours and minutes.
OLMD	4	Operation latest start time (month and day), in MMDD format.
OLMM	2	Operation latest start month, in MM format.
OLWK	2	Operation latest start week (week in the year) in WW format.
OLYMD	6	Operation latest start date in YYMMDD format.

**Table 4. Operation-related JCL variables (continued)**

Variable name	Length (in bytes)	Description
OLYYDDD	5	Operation latest start in Julian date format (YYDDD).
OOPNO	3	Operation number within the occurrence, right-justified and padded with zeros.
OWSID	4	Workstation ID for current operation.
OXJOBNAM	54	Extended job name set in the current plan at the operation level. If you enter blanks in the Extended Job Name field, the OXJOBNAM value is truncated at the first blank.

## Date-related JCL variables

**Table 5. Date-related JCL variables**

Variable name	Length (bytes)	Description
CDAY	1	Current day of the week; 1 represents Monday, 7 represents Sunday
CDD	2	Current day of month in DD format
CDDD	3	Day number in the current year
CDDMMYY	6	Current date in DDMMYY format
CHH	2	Current time in HH format
CHHMM	4	Current hour and minute in HHMM format
CHHMMSSX	8	Current hour, minute, second, and hundredths of seconds in HHMMSSXX format
CMM	2	Current month in MM format
CMMYY	4	Current month within year in MMY format
CWW	2	Week number in the current year
CWWD	3	Current day within week in WWD format, where WW is the week number within the year and D is the day within the week
CYMD	8	Current date in YYYYMMDD format
CYY	2	Current year in YY format
CYYDDD	5	Current Julian date in YYDDD format
CYYMM	4	Current month within year in YYMM format
CYYMMDD	6	Current date in YYMMDD format

**Table 5. Date-related JCL variables (continued)**

Variable name	Length (bytes)	Description
CYYYY	4	Current year in YYYY format, for example, 1990
CYYYYMM	6	Current month within year in YYYYMM format

## Dynamic-format supplied variables

**Table 6. Dynamic-format date-related supplied variables**

Variable name	Description
CDATE	Current date
CTIME	Current time
OCDATE	Occurrence IA date
OCFRSTC	First calendar day in month of the occurrence IA date
OCFRSTW	First work day in the month of the occurrence IA date
OCFRSTWY	First work day in the year of the occurrence IA date
OCLASTC	Last calendar day in the month of the occurrence IA date
OCLASTW	Last work day in the month of the occurrence IA date
OCLASTWY	Last work day in the year of the occurrence IA date
OCTIME	Occurrence IA time (hours and minutes)
OPIADATE	Operation IA date (if blank, this takes the value of the occurrence IA date)
OPIATIME	Operation IA time (if blank, this takes the value of the occurrence IA time)
OPLSDATE	Operation latest start date
OPLSTIME	Operation latest start time

# Chapter 7. Codes

The following is a list of the occurrence status codes:

## **C**

Complete

## **D**

Deleted

## **E**

An operation in the occurrence has ended-in-error

## **P**

A pending predecessor exists for the occurrence

## **S**

Started

## **U**

Undecided (the status is not known)

## **W**

No operations in the occurrence have started.

## Error codes

The scheduler assigns error codes to certain operations and to job and started task steps. These codes are used by the automatic job recovery function to decide a recovery action.

### **CAN**

The job or started task was canceled by the operator or by a TSO user before execution. This code is also possible if the job-termination event (type 3P) is missing.

### **CCUN**

The completion code is unknown. The job or started task has ended, but no completion code is available. This code is also possible if the job-end event (type 3J) is missing.

Check the job log and SYSLOG.

### **CLNA**

A failure occurred when IBM® Z Workload Scheduler attempted to complete the JCL tailoring during the restart and cleanup process.

### **CLNC**

A failure occurred when IBM® Z Workload Scheduler attempted to execute the data set cleanup during the restart and cleanup process.

**CLNO**

A failure occurred when IBM® Z Workload Scheduler attempted to retrieve the historical job log data during the restart and cleanup process. *nnnn* Step return code. **S** *xxx* System abend code. **U** *xxx* User abend code in hexadecimal notation. For example, user abend 2750 is represented in IBM® Z Workload Scheduler as UABE. *xxxx* User-defined error code.

**CLNP**

A failure occurred in the EQQCLEAN step, during the run of a restarted job.

**JCCE**

An error during JCC (job completion checker) processing prevented the JCC from determining an error code for the operation.

**FBND**

The request to bind the shadow job with a real instance of the remote plan failed.

**JCL**

A JCL error was recognized after the job or started task began to execute, or a JCL error was recognized after syntax checking in the internal reader.

**JCLI**

A JCL error occurred immediately; that is, the error was detected before the job or started task began. This code is also possible when both the job-start event (type 2) and the job-end event (type 3J) are missing. On IBM Z Workload Scheduler Agent workstations, this code is returned when a parsing error occurs in the JOBREC statement.

**LOOP**

The workstation analyzer task has tried to start the same operation repeatedly and message EQQW534E has been stored in the controller MLOG to signal a loop. To stop the loop, the operation was set to error with error code LOOP.

**MCP**

The operation was manually set to error in the MCP panel.

**OAUT**

While running the System Automation command specified with the operation, the System Automation exit EQQUXSAZ issued a return code different from 0. The operation status is set to E. Check for System Automation messages in the log specified in the AUTOMATIONMSG parameter of the OPCOPTS statement.

**ODEA**

The operation did not complete by the time settings that you specified in Deadline Action.

**OLAT**

The operation did not start by the time settings that you specified in Not Started Action.

**OFxx**

The system that the operation is defined on has gone offline. The WSOFFLINE parameter on the JTOPTS initialization statement specifies that started operations should be marked as ended-in-error. xx is the status and extended status of the failing operation. Operations that were running (status SS) have a step-code error status of OFFL.

**OJCV**

An error occurred during JCL-variable substitution when the job or started task was submitted, or IBM® Z Workload Scheduler detected an error in the RECOVER statement during automatic recovery. Browse the JCL for the operation or the EQQMLOG data set to find more information about the failure.

This error code can also be issued when an error occurred during variable substitution in a System Automation command text. The operation status could be set to E, based on the SAVARFAIL parameter set in the OPCOPTS initialization statement. To identify the variables not resolved and the type of error occurred, check for the messages related to occurrence variable substitution in EQQMLOG.

**OSEQ**

A job or started task began to execute before all its predecessors have completed. This can occur only if the job was not submitted by IBM® Z Workload Scheduler and if either HOLDJOB(NO) or HOLDJOB(USER) is specified in the EWTROPTS statement. For fault-tolerant agents, the OSEQ code can indicate that a dependency on another operation or a special resource was added after the job started, but before the event reached the controller.

For details about EWTROPTS, see *Customization and Tuning*.

**OSIN**

Default value for operations set in error through the OPSTAT command or EQQUSIN subroutine.

**OSUB**

A failure occurred when IBM® Z Workload Scheduler attempted to submit a job or start a started task. In the case of a started task, it could be that the started task is a subsystem that is not started by JES, or IBM® Z Workload Scheduler subsystem EQQSTC dname is not allocated to a JES-defined procedure library. The operation should be marked as ended-in-error.

For jobs running on:

- z-centric agent workstations and z/OS® shadow jobs, OSUB indicates that the job submission failed.
- For file transfer jobs running on z-centric agent workstations, OSUB could indicate that the job submission failed for security reasons. In this case, the error code and reason code are provided in the EQQHT83W message stored in the controller MLOG.

**OSUF**

A failure occurred when IBM® Z Workload Scheduler attempted to retrieve the JCL for a job or started task. This code is set if the SUBFAILACTION keyword of the JTOPTS initialization statement specifies that the

operation should be marked as ended-in-error. This code is also caused if you have JOBCHECK(SAME) and the job name in the application description does not match the one on the job card. Another reason for this code is a job is missing JCL that was packed by ISPF in EQQJBLIB.

In an end-to-end scheduling with fault tolerance capabilities, this code indicates that an error occurred while the scheduler was queuing the submission event for the job. For jobs with centralized scripts, it might also show that IBM Z Workload Scheduler could not download the script to the distributed agent.

#### **OSUP**

A time operation is late, and the SUPPRESSACTION parameter of the JTOPTS initialization statement specified that the operation should be marked as ended-in-error. **OSxx** The system on which the operation is defined has failed. The WSFAILURE parameter on the JTOPTS initialization statement specifies that started operations should be marked as ended-in-error. *xx* is the status and extended status of the failing operation. Operations that were running (status SS) have a step-code error of OSYS.

#### **OSxx**

The system on which the operation is defined has failed. The WSFAILURE parameter on the JTOPTS initialization statement specifies that started operations should be marked as ended-in-error. *xx* is the status and extended status of the failing operation. Operations that were running (status SS) have a step-code error of OSYS.

#### **PCAN**

A print operation was canceled by the operator.

#### ***nnnn***

Step return code.

#### **SERC**

An operation submitted in a Restart and Clean up path (via dialogue but also automatically, for example, when clean up was set to AUTOMATIC) was not submitted because the required scheduling environment was not available..

#### **SEUN**

An operation required a scheduling environment that is unknown to WLM; for this reason it was not submitted.

#### **SHPF**

The request to bind the shadow job is rejected.

#### **Sxxx**

System abend code.

#### **UNKN**

When a connection problem occurs between the controller and a z-centric agent where a job is running, if the query status for that job is unknown, the job is set to Error with UNKN.

**Uxxx**

User abend code in hexadecimal notation. For example user abend 2750 is represented in IBM® Z Workload Scheduler as UABE.

**xxxx**

User-defined error code.

## Operation status codes

When IBM® Z Workload Scheduler displays the status of an operation, it uses the format *xy*, where *x* is the status code and *y*, if present, is the extended status code. For quick recognition of the operation status, each status has a default color. You can customize the status color in the ISPF options (see Setting options).

The following is a list of the operation status codes and the corresponding default colors:

**A**

Arriving; the operation is ready for processing; no predecessors were defined. The default status color is green.

**C**

Complete. The default status color is blue.

**D**

Deleted. The default status color is blue.

**E**

The operation has ended-in-error. The default status color is red.

**I**

The operation is interrupted. The default status color is blue.

**R**

Ready for processing; all predecessors are complete. The default status color is green.

**S**

Started. The default status color is turquoise.

**U**

Undecided; the operation status is not known. The default status color is red.

**W**

The operation is waiting for a predecessor to complete. The default status color is yellow.

**X**

The operation is suppressed by condition. The default status color is blue.

\*

Ready; at least one predecessor is defined on a nonreporting workstation; all predecessors are complete. The default status color is green.

## Extended status codes

Together with the normal status codes, IBM® Z Workload Scheduler maintains extended status codes that provide additional information about the status of operations. The extended status code is not always present.

The following is a list of the extended status codes:

**3**

The scheduler is sending an HTTP or HTTPS request to bind the shadow job to a real instance in the remote plan.

**4**

The scheduler is waiting to receive the result of the HTTP or HTTPS request that was sent to bind the shadow job to a real instance in the remote plan.

**5**

The bind between the shadow job and a real instance in the remote plan was established.

**7**

Job recovery is in progress.

**8**

The job execution has been suspended.

**9**

Smooth submission delay added to the operation.

**A**

The job is waiting for a manual cleanup action to be initiated or discarded by a panel user (the cleanup type is manual).

**B**

The job is waiting for a cleanup action to be started (the cleanup type is automatic or immediate).

**C**

A restart and cleanup process is in progress (data set cleanup or step restart, or both). The job is waiting for the process to be completed.

**D**

Close down is in progress.

**E**

An error occurred during job submission or release.

**G**

The operation is running on a WAIT workstation (it is a dummy operation waiting for the delay period to elapse).

**H**

A panel user has used the HOLD command on the operation.

**L**

The operation is a late time-dependent operation with the suppress-if-late attribute.

**M**

The status of the operation has been manually set.

**N**

A panel user has used the NOP command on the operation.

**O**

Workstation is offline.

**Q**

For z/OS® jobs the job has been added to the JES job queue. For fault-tolerant workstations, it is waiting for submission.

**R**

The operation has ended in error but was automatically reset (the completion code is defined in the installation options to be automatically reset).

**S**

The job or started task is executing.

**T**

Waiting until a particular time.

**U**

Submit is in progress.

**V**

The limit value of this fault-tolerant workstation was reached.

**W**

Waiting for scheduling environment.

**X**

Waiting for resource.

**Y**

The job ended with an error code matching a NOERROR entry.

For operations on computer workstations, a blank extended status has a particular meaning for the following statuses:

**Arrived (A) or Ready (\* or R)**

The scheduler is in the process of submitting this job. The scheduler is waiting for the availability of a parallel server or a critical resource, or the operation is not to be submitted automatically.

**Started (S)**

The job has been successfully submitted but has not yet been reported as added to the JES job queue.

## Job log retrieval status codes

When the job log retrieval function is used, IBM® Z Workload Scheduler maintains status information to report on the retrieval of the log. The following status codes are possible:

**C**

Completed; the controller has received the log.

**E**

Error. There was an error retrieving the log.

**I**

Initiated. The controller has sent a retrieval request to the tracker, but the tracker has not yet processed the request.

**S**

Started. The controller has sent a retrieval request to the tracker, and the tracker has started to retrieve the log.

**blank**

The controller has not sent any retrieval request to the tracker.

## Operation reason codes

If your ready list layout includes the RSNC field, you can see these operation reason codes. Note that the codes are listed in hierarchical order. For example, if job submission failed, and job submission is deactivated, code D is obtained, not code F.

**A**

Automatic reset error condition

**C**

Workstation is closed

**D**

Job submission deactivated

**F**

Job submission failed

**H**

Closedown in progress

**J**

No automatic job submission

**L**

Job is late

**O**

Work station is offline

**P**

All parallel servers in use

**S**

Waiting for special resource

**T**

Start time not reached

**U**

Work station is unlinked

**W**

Waiting for scheduling environment

**1**

Not enough free WS resource 1

**2**

Not enough free WS resource 2

# Chapter 8. Free-day rule

## About this task

The possible values of the free-day rule are:

**E**

Count only work days when using the rule or offset. That is, free days are excluded. This option ensures that the scheduled day will always be a work day. This is the default for offset-based run cycles.

**1**

Count work days and free days when using the rule or offset. If this gives a free day, schedule the application on the closest work day *before* the free day.

**2**

Count work days and free days when using the rule or offset. If this gives a free day, schedule the application on the closest work day *after* the free day.

**3**

Count work days and free days when using the rule or offset. If this gives a free day, schedule the application *on* the free day. This is the default for rule-based run cycles.

**4**

Count work days and free days when using the rule or offset. If this gives a free day, *do not* schedule the application at all.

The free-day rule provides the flexibility to schedule your applications precisely when they are required. Sometimes you will find that you must work out on paper which free-day rule you should select. When you do this, consider what would happen if a normal work day is declared a holiday and, therefore, is defined in the calendar as a free day.

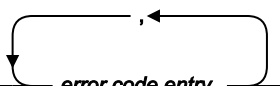
When an application is normally due to run but the calendar definition identifies the day as free, the free-day rule in the run cycle for that application determines the effect.

## Chapter 9. Defining day to day workload

### Job related

#### NOERROR syntax for JCL return code management

▶▶ NOERROR — LIST — ( — *error code entry* — ) ▶▶



For the parameter descriptions, see *Customization and Tuning*.

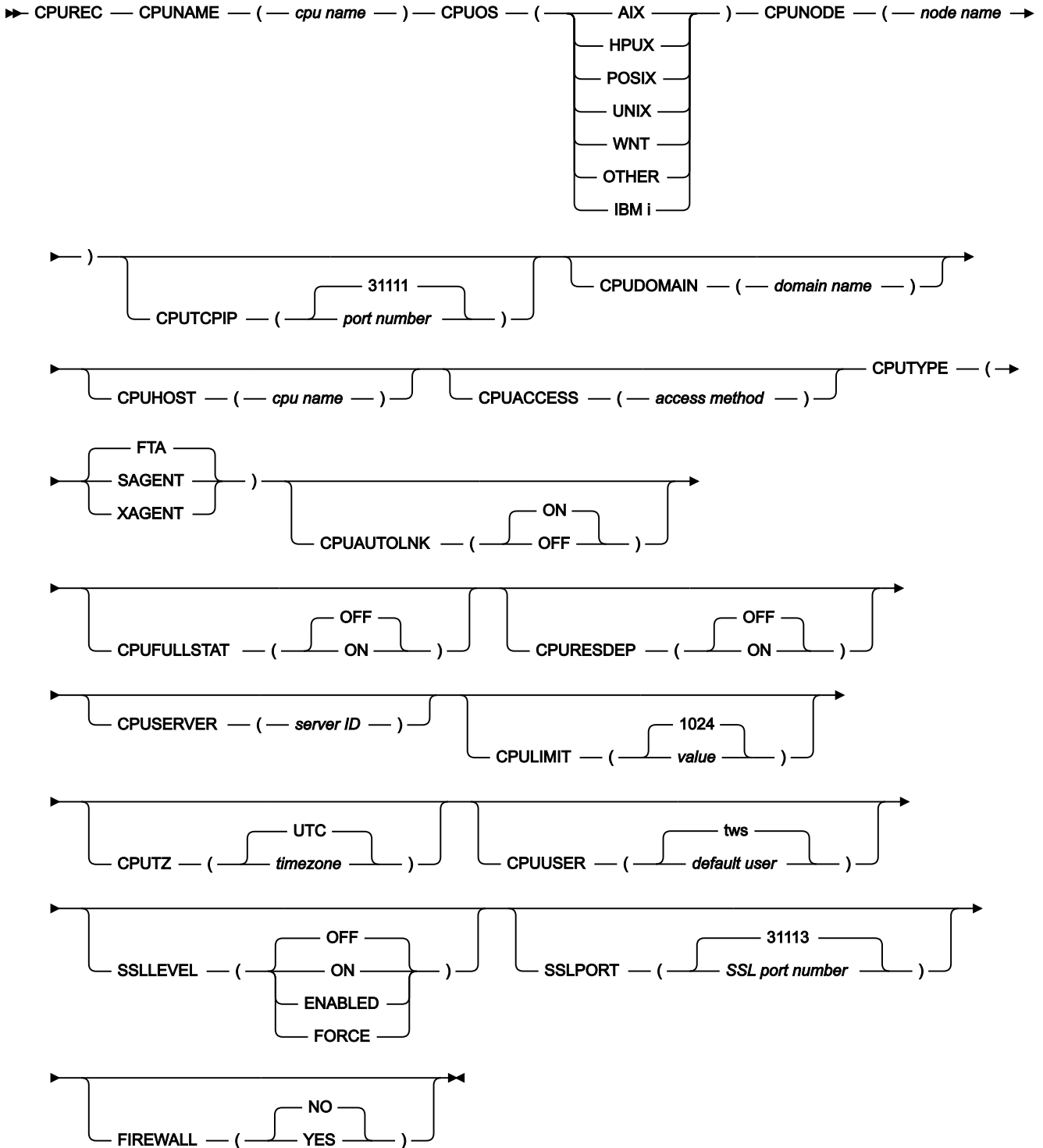
#### USRREC syntax for end-to-end scheduling

▶▶ USRREC — USRCPU — ( — *cpu name* — ) — USRNAM — ( — *logon ID* — ) — USRPSW — ( — *password* — ) ▶▶

For the parameter descriptions, see *Scheduling End-to-end with Fault Tolerance Capabilities*.

### Modeling related for end-to-end scheduling

## CPUREC



For the parameter descriptions, see *Scheduling End-to-end with Fault Tolerance Capabilities*.

## DOMREC

▶▶ DOMREC — DOMAIN — ( — *domain name* — ) — DOMMNGR — ( — *domain manager name* — ) — DOMPARENT — ( →  
▶ — *parent domain* — ) ▶◀

For the parameter description, see *Scheduling End-to-end with Fault Tolerance Capabilities*.

# Notices

This document provides information about copyright, trademarks, terms and conditions for product documentation.

© Copyright IBM Corporation 1993, 2016 / © Copyright HCL Technologies Limited 2016, 2025

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing*  
*IBM Corporation*  
*North Castle Drive, MD-NC119*  
*Armonk, NY 10504-1785*  
*US*

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

*Intellectual Property Licensing*  
*Legal and Intellectual Property Law*  
*IBM Japan Ltd.*  
*19-21, Nihonbashi-Hakozakicho, Chuo-ku*  
*Tokyo 103-8510, Japan*

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

*IBM Director of Licensing*

*IBM Corporation*

*North Castle Drive, MD-NC119*

*Armonk, NY 10504-1785*

*US*

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

The performance data discussed herein is presented as derived under specific operating conditions. Actual results may vary.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

#### COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

© (your company name) (year).

Portions of this code are derived from IBM Corp. Sample Programs.

© Copyright IBM Corp. 2025

## Trademarks

IBM, the IBM logo, and [ibm.com](http://ibm.com) are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM® or other companies. A current list of IBM® trademarks is available on the web at "Copyright and trademark information" at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Adobe™, the Adobe™ logo, PostScript™, and the PostScript™ logo are either registered trademarks or trademarks of Adobe™ Systems Incorporated in the United States, and/or other countries.

IT Infrastructure Library™ is a Registered Trade Mark of AXELOS Limited.

Linear Tape-Open™, LTO™, the LTO™ Logo, Ultrium™, and the Ultrium™ logo are trademarks of HP, IBM® Corp. and Quantum in the U.S. and other countries.

Intel™, Intel™ logo, Intel Inside™, Intel Inside™ logo, Intel Centrino™, Intel Centrino™ logo, Celeron™, Intel Xeon™, Intel SpeedStep™, Itanium™, and Pentium™ are trademarks or registered trademarks of Intel™ Corporation or its subsidiaries in the United States and other countries.

Linux™ is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft™, Windows™, Windows NT™, and the Windows™ logo are trademarks of Microsoft™ Corporation in the United States, other countries, or both.



Java™ and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Cell Broadband Engine™ is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

ITIL™ is a Registered Trade Mark of AXELOS Limited.

UNIX™ is a registered trademark of The Open Group in the United States and other countries.

## Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

**Applicability**

These terms and conditions are in addition to any terms of use for the IBM website.

**Personal use**

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

**Commercial use**

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

**Rights**

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

# Index

## Special Characters

\* (asterisk) operation status code 69

## Numerics

3 extended status code 70  
4 extended status code 70  
5 extended status code 70  
7 extended status code 70  
8 extended status code 70  
9 extended status code 70

## A

A extended status code 70  
A operation status code 69  
APARs 32  
  PH11184 24  
  PH12019 24  
  PH29226 24, 67  
  PI45919 70  
  PI58955 35  
  PK24559 73  
  PK69328 34  
  PK71539 40, 40  
  PK73549 50  
  PK87319 73  
  PM01090 45  
  PM04927 66  
  PQ6977 58, 59, 62  
  PQ87904 66  
  PQ93442 34  
asterisk (\*) operation status code 69  
availability  
  special resources 28

## B

B extended status code 70  
BACKUP command 10  
BACKUP keyword of JTOPTS 10  
blank, extended status code 72  
BULKDISC command 13  
BULKDISC keyword of MONOPTS 13

## C

C extended status code 70  
C occurrence status code 65  
C operation status code 69  
CAN error code 65  
CANCEL command, z/OS 38  
CCUN error code 65  
CDATE, JCL variable 64  
CDAY, JCL variable 63  
CDD, JCL variable 63  
CDDD, JCL variable 63  
CDDMMYY, JCL variable 63  
CHH, JCL variable 63  
CHHMM, JCL variable 63  
CLNA error code 65  
CLNC error code 65  
CLNO error code 66  
CMM, JCL variable 63  
CMMYY, JCL variable 63  
codes  
  error 65  
  extended status codes 70  
  job log retrieval 72  
  status codes for occurrences 65, 69  
  status codes for operations 69  
commands

z/OS  
  CANCEL 38  
  MODIFY 38  
  START 37  
  STOP 37  
CTIME, JCL variable 64  
current plan  
  on-request BACKUP 10  
  on-request BULKDISC 13  
  on-request JSUACT 15  
CWW, JCL variable 63  
CWWDD, JCL variable 63  
CYMD, JCL variable 63  
CYY, JCL variable 63  
CYYDDD, JCL variable 63  
CYYMM, JCL variable 63  
CYYMMDD, JCL variable 63  
CYYYY, JCL variable 64  
CYYYYMM, JCL variable 64

## D

D extended status code 70  
D occurrence status code 65  
D operation status code 69  
deviation of special resources 29

## E

E extended status code 70  
E occurrence status code 65  
E operation status code 69  
EQQUSIN subroutine 21  
error codes  
  automatic recovery 65  
  job log retrieval 72  
extended status codes 70

## F

FBND error code 66

## G

G extended status code 70

## H

H extended status code 70

## I

I operation status code 69  
in-progress operation 18

## J

JCCE error code 66  
JCL  
  error code 66  
  on-request BACKUP of repository file 10  
  on-request BULKDISC 13  
  on-request JSUACT of repository file 15  
JCLI error code 66  
job log  
  retrieval status codes 72  
JS file backup 10  
JS file jsuact 15  
JSUACT command 15  
JSUACT keyword of JTOPTS 15  
JTOPTS initialization statement  
  BACKUP keyword 10  
  JSUACT keyword 15  
  MAXJSFILE keyword 10  
  OPINFOSCOPE keyword 18  
  SUBFAILACTION keyword 68  
  SUPPRESSACTION keyword 68

WSFAILURE keyword 68  
WSOFFLINE keyword 67

## L

L extended status code 70  
lifespan of special resources 30  
LOOP error code 66

## M

M extended status code 70  
MAXJSFILE keyword of JTOPTS 11  
MCP error code 66  
MODIFY command, z/OS 38  
MONOPTS statement 13

## N

N extended status code 70  
nnnn error code 68  
normal mode manager 40

## O

O extended status code 70  
OADID, JCL variable 61  
OADOWNER, JCL variable 81  
OAGROUP, JCL variable 61  
OAUT error code 66  
OCALID, JCL variable 61  
occurrences  
  status codes 65  
OCCDATE, JCL variable 64  
OCFRSTC, JCL variable 64  
OCFRSTW, JCL variable 64  
OCFRSTWY, JCL variable 64  
OCLASTC, JCL variable 64  
OCLASTW, JCL variable 64  
OCLASTWY, JCL variable 64  
OCTIME, JCL variable 64  
ODAY, JCL variable 61  
ODD, JCL variable 61  
ODDD, JCL variable 61  
ODEA error code 66  
ODMY1, JCL variable 61  
ODMY2, JCL variable 61  
OFREEDAY, JCL variable 61  
OFxx error code 67  
OHH, JCL variable 61  
OHHMM, JCL variable 61  
OJCV error code 67  
OJOBNAME, JCL variable 62  
OLAT error code 66  
OLDAY, JCL variable 62  
OLDD, JCL variable 62  
OLHH, JCL variable 62  
OLHHMM, JCL variable 62  
OLMD, JCL variable 62  
OLMM, JCL variable 62  
OLWK, JCL variable 62  
OLYMD, JCL variable 62  
OLYYDDD, JCL variable 63  
OMM, JCL variable 61  
OMMY, JCL variable 61  
OOPNO, JCL variable 63  
operations  
  extended status codes 70  
  feedback of information 17  
  reason codes 72  
  status (OPSTAT) command 21  
  status codes 69  
  user data 17

- operator commands,
  - z/OS
    - 37
- OPIADATE, JCL variable 64
- OPIATIME, JCL variable 64
- OPINFO command
  - description 17
  - security considerations 18
  - syntax 17
- OPINFOSCOPE keyword of JTOPTS 18
- OPLSDATE, JCL variable 64
- OPLSTIME, JCL variable 64
- OPSTAT command
  - introduction 21
- OSEQ error code 67
- OSIN error code 67
- OSUB error code 67
- OSUF error code 67
- OSUP error code 68
- OSxx error code 68
- OWSID, JCL variable 63
- OWW, JCL variable 61
- OWWD, JCL variable 61
- OWWLAST, JCL variable 61
- OWWMONTH, JCL variable 61
- OXJOBNAM, JCL variable 63
- OYM, JCL variable 62
- OYMD, JCL variable 62
- OYMD1, JCL variable 62
- OYMD2, JCL variable 62
- OYMD3, JCL variable 62
- OYY, JCL variable 62
- OYYDDD, JCL variable 62
- OYYMM, JCL variable 62
- OYYYY, JCL variable 62

## P

- P occurrence status code 65
- PCAN error code 68
- PQ93442 32

## Q

- Q extended status code 70
- quantity of special resources 30

## R

- R extended status code 70
- R operation status code 69
- reason codes 72
- resources
  - changing the status 26
- restart and cleanup
  - error codes 66, 68, 68

## S

- S extended status code 70
- S occurrence status code 65
- S operation status code 69
- scope of OPINFO 18
- security
  - BACKUP command 11
  - OPINFO command 18
  - OPSTAT command 22
  - SRSTAT command 27
  - WSSTAT command 32
- SERC error code 68, 68
- SHPF error code 68
- special resources
  - status (SRSTAT) command 26
- SRSTAT command
  - introduction 26
- START command, z/OS 37

- status codes
  - extended 70
  - job log retrieval 72
  - occurrence 65
  - operation 69
- STOP command, z/OS 37
- SUBFAILACTION keyword of JTOPTS 68
- subsystem
  - cancelling with z/OS 38
  - modifying with z/OS 38
  - starting with z/OS 37
  - stopping with z/OS 37
- SUPPRESSACTION keyword of JTOPTS 68
- Sxxx error code 68

## T

- T extended status code 70
- TSO commands
  - BACKUP 10
  - BULKDISC 13
  - JSUACT 15
  - OPINFO 17
  - OPSTAT 21
  - SRSTAT 26
  - WSSTAT 32

## U

- U extended status code 70
- U occurrence status code 65
- U operation status code 69
- UNKN error code 68
- user data 17
- Uxxx error code 69

## V

- V extended status code 70

## W

- W extended status code 70
- W occurrence status code 65
- W operation status code 69
- WSFAILURE keyword of JTOPTS 68
- WSOFFLINE keyword of JTOPTS 67
- WSSTAT command 32

## X

- X extended status code 70
- xxxx error code 69

## Y

- Y extended status code 70

## Z

- z/OS
  - CANCEL command 38
  - MODIFY command 38
  - operator commands 37
  - START command 37
  - STOP command 37