FUJITSU Software
Systemwalker Operation Manager

Reference Guide

UNIX/Windows(R)

Preface

Purpose of This Document

This document provides reference information on the commands, JCL, etc. used in Systemwalker Operation Manager.

Intended Readers

This document is intended for those who will schedule, operate, monitor and manage jobs using Systemwalker Operation Manager.

This document assumes that you are familiar with how to work with operating Systems and GUIs.

Abbreviations and Generic Terms Used

- The term "Windows Server 2012 R2" is used to refer to all of the following products:

    - Microsoft(R) Windows Server(R) 2012 R2 Foundation (x64)

    - Microsoft(R) Windows Server(R) 2012 R2 Standard (x64)

    - Microsoft(R) Windows Server(R) 2012 R2 Datacenter (x64)

- The term "Windows Server 2012" is used to refer to all of the following products:

    - Microsoft(R) Windows Server(R) 2012 Foundation (x64)

    - Microsoft(R) Windows Server(R) 2012 Standard (x64)

    - Microsoft(R) Windows Server(R) 2012 Datacenter (x64)

    - Microsoft(R) Windows Server(R) 2012 R2 Foundation (x64)

    - Microsoft(R) Windows Server(R) 2012 R2 Standard (x64)

    - Microsoft(R) Windows Server(R) 2012 R2 Datacenter (x64)

- The term "Windows Server 2008 R2" is used to refer to all of the following products:

    - Microsoft(R) Windows Server(R) 2008 R2 Foundation

    - Microsoft(R) Windows Server(R) 2008 R2 Standard

    - Microsoft(R) Windows Server(R) 2008 R2 Enterprise

    - Microsoft(R) Windows Server(R) 2008 R2 Datacenter

- The term "Windows Server 2008 Foundation" is used to refer to all of the following products:

    - Microsoft(R) Windows Server(R) 2008 R2 Foundation

    - Microsoft(R) Windows Server(R) 2008 Foundation

- The term "Server Core" is used to refer to all of the following products:

    - Microsoft(R) Windows Server(R) 2012 Standard Server Core

    - Microsoft(R) Windows Server(R) 2012 Datacenter Server Core

    - Microsoft(R) Windows Server(R) 2008 Standard Server Core

    - Microsoft(R) Windows Server(R) 2008 Standard without Hyper-V(TM) Server Core

    - Microsoft(R) Windows Server(R) 2008 Enterprise Server Core

    - Microsoft(R) Windows Server(R) 2008 Enterprise without Hyper-V(TM) Server Core

    - Microsoft(R) Windows Server(R) 2008 Datacenter Server Core

- Microsoft(R) Windows Server(R) 2008 Datacenter without Hyper-V(TM) Server Core

- The term "Windows Server 2008 STD" is used to refer to all of the following products:

    - Microsoft(R) Windows Server(R) 2008 Standard (x86)/(x64)

    - Microsoft(R) Windows Server(R) 2008 Standard without Hyper-V(TM)

- The term "Windows Server 2008 DTC" is used to refer to all of the following products:

    - Microsoft(R) Windows Server(R) 2008 Datacenter (x86)/(x64)

    - Microsoft(R) Windows Server(R) 2008 Datacenter without Hyper-V(TM)

- The term "Windows Server 2008 EE" is used to refer to all of the following products:

    - Microsoft(R) Windows Server(R) 2008 Enterprise (x86)/(x64)

    - Microsoft(R) Windows Server(R) 2008 Enterprise without Hyper-V(TM)

- The term "Windows Server 2008" is used to refer to all of the following products:

    - Microsoft(R) Windows Server(R) 2008 Standard (x86)/(x64)

    - Microsoft(R) Windows Server(R) 2008 Enterprise (x86)/(x64)

    - Microsoft(R) Windows Server(R) 2008 Datacenter (x86)/(x64)

    - Microsoft(R) Windows Server(R) 2008 Foundation (x64)

    - Microsoft(R) Windows Server(R) 2008 Standard without Hyper-V(TM) (x86)/(x64)

    - Microsoft(R) Windows Server(R) 2008 Enterprise without Hyper-V(TM) (x86)/(x64)

    - Microsoft(R) Windows Server(R) 2008 Datacenter without Hyper-V(TM) (x86)/(x64)

    - Microsoft(R) Windows Server(R) 2008 R2 Foundation (x64)

    - Microsoft(R) Windows Server(R) 2008 R2 Standard (x64)

    - Microsoft(R) Windows Server(R) 2008 R2 Enterprise (x64)

    - Microsoft(R) Windows Server(R) 2008 R2 Datacenter (x64)

- The term "Windows Server 2003 STD" is used to refer to all of the following products:

    - Microsoft(R) Windows Server(R) 2003 R2, Standard x64 Edition

    - Microsoft(R) Windows Server(R) 2003 R2, Standard Edition

    - Microsoft(R) Windows Server(R) 2003, Standard x64 Edition

    - Microsoft(R) Windows Server(R) 2003, Standard Edition

- The term "Windows Server 2003 DTC" is used to refer to all of the following products:

    - Microsoft(R) Windows Server(R) 2003 R2, Datacenter x64 Edition

    - Microsoft(R) Windows Server(R) 2003 R2, Datacenter Edition

    - Microsoft(R) Windows Server(R) 2003, Datacenter x64 Edition

    - Microsoft(R) Windows Server(R) 2003, Datacenter Edition for Itanium-based Systems

    - Microsoft(R) Windows Server(R) 2003, Datacenter Edition

- The term "Windows Server 2003 EE" is used to refer to all of the following products:

    - Microsoft(R) Windows Server(R) 2003 R2, Enterprise x64 Edition

    - Microsoft(R) Windows Server(R) 2003 R2, Enterprise Edition

    - Microsoft(R) Windows Server(R) 2003, Enterprise x64 Edition

    - Microsoft(R) Windows Server(R) 2003, Enterprise Edition for Itanium-based Systems

- Microsoft(R) Windows Server(R) 2003, Enterprise Edition

- The term "Windows(R) 2000" is used to refer to all of the following products:

    - Microsoft(R) Windows(R) 2000 Professional

    - Microsoft(R) Windows(R) 2000 Server

    - Microsoft(R) Windows(R) 2000 Advanced Server

    - Microsoft(R) Windows(R) 2000 Datacenter Server

- The term "Windows NT(R)" is used to refer to all of the following products:

    - Microsoft(R) Windows NT(R) Server network operating system Version 4.0

    - Microsoft(R) Windows NT(R) Workstation operating system Version 4.0

- The term "Windows(R) 8.1" is used to refer to all of the following products:

    - Windows(R) 8.1 (x86)

    - Windows(R) 8.1 Pro (x86)

    - Windows(R) 8.1 Enterprise (x86)

    - Windows(R) 8.1 (x64)

    - Windows(R) 8.1 Pro (x64)

    - Windows(R) 8.1 Enterprise (x64)

- The term "Windows(R) 8" is used to refer to all of the following products:

    - Windows(R) 8 (x86)

    - Windows(R) 8 Pro (x86)

    - Windows(R) 8 Enterprise (x86)

    - Windows(R) 8 (x64)

    - Windows(R) 8 Pro (x64)

    - Windows(R) 8 Enterprise (x64)

    - Windows(R) 8.1 (x86)

    - Windows(R) 8.1 Pro (x86)

    - Windows(R) 8.1 Enterprise (x86)

    - Windows(R) 8.1 (x64)

    - Windows(R) 8.1 Pro (x64)

    - Windows(R) 8.1 Enterprise (x64)

- The term "Windows(R) 7" is used to refer to all of the following products:

    - Windows(R) 7 Home Premium (x86)

    - Windows(R) 7 Professional (x86)

    - Windows(R) 7 Enterprise (x86)

    - Windows(R) 7 Ultimate (x86)

    - Windows(R) 7 Home Premium (x64)

    - Windows(R) 7 Professional (x64)

    - Windows(R) 7 Enterprise (x64)

    - Windows(R) 7 Ultimate (x64)

- The term "Windows Vista(R)" is used to refer to all of the following products:

  - Windows Vista(R) Home Basic (x86)

  - Windows Vista(R) Home Premium (x86)

  - Windows Vista(R) Business (x86)

  - Windows Vista(R) Enterprise (x86)

  - Windows Vista(R) Ultimate (x86)

  - Windows Vista(R) Home Basic (x64)

  - Windows Vista(R) Home Premium (x64)

  - Windows Vista(R) Business (x64)

  - Windows Vista(R) Enterprise (x64)

  - Windows Vista(R) Ultimate (x64)

- The term "Windows(R) XP" is used to refer to all of the following products:

  - Microsoft(R) Windows(R) XP Professional x64 Edition

  - Microsoft(R) Windows(R) XP Professional

  - Microsoft(R) Windows(R) XP Home Edition

- Microsoft(R) Windows(R) Millennium Edition is abbreviated as "Windows(R) Me".

- Microsoft(R) Windows(R) 98 operating system is abbreviated as "Windows(R) 98".

- Microsoft(R) Windows(R) 2000 Server is abbreviated as "Windows(R) 2000 Server".

- Windows Internet Explorer(R) is abbreviated as "Internet Explorer".

- Versions of Systemwalker Operation Manager that run on all of the following operating systems are referred to as "Windows versions of Systemwalker Operation Manager" or simply "Windows versions":

  - Windows

  - 64-bit versions of Windows, except Itanium

- Articles specific to the version of Systemwalker Operation Manager that runs on 32-bit versions of Windows are referred to as "Windows x86 version".

- Articles specific to the version of Systemwalker Operation Manager that runs on Itanium-compatible versions of Windows are referred to as "Windows for Itanium version".

- Articles specific to the version of Systemwalker Operation Manager that runs on 64-bit versions of Windows, except Itanium, are referred to as "Windows x64 version".

- Windows(R) 2000, Windows Server 2003 STD, Windows Server 2003 DTC, Windows Server 2003 EE, Windows Server 2008, Windows Server 2008 EE, Windows Server 2008 DTC, Windows Server 2008 STD, Windows Server 2008 Foundation, Windows Server 2008 R2, Server Core, Windows Server 2012, and Windows Server 2012 R2 may be abbreviated as "Windows servers".

- Oracle Solaris may be referred to as Solaris, Solaris Operating System or Solaris OS.

- Versions of Systemwalker Operation Manager that run on Solaris are referred to as "Solaris versions of Systemwalker Operation Manager" or simply "Solaris versions".

- Articles specific to the version of Systemwalker Operation Manager that runs on 32-bit versions of Solaris are referred to as "Solaris 32-bit version".

- Articles specific to the version of Systemwalker Operation Manager that runs on 64-bit versions of Solaris are referred to as "Solaris 64-bit version".

- Versions of Systemwalker Operation Manager that run on HP-UX are referred to as "HP-UX versions of Systemwalker Operation Manager" or simply "HP-UX versions".

- Versions of Systemwalker Operation Manager that run on AIX are referred to as "AIX versions of Systemwalker Operation Manager" or simply "AIX versions".

- Versions of Systemwalker Operation Manager that run on the following operating system are referred to as "Linux versions of Systemwalker Operation Manager" or simply "Linux versions":

    - Linux

    - 64-bit versions of Linux, except Itanium

- Articles specific to the version of Systemwalker Operation Manager that runs on 32-bit versions of Linux are referred to as "Linux x86 version".

- Articles specific to the version of Systemwalker Operation Manager that runs on Itanium-compatible version of Linux are referred to as "Linux for Itanium version".

- Articles specific to the version of Systemwalker Operation Manager that runs on 64-bit versions of Linux, except Itanium, are referred to as "Linux x64 version".

- Solaris, HP-UX, AIX, Linux and Linux for Itanium versions of Systemwalker Operation Manager are referred to collectively as "UNIX versions of Systemwalker Operation Manager" or simply "UNIX versions".

- Solaris, HP-UX, AIX and Linux may be referred to as "UNIX servers".

- Systemwalker Operation Manager Standard Edition may be abbreviated as "SE".

- Systemwalker Operation Manager Enterprise Edition may be abbreviated as "EE".

- Standard Edition may be abbreviated as "SE" and Enterprise Edition may be abbreviated as "EE".

- BrightStor(R) ARCserve(R) Backup for Windows is abbreviated as "ARCserve".

- Microsoft(R)-Mail that is provided as a standard feature with Windows NT(R) is abbreviated as "MS-Mail".

Export Restriction

Exportation/release of this document may require necessary procedures in accordance with the regulations of your resident country and/or US export control laws.

Trademarks

APC and PowerChute are trademarks or registered trademarks of American Power Conversion Corporation.

ARCserve is a registered trademark of CA, Inc. or one of its subsidiaries.

HP-UX is a registered trademark of Hewlett-Packard Development Company.

IBM, the IBM logo, AIX, HACMP, and ViaVoice are trademarks or registered trademarks of International Business Machines Corporation in the United States and/or other countries.

Intel and Itanium are trademarks of Intel Corporation in the U.S. and/or other countries.

MC/ServiceGuard is a registered trademark of Hewlett-Packard Company.

Microsoft, Windows, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo and JBoss are registered trademarks of Red Hat, Inc. in the U.S. and other countries.

Linux is a registered trademark of Linus Torvalds.

Oracle and Java are registered trademarks of Oracle and/or its affiliates.

R/3 and SAP are registered trademarks of SAP AG in Germany and in several other countries.

Tcl/Tk is free software developed by the University of California, Sun Microsystems, Inc., Scriptics Corporation, and other parties.

UNIX is a registered trademark of The Open Group.

VMware, the VMware logo, and VMotion are registered trademarks or trademarks of VMware, Inc. in the United States and/or other jurisdictions.

Short Mail is a registered trademark of NTT DoCoMo, Inc.

Other names may be trademarks or registered trademarks of their respective owners.

The use of screenshots follows the guidelines of Microsoft Corporation.

May 2015

Contents

Chapter 1 Shared Commands

This chapter describes the shared commands provided by Systemwalker Operation Manager. You must run the commands except the swsilent command on servers in which Systemwalker Operation Manager server functions are installed.

🔔 Note

If commands are executed in an environment where the UAC function for Windows Server 2008 or later has been enabled, then a dialog box asking a user's privileges to be promoted may be displayed depending on the privileges of the user that logged in to the operating system, and action may be required.

Refer to "Notes on Executing Commands in Windows [Windows version]" in the *Systemwalker Operation Manager User's Guide* for more information.

1.1 soperationmgr Service/Daemon Startup Command

Synopsis

[Windows]

```
soperationmgr [/a]
```

[UNIX]

```
soperationmgr [-a]
```

Description

This command starts Systemwalker Operation Manager services/daemons in batch.

Options

/a [Windows]
-a [UNIX]

Specified to simultaneously start the services/daemons of Systemwalker Operation Manager and Systemwalker Centric Manager when the two systems coexist on the same machine. If this option is omitted, only the Systemwalker Operation Manager services/daemons including common services/daemons are started.

If Systemwalker Centric Manager is not installed, the Systemwalker Operation Manager services/daemons are started, regardless of whether this option is specified or not.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/opt/systemwalker/bin

Privilege Required

System administrator (user belonging to the Administrators group or superuser) privileges are required.

Cautions

- A full path must be specified with this command.

- If the services/daemons common to Systemwalker Operation Manager and Systemwalker Centric Manager are already active in Systemwalker Centric Manager when the startup request is made with this command, this request is accepted because multiple startup is ignored.

- If an error occurs during service/daemon activation, correct the error and then reexecute the soperationmgr command to restart the services/daemons.

- Service designated as automatic in [Service] on [Control Panel] can be started up with this command. To activate a service, set the service to be activated automatically in [Service] on [Control Panel] in advance. [Windows]

- If SystemWalker/CentricMGR V5.0L10 or earlier has been installed, its services can also be started up with this command. [Windows]

- If you execute this command using a Systemwalker Operation Manager job, the operation of Systemwalker Operation Manager and Systemwalker Centric Manager cannot be guaranteed. See "Commands that Should Not Be Executed as Jobs" in the *Systemwalker Operation Manager User's Guide* for details.

- In Red Hat Enterprise Linux 7, the operating system may need to be stopped or restarted even if the daemons are configured to stop automatically when the operating system is stopped.
  In this case, manually execute the poperationmgr service/daemon termination command in advance to stop the Systemwalker Operation Manager daemons. [Linux]

1.2 poperationmgr Service/Daemon Termination Command

Synopsis

[Windows]

```
poperationmgr [/a]
```

[UNIX]

```
poperationmgr [-s|-a]
```

Description

This command stops Systemwalker Operation Manager services/daemons in batch.

Options

/a [Windows]
-a [UNIX]

Specify to simultaneously stop both services/daemons of Systemwalker Centric Manager and Systemwalker Operation Manager when the two systems coexist on the same machine. If this option is omitted, only the Systemwalker Operation Manager services/daemons that do not affect Systemwalker Centric Manager operations are stopped.

If Systemwalker Centric Manager is not installed, the Systemwalker Operation Manager services/daemons are stopped, regardless of whether this option is specified or not.

-s [UNIX]

Specify to stop ACL manager daemon if Systemwalker Centric Manager and Systemwalker Operation Manager are installed on the same machine. If this option is omitted, ACL manager daemon will not be stopped.

If Systemwalker Centric Manager is not installed, ACL manager daemon will be stopped regardless of whether this option is specified.

This option cannot be specified together with the -a option.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin

Solaris HP-UX AIX Linux	/opt/systemwalker/bin

Privilege Required

System administrator (user belonging to the Administrators group or superuser) privileges are required.

Cautions

- A full path must be specified with this command.

- Confirm that there are no job nets running before stopping the services/daemons.

- If an error occurs while stopping the services/daemons, correct the error and then reexecute the poperationmgr command to stop the services/daemons.

- If SystemWalker/CentricMGR V5.0L10 or earlier has been installed, its services can be stopped with this command. [Windows]

- In the system where Systemwalker Operation Manager and Systemwalker Centric Manager are both installed, the following Systemwalker Centric Manager functions cannot be used while ACL manager daemon is stopped (by specifying the -s option). [UNIX]

    - GUI manipulation

    - Resource Distribution

    In addition, when ACL manager stops, the ACL manager daemon process (MpFwsec) which is the common function to Systemwalker Centric Manager and Systemwalker Operation Manager also stops. In response to this event, the process monitoring function of Systemwalker Centric Manager outputs the following message. However, no action against the message is required since it gives no impacts to the system operation.

    [Message]

    > Check if the Systemwalker Centric Manager process (MpFwSec) is running properly.

- If you execute this command using a Systemwalker Operation Manager job, the operation of Systemwalker Operation Manager and Systemwalker Centric Manager cannot be guaranteed. See "Commands that Should Not Be Executed as Jobs" in the *Systemwalker Operation Manager User's Guide* for details.

- In Red Hat Enterprise Linux 7, the operating system may need to be stopped or restarted even if the daemons are configured to stop automatically when the operating system is stopped.
  In this case, manually execute the poperationmgr service/daemon termination command in advance to stop the Systemwalker Operation Manager daemons. [Linux]

1.3 swstop Service/Daemon Startup Suppression Command

Synopsis

```
swstop
```

Description

This command suppresses startup of Systemwalker Operation Manager services/daemons if the system is restarted after installation of Systemwalker Operation Manager. If Systemwalker Centric Manager V11.0L10/V11.0 or later coexists, this command also suppresses the startup of Systemwalker Centric Manager.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Command Location

Windows	<*Systemwalker Operation Manager installation directory*>\mpwalker.jm\mpcmtool\bin
Solaris HP-UX AIX Linux	/opt/FJSVftlo/bin

Privilege Required

System administrator (user belonging to the Administrators group or superuser) privileges are required.

Cautions

- This command can be executed on servers.

- Because this command is provided in Systemwalker Centric Manager V11.0L10/V11.0 or later, it cannot suppress the startup of Systemwalker Centric Manager services if the system coexists with Systemwalker CentricMGR V10.0L21/V10.1 or earlier.

- If Systemwalker Centric Manager V11.0L10 or 11.0 coexists, cancel the settings for startup suppression with swstart (service/daemon startup suppression release command) before installing or uninstalling Systemwalker Operation Manager. See "1.4 swstart Service/Daemon Startup Suppression Release Command" for details.

Examples

[Windows]

```
d:\win32app\mpwalker.jm\mpcmtool\bin\swstop
```

[UNIX]

```
/opt/FJSVftlo/bin/swstop
```

1.4 swstart Service/Daemon Startup Suppression Release Command

Synopsis

```
swstart
```

Description

This command controls startup of Systemwalker Operation Manager services/daemons and starts them up. Also, this command controls startup of Systemwalker Centric Manager services/daemons if Systemwalker Centric Manager V11.0L10/V11.0 or later coexists, and starts them up.

Return Values

Return value	Meaning
0	Normal termination

Return value	Meaning
Value other than 0	Abnormal termination

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\mpwalker.jm\mpcmtool\bin
Solaris HP-UX AIX Linux	/opt/FJSVftlo/bin

Privilege Required

System administrator (user belonging to the Administrators group or superuser) privileges are required.

Caution

- This command can be executed on servers.

- If you execute this command using a Systemwalker Operation Manager job, the operation of Systemwalker Operation Manager and Systemwalker Centric Manager cannot be guaranteed. See "Commands that Should Not Be Executed as Jobs" in the *Systemwalker Operation Manager User's Guide* for details.

Examples

[Windows]

```
d:\win32app\mpwalker.jm\mpcmtool\bin\swstart
```

[UNIX]

```
/opt/FJSVftlo/bin/swstart
```

1.5 swsilent Silent Installation Command

Synopsis

[Windows]

```
swsilent.exe -i File [-y]
```

[UNIX]

<Server>

```
swsilent.sh -i File [-y]
```

<Client>

```
swsilent.exe -i File [-y]
```

Description

This command performs silent installation of Systemwalker Operation Manager.

Options

-i *File*

Specify silent installation information in full path. If blanks are contained in the middle, enclose them in double quotation marks ("").

-y

Specify when automatically restarting the system after silent installation.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Command Location

Windows x86	<*drive*>\MAIN\tool\sscmd
Windows x64 (server)	<*drive*>\MAIN\tool\sscmd
Windows x64 (client)	<*drive*>\Client\tool\sscmd
Solaris	<*drive*>/Solaris/tool_unx/sscmd
Linux	<*drive*>/Linux/tool_unx/sscmd
HP-UX AIX	<*drive*>/tool_unx/sscmd

Privilege Required

- System administrator (user belonging to the Administrators group or superuser) privileges are required.

- When executing this command on Windows operating systems, do it on a Command Prompt opened with system administrator privileges.

Cautions

- This command can be executed on servers and clients.

- Since this command is silent, if an option error occurs, it is recorded in the following logs.

[Windows]

```
Windows directory\swsilent.log
```

[Solaris]

<Server>

```
/var/tmp/swsilent.log
```

<Client>

```
Windows directory\swsilent.log
```

[HP-UX, AIX, Linux]

<Server>

```
/tmp/swsilent.log
```

<Client>

```
Windows directory\swsilent.log
```

- Systemwalker Operation Manager will start automatically when installation is completed (Windows) for some conditions.

  Systemwalker Operation Manager will not start when installation is completed under any of the following conditions:

  - Where Systemwalker Centric Manager has been installed

  - It is not a new installation

  - When Restart is required because the Microsoft Visual C++ 2005 Redistributable package installed is older than the version bundled with this product

Examples

The following examples perform silent installation according to the specified installation information file and automatically restart the system after installation.

[Windows x86]

```
f:\MAIN\tool\sscmd\swsilent.exe -i c:\swinst.ini -y
```

[Solaris]

```
/dev/cd0/Solaris/tool_unx/sscmd/swsilent.sh -i /work/swinst.ini -y
```

[Linux]

```
/dev/cd0/Linux/tool_unx/sscmd/swsilent.sh -i /work/swinst.ini -y
```

1.6 jmmode Continuous Execution Mode Switching Command

Synopsis

```
jmmode {continue|cancel|check} [-func {JOB|WRT}]
[-sys {all|subsystem number}] (*)
```

*:

[-sys {all|subsystem number}] is the option of Systemwalker Operation Manager EE.

Description

This command defines whether or not the following two high-reliability functions are to be enabled respectively, as required.

- Continuous execution of the network jobs when the schedule server experiences system down.

- Duplication of the schedule information file (writing to backup file).

Options

{continue|cancel|check}

Specify whether or not the high-reliability functions are enabled.

continue

Enables the high-reliability function specified with the -func option. Enables all the high-reliability functions if the -func option is omitted.

cancel

Disables the high-reliability function specified with the -func option. Disables all the high-reliability functions if the -func option is omitted.

Cancel is the default value for both JOB and WRT.

check

Displays the status of the current high-reliability function. If this option is specified, settings of all other options are ignored, only providing status display.

-func {JOB|WRT}

Specify whether or not the high-reliability functions are enabled. The continue and cancel options are used for the enable/disable setting, respectively. If this option is omitted, all high-reliability functions are either enabled or disabled altogether.

JOB

Specifies continuous execution of the network job.

You need to specify the same status of this option on the schedule server and the execution server. If the status of all the functions is not identical, jobs will not run properly.

When enabled with the continue option, execution of the network job is not interrupted in case of system down on the schedule server.

When disabled with the cancel option, execution of the network job is forced to terminate in case of system down on the schedule server.

WRT

Specifies duplication of the schedule information file.

When enabled with the continue option, the schedule information file is duplicated by writing the schedule information file into the original file as well as into the backup file.

When disabled with the cancel option, the schedule information file is not written into a backup file. Despite this rule, however, the file is written into a backup file under the following conditions:

  - When registration information (job net/group information) is added/updated/deleted.

`EE`

-sys {all|subsystem number}

In systems with multiple operating subsystems, this option is used to specify a subsystem as operating target. If this option is omitted, subsystem number 0 will be the operating target.

all

Specify to include all subsystems as operating targets.

subsystem number

Specify operating target subsystem number using a range of 0 to 9.

Output Example

The following is an example of display with the check option specified. "on" appears if the option is enabled, and "off" if disabled.

```
JOB on
WRT off
```

`EE`

The following is an example of display with the EE version.

```
SYS 0
JOB on
WRT off
```

```
SYS 1
JOB off
WRT off
```

Cautions

When specifying the "continue" or "cancel" option, services/daemons need to be stopped. Stop and start the following services/daemons from the [Systemwalker Operation Manager Environment Setup] window.

- Jobscheduler

- Job Execution Control

1.7 mpbko Backup Command

Synopsis

[Windows]

```
mpbko drive name:directory name [/Ca|/Cs] [/d] [/SN]
```

[UNIX]

```
mpbko -b Directory [-d] [-SN]
```

Description

This command backs up operation management information such as environment definition information and job scheduling information stored on the Systemwalker Operation Manager server to the specified directory.

Options

drive name:directory name [Windows]

Specify the backup directory. Specify this option with up to 32 alphanumeric characters. Blanks cannot be used.

drive name

Specify the drive name of the backup destination hard disk (which should be write-enabled).

directory name

Specify the backup directory.

-b *Directory* [UNIX]

Specify the full path of the backup directory in alphanumeric characters. If the specified directory does not exist, it will be created automatically. Blanks cannot be contained in the directory name of the backup destination.

/Ca|/Cs [Windows]

Use this option for backup on a cluster system. Omit this option if you wish to back up information in a normal system for non-cluster operations. You must specify this option after "drive name: directory name."

/Ca

Use this option for backup on a running node. Information stored on a running node and shared disk will be backed up.

/Cs

Use this option for backup on a standby node. Information stored on a standby node will be backed up.

/d [Windows]
-d [UNIX]

Use this option for backup of resources in silent mode without outputting any confirmation message at backup. When specifying this option, the backup directory must be empty and if not, this command will terminate with an error.

/SN [Windows]
-SN [UNIX]

Use this option for backup without stopping Systemwalker Operation Manager services/daemons.

Return Values

Return value	Meaning
0	Normal termination
1	Abnormal termination

Command Location

Windows	<*Systemwalker Operation Manager installation directory*>\MpWalker.JM\MpCmtool \backup\bin
Solaris HP-UX AIX Linux	/opt/systemwalker/bin

Privilege Required

System administrator (user belonging to the Administrators group or superuser) privileges are required.

Cautions

- When specifying a backup directory with this command, it must be specified with a full path name.

- If you execute this command without the /SN or -SN option, any active service/daemon temporarily stops. In an environment where Systemwalker Centric Manager and Systemwalker Operation Manager coexist, the services/ daemons of the two systems stop simultaneously.

- It is recommended that the Operation Environment Maintenance Wizard be used to back up the operation environment, in order to avoid operating errors, errors entering the command, or similar errors. Refer to "Backing Up or Restoring Operation Environment" in the *Systemwalker Operation Manager Installation Guide* for details on the procedure. When executing the command, take care to avoid operating errors and errors entering the command. [Windows]

- Before executing this command, be sure to check the following:

    - Access privileges to the backup directory are granted.

    - There is plenty of free space in the backup directory. See "Backup Capacity" for required space.

- If the backup was not successful, check the following. And then execute the mpbko command again after taking an appropriate action.

    - Access privileges to the backup directory are granted

    - There is plenty of free space in the backup directory. See "Backup Capacity" for required space.

- If a backup is made and the information is restored while a job, job net or group is running, the job, job net or group will be terminated abnormally with completion code 239. We therefore recommend to make a backup during the time when a job, job net or group is not running.

- Directory name located on the network drive cannot be specified.

Backup Capacity

[Windows]

The capacity required on the backup directory varies depending on the environment or on the amount of registered data. Estimate the required capacity on the basis of the following values: Calculate the capacity of the variable part and add it to the capacity of the fixed part.

Fixed Part

The fixed part has a capacity of 586 Kilobytes.

Variable Part

The capacity of the variable part is as follows:

- When the calendar function is used:

    - Number of calendars x 6 Kilobytes

- When the event monitoring function is used:

    - Number of event lines defined x 2 Kilobytes

    This number is equal to the number of lines in a list, which is displayed as the line count in the [Monitored Event Table] window.

- When Jobscheduler is used:

    - Number of projects registered x 1.5 Kilobytes

    - Number of job nets x 8 Kilobytes

    - Number of jobs x 2 Kilobytes

    - Number of groups x 2.4 Kilobytes

    - Number of job nets in each group x 12 Kilobytes

    - Total capacity of various exits

    - Total capacity for output (stdout/stderr) from all jobs

    - Specified log size x 3 bytes

- When Job Execution Control is used:

    - Number of job folders x 1 Kilobyte

    - Number of job owners x 0.5 Kilobyte

    - Number of queues x 4 Kilobytes

    - Maximum number of jobs acceptable by the system x 40 Kilobytes

    - debug.log file size (not fixed)

[UNIX]

See the following calculation formula for required space on the backup

Total size of directories under "/etc/opt/FJSV*" + Total space of the directories under "/var/opt/FJSV*"

FJSV*: Package name of component that is contained in the product.

FJSV* needs to be replaced depending on the OS as follows. Calculate the capacity after replacing it with the corresponding value.

```
Solaris version--> FJSV*

Linux version--> FJSV*

HP-UX version--> FHP*
```

```
AIX version--> FAIX*
```

Examples

[Windows]

The following is a setting example for backing up the operation management information and log information such as the environment definition information for the Systemwalker Operation Manager server and job scheduling information in the "d:\tmp\OMGRBack" directory if Systemwalker Operation Manager is installed under "c:\win32app."

```
cd c:\win32app\MpWalker.JM\MpCmtool\backup\bin
mpbko d:\tmp\OMGRBack
```

[UNIX]

The following is an example for backing up the resources to the "/backup" directory.

```
/opt/systemwalker/bin/mpbko -b /backup
```

Information

Registering mpbko command as a job

The mpbko command can be registered as a job. If it is actually registered as a job, ensure you do the following:

- Specify the /SN or -SN option so as not to stop the services or daemons during the backup.

- Specify the /d or -d option to the command so as not to output any confirmation messages during the backup.

- Check that the backup destination directory is empty before performing the backup.

1.8 mprso Restore Command

Synopsis

[Windows]

```
mprso drive name:directory name [/Ln|/Lo|/Le] [/Ca|/Cs]
```

[UNIX]

```
mprso -b Directory
```

Description

This command restores Systemwalker Operation Manager server's environment definition information or operation management information that has been backed up with the mpbko command to the same version/level of Systemwalker Operation Manager.

Options

drive name:directory name [Windows]

Specify where to store the information backed up by the mpbko command. Specify this option with up to 32 alphanumeric characters. Blanks can not be used.

drive name

Specify the drive name of the hard disk to store the information backed up by the mpbko command.

directory name

Specify the name of the directory to store the information backed up by the mpbko command.

-b *Directory* [UNIX]

Specify the full path of the directory to which the resources have been backed up in alphanumeric characters. Blanks cannot be contained in the directory name.

/Ln|/Lo|/Le [Windows]

Specify which type of information should be restored, either log information or definition information, or both. If omitted, /Le is assumed. You must specify this option after "drive name:directory name."

/Ln

Specify to restore only definition information. Log information is not restored.

/Lo

Specify to restore only log information. Definition information is not restored.

/Le

Specify to restore both log and definition information.

/Ca|/Cs [Windows]

Use this option for restoration on a cluster system. Omit this option if you restore information in a normal system for operations other than cluster operations. Specify this option after the /Ln, /Lo, or /Le option.

/Ca

Use this option for restoration on a running node. Information is restored on a running node and a shared disk. You cannot specify this option for restoring the backup that was obtained through the mpbko command with the /Cs option specified.

/Cs

Use this option for restoration on a standby node. Information is restored on a standby node but not on shared disk.

Return Values

Return value	Meaning
0	Normal termination
1	Abnormal termination

Reference

mpbko

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\MpCmtool\backup\bin
Solaris HP-UX AIX Linux	/opt/systemwalker/bin

Privilege Required

System administrator (user belonging to the Administrators group or superuser) privileges are required.

Cautions

- When specifying a storage directory with this command, it must be specified with a full path name.

- It is recommended to use the Operation Environment Maintenance Wizard to restore the operation environment, in order to avoid operating errors, errors entering the command, or similar errors. Refer to "Backing Up or Restoring Operation Environment" in the *Systemwalker Operation Manager Installation Guide* for details on the procedure. When executing the command, take care to avoid operating errors and errors entering the command. [Windows]

- If you execute this command, any active services/daemons stop. If you execute this command in an environment where Systemwalker Centric Manager coexists, the services/daemons of the two systems stop simultaneously.

- Be sure to confirm the following before executing this command:

  - Access privileges to the backup directory and restoration directory are granted.

  - There is plenty of free space in the restoration destination directory.

- Only the resources backed up by the mpbko command in this version/level can be restored.

- Operating system, installation type, installation directory, and character code must be the same between the backup source and restore destination. Otherwise, restoration cannot be performed.

- Directory name located on the network drive cannot be specified.

- If you execute this command using a Systemwalker Operation Manager job, the operation of Systemwalker Operation Manager and Systemwalker Centric Manager cannot be guaranteed. See "Commands that Should Not Be Executed as Jobs" in the *Systemwalker Operation Manager User's Guide* for details.

Examples

[Windows]

The following is a setting example for restoring the operation management information and log information backed up in the "d:\tmp\OMGRBack" directory if Systemwalker Operation Manager is installed under "c:\win32app."

```
cd c:\win32app\MpWalker.JM\MpCmtool\backup\bin
mprso d:\tmp\OMGRBack
```

[UNIX]

The following is an example for restoring the server resources that have been backed up to "/backup."

```
/opt/systemwalker/bin/mprso -b /backup
```

1.9 omgrmonitor Monitored Host Information Import Command

Synopsis

[Windows]

```
omgrmonitor {-export|-get|-import} file_name [-name monitored host configuration name]
```

[UNIX]

```
omgrmonitor {-export|-import} file_name [-name monitored host configuration name]
```

Description

This command outputs information to a file in CSV format.

Also this command enables you to retrieve CSV format files as the monitored host information of Systemwalker Operation Manager.

The information on the basic tree registered in the configuration information database of Systemwalker Centric Manager can be output to a CSV format file if a Systemwalker Centric Manager Operation Management Server is installed. [Windows]

Options

-export *file_name*

Exports monitored host information to the specified file in CSV format.

-get *file_name* [Windows]

Outputs basic tree information from Systemwalker Centric Manager's configuration information database to the specified file in CSV format.

-import *file_name*

Imports the specified file in CSV format as Systemwalker Operation Manager's monitored host information.

-name *monitored host configuration name*

Specifies the monitored host configuration name to be imported. If this option is omitted, "DEFAULT" information is output or imported.

Caution

- If you specify the -get option to output the basic tree information registered in the configuration information database of Systemwalker Centric Manager, execute this command on Systemwalker Centric Manager's Operation Management Server. [Windows]

Example Output of Basic Tree Information

```
Folder number,Previous folder number,Folder name,Display name,Host name,IP address
1,,"root",,,
2,1,"astroNT",,,
3,2,"Local department 1",,,
4,3,"10.11.11.0",,,
,4,,"Operation Management Server","astroNT",10.11.111.111
5,1,"Local department2",,,
6,5,"10.111.111.0",,,
,6,,"111","111",10.111.111.111
,6,,"222","222",10.111.111.112
,6,,"333","333",10.111.111.113
7,5,"10.111.222.0",,,
```

If you display the above files in spreadsheet software, they appear in the following format:

Example Display of Basic Tree Information

	Folder number	Previous folder number	Folder name	Display name	Host name	IP address
1	Folder number	Previous folder number	Folder name	Display name	Host name	IP address
2	1		root			
3	2	1	astroNT			
4	3	2	Local department 1			
5	4	3	10.11.11.0			
6		4		Operation Management Server	astroNT	10.11.11.111
7	5	1	Local department 2			
8	6	5	10.111.111.0			
9		6		111	111	10.111.111.111
10		6		222	222	10.111.111.112
11		6		333	333	10.111.111.113
12	7	5	10.111.222.0			

Notes for creating monitored host information in a CSV format file

The title in the first line is a mandatory item. Do not change it.

The root folder in the second line is a mandatory item. Do not change it.

Write the third and subsequent lines as follows:

When specifying a folder:

1. Specify a folder number. This number must be unique.

2. Specify the previous folder number

3. Specify the folder name of a folder.

The folder number, previous folder number and folder name must be specified.

Do not specify any other information.

When specifying a host:

1. Specify the previous folder number.

2. Specify the display name (which is displayed in the Connected host name combo box). Specify it within 128 bytes.

3. Specify the host name (DNS name) in the following format: "host name" or "host name.domain name" (with a full stop between host name and domain name).

4. Specify the IP address.

The previous folder number, display name, host name, and IP address must be specified.

Do not specify any other information.

🔰 Note

. . . .

When importing a CSV format file into spreadsheet software, all-number character strings may be converted as follows:

Notation in CSV file: 0010 -> Notation in spreadsheet software: 10

This is a side effect where the spreadsheet software mistakenly recognizes a string which consists only of numeric digits as a number, rather than as a string. In such case, first change the format setting for the relevant cell so as to have it handled as a character string, then correct the character string to match the notation in the CSV file.

. . . .

1.10 createsubsystem.exe/sh Subsystem Environment Create Command

Synopsis

[Windows]

```
createsubsystem.exe -sys subsystem-number
```

[UNIX]

```
createsubsystem.sh -sys subsystem-number
```

Description

This command creates a subsystem environment.

Options

-sys subsystem-number

Specify the subsystem number to be created, ranging from 1 to 9.

Return Values

[Windows]

Return value	Meaning
0	Normal termination
1	Failed due to argument error, or privilege error.
2	Failed to create a subsystem environment with Jobscheduler.
3	Failed to create a subsystem environment with Job Execution Control.
4	Failed to create an environment of Jobscheduler and Job Execution Control.

[UNIX]

Return value	Meaning
0	Normal termination
1	Failed due to argument error or privilege error.

Reference

[Windows]

deletesubsystem.exe

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin
Solaris Linux	/opt/FJSVJMCMN/bin/
HP-UX	/opt/FHPJMCMN/bin/
AIX	/opt/FAIXJMCMN/bin/

Privilege Required

A user who belongs to the Administrators group and who has privilege of functioning as part of the operating system or superuser is required.

Example

[Windows]

```
createsubsystem.exe -sys 1
```

1.11 deletesubsystem.exe Subsystem Environment Delete Command [Windows]

Synopsis

```
deletesubsystem.exe -sys subsystem-number
```

Description

This command deletes a subsystem environment.

Option

-sys subsystem number

Specify subsystem number to be deleted using a range from 1 to 9.

Return Values

Return value	Meaning
0	Normal termination
1	Failed due to argument error or privilege error.
2	Failed to delete a subsystem environment with Jobscheduler.
3	Failed to delete a subsystem environment with Job Execution Control.
4	Failed to delete an environment of Jobscheduler and Job Execution Control.

Reference

createsubsystem.exe

Command Location

Windows	<*Systemwalker Operation Manager installation directory*>\MpWalker.JM\bin

Privilege Required

System administrator (user belonging to the Administrators group) privileges are required.

Caution

Execute this command after stopping services of Jobscheduler and Job Execution Control corresponding to the subsystem to be deleted.

Example

```
deletesubsystem.exe -sys 1
```

1.12 swmove Backup Command for Migration [UNIX]

Synopsis

```
swmove -b directory
```

Description

This command backs up the resources of older version of Systemwalker Operation Manager to migrate them to this version. The specified resources are backed up to the specified backup directory.

Option

-b directory:

Specify the full path of the backup directory in alphanumeric characters. If the specified directory does not exist, it will be created automatically. Blanks cannot be contained in the directory name.

Return Values

Return value	Meaning
0	Normal termination
1	Abnormal termination

Reference

rmpackage.inf

This file contains a list of old packages that are required to be deleted during the Systemwalker Operation Manager migration. If the deletion fails during the migration, you should delete the old packages by referring to this file. This file is created directly under the backup directory specified with the -b option.

Command Location

Solaris	*<drive>*/*<mount point>*/Solaris/unx/tool
Linux	*<drive>*/*<mount point>*/Linux/unx/tool
HP-UX AIX	*<drive>*/*<mount point>*/unx/tool

Privilege Required

System administrator (superuser) privileges are required.

Cautions

- To back up the resources, you must follow the procedures described in the *Systemwalker Operation Manager Upgrade Guide*.

- Make sure to confirm the following before executing this command:

    - The backup directory is writable.

    - The backup directory has sufficient free space.

  Error handling:

      In case of insufficient space, you should create sufficient space by deleting unnecessary files or by specifying a different directory, and then reexecute the command. If an error occurs while deleting old packages, you should also reexecute the command. You can delete old packages on your own responsibility by referring to rmpackage.inf.

- Directory name located on the network directory cannot be specified.

- When executing this command, Systemwalker Operation Manager is stopped automatically. If Systemwalker Centric Manager coexists, each daemon of Systemwalker Centric Manager will also be stopped automatically.

Example

The following example backs up the migration data to "/work/move":

[Solaris]

<*drive*>/<*mount point*>/Solaris/unx/tool/swmove -b /work/move

[Linux]

<*drive*>/<*mount point*>/Linux/unx/tool/swmove -b /work/move

1.13 swtrans Conversion/Registration Command for Migration [UNIX]

Synopsis

```
swtrans -b directory
```

Description

This command migrates the resources of older version of Systemwalker Operation Manager that have been backed up with the swmove command to this version. By executing this command, the migration resources will be restored from the specified backup directory.

Option

-b directory

Specify the full path of the backup directory to which the migration resources have been backed up in alphanumeric characters. Blanks cannot be contained in the directory name.

Return Values

Return value	Meaning
0	Normal termination
1	Abnormal termination

Command Location

Solaris	<*drive*>/<*mount point*>/Solaris/unx/tool
Linux	<*drive*>/<*mount point*>/Linux/unx/tool
HP-UX AIX	<*drive*>/<*mount point*>/unx/tool

Privilege Required

System administrator (superuser) privileges are required.

Cautions

- To migrate the resources, you must follow the procedures described in the *Systemwalker Operation Manager Upgrade Guide*.

- The operating system, installation type, installation directory, and character codes must be the same on the migration source and destination.

- Make sure to confirm the following before executing this command:

    - Access privileges to the backup directory and installation directory are granted.

    - The installation directory on the migration destination has sufficient free space.

- Directory name located on the network drive cannot be specified.

- When executing this command, Systemwalker Operation Manager is stopped automatically. If Systemwalker Centric Manager coexists, each daemon of Systemwalker Centric Manager will also be stopped automatically.

- The following message is output when this command is executed.

```
May I shift by this backup data? [y or n](Default: n)
```

To restore the specified asset, enter "y".

Example

The following example restores the migration data from "/work/move":

[Solaris]

```
<drive>/<mount point>/Solaris/unx/tool/swtrans -b /work/move
```

[Linux]

```
<drive>/<mount point>/Linux/unx/tool/swtrans -b /work/move
```

EE

1.14 mpsetcluster Cluster Information Registration/Deletion Command

Synopsis

[Windows]

```
mpsetcluster.exe
```

[UNIX]

```
mpsetcluster -c|-d
```

Description

This command registers or deletes the cluster information of Systemwalker Operation Manager.

Options

-c [UNIX]

Registers the cluster information.

-d [UNIX]

Deletes the cluster information.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\mpwalker.jm\mpcmtool \bin
Solaris HP-UX AIX Linux	/opt/systemwalker/bin

Privilege Required

System administrator (user belonging to the Administrators group or superuser) privileges are required.

Cautions

- If the cluster information is not registered by using this command, backup/restore function does not work properly because it cannot refer to the shared disk information.

- If the cluster information is already registered, you have to delete it by using this command before uninstalling Systemwalker Operation Manager.

Examples

[Windows]

```
mpsetcluster.exe
```

By executing this command, the [Cluster Registration/Remove] window is displayed.
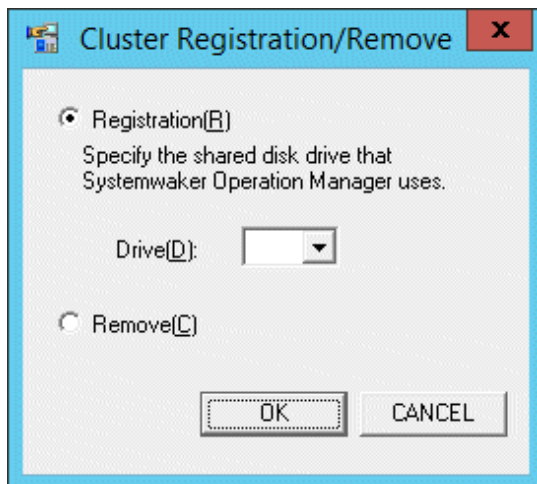
[Registration]

Registers cluster information

[Remove]

Removes cluster information

[UNIX]

The following example registers the cluster information:

```
mpsetcluster -c
```

The following example deletes the cluster information:

```
mpsetcluster -d
```

1.15 swpkginfo Product Information Display Command (for Server) [UNIX]

Synopsis

```
swpkginfo [-l]
```

Description

This command checks the product information of the currently installed Systemwalker Centric Manager and Systemwalker Operation Manager using the simple checkout method.

In Linux and Solaris, it can be confirmed whether the installation media is 32bit or 64bit.

Option

-l:

Outputs a list of installed packages.

Command Location

Solaris HP-UX AIX Linux	/opt/FJSVftlo/bin/swpkginfo

Cautions

If an incorrect option is specified, the command description (USAGE) is output.

Example

```
/opt/FJSVftlo/bin/swpkginfo
```

Execution Result

If the option is not specified:

```
*************************************************
Systemwalker Operation Manager Information
*************************************************
PRODUCTNAME   : Systemwalker Operation Manager Enterprise Edition (32bit)
VERSION       : V13.8.0
VENDER        : COPYRIGHT FUJITSU LIMITED 1995-2014
CODE          : ASCII
INSTALLDATE   : 2015.05.15
PLATFORM      : Linux
INSTALLCLASS  : Server
INSTALLPATH   : /opt
                /etc/opt
                /var/opt
```

If "-l:" is specified:

```
*************************************************
Systemwalker Operation Manager Information
*************************************************
PRODUCTNAME   : Systemwalker Operation Manager Enterprise Edition (32bit)
VERSION       : V13.8.0
VENDER        : COPYRIGHT FUJITSU LIMITED 1995-2014
```

```
CODE          : ASCII
INSTALLDATE   : 2015.05.15
PLATFORM      : Linux
INSTALLCLASS  : Server
INSTALLPATH   : /opt
                /etc/opt
                /var/opt

PACKAGES      : Install of Packages as follows.
                FJSVswaic
                 Systemwalker Authentication and ID Management Client
                FJSVftlo
                 Systemwalker Operation Manager - FrameWork Tool Operation Manager
                FJSVfwtco
                 Systemwalker Operation Manager - Script Base
                FJSVJMCMN
                 Systemwalker Operation Manager - Operation Manager Common
                FJSVfwseo
                 Systemwalker Operation Manager - ACL Manager
                FJSVJOBSC
                 Systemwalker Operation Manager - JOB(command/shell script) Scheduling
                FJSVMJS
                 Job Execution Control Server
                FJSVsnjss
                 Systemwalker Operation Manager - NeTJobServe for Operation Manager Server
                FJSVjmcal
                 Systemwalker Operation Manager - Calendar Services
                FJSVstem
                 Systemwalker Operation Manager - Solution Template
                FJSVjmweb
                 Systemwalker Operation Manager - Web Console
                FJSVjmman
                 Systemwalker Operation Manager - Web Manual
```

1.16 F3crfver Product Information Display Command

Synopsis

```
F3crfver.exe [/OPER | -f result-destination-filename]
```

Description

This command checks the product information of Systemwalker Operation Manager currently installed using the simple checkout method.

Options

Not specified

Displays an overview of Systemwalker Operation Manager product information in the dialog box.

If Systemwalker Centric Manager is installed, its product information is also displayed simultaneously.

/OPER

Displays the details of Systemwalker Operation Manager product information in the dialog box.

-f result-destination-filename

Specify this option to output an overview of the product information to a file.

When specifying this option, do not enter a space between "-f" and the file name that follows the -f, and specify this file name with its full path name.

Do not specify a path which contains a non-existing directory.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>* \mpwalker.jm\mpcmtool\bin \F3crfver.exe

Caution

If installation type is "client," no edition is displayed.

Example

The following example displays the details of Systemwalker Operation Manager product information.

```
Systemwalker Operation Manager installation directory\mpwalker.jm\mpcmtool\bin
\F3crfver.exe /OPER
```

Execution Results/Output Format

The following shows the execution results of this command.

- When the option is not specified:

  [Windows]

[UNIX]

Version Information

FUJITSU

Systemwalker Centric Manager
V15.0.0 (Build-20131024)
Client License
Operation Management Client

FUJITSU Software
Systemwalker Operation Manager
Vxx.x.x (Build-20141205)
Client

Copyright 1995 - 2014 FUJITSU LIMITED

OK
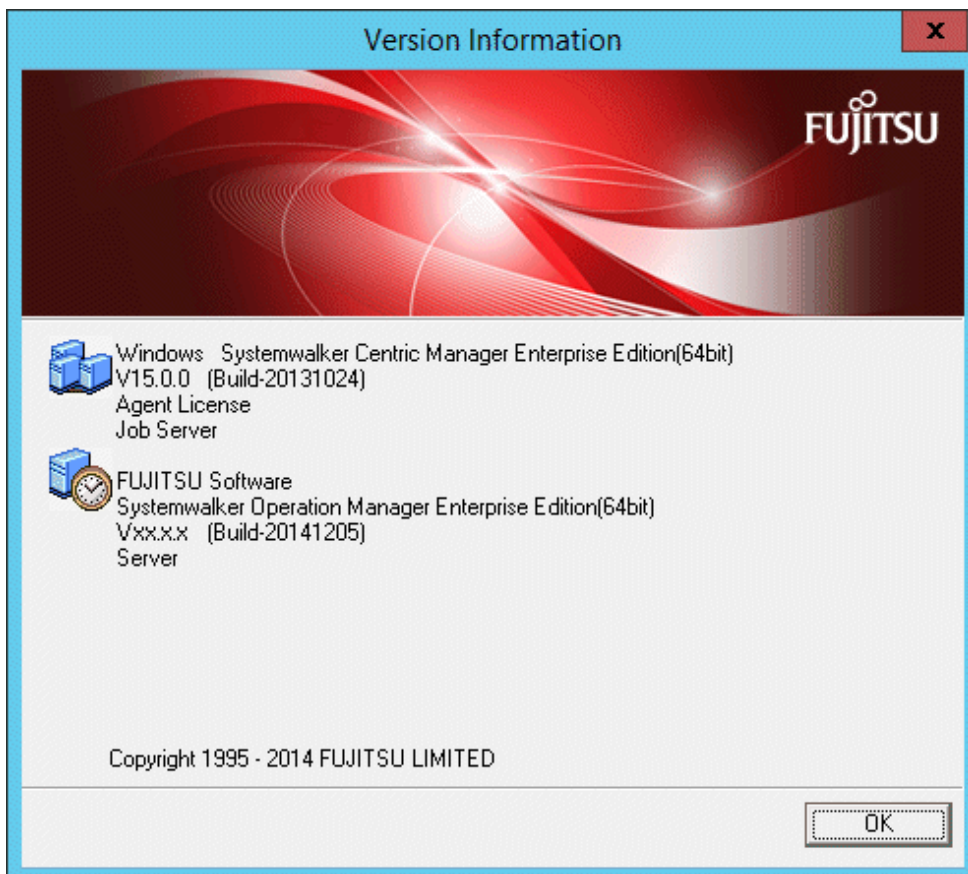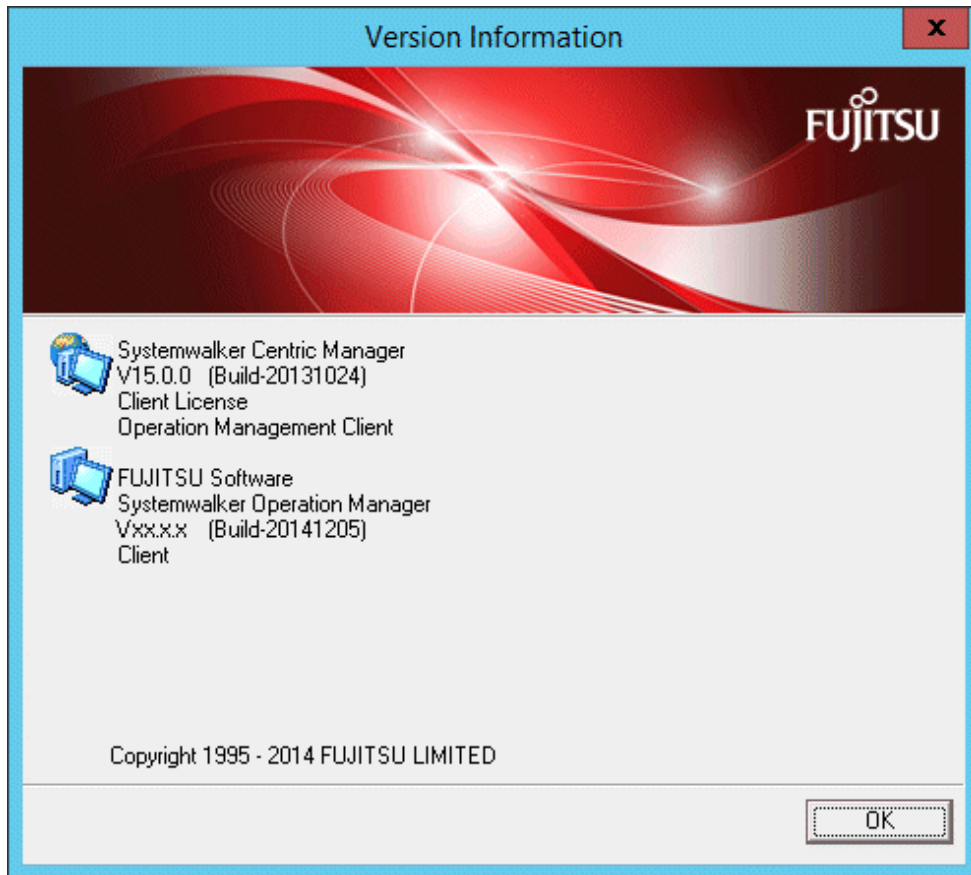
- When "/OPER" is specified

[Windows]
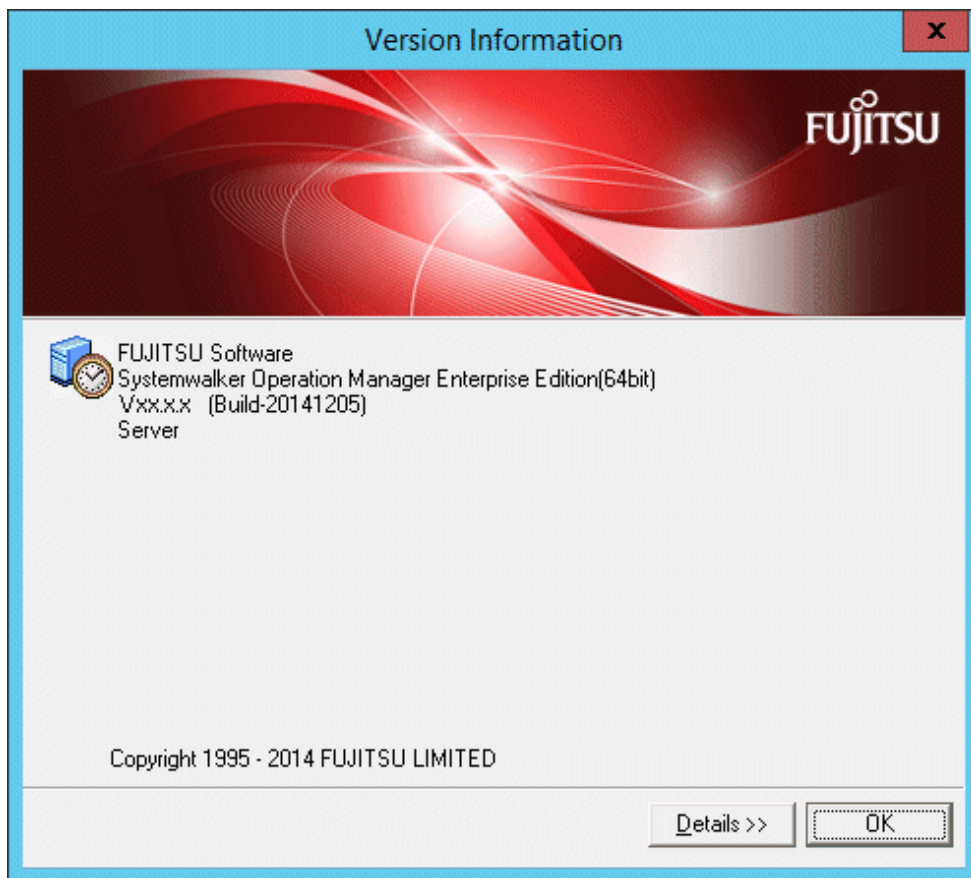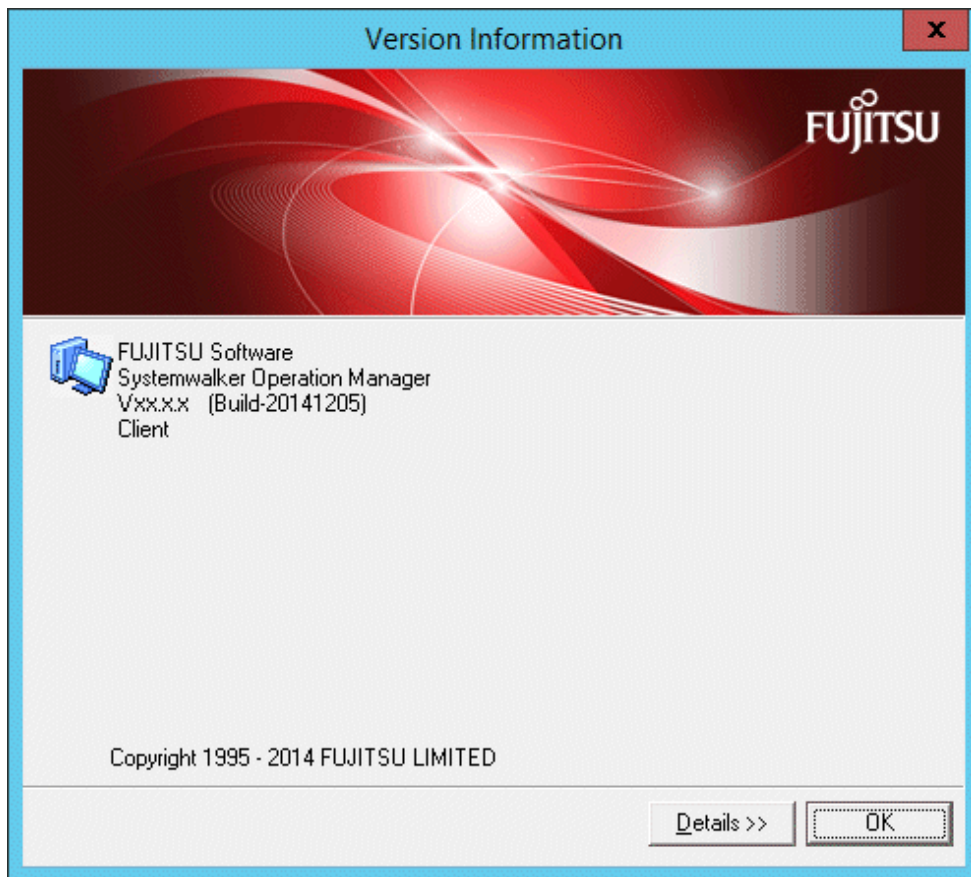
[UNIX]

You can check the installed functions by clicking the Details button.

[Windows]
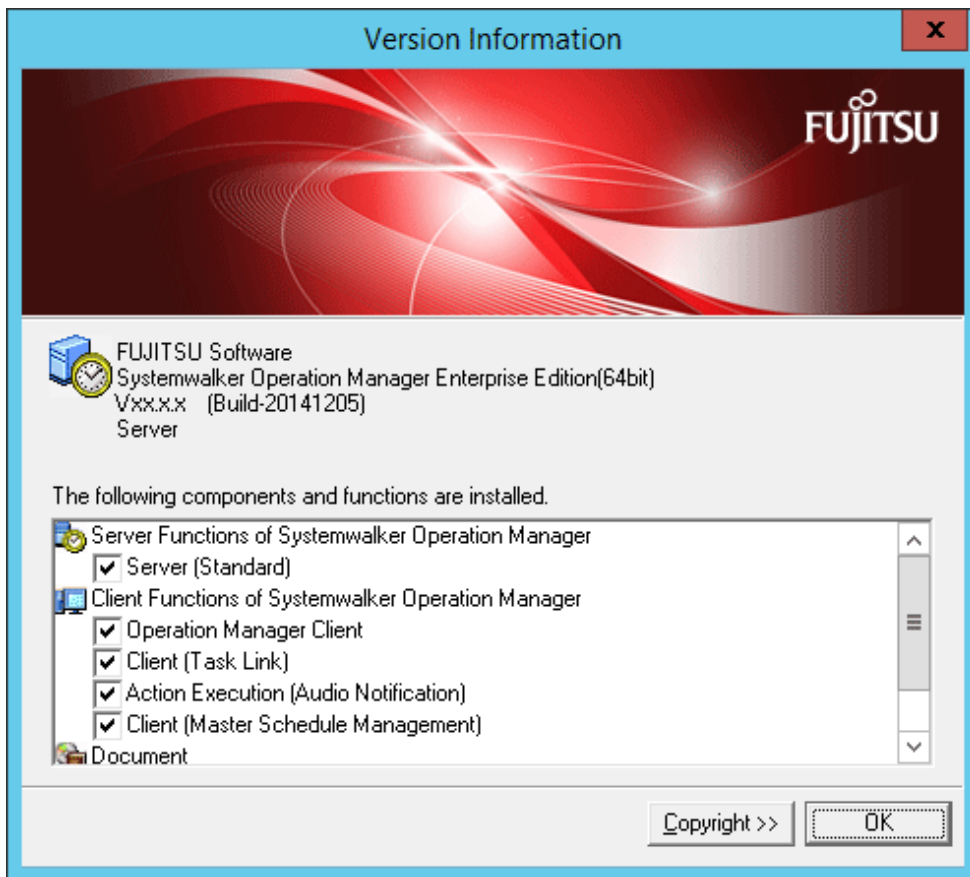
[UNIX]
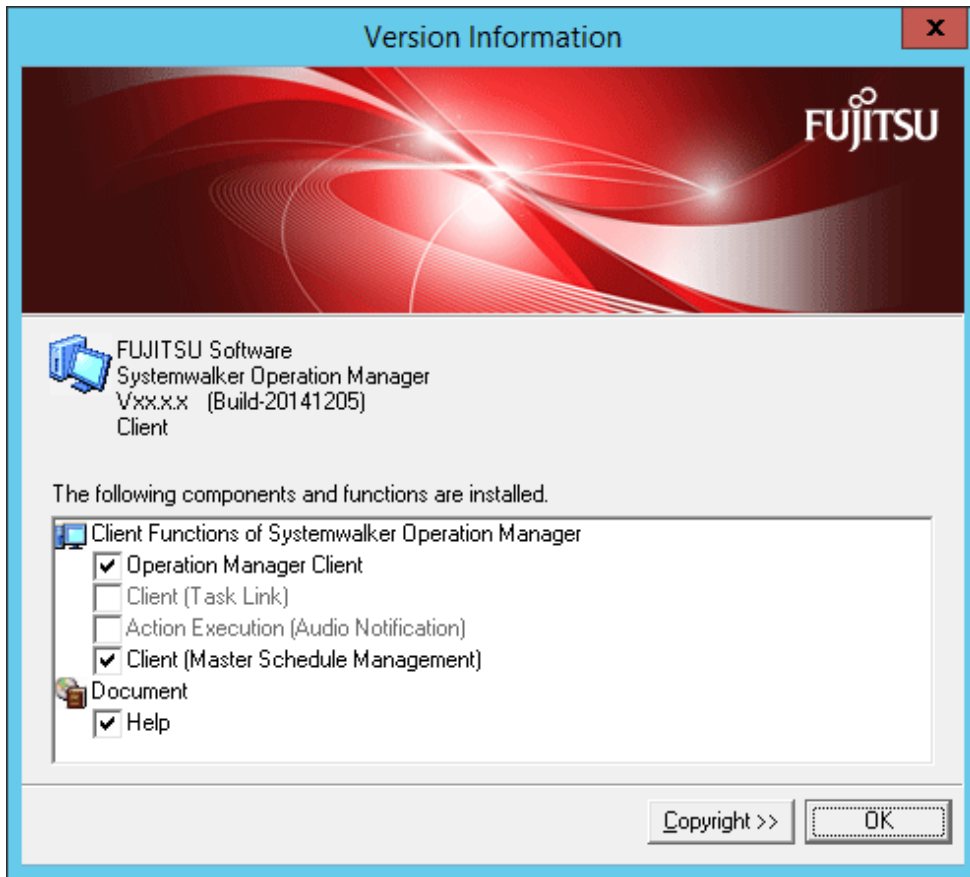
- When "-f" is specified:

[Windows]

```
Windows version   Systemwalker Centric Manager Enterprise Edition (64bit)
V15.0.0   (Build-20131024)
Agent licnese
Job Server

FUJITSU Software
Systemwalker Operation Manager Enterprise Edition(64bit)
V13.8.0 (Build-20141020)
Server
```

[UNIX]

```
Systemwalker Centric Manager
V15.0.0   (Build-20131024)
Client license
Operation Management Client

FUJITSU Software
Systemwalker Operation Manager
V13.8.0   (Build-20141020)
Client
```

If the installation type is "client", the edition will not be displayed.

1.17 setup Systemwalker Operation Manager Installation Command [Windows]

Synopsis

```
setup.exe
```

Description

Start the Systemwalker Operation Manager installer. When this command is executed, the **Systemwalker Setup** window is displayed. Systemwalker Operation Manager can be installed by following the windows and specifying the settings displayed.

Return Values

The GUI will appear immediately after the command is executed, even if the installation process is not complete. A return value will be produced, however this value does not provide any relevant information about the installation process.

Command Location

Windows x86	<*drive*>\MAIN\win32
Windows x64 (server)	<*drive*>\MAIN\win32
Windows x64 (client)	<*drive*>\Client\win32

Privilege Required

System administrator (user belonging to the Administrators group) privileges are required.

Cautions

This command cannot be used to perform a silent installation. Refer to "Silent Installation" in the *Systemwalker Operation Manager Installation Guide* for more information on silent installations.

Examples

[Windows x86]

```
<drive>\MAIN\win32\setup.exe
```

1.18 swuset Systemwalker Operation Manager Uninstallation Command [Windows]

Synopsis

```
swuset.exe
```

Description

This command uninstalls Systemwalker Operation Manager.

Return Values

The GUI will appear immediately after the command is executed, even if the uninstallation process is not complete. A return value will be produced, however this value does not provide any relevant information about the uninstallation process.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\mpwalker.jm \jmunins

Privilege Required

System administrator (user belonging to the Administrators group) privileges are required.

Cautions

Refer to "Notes Prior to Uninstallation", "Tasks Prior to Uninstallation", and "Systemwalker Uninstallation from the Windows System" in the *Systemwalker Operation Manager Installation Guide* for notes and cautions.

Examples

<Systemwalker Operation Manager installation directory>\mpwalker.jm\jmunins \swuset.exe

1.19 mpowebsetup Web Server Setup Command

Synopsis

[Windows]

mpowebsetup.bat [-u]

[UNIX]

mpowebsetup [-u]

Description

Use this command to set up a Web server.

Options

-u

Removes the Web server setup.

Return Values

Return value	Meaning
0	Normal termination
1 or more	Abnormal termination

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\mpwalker.jm\mpjmweb\bin

Solaris Linux	/opt/FJSVjmweb/bin

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- If the mpowebsetup command is executed after the setup has already been performed, such as immediately after installing this product, the command will end normally without performing the setup. Similarly, if "mpowebsteup -u" is executed after the setup has been removed, the command will end normally without removing the setup.

- Before executing the mpowebsetup command, stop Systemwalker Operation Manager by using the poperationmgr command. Immediately after executing the mpowebsetup command, start Systemwalker Operation Manager by using the soperationmgr command for the settings to take effect.

Examples

Set up a Web server.

[Windows]

```
mpowebsetup.bat
```

[UNIX]

```
mpowebsetup
```

Remove the Web server setup.

[Windows]

```
mpowebsetup.bat -u
```

[UNIX]

```
mpowebsetup -u
```

Execution Results/Output Format

The following message is output to the standard output when the command terminates normally:

```
mpowebsetup command has been completed successfully.
```

The following message is output to the standard error output when the command terminates abnormally:

```
Web server setup failed.
```

1.20 mpowebconfig Web Server Configuration Update Command

Synopsis

[Windows]

```
mpowebconfig.bat -o <file name>|-i <file name>
```

[UNIX]

```
mpowebconfig -o <file name>|-i <file name>
```

Description

This command updates the information about port numbers on the Web server of the system where the command is executed.

Options

-o *<file name>*

Outputs the port number definitions currently set for the Web server to the Web server port definition file in XML format. Specify the full path of the file to be output in *<file name>*.

-i *<file name>*

Reflects the port number definition information in the XML-format Web server port definition file to the Web server. Specify the full path of the file name in *<file name>*.

Return values

Return value	Meaning
0	Normal termination
1 or more	Abnormal termination

Reference

Web server port definition file

Refer to "18.7 Web Server Port Definition File" for details.

Command location

Windows	*<Systemwalker Operation Manager installation directory>*\mpwalker.jm\mpjmweb\bin
Solaris Linux	/opt/FJSVjmweb/bin

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- Before updating port numbers with the -i option after the Web server has been set up, use the poperationmgr command to stop Systemwalker Operation Manager. Restart Systemwalker Operation Manager by using the soperationmgr command for the updated port numbers to take effect.

- If a Web server has not been set up when the port numbers are updated with the -i option, set up the Web server after executing this command, by executing mpowebsetup (web server setup command).

Examples

Output the port number definitions currently set for the Web server to the Web server port definition file in XML format.

[Windows]

```
mpowebconfig.bat -o C:\tmp\port.xml
```

[Solaris]

```
mpowebconfig -o /var/tmp/port.xml
```

[Linux]

```
mpowebconfig -o /tmp/port.xml
```

Reflect the port number definition information in the XML-format Web server port definition file to the Web server.

[Windows]

```
mpowebconfig.bat -i C:\tmp\port.xml
```

[Solaris]

```
mpowebconfig -i /var/tmp/port.xml
```

[Linux]

```
mpowebconfig -i /tmp/port.xml
```

Execution Results/Output Format

The following message is output to the standard output when the command terminates normally:

```
mpowebconfig command has been completed successfully.
```

The following message is output to the standard error output when the command terminates abnormally:

- Execution result 1

```
Failed to open the file (%1).
```

```
%1: Input file path
```

- Execution result 2

```
Format of input file (%1) is invalid.
```

```
%1: Input file path
```

- Execution result 3

```
Value (%2) in input file (%1) is duplicated.
```

```
%1: Input file path
%2: Value of the port number that is duplicated
```

- Execution result 4

```
Failed to open the output file (%1).
```

```
%1: File path specified as the output directory
```

Chapter 2 Security Commands

This chapter describes the commands for setting security provided by Systemwalker Operation Manager. Accordingly, you must run the commands on servers in which Systemwalker Operation Manager server functions are installed.

Note

....

If commands are executed in an environment where the UAC function for Windows Server 2008 or later has been enabled, then a dialog box asking a user's privileges to be promoted may be displayed depending on the privileges of the user that logged in to the operating system, and action may be required.

Refer to "Notes on Executing Commands in Windows [Windows version]" in the *Systemwalker Operation Manager User's Guide* for more information.

....

2.1 mpsetacl Command

Synopsis

[Windows]

```
[To set/modify]
mpsetacl -a -t Target {-u User | -g Group | -w Wellknown } [-s Domain] -p {w | c | x | r}
mpsetacl -a -t Target {-U SwUser | -G SwGroup }-p {w | c | x | r}
[To delete]
mpsetacl -d -t Target [ [ {-u User | -g Group | -w Wellknown } ] [-s Domain] ]
mpsetacl -d -t Target [-U SwUser | -G SwGroup]
```

[UNIX]

```
[To set/modify]
mpsetacl -a -t Target {-u User | -g Group } -p {w | c | x | r}
mpsetacl -a -t Target -U SwUser | -G SwGroup } -p {w | c | x | r}
[To delete]
mpsetacl -d -t Target [-u User | -g Group ]
mpsetacl -d -t Target [-U SwUser | -G SwGroup]
```

Description

This command sets, modifies, or deletes user or group access rights for projects specified with *Target*.

Options

-a

Specify when setting or changing user or group access rights for a project.

-t *Target*

Specify the project name for which access rights are set, modified, or deleted in the following format.

[Windows]

```
mpjobsch\schedule\project name
```

During multiple subsystem operation, specify a project in the following format. (Note)

```
mpjobsch\schedulen\project name
```

```
n: Subsystem number (1 to 9)
```

[UNIX]

```
mpjobsch/schedule/project name
```

EE

```
mpjobsch/schedulen/project name
```

```
n: Subsystem number (1 to 9)
```

-u *User*

Specify the name of the user to whom access rights are granted. However, use the "-U option" when access rights are granted to the user in the Systemwalker authentication repository.

If the user name includes any space characters, enclose the entire user name in double quotations ("").

The user's principal name cannot be specified. [Windows]

-g *Group*

Specify the name of the group to which access rights are granted.

If the group name includes any space characters, enclose the entire group name in double quotations ("").

-w *Wellknown* **[Windows]**

Specify the name of the well-known group to whom access rights are granted. The well-known group name indicates the groups such as "Everyone" and "NETWORK" that are registered on the system beforehand.

If the well-known group name includes any space characters, enclose the entire well-known group name in double quotations ("").

-s *Domain* **[Windows]**

Specify the domain of the user specified with "-u" or the group specified with "-g." The default for this option is the domain name of the user or group on the computer where this command is executed.

-p

Specifies the access right to be granted. One of the following operands must be specified: w, x, r or c. Update rights include change rights, operation rights and reference rights. Similarly, both change rights and operation rights include reference rights.

Specify the access rights to be granted. Make sure to specify one of the following:

- w: Update right
- c: Change rights
- x: Operating right
- r: Reference right

-d

Specify when deleting a user's or a group's access rights for a project.

Note that if you specify this option but omit the user (-u option) or group (-g option), all access rights for the project specified with *Target* are deleted. If a user or a group is specified, all access rights for the specified user or group are deleted from the project specified with *Target*.

-U *SwUser*

When using the Systemwalker authentication repository, specify the name of the user in the Systemwalker authentication repository to whom access rights are granted.

If the user name includes any space characters, enclose the entire user name in double quotations ("").

-G *SwGroup*

When using the Systemwalker authentication repository, specify the name of the group in the Systemwalker authentication repository to whom access rights are granted.

If the group name includes any space characters, enclose the entire user name in double quotations ("").

Command Location

Windows	Systemwalker Operation Manager installation directory\mpwalker\bin
Solaris HP-UX AIX Linux	Systemwalker Operation Manager installation directory/FJSVfwseo/bin

Privilege Required

System administrator (user belonging to the Administrators group or superuser) privileges are required.

Cautions

- If "-a" is specified, the user (-u option), group (-g option) or well-known group (-w option) must be specified.

- In cluster environments, execute this command only on the active node (the node where the shared disk is mounted).

  The settings configured with this command on the active node will be applied to the standby node at the time of next failover. Check which nodes are the active node or standby node in the cluster system currently used.

Examples

When setting access rights:

The following example shows how to state this command in order to establish the update right as the access right of the user "user1" set in the project "proj1."

[Windows]

```
mpsetacl -a -t mpjobsch\schedule\proj1 -u user1 -p w
```

[UNIX]

```
mpsetacl -a -t mpjobsch/schedule/proj1 -u user1 -p w
```

When setting access rights during multi-subsystem operation:

The following example shows how to state this command in order to establish the update right as the access right of user "user1" being set in project "proj1" of subsystem number 2.

[Windows]

```
mpsetacl -a -t mpjobsch\schedule2\proj1 -u user1 -p w
```

[UNIX]

```
mpsetacl -a -t mpjobsch/schedule2/proj1 -u user1 -p w
```

When deleting access rights:

The following example shows how to state this command in order to delete the access rights of the user "user1" set in the project "proj1".

[Windows]

```
mpsetacl -d -t mpjobsch\schedule\proj1 -u user1
```

[UNIX]

```
mpsetacl -d -t mpjobsch/schedule/proj1 -u user1
```

When deleting a project:

If the user (or group) name is omitted, all access rights set for the project "proj1" are removed.

[Windows]

```
mpsetacl -d -t mpjobsch\schedule\proj1
```

[UNIX]

```
mpsetacl -d -t mpjobsch/schedule/proj1
```

2.2 mpchown Command

Synopsis

[Windows]

```
mpchown -t Target [-d Domain] -o Owner [-g|-u|-w
```

[UNIX]

```
mpchown -t Target -o Owner [-g|-u]
```

Description

This command sets access rights (update right) to the project owner.

Options

-t *Target*

Specify the name of the project in which the owner's access rights are set.

Specify a project in the following format:

[Windows]

```
mpjobsch\schedule\project name
```

EE

During multiple subsystem operation, specify the project in the following format.(Note)

```
mpjobsch\schedulen\project name
```

```
n: Subsystem number (1 to 9)
```

[UNIX]

```
mpjobsch/schedule/project name
```

EE

During multiple subsystem operation, specify the project in the following format.(Note)

```
mpjobsch\schedulen\project name
```

```
n: Subsystem number (1 to 9)
```

-d *Domain* [Windows]

Specify the domain name for the owner specified with "-o." The default for this option is the domain name of the user or group on the computer where this command is executed.

-o *Owner*

Specify the owner name.

The user's principal name cannot be specified.

-g

Used to specify the owner name specified with "-o" as a group.

-u

Used to specify the owner name specified with "-o" as a user.

-w [Windows]

Used to specify the owner name specified with "-o" as a well-known group. The well-known group name indicates the groups such as "Everyone" and "NETWORK" that are registered on the system beforehand and cannot be changed and deleted.

Command Location

Windows	Systemwalker Operation Manager installation directory\mpwalker\bin
Solaris HP-UX AIX Linux	Systemwalker Operation Manager installation directory/FJSVfwseo/bin

Privilege Required

System administrator (user belonging to the Administrators group or superuser) privileges are required.

Cautions

- If all of "-g (group)", "-u (user)", and "-w (well known group)[Windows]" are omitted, the access rights for the owner specified with "-o" are set if that owner is registered with the OS either as a user or a group.

    However, if "-d (domain)" is omitted and yet a user or group having the same name with the well-known group is specified, the access rights for the well-known group are set. [Windows]

If the owner is registered both as a group and user, user's access rights are set. [UNIX]

- When the schedule information is distributed by the policy extraction/distribution, the mpchown command is to be used to set the update right for the project owner. This command is also executed automatically and internally when the policy is applied.

To change the project owner, use the "4.5 jobschsetproject Command". For details, see jobschsetproject Command.

- In cluster environments, execute this command only on the active node (the node where the shared disk is mounted).

The settings configured with this command on the active node will be applied to the standby node at the time of next failover. Check which nodes are the active node or standby node in the cluster system currently used.

2.3 mkbat Command

Synopsis

[Windows]

```
mkbat -f file [-j] [-b]
```

[UNIX]

```
mkbat -f file [-j]
```

Description

This command creates command sequences of the access rights setup command (mpsetacl) and owner access rights setup command (mpchown) based on the access control information already set in the project. Use this command when servers share the same definitions. You need to edit the batch file or shell script that is output if the user name specified for the "-u" option or the group name specified for the "-g" option of the access right setup command (mpsetacl) and the user name or the group name specified for the "-o" option of the owner access right setup command (mpchown) differ on each server.

Options

-f *file*

Specify the name of the batch file/shell script to be created.

Note that the specified file will be overwritten if it is an existing file.

-j

If "-j" is specified, the definition information on the Systemwalker Operation Manager access rights is created with the command sequence.

If "-j" is omitted, the definition information on the Systemwalker Operation Manager and Systemwalker Centric Manager access rights is created with the command sequence.

If Systemwalker Centric Manager has not been installed, specify the "-j" option when executing this command.

-b [Windows]

Specified to back up the access right definition information which has been created by the "-j" option for the local computer.

Command Location

Windows	Systemwalker Operation Manager installation directory\mpwalker\bin
Solaris HP-UX AIX Linux	Systemwalker Operation Manager installation directory/FJSVfwseo/bin

Privilege Required

System administrator (user belonging to the Administrators group or superuser) privileges are required.

Cautions

Note the following points when creating the access right definition information for Systemwalker Centric Manager as well.

- In accordance with the change of Systemwalker Centric Manger security roles, the following actions are required to create and apply the access definition information with the mkbat command between Systemwalker Centric Manager V13.3.1 or earlier and V13.4.0 or later.

  - If there are users registered for the Systemwalker Centric Manager security role ("SecurityAdmin" or "SecurityAuditor") on the destination server (to create and apply the access definition information), delete all those users and then extract and apply the security information with the mkbat command.

  - After creating or applying the access definition information with the mkbat command, register users for "SecurityAdmin" or "SecurityAuditor" as necessary.

In cluster environments, execute batch files and shell scripts that are created with this command only on the active node. The settings configured with this command on the active node will be applied to the standby node at the time of next failover. Check which nodes are the active node or standby node in the cluster system currently used.

EE

2.4 mpaclcls Command

Synopsis

[Windows]

```
mpaclcls [-u [-s subsystem-number]] | [-s subsystem-number [-w]] directory-of-shared-disk
```

[UNIX]

```
mpaclcls [-u ] [-s subsystem number]
```

Description

This command sets the shared disk information to save the access rights information that has been set for the project in a cluster environment.

This command can also create a directory on the shared disk to save the access rights information in it. [Windows]

Options

-u

Specify to delete a cluster environment. If a subsystem number is specified, the cluster environment of the specified subsystem number is removed.

-s subsystem-number

Specify a subsystem number. Use this option for only 1:1 active node/standby node operations (with subsystems and partial cluster operations) or two-node mutual standby operations.

For 1:1 active node/standby node operations (with subsystems and partial cluster operations) and two-node mutual standby operations, an error will occur if this option is omitted. Conversely, an error will occur if this option is specified for configurations other than 1:1 active node/standby node operations (with subsystems and partial cluster operations) or two-node mutual standby operations. [UNIX]

-w [Windows]

Specify to set a subsystem number of the standby node system. When specifying this option, you must always specify the "-s" option.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Command Location

Windows	Systemwalker Operation Manager installation directory\mpwalker\bin
Solaris HP-UX Linux	Systemwalker Operation Manager installation directory/FJSVfwseo/bin

Privilege Required

System administrator (user belonging to the Administrators group or superuser) privileges are required.

Examples

[Windows]

The following example sets the shared disk "d:\share" in 1:1 active/standby node configurations (without subsystems) and 1:1 active/standby node configurations (with subsystems):

```
mpaclcls d:\share
```

The following example sets "d:\omgr1" of the shared disk on the node operating in subsystem 1 during two-node mutual standby operation:

```
mpaclcls -s 1 d:\omgr1
```

The following example cancels setup of the shared disk "d:\share" in 1:1 active/standby node configurations (without subsystems) and 1:1 active/standby node configurations (with subsystems):

```
mpaclcls -u d:\share
```

The following example cancels the setup of "d:\omgr1" shared disk on the node operating in subsystem 1 during two-node mutual standby operation:

```
mpaclcls -u -s 1 d:\omgr1
```

The following example sets the "e:\omgr2" shared disk on the node operating in subsystem 2 in the standby node during two-node mutual standby operation:

```
mpaclcls -s 1 -w e:\omgr2
```

[UNIX]

The following example sets the shared disk information in 1:1 active/standby node configurations (without subsystems) and 1:1 active/standby node configurations (with subsystems):

```
mpaclcls
```

The following example sets the shared disk information for the nodes where subsystem 1 is operating during 1:1 active node/standby node operations (with subsystems and partial cluster operations):

```
mpaclcls -s 1
```

The following example removes the shared disk information in 1:1 active/standby node configurations (without subsystems) and 1:1 active/standby node configurations (with subsystems):

```
mpaclcls -u
```

The following example removes the shared disk information for the nodes where subsystem 1 is operating during 1:1 active node/standby node operations (with subsystems and partial cluster operations):

```
mpaclcls -u -s 1
```

Execution Results/Output Format

An error message is output to the standard error output if this command terminates abnormally.

EE

2.5 mpcssave Command

Synopsis

```
mpcssave [-s subsystem-number]
```

Description

This command saves access rights information and Operation Manager user information set for projects to the shared disk of the cluster in a cluster environment.

Execute this command only on the active node.

Option

-s subsystem-number

Specify a subsystem number between 0 and 9. The access rights information of the specified subsystem number's projects is saved to the shared disk of the cluster in a cluster environment.

This option is required for 1:1 active/standby node configurations (with subsystems and partial cluster operations) and for dual node mutual standby configurations).

If this option is omitted, the command is applied to all subsystems.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Command Location

| Windows | Systemwalker Operation Manager installation directory\mpwalker\bin |
| Solaris
HP-UX
Linux | Systemwalker Operation Manager installation directory/FJSVfwseo/bin |

Privilege Required

System administrator (user belonging to the Administrators group or superuser) privileges are required.

Cautions

[Windows]

- Execute this command when the following message is output to the event log:

  "Access control information was saved incompletely. For details, see *filename*."

[UNIX]

- Execute this command when the following message is output to the syslog:

"The system could not backup the access control information for some functions. For details, see *filename*."

Examples

The following example saves the access rights information of projects to the shared disk:

```
mpcssave
```

The following example saves the access rights information of projects to the shared disk on the server operating in the subsystem 2:

```
mpcssave -s 2
```

2.6 mplstacluser Command

Synopsis

```
mplstacluser [-s subsystem-number]
```

Description

This command displays the list of the accounts of projects to which access rights have been set.

Option

-s subsystem-number

Of the projects registered in the specified subsystem, displays the list of users belonging to the projects to which access rights have been set.

If this option is omitted, the project with subsystem number 0 will be the target.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Command Location

Windows	Systemwalker Operation Manager installation directory\mpwalker\bin
Solaris HP-UX AIX Linux	Systemwalker Operation Manager installation directory/FJSVfwseo/bin

Privilege Required

System administrator (user belonging to the Administrators group or superuser) privileges are required.

Cautions

Well-known group names are not displayed even if access rights have been set to their projects. The well-known group name indicates the groups such as "Everyone" and "NETWORK" that have been registered on a system beforehand. [Windows]

If an Operation Manager user is in use, the OS user allocated to the Operation Manager user will also be displayed. [UNIX]

Examples

The following example displays the list of users belonging to projects to which access rights have been set:

```
mplstacluser
```

The following example displays the list of users belonging to projects to which access rights have been set and which have been registered in the subsystem 1:

```
mplstacluser -s 1
```

Execution Results/Output Format

[Windows]

```
[USER]      : Computer name\Administrator
[USER]      : Computer name\swuser1
[USER]      : Computer name\swuser2
[USER]      : Domain name\domuser1
[USER]      : Domain name\domuser2
[GROUP]     : Computer name\swgrp1
[GROUP]     : Computer name\swgrp2
[SW-USER]   : repouser1
[SW-GROUP]  : repogrp1
```

[UNIX]

```
[USER]      : root
[USER]      : swuser1
[USER]      : swuser2
[EX-USER]   : omgruser1|
[EX-USER]   : omgruser2
[GROUP]     : swgrp1
[GROUP]     : swgrp2
```

Remark:

USER:OS user
GROUP:OS group
EX-USER:Operation Manager user (only when the extended user management function is enabled)

SW-USER:Systemwalker authentication repository user (only when the Systemwalker authentication repository is enabled)
SW-GROUP:Systemwalker authentication repository group (only when the Systemwalker authentication repository is enabled)

2.7 mpsetusermode Command [UNIX]

Synopsis

```
mpsetusermode -s {on|off}
```

Description

This command enables and disables the extended user management function.

Options

-s on

Specified to enable the extended user management function.

-s off

Specified to disable the extended user management function. This option is set to off just after installation.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Reference

mpusermode, mpadduser, mpdeluser, mpmoduser, mpsetpasswd, mpusers

Command Location

Solaris HP-UX AIX Linux	Systemwalker installation directory/FJSVfwseo/bin

Privilege Required

System administrator (superuser) privileges are required.

Cautions

- You cannot change the enable/disable state of the extended user management function during running of the Systemwalker Operation Manager daemons. This command must be executed after the Systemwalker Operation Manager daemons are stopped.

- When the extended user management function is enabled after an OS user created a project, only the administrator can have the update right for the created project. Set access rights to the project as required. Owner information is inherited as it is.

- When the extended user management function is disabled after an Operation Manager user created a project, only the system administrator (superuser) and project owner can have the update right for the created project. Set access rights to the project as required. Owner information is inherited as it is.

Examples

The following example enables the extended user management function:

```
mpsetusermode -s on
```

The following example disables the extended user management function:

```
mpsetusermode -s off
```

Execution Results/Output Format

Normal termination	No message is output.
Abnormal termination	An error message is output.

2.8 mpusermode Command [UNIX]

```
mpusermode
```

Description

This command displays whether the extended user management function is enabled or disabled.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Reference

mpsetusermode

Command Location

Solaris HP-UX AIX Linux	Systemwalker installation directory/FJSVfwseo/bin

Privilege Required

System administrator (superuser) privileges are required.

Example

The following example displays whether the extended user management function is enabled or disabled:

```
mpusermode
```

Execution Results/Output Format

When the extended user management function is enabled:

```
Operation Manager user mode is ON.
```

When the extended user management function is disabled:

```
Operation Manager user mode is OFF.
```

2.9 mpadduser Command [UNIX]

Synopsis

```
mpadduser -o OS user name [-d description] [-a] username
```

Description

This command registers an Operation Manager user to Systemwalker Operation Manager.

Options

-o *OS user name*

Specify an OS user to be assigned to the Operation Manager user. When registering an administrator, specify a system administrator (superuser) name. When registering a non-administrator, specify a general user name. When the -a option is specified, you cannot register administrators by specifying any general user names in this option. When the -a option is not specified (registering a non-administrator), you cannot register any non-administrators by specifying any system administrator (superuser) names in this option.

-d *description*

Specified to describe the Operation Manager user to be registered. This option can be omitted.

-a

Specified to register an Operation Manager user as an administrator. The administrator has the update right for all the projects. The OS user assigned to an administrator is a system administrator (superuser).

If this operation is not specified, a general user is registered.

username

Specify the Operation Manager user name to be registered within 20 bytes.

The Operation Manager user named "root" cannot be specified because it is created at installation.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Reference

mpdeluser, mpmoduser, mpsetpasswd, mpusers

Command Location

Solaris HP-UX AIX Linux	Systemwalker installation directory/FJSVfwseo/bin

Privilege Required

System administrator (superuser) privileges are required.

Cautions

- When the extended user management function is disabled, the registered Operation Manager user is not available.

- The following characters are available for the Operation Manager user name:

Category	Character
Alphabetic character	A - Z a - z
Numeric character	0 - 9
Symbol	-

- In cluster environments, execute this command only on the active node (the node where the shared disk is mounted).

The settings configured with this command on the active node will be applied to the standby node at the time of next failover. Check which nodes are the active node or standby node in the cluster system currently used.

Examples

The following example registers the Operation Manager user named "swroot" as an administrator:

```
mpadduser -o root -d "Operation Manager Administrator" -a swroot
```

The following example registers the Operation Manager user named "swuser": On the OS, the user right is set to user:

```
mpadduser -o user swuser
```

Execution Results/Output Format

| Normal termination | No message is output. |
| Abnormal termination | An error message is output. |

2.10 mpdeluser Command [UNIX]

Synopsis

```
mpdeluser username [username...] | -all
```

Description

This command deletes the Operation Manager user registered to Systemwalker Operation Manager.

Options

username

Specify the Operation Manager user name to be deleted. Multiple user names can be specified.

-all

Specify to delete all the Operation Manager users registered to Systemwalker Operation Manager.

This option cannot be specified during running of the Systemwalker Operation Manager daemons. When specifying this option, execute this command after stopping the Systemwalker Operation Manager daemons.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Reference

mpusermode, mpadduser, mpmoduser, mpsetpasswd, mpusers

Command Location

| Solaris
HP-UX
AIX
Linux | Systemwalker installation directory/FJSVfwseo/bin |

Privilege Required

System administrator (superuser) privileges are required.

Cautions

- When the -all option is specified, the Systemwalker Operation Manager daemons must be stopped.

- Do not delete the user, who is using Systemwalker Operation Manager during running of Systemwalker Operation Manager. The deleted user is handled without access rights for all the projects.

Examples

The following example deletes the Operation Manager user "swroot" registered to Systemwalker Operation Manager:

```
mpdeluser swroot
```

The following example deletes the Operation Manager users "swroot" and "swuser" registered to Systemwalker Operation Manager:

```
mpdeluser swroot swuser
```

The following example deletes all the Operation Manager users registered to Systemwalker Operation Manager:

```
mpdeluser -all
```

Execution Results/Output Format

Normal termination	No message is output.
Abnormal termination	An error message is output.

2.11 mpmoduser Command [UNIX]

Synopsis

```
mpmoduser {-o OS user name | -d description} username
```

Description

This command sets the OS user name to be assigned to the Operation Manager user which is registered with Systemwalker Operation Manager.

Options

-o OS user name

Specify this option when changing the OS user name that has been assigned to the Operation Manager user by using the mpadduser command.

If the OS user name that has been associated with the Operation Manager user is the user name of the system administrator, the OS user name to which you change must be a user name of the system administrator. And if the OS user name that has been associated with the Operation Manager user is the user name of other than the system administrator, the OS user name to which you change must be a user name of other than the system administrator.

-d description

Specified to change the description about the Operation Manager user.

username

Specify the name of the Operation Manager user whose attribute is to be changed.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Reference

mpadduser, mpsetpasswd, mpusers

Command Location

Solaris HP-UX AIX Linux	Systemwalker installation directory/FJSVfwseo/bin

Privilege Required

System administrator (superuser) privileges are required.

Cautions

- In cluster environments, execute this command only on the active node (the node where the shared disk is mounted).

  The settings configured with this command on the active node will be applied to the standby node at the time of next failover. Check which nodes are the active node or standby node in the cluster system currently used.

Examples

The following example changes the OS user assigned to the Operation Manager user "swuser" to "user2":

```
mpmoduser -o user2 swuser
```

The following example changes the description about the Operation Manager user "swuser" to "Operation Manager User":

```
mpmoduser -d "Operation Manager User" swuser
```

Execution Results/Output Format

Normal termination	No message is output.
Abnormal termination	An error message is output.

2.12 mpsetpasswd Command [UNIX]

Synopsis

```
mpsetpasswd [-d] username
```

Description

This command sets, modifies, or deletes the password of the Operation Manager user registered to Systemwalker Operation Manager. The Operation Manager user with the password deleted cannot log in. This command also deletes the password of the Operation Manager user named "root."

Options

-d

Specify to delete the password. The user with the password deleted cannot log in.

username

Specify the name of the Operation Manager user whose password is to be set, modified, or deleted.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Reference

mpadduser, mpmoduser, mpusers

Command Location

Solaris HP-UX AIX Linux	Systemwalker installation directory/FJSVfwseo/bin

Privilege Required

System administrator (superuser) privileges are required.

Cautions

- The following characters are available for the password:

Category	Character
Alphabetic character	A - Z
	a - z
Numeric character	0 - 9
Symbol	! " # % & ' ( ) * + , - . / : ; < = > ? [\ ] ^ _ {	} ~
Blank	' '

- Password can be between 5 and 50 alphanumeric characters. However, only up to 8 characters are valid as the password.

- In cluster environments, execute this command only on the active node (the node where the shared disk is mounted).

  The settings configured with this command on the active node will be applied to the standby node at the time of next failover. Check which nodes are the active node or standby node in the cluster system currently used.

Examples

The following example sets and modifies the password of the Operation Manager user named "swuser":

```
mpsetpasswd swuser
Password:  <-- Type a password to be set and modified
                      (which will not be echoed back)
Retype: <-- Retype the password to be set and modified
                      (which will not be echoed back)
```

The following example deletes the password of the Operation Manager user named "root":

```
mpsetpasswd -d root
```

Execution Results/Output Format

Normal termination	No message is output.
Abnormal termination	An error message is output.

2.13 mpusers Command [UNIX]

Synopsis

```
mpusers [-a]
```

Description

This command displays the list of the Operation Manager users registered to Systemwalker Operation Manager.

Option

-a:

Displays the following attributes of the registered Operation Manager user in addition to the name:

- Description

- Assigned OS user name

- Whether the user is an administrator or not

If this option is omitted, only the user name is displayed.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Reference

mpadduser, mpdeluser, mpmoduser

Command Location

Solaris HP-UX AIX Linux	Systemwalker installation directory/FJSVfwseo/bin

Privilege Required

System administrator (superuser) privileges are required.

Examples

The following example displays the list of the Operation Manager users registered to Systemwalker Operation Manager:

```
mpusers
```

The following example displays the list of the Operation Manager users registered to Systemwalker Operation Manager. The user attribute is also displayed.

```
mpusers -a
```

Execution Results/Output Format

If -a option is not specified:

```
# mpusers
swuser1
swuser2
swuser3
#
```

If -a option is specified:

```
# mpusers -a
NAME      OS USERNAME    ADMIN   DESCRIPTION
swroot1   root           Yes     Operation Manager Administrator
swroot2   root           Yes     Operation Manager Administrator
swuser1   user           No      Operation Manager User
```

Output Items

Output items are described below.

If the -a option is omitted, only the registered Operation Manager user names are displayed.

NAME

Displays the registered Operation Manager user name.

OS USERNAME

Displays the OS user name assigned to the Operation Manager user.

ADMIN

Displays "Yes" for administrator and "No" for non-administrator.

DESCRIPTION

Displays the registered description.

2.14 mpsetupolmode Command [UNIX]

Synopsis

```
mpsetupolmode -u {clear|keep}
```

Description

This command specifies whether to delete or overwrite the user information registered in the destination server when distributing the Operation Manager user information as a policy. This command is used in the policy extraction source server when policies are extracted.

Options

-u clear

Specify when deleting the Operation Manager user information registered to the policy distribution destination server, and then distribution a policy.

-u keep

Specify when performing policy distribution without deleting the Operation Manager user information registered to the policy distribution destination server. When the same Operation Manager user name exists, user information is overwritten to the existing one.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Reference

mpusermode, mpadduser, mpdeluser, mpmoduser, mpusers

Command Location

Solaris HP-UX AIX Linux	Systemwalker installation directory/FJSVfwseo/bin

Privilege Required

System administrator (superuser) privileges are required.

Cautions

- This setting must be performed in the policy source server before policies are extracted. Even if this command is executed, the setting is not available when:

    - the setting was performed in the policy source server after policies were extracted, and

    - the setting was performed in the policy destination server.

- "keep" (not to delete) is set just after installation.

Examples

The following example deletes the Operation Manager user information in the destination before extracting policies of Operation Manager user information. If policies are extracted or distributed after this setting, the Operation Manager user information in the destination will be deleted.

```
mpsetupolmode -u clear
```

The following example retains the Operation Manager user information in the destination before extracting policies of Operation Manager user information. If policies are extracted or distributed after this setting, the Operation Manager user information in the destination will not be deleted. When the same Operation Manager user exists, user information is overwritten to the existing one.

```
mpsetupolmode -u keep
```

Execution Results/Output Format

Normal termination	No message is output.
Abnormal termination	An error message is output.

2.15 mpupolmode Command [UNIX]

Synopsis

```
mpupolmode
```

Description

This command displays whether or not to delete the Operation Manager user information registered in the destination server when distributing Operation Manager user information as a policy. This command is used in the policy extraction source server.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Reference

mpsetupolmode

Command Location

Solaris HP-UX AIX Linux	Systemwalker installation directory/FJSVfwseo/bin

Privilege Required

System administrator (superuser) privileges are required.

Example

The following example displays whether or not to delete the Operation Manager user information from the destination on the policy extraction source server when distributing policies:

```
mpupolmode
```

Execution Results/Output Format

When user information is not deleted in the policy destination:

```
Operation Manager user policy distribution mode is KEEP.
```

When user information is deleted in the policy destination:

```
Operation Manager user policy distribution mode is CLEAR.
```

2.16 mpprjcmdacl Command [UNIX]

Synopsis

```
mpprjcmdacl [-s subsystem number] project name
```

Description

This command displays the access rights information of the OS user for a project.

Options

EE

-s subsystem number

Specify a subsystem to be displayed in a system where multiple subsystems run. Specify the subsystem number to be displayed between 0 and 9. If this option is omitted, subsystem 0 is assumed.

project name

Specify the project of which the list of access rights of OS user is displayed.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Reference

mpsetusermode, mpusermode

Command Location

Solaris HP-UX AIX Linux	Systemwalker installation directory/FJSVfwseo/bin

Privilege Required

System administrator (superuser) privileges are required.

Example

The following example lists the access rights of the OS user for project "project":

```
mpprjcmdacl project
```

Execution Results/Output Format

```
# mpprjcmdacl project
OS USERNAME    ACCESS RIGHTS
root           Update    Right
user           Change    Right
user2          Operation Right
user3          Reference Right
```

Output Items

Output items are described below.

OS USERNAME

Displays the OS user name.

ACCESS RIGHTS

Displays the access rights:

Update Right
Change Rights
Operation Right
Reference Right

2.17 mpmkpwbat Command [UNIX]

Synopsis

```
mpmkpwbat [-c] [-u username] -f filename
```

Description

This command extracts the password information of the Operation Manager user registered to Systemwalker Operation Manager to a file.

Options

-c

Specify to include password information of a user with password omitted or deleted in the extraction target information. When password information is extracted and executed as a script with this option, the password information of the same Operation Manager user name registered in the destination will be deleted. If this option is omitted, password information of a user with password omitted is not extracted.

-u *username*

Specify to extract only the password information of the specified user. If this option is omitted, password information of all users is extracted.

-f *filename*

Specify a file name in the extraction destination. Specifying an existing file name causes information to be overwritten to the existing file.

Executing this file as a script, you can change password information of the Operation Manager user on the server, in which this command is to be executed, to password information of the Operation Manager user registered in the source server. Only the system administrator (superuser) can execute this file as a script.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Reference

mpsetusermode, mpusermode

Command Location

Solaris HP-UX AIX Linux	Systemwalker installation directory/FJSVfwseo/bin

Privilege Required

System administrator (superuser) privileges are required.

- Password information is encrypted when it is extracted.

- The file extracted with this command contains password information of user. You should take great care to handle this file.

Examples

The following example extracts the password information of the Operation Manager user registered to Systemwalker Operation Manager to file "users.sh":

```
mpmkpwbat -f users.sh
```

The following example extracts the password information of the Operation Manager user "swuser1" registered to Systemwalker Operation Manager to file "swuser1.sh":

```
mpmkpwbat -u swuser1 -f swuser1.sh
```

Execution Results/Output Format

Normal termination	No message is output.
Abnormal termination	An error message is output.

2.18 mpsetlogsend_omgr Command

Synopsis

```
mpsetlogsend_omgr
```

```
mpsetlogsend_omgr {-n|-y}
```

```
mpsetlogsend_omgr [{-n|-y}] -f <output file> -k {<number of storage
days>|unlimit}
```

```
mpsetlogsend_omgr -d
```

Description

This command configures the settings for outputting the audit logs that are output by Systemwalker Operation Manager.

Options

No option specified

Outputs the current settings for audit log output. The following information is output:

```
LOG NAME          : Systemwalker Operation Manager audit log
LOG OUTPUT        : ON
LOG KEEP DAYS     : 31 day(s)
OUTPUT FILE       : /var/opt/FJSVftlo/audit/log/mp_omgr_auditYYMMDD.log
```

The output content is as follows:

LOG NAME	This item is the name of the target audit log.
LOG OUTPUT	"ON" is displayed if settings have been made so that audit logs are output, and "OFF" is displayed if such settings have not been made.
LOG KEEP DAYS	This item displays the number of days for which audit logs that have been output in the past will be stored. Information will be displayed as "unlimited", "1 day(s)", "2 day(s)", ... "99 day(s)"

OUTPUT FILE	This item displays the output file for audit logs. In the content displayed, the "YYMMDD" part is actually replaced by the date (where "YY" is the last two digits of the year, "MM" is the month and "DD" is the day of the month).
LOG COLLECTION (*1)	"ON" is displayed if settings have been made so that audit logs are collected by the Systemwalker Centric Manager audit log management function, and "OFF" is displayed if such settings have not been made.
COLLECTION FILE (*1)	This item outputs the files to be collected by the Systemwalker Centric Manager audit log management function. If log collection settings have never been made, "-" will be output.

*1: The audit log management function can be used only if Systemwalker Operation Manager coexists with Operation Management Servers, Asset Management Servers, Section Management Servers, Job Servers and Operation Management Clients running Systemwalker Centric Manager V13.2.0 or later.

-f <output file>

Specify the output file for audit logs using the full path specification.

Up to 246 bytes' worth of characters can be specified for the output destination.

Log files will be output with the name specified by the "-f" option plus "YYMMDD.log", where YYMMDD indicates the date.

Example:

If "/var/audit/audit_log" has been specified for the "-f" option, the log file that is actually output on May 19, 2006 will be "/var/audit/audit_log060519.log".

-k <number of storage days>

Specify the maximum number of days for which audit logs will be held. After "-k", specify either a number between 1 and 99 (units are "days") or "unlimit". The command will terminate with a parameter error if a value of 100 or more is specified. If "unlimit" is specified, the audit log files that have been output will not be deleted.

-d

Specify this option to stop outputting audit logs. If this option is specified, Systemwalker Operation Manager audit logs will no longer be output.

At the same time, collection using the Systemwalker Centric Manager audit log management function will also stop. To only stop collection using the audit log management function, use the "-n" option. (*1)

-y (*1)

Specify this option to collect audit log files onto an Operation Management Server using the Systemwalker Centric Manager audit log management function. If this option is specified, audit logs that have been output will be automatically collected on the Operation Management Server.

Collection settings for the files collected by the audit log management function are also updated by this option.

-n (*1)

Specify this option to pause the collection of audit log files on the Operation Management Server using the Systemwalker Centric Manager audit log management function. If this option is specified, audit logs that have been output will stop being collected on the Operation Management Server. To start collecting audit logs again, execute this command again, specifying the "-y" option.

*1:

These options can be used only if Systemwalker Operation Manager coexists with Operation Management Servers, Asset Management Servers, Section Management Servers, Job Servers and Operation Management Clients running Systemwalker Centric Manager V13.2.0 or later.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Command Location

Windows	*Systemwalker Operation Manager installation directory*\mpwalker.jm\bin
Solaris HP-UX AIX Linux	/usr/bin

Privilege Required

This command can be executed by only system administrators (superusers or users belonging to the Administrators group).

Cautions

- If log collection settings for the Systemwalker Centric Manager audit log management function have not been updated since the audit log output settings were changed (that is, if this command has been executed without the "-y" option specified), the content of the audit log output destination (OUTPUT FILE) and the content that is actually collected on the Operation Management Server may be different. To resume log collection correctly, execute this command again specifying the "-y" option.

- The access rights to the output destination directory for audit logs must be set up again when the output destination directory is changed. Refer to "Define user restrictions" in the *Systemwalker Operation Manager Installation Guide* for details.

- When both Systemwalker Operation Manager and Centric Manager are installed on Linux 64, as shown in the table below, the audit logs of the ACL Manager will be output in two separate files.

		Centric Manager	
		V13.4.0 or later (Linux x86)	**V13.4.0 or later (Linux x64)**
Operation Manager	**V13.2.0 - V13.3.1**	No issues.	The audit logs of the ACL Manager will be output in two separate files regardless of the installation order of Centric Manager and Operation Manager.
	V13.8.0 or later (Linux x86)	No issues.	If Centric Manager has been installed after Operation Manager, the audit logs of the ACL Manager will be output in two separate files.
	V13.8.0 or later (Linux x64)	If Centric Manager has been installed after Operation Manager, the audit logs of the ACL Manager will be output in two separate files.	No issues.

The file name of the output destination is as follows:

- The output destination displayed by mpsetlogsend_omgr (audit log setup command)

- The file that "_acl" was added to before the date of the above file
  (Example: /var/opt/FJSVftlo/audit/log/mp_omgr_audit_aclYYMMDD.log)

The files that "_acl" is added to are not displayed as the execution result of mpsetlogsend_omgr (audit log setup command), however the two files have the same audit log output settings (ON or OFF, storage period and if they are the collection target).

2.19 mpsetssoinfoo Command

Synopsis

mpsetssoinfoo {SET SVSSO -h <*host name*> [-p <*port number*>] | DELETE SVSSO
| REFER SVSSO}

Description

This command configures the settings required to link with the ServerView Operations Manager Single Sign-On environment.

Options

SET SVSSO

Links with the ServerView Operations Manager Single Sign-On environment. The following parameters can be specified:

-h <*host name*>

Specify the ServerView Operations Manager FQDN.

Specify the FQDN of the admin server defined in the hosts file when installing the ServerView Resource Orchestrator admin server.

This cannot be omitted.

-p <*port number*>

Specify the port number of the URL used to authenticate ServerView Operations Manager.

If omitted, "3170" is used by default.

DELETE SVSSO

Removes the link with the ServerView Operations Manager Single Sign-On environment.

REFER SVSSO

Displays the settings for the link with the ServerView Operations Manager Single Sign-On environment.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Command Location

| Windows | <*Systemwalker installation directory*>\mpwalker\mpaclmgr\bin |

Solaris Linux	/opt/FJSVfwseo/bin

Privilege Required/Execution Environment

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- This command can be executed on a server.

- After the command is executed, restart the Systemwalker Operation Manager service/daemon by using the poperationmgr and soperationmgr commands for the settings to take effect.

- This command cannot be executed on Solaris versions.

Examples

Configure the linkage settings for ServerView Operations Manager by setting the host name as "svom.fujitsu.com" and the port number as "13170".

```
mpsetssoinfoo SET SVSSO -h "svom.fujitsu.com" -p 13170
```

Remove the ServerView Operations Manager link.

```
mpsetssoinfoo DELETE SVSSO
```

Display the ServerView Operations Manager link settings.

```
mpsetssoinfoo REFER SVSSO
```

Execution Results/Output Format

When the command terminates normally (return value is 0):

The result is output to the standard output in the format shown below.

- Example of the SET SVSSO option

```
MpSSO: INFO: 10000: ServerView Operations Manager Single Sign-On linkage setup has
started.
MpSSO: INFO: 10001: ServerView Operations Manager Single Sign-On linkage setup
completed successfully.
```

- Example of the REFER SVSSO option

```
MpSSO: INFO: 10200: The ServerView Operations Manager Single Sign-On linkage settings
display has started.
Setting status: Enabled
Host name: svom.fujitsu.com
Port number: 3170
MpSSO: INFO: 10201: The ServerView Operations Manager Single Sign-On linkage settings
display completed successfully.
```

When the command terminates abnormally (return value is other than 0):

The result is output to the standard output in the format shown below.

- Example of the SET SVSSO option

```
MpSSO: INFO: 10000: ServerView Operations Manager Single Sign-On linkage setup has
started.
```

```
<Error message>
MpSSO: ERROR: 50000: Failed to complete the ServerView Operations Manager Single Sign-
On linkage setup
```

Chapter 3 Calendar/Power Control Commands

This chapter describes the Calendar commands and Power Control commands provided by Systemwalker Operation Manager. Accordingly, you must run the commands on servers in which Systemwalker Operation Manager server functions are installed.

🔓 Note

・・・

If commands are executed in an environment where the UAC function for Windows Server 2008 or later has been enabled, then a dialog box asking a user's privileges to be promoted may be displayed depending on the privileges of the user that logged in to the operating system, and action may be required.

Refer to "Notes on Executing Commands in Windows [Windows version]" in the *Systemwalker Operation Manager User's Guide* for more information.

・・・

3.1 f3crheet Queuing Completion Notification Command

Synopsis

```
f3crheet queuing completion notification
```

Description

Use this command to issue a queuing completion notification when you want to postpone the shutdown/power off until a specific processing or an application terminates. To do this, you need to define the character string (with the option of this command) in the [Queuing Completion Notification Definition] dialog box beforehand. To use this command in the target host with batch power control system, the definition must be made in the [Queuing Completion Notification Definition] dialog box of the control host. In addition, this command is supported by Windows, Solaris, and Linux versions only.

Option

queuing completion notification

Specify the character string indicating termination of specific processing or an application.

This option can be specified with up to 100 characters. Blank space and comma cannot be included in it.

Do not allow this to be mixed up with character string "queuing completion notification" specified in other processing, applications, and other nodes.

Usage (Performing shutdown after termination of job net)

When executing this command as the job to be activated or terminated at the end of operation, a queuing completion notification is issued when the job net terminates. By defining this queuing completion notification in the [Queuing Completion Notification Definition] dialog box, shutdown processing is postponed until the queuing completion notification specified with this command is issued. Then, power off is performed on the system by the power control function after the shutdown processing has been completed.

This command is set on the shutdown exit of the shutdown job net of the standard Jobscheduler. [Windows]

This command can be useful when you use it also at points other than the shutdown exit of the shutdown job net in Jobscheduler since you can have the system wait events other than job nets simultaneously, and then shut down the system. [Windows]

Return Values

Return value	Meaning
0	Normal termination

Return value	Meaning
10	An error has occurred. Major causes might be the following: - There is an error in parameter setting. - Queuing has already terminated.
20	A communication error has occurred between nodes.

Command Location

Windows	Systemwalker Operation Manager installation directory\mpwalker\bin
Solaris Linux	/opt/FJSVjmcal/bin

EE

3.2 calregpath Register Shared Disk Directory of Calendar Auto Reflection Command [Windows]

Synopsis

```
calregpath -add {-sys subsystem-number | -all} [-pm] -p shared disk directory
calregpath -del {-sys subsystem-number | -all}
calregpath -ref
```

Description

Use this command to register the directory path information located on the shared disk to the calendar service and to delete the registered information from the calendar service at cluster operation. Information to be registered varies depending on the cluster operation style.

This command needs to be executed on each node that has cluster configuration.

Options

-add

Used to register the directory located on the shard disk.

If this option is specified, either the -sys option or the -all option and the -p option need to be specified.

The -all option and the -sys option cannot be specified together.

-sys subsystem-number

If the cluster system type is either 1:1 active node/standby node operation (with subsystems and partial cluster operations) or one of the following mutual standby operations, specify the subsystem number that is being treated as a cluster by the Jobscheduler. The path information of the subsystem number is edited.

- Two-node mutual standby

- Failovering

- Random

This option needs to be executed corresponding to the number of subsystems.

If specifying it at specifying the -del option, the specified subsystem number returns to the initial status.

-all

Specify this option when the cluster style is any of the following standby operations:

- 1:1 Active/Standby (without subsystems) and 1:1 Active/Standby (with subsystems)

- N:1 Active/Standby

- Cascade

If specifying it at specifying the -del option, the path information of all the subsystems registered returns to the initial status.

-pm

Use this option when specifying the -add option.

When specifying this option, data stored in the directory before change is moved to the directory specified with the -p option.

Specify this option when migrating data to the shared disk etc. at cluster environment setup. If omitted, only the information of the destination to which the directory refers to is updated.

When specifying this option, you must confirm that both the reference destination directory currently set and the one specified with the -p option can be referred to.

-p shared disk directory name

Specify the directory on the shared disk created at cluster installation. Specify the path name of the directory within 255 bytes. This option must be specified when the -add option is specified.

-del

Specify to return the path information currently registered to the initial status.

Specify this option in combination with the -sys option or the -all option.

-ref

Specify to display the path information of directories currently registered to standard output in list format.

In the initial status, the following initial information is displayed by each subsystem:

- Systemwalker installation directory\MPWALKER.JM\mpjmcal\post\sys(*n*)

(*n*): Subsystem number

Return Values

Return value	Meaning
0	Normal termination
1	Abnormal termination

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin

Cautions

- System administrator (user belonging to the Administrators group) privileges are required.

- When registering or deleting directories, execute this command after stopping the calendar service.

EE

3.3 calsetcluster Calendar Auto Reflection Setup Command

Synopsis

```
calsetcluster [-p] [-a] {-type cluster system style}
calsetcluster -d
calsetcluster -v
```

Description

Use this command to validate the function that automatically reflects calendar information at cluster installation.

Execute this command after setting the shared directory with the calregpath command to be described. [Windows]

The following kinds of information are automatically reflected:

- Day change time

- Holiday information of the calendar

- Information on the SYSTEM_CALENDAR schedule, power on/off schedule, completion monitoring (selectable at setting)

- Information on service, application and application startup (selectable at setup)

Options

-p

When specifying this option, information on the schedule, power on/off schedule, completion monitoring of SYSTEM_CALENDAR are removed from targets to be reflected. If omitted, they are included in auto-reflection targets.

To reflect such information, the setting that allows update of the information (i.e. the -p option should not be specified) must be defined on both the reflection source server where update is executed and reflection destination server.

Note

This option must be specified on all the reflection target servers when the power control is in either of the following environments:

- Batch power control is executed

- Power control settings do not match among all the information reflection destination servers.

-a

If this option is specified, information on service, application and application startup will be removed from reflection targets. If omitted, it will be reflected.

To reflect this information, the setting that allows update of the information (i.e. the -a option is not specified) must be defined on both the reflection source server where update is executed and the reflection destination server.

-type cluster system style

If this option is specified, the auto reflection function is activated. Specify it as shown below depending on the cluster style to be in operation.

[-type s]

1:1 Active/Standby

[-type n]

N:1 Active/Standby

[-type e]

Two-node mutual standby

[-type r]

Failovering

[-type d]

Random

[-type c]

Cascade

-d

If this option is specified, the auto reflection function will be deactivated.

📋 **Note**

....

If this option is specified, the auto reflection function will be deactivated.

When uninstalling Systemwalker Operation Manager from a cluster system, deactivate the auto reflection function first. If not, the following symbolic link will remain and you have to delete it manually. [UNIX]

Solaris: /etc/rc3.d/S28JMCAL

HP: /sbin/rc3.d/S28JMCAL

Linux: /etc/rc3.d/S28JMCAL

AIX: Symbolic links cannot be created.

....

-v

Specify to display the current settings.

Return Values

Return value	Meaning
0	Normal termination
1	Abnormal termination

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris Linux	/opt/FJSVjmcal/bin
HP-UX	/opt/FHPjmcal/bin
AIX	/opt/FAIXjmcal/bin

Cautions

- System administrator (user belonging to the Administrators group or superuser) privileges are required.

- Execute this command after stopping the calendar service/daemon.

EE

3.4 calrecpl Reread Definition of Calendar Reflection Destination Host Command

Synopsis

```
calrecpl.exe
```

Description

Use this command for cluster operation.

When changing definitions on the reflection destination host, the changes can be reflected without stopping the calendar service/daemon by executing this command.

Return Values

Return value	Meaning
0	Normal termination
1	Abnormal termination

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris Linux	/opt/FJSVjmcal/bin
HP-UX	/opt/FHPjmcal/bin
AIX	/opt/FAIXjmcal/bin

Cautions

- System administrator (user belonging to the Administrators group or superuser) privileges are required.

- When executing this command, the calendar service/daemon must be running.

Chapter 4 Jobscheduler Commands

This chapter describes the Jobscheduler commands provided by Systemwalker Operation Manager. Note that the commands must be executed on the server where the Systemwalker Operation Manager server function is installed. For the Jobscheduler commands that are used to output information, see Chapter 5 Jobscheduler Commands (For Printing).

Note
....

If commands are executed in an environment where the UAC function for Windows Server 2008 or later has been enabled, then a dialog box asking a user's privileges to be promoted may be displayed depending on the privileges of the user that logged in to the operating system, and action may be required.

Refer to "Notes on Executing Commands in Windows [Windows version]" in the *Systemwalker Operation Manager User's Guide* for more information.
....

Note
....

Notes when you execute Jobscheduler commands

When executing a command, make sure the following conditions are satisfied:

- The Jobscheduler service/daemon must be running (the jobschconvert command is not this case).

- The standard output and standard error output should not be closed.

- The system resource availability (maximum file size limit) for respective user should not be limited by the user who executes the command. (Otherwise, the command may not be executed correctly. Additionally, the maximum file size limit may cause the schedule information file to be damaged.) [UNIX]

The commands can be executed by a user who is already logged on to the Windows server/UNIX server.
....

4.1 List of Commands

The following are the commands used in the Jobscheduler function.

Commands executed with system administrator privileges

The following commands and the related control statements provided with Jobscheduler and executed with the system administrator privileges are discussed in the sections from 4.1 to 4.7 in this chapter.

jobschsethol command

Registers, changes and deletes holidays in the calendar.

Holiday information control statement

Specifies holiday information.

jobschsetcal command

Registers, changes and deletes a schedule pattern.

Schedule pattern control statement

Specifies a schedule pattern.

jobschcod command

Changes the day change time and reschedules groups and job nets.

jobschsetproject command

Registers, changes and deletes project information.

Project information control statement

Sets up project information.

EE **jobsch_start command [UNIX]**

Starts the Jobscheduler daemons.

EE **jobsch_end command [UNIX]**

Stops the Jobscheduler daemons.

Commands executed with the general users privileges

The following commands and the related control statements provided with Jobscheduler and executed with the general users privileges are discussed in the sections from 4.8 to 4.22. Also note that projects that can be operated by each command are determined by the access rights set to respective project.

P Point
....

When the Extended User Management function is enabled, the access right authentication depends on the OS user that executed the following commands:

a. If the OS user is a system administrator (user belonging to the Administrators group in Windows or superuser in UNIX), the commands are executed with the Operation Manager administrator privileges (has update rights for all projects).

b. If the OS user is a project owner, the commands are executed with the assumption that the owner has update rights over their projects.

c. If the OS user is neither a system administrator nor a project owner, the OS user that executed the commands is included in the OS users corresponding to all Operation Manager users that have access rights to the project, and the commands are executed if the user has the privileges. If the OS user that executed the commands corresponds to more than one Operation Manager user, the OS user will be given the most powerful privileges (update rights > change and operation rights > reference rights) among the privileges that the corresponding Operation Manager users have. If the corresponding Operation Manager user has both change rights and operation rights, then the OS user will also have both privileges.

When the Systemwalker authentication repository is enabled, the access right authentication depends on the OS user that executes the following commands:

a. If the OS user is a system administrator (user belonging to the Administrators group in Windows or superuser in UNIX), the commands are executed with the Operation Manager administrator privileges (has update rights for all projects).

b. If the OS user is a project owner, the commands are executed with update rights over the project they are the owner of.

c. If the OS user is neither a system administrator nor a project owner, the commands are executed if the OS user has access rights over the project (update rights, change rights, operation rights, and reference rights).

- jobschsetnet command

- jobschsetgrp command

- jobschctljob command

- jobschcontrol command

- jobschctlgrp command

jobschsetnet command

Registers, changes, deletes, and copies job nets. Can also be used to register or delete the startup days/message events/ temporary changes information for the job net.

Job net control statement

Sets registration information of job nets.

Job control statement

Sets registration information of jobs.

Startup days control statement

Sets job net startup days.

Message event control statement

Sets message events.

Temporary changes control statement

Sets temporary change information of job nets.

jobschsetgrp command

Registers, changes and deletes groups.

Group control statement

Sets group registration information.

Group job net control statement

Sets registration information of the job net in a group.

jobschctljob command

Controls jobs.

jobschcontrol command

Controls job nets.

jobschctlgrp command

Controls groups.

jobschmove command

Changes the registration order of groups/job nets.

jobschmsgclear command

Waits once again for already generated message events.

jobschmsgevent command

Generates message events.

jobschcancelnet command [Windows]

Deletes the job net from the job net execution queue.

jobschdeftocsv command

Converts the registration information for the job net, group, holidays in the calendar and schedule pattern from the control statement format into the CSV format. The conversion results are output to the standard output (stdout).

jobschcsvtodef command

Converts the registration information for the job net, group, holidays in the calendar and schedule pattern from the CSV format into the control statement format. The conversion results are output to the standard output (stdout).

jobschchecktime command

Waits until the specified time comes.

jobschchecklog command

Checks whether a log file in text format contains a specific regular expression.

chkfile/jobschchkfile command

Checks the existence of files, and waits for files to be created/updated/deleted/resized.

jobschnetmemo command

Registers/deletes memo to/from a job net. Also outputs and searches the contents of the memo.

jobschchkmsgevt command

Called by Wait for Message Event jobs.

jobschchknetvar command

Called by jobs that check the values of job net variables.

jobschsleep command

Waits for the specified time period.

Commands for migration from earlier versions

In the section 4.23, the command used for migration from the earlier versions is discussed.

jobschconvert command

Converts the Jobscheduler database from the earlier version to the type that can be used in the current version.

About This Control Statement

When you create a control statement, use an editor, such as Notepad. A file containing control statements is referred to as a definition file.

The synopsis and notes for control statements are described below.

Synopsis

Write a control statement in the following format:

```
KEYWORD                            <- The keyword representing the control statement
operand1 Followed by a parameter    <- An option and associated parameter
operand2 Followed by a parameter
......
operandn Followed by a parameter
;                                   <- End of the control statement
```

Cautions

- A unit of control statement starts with a keyword that represents the control statement and ends with a semicolon (;).

- Enter one keyword or one option per line using a line break.

- Insert one space character between an operand and a parameter.

- Any incorrect specification causes registration of the control statement to fail.

- Comments can also be included. Use the hash "#" symbol to enter comments. Any text after the hash "#" symbol will be treated as a comment.

4.2 jobschsethol Command

Synopsis

```
[To register/change]
  jobschsethol -ent definition file
[To delete]
  jobschsethol -del calendar name
```

Description

The jobschsethol command registers, changes and deletes holidays in the calendar.

Options

-ent

Based on the information in the definition file, registers and changes holidays in the calendar.

definition file

Specifies the name of the file storing the holiday information control statement.

For more details on the holiday information control statement, see "4.2.1 Holiday Information Control Statement".

-del

Deletes calendars other than the SYSTEM_CALENDAR.

calendar name

Specifies the name of the calendar to be deleted, excluding the SYSTEM_CALENDAR.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- Holidays in the Calendar can be registered/changed/deleted only by a system administrator (user belonging to the Administrator group or superuser). General users cannot use the jobschsethol command.

- The Calendar being quoted by a job net cannot be deleted.

- If the SYSTEM_CALENDAR is changed from a Working day to a holiday, the pattern information for the associated power control will be lost. Also, if it is changed from a holiday to a Working day, the pattern of power control is set to the standard pattern (except for HP-UX, AIX).

- This command overwrites existing holidays in the Calendar and recreates new holidays. Therefore, the existing holidays information individually is cleared. Set up the holidays information again as necessary.

- You cannot set the option of "Specify the reschedule date" by using this command.

If the [Halt recreation of startup days] option is not selected in the [Use function2] sheet in the [Define Jobscheduler Startup Parameters] window, note the following:

- If the Calendar being quoted by a job net is changed, the job net startup days are reset based on the new Calendar. After changing the Calendar, make sure and modify the job net startup days (settings for exception days). In addition, even if the "Specify the reschedule date" option is specified, startup days of the job net will be reset based on the new Calendar regardless of the specified reschedule date.

- This command only requests the server to change the holiday calendar, which means that completing the command does not guarantee that these changes have been applied to the startup days of each job net on the server. Therefore, after executing this command, the changes may be reflected in the startup days of each job net after an extended period.

- By using this command, when you create a new holiday calendar with the same name as the one already existing, the already existing calendar will be overwritten and recreated. Therefore, this makes the contents of the existing calendar the same as the contents of control statement in the newly created calendar.

4.2.1 Holiday Information Control Statement

This control statement is used to set holiday information.

Synopsis

```
HOL                                 ---------- (1)
calendarname Calendar name          ---------- (2)   (*)   (Note)
yearset Year setting                ---------- (3)   (*)
monthset Month setting              ---------- (4)   (*)
dayset Day setting                  ---------- (5)   (**)
weekset Day of the week setting     ---------- (6)   (**)
daycancel Day setting cancellation  ---------- (7)
;
```

Note:

The calendarname operand should be specified immediately after the HOL keyword.

*:

This must be specified.

**:

One or both of the operands marked with (**) must be specified.

About This Control Statement

(1) HOL

This is the keyword that indicates the beginning of a holiday information control statement.

(2) calendarname Calendar name

Specify the calendar name (for the SYSTEM_CALENDAR or some other) within 24 bytes. The following characters should not be used to specify a calendar name: /?*;,\"<>|:, space, or any user-defined character. Note that, if a name of an existing calendar that has already been set up is specified, the previous setting information is overwritten.

(3) yearset Year setting

Specify the year(s) in which holidays are set in four digits, using numerals.

Holidays can be defined for three years (current year, next year, year after next). To define the same holiday pattern for every year, specify "EVERY".

Year Setting Example

The following gives an example to set the current year and the next year when the current year is 2005.

```
yearset 2005,2006
```

(4) monthset Month setting

Specify the month(s) in which holidays are set, using numerals. To specify more than one month, the format should look similar to "3,5,10,....." To define the same holiday pattern for every month, specify "EVERY."

(5) dayset Day setting

Specify the day(s) which is (are) set to holiday(s), using numerals. To specify more than one day, the format should look similar to "1,2,15,....."

(6) weekset Day of the week setting

Specify holidays by day of the week. A day of the week should be specified by using the first three letters of the day. To specify more than one day of the week, the format should look similar to "SUN,MON,TUE,WED,THU,FRI,SAT".

(7) daycancel Day setting cancellation

Used to cancel the holiday setting for a specific date. Use the format "MM,DD(,DD),YYYY", all in numerals. "EVERY" cannot be specified for this operand.

Example of day setting cancellation

The following is an example of how to specify the day setting cancellation.

To cancel holiday setting for November 1, 2005:

```
daycancel 11,01,2005
```

To cancel holiday setting for November 1 and 15, 2005:

```
daycancel 11,01,15,2005
```

To cancel holiday setting for November 1 and December 20, 2005:

```
daycancel 11,01,2005
daycancel 12,20,2005
```

To cancel holiday setting for November 3, 2005 and November 3, 2006:

```
daycancel 11,03,2005
daycancel 11,03,2006
```

Example of holiday information control statement

The following is an example of the holiday information control statement when setting the dates below as holidays.

- November 1, 2005

- December 20, 2005

```
HOL
calendarname MainOfficeCalendar
yearset 2005
monthset 11
dayset 1
yearset 2005
monthset 12
dayset 20
;
```

Cautions

- The "Month setting" should be placed after the "Year setting."

- The "Day setting" and the "Day of the week" should be placed after the "Month setting."

- To specify more than one holiday pattern, combine a "Year setting" and "Month setting" to form one holiday pattern, and specify such combinations as many as required.

4.3 jobschsetcal Command

Synopsis

```
[To register/change]
  jobschsetcal -ent definition file name [-sys subsystem-number](*1)
[To delete]
  jobschsetcal -del schedule pattern name [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschsetcal command registers, changes and deletes a schedule pattern. By using the schedule pattern, you can easily set the startup days of job net. This is particularly useful when registering a job net having startup days scheduled in a cyclic pattern or when setting similar startup days to multiple job nets.

Options

-ent

Based on the information in the definition file, registers and changes the schedule pattern.

definition file name

Indicates the name of the file storing the schedule pattern control statement.

For more details on the schedule pattern control statement, see the "4.3.1 Schedule Pattern Control Statement".

-del

Deletes schedule pattern information.

schedule pattern name

Specifies the name of a schedule pattern to be deleted.

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- The schedule pattern can be registered/changed/deleted only by a system administrator (user belonging to the Administrator group or superuser). General users cannot use the jobschsetcal command.

If the [Halt recreation of startup days] option is not selected in the [Use function2] sheet in the [Define Jobscheduler Startup Parameters] window, note the following:

- If a registered schedule pattern is changed, the startup days of the job net for which the pattern is set are reset. After changing the schedule pattern, make sure or correct the startup days.

- Completing this command does not guarantee the completion of processing requests. Thus, after executing this command, it may take time to reflect settings to startup days of each job net on the server.

- By using this command, when you add a new schedule pattern with the same name as the one already existing, the already existing schedule pattern will be overwritten and recreated. Therefore, this makes the contents of the existing schedule pattern the same as the contents of control statement in the newly created schedule pattern.

4.3.1 Schedule Pattern Control Statement

This control statement is used to set a schedule pattern.

Synopsis

```
CAL                                                    ---------- (1)
execpatname Schedule pattern name                      ---------- (2) (*) (Note)
monthset Month setting                                 ---------- (3) (**)
dayset Day setting                                     ---------- (4) (**)
weekset Day of the week setting                        ---------- (5) (**)
daycancel Day setting cancellation                     ---------- (6) (**)
monthlyday Date setting for every month                ---------- (7) (**)
```

```
monthlyweek Day of the week setting for every month         ---------- (8) (**)
bnsmode Criteria of business day basis                       ---------- (9) (**)
bnsbase Month's start date of business day basis             ---------- (10)(**)
bnsday Business days                                         ---------- (11)(**)
bnslimit Set only business days of the month as startup days ---------- (12)(**)
;
```

Note:

The execpatname operand should be specified immediately after the CAL keyword.

*:

This must be specified.

**:

Select one of the following; a. (Yearly standards) /b. (Monthly standards) /c. (Business day standards).
a. monthset, dayset, weekset, daycancel (monthset is mandatory. Either or both dayset and weekset can be selected. daycancel can only be selected if dayset is selected.)
b. A combination of monthlyday and monthlyweek (either or both)
c. bnsmode, bnsbase, bnsday, bnslimit (The bnsmode operand and the bnsbase operand must be specified if this operand group is specified.)

About This Control Statement

(1) CAL

This is the keyword that indicates the beginning of a schedule pattern control statement.

(2) execpatname Schedule pattern name

Specify a schedule pattern name within 24 bytes. The name should not include a blank space or a user-defined character. Note that, if a schedule pattern name that has already been setup is specified, the previous setting information is cleared.

(3) monthset Month setting

When defining the schedule pattern information using the yearly standards, specify the month(s) using numerals. This must be specified. To specify more than one month, the format should look similar to "3,5,10,....." To define the same schedule pattern for every month, specify "EVERY".

(4) dayset Day setting

When defining the schedule pattern information using the yearly standards, specify the day(s) using numerals. To specify more than one day, the formal should look similar to "1,2,15,21,...."

(5) weekset Day of the week setting

When defining the schedule pattern information using the yearly standards, this is used to specify the schedule pattern by day of the week. A day of the week should be specified by using the first three letters of the day. To specify more than one day of the week, the format should look similar to "SUN,MON,TUE,WED,THU,FRI,SAT".

(6) daycancel Day setting cancellation

When defining the schedule pattern information using the yearly standards, this is used to cancel schedule pattern setting for a specific day. Specify the month and the date on which you want to cancel the schedule pattern, using the format "MMDD" with two numerals respectively.

Example of day setting cancellation

The following is an example of how to specify the day setting cancellation.

To cancel the schedule pattern set for January 1:

```
daycancel 0101
```

To cancel schedule patterns set for January 1 and 10 and February 10:

```
daycancel 0101,0110,0210
```

(7) monthlyday Date setting for every month

When defining the schedule pattern information using the monthly standards, specify the date(s) in which the job net is executed for every month. To set DD (day) of every month as the startup day, specify "DD". More than one startup day can be set. To execute the job net at the end of every month, enter "LAST".

(8) monthlyweek Day of the week setting for every month

When defining the schedule pattern information using the monthly standards, specify the day(s) of the week on a weekly basis in which the job net is executed for every month. It is only one week that can be specified with one operand. To set more than one week, use this operand repeatedly. To specify this operand, enter the ordinal number of the week (where 1 - 5 represents the first through the fifth week, 0 indicates that every week is specified, and 6 represents the last week of the month) with a number between 0 and 6 that is followed by the first three letters of the day of the week. The format should look similar to the following: 5,SUN,MON,TUE,WED,THU,FRI,SAT.

(9) bnsmode Criteria of business day basis

When defining the schedule pattern information using business day standards, specify either "Month's start date" or "Month's end date" as the selection criteria of business days.

START

Counts the business days from the Month's start date.

END

Counts the business days back from the Month's end date.

(10) bnsbase Month's start date of business day basis

When defining the schedule pattern information using business day standards, specify "Month's start date". "0" is invalid.

1 to 28

Indicates 1st to 28th of the month.

LAST

Indicates the end of the month.

(11) bnsday Business days

When defining the schedule pattern information using business day standards, specify business day. To specify more than one business day, use a comma (,) as a separator. "0" is invalid.

1 to 31

The business days that are specified by counting from the 1st business day (the first business day of the month) using a range from 1 to 31. This parameter cannot be specified if "END" is set in the bnsmode operand.

-1 to -31

The business days that are specified by calculating back from the -1st business day (the last business day of the month) using a range from -1 to -31. This parameter cannot be specified if "START" is set in the bnsmode operand.

(12) bnslimit Set only business days of the month as startup days

When defining the schedule pattern information using business day standards, specify whether "Set only business days of the month as startup days" is set to ON or OFF. If the bnsmode operand is specified, the default is "ON".

ON

"Set only business days of the month as startup days" is selected.

OFF

"Set only business days of the month as startup days" is not selected.

Example of schedule pattern control statement

The following is an example of setting the schedule pattern with the dates below as startup days.

- January 1

- February 10

```
CAL
execpatname Sample
monthset 1
dayset 1
monthset 2
dayset 10
;
```

Cautions

- The "Day setting" and the "Day of the week setting" should be placed after the "Month setting."

- If more than one pattern is specified for "Day setting" and "Day of the week setting", be sure to include "Month setting" for each of schedule patterns to be specified

4.4 jobschcod Command

Synopsis

```
jobschcod day change time
```

Description

The jobschcod command changes the day change time.

Options

day change time

The day change time is set. Specify this option in the format "hhmm" in four digits. "hh" is set in a range "from 00 to 23" (hour) and "mm" is set in a range "from 00 to 59" (minute).

Example Setting

The following shows an example where the day change time is set to 9.05 a.m.

```
jobschcod 0905
```

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- The day change time can be changed only by a system administrator (user belonging to the Administrator group or superuser). General users cannot use the jobschcod command.

- The day change time is stored in SYSTEM_CALENDAR. You should be aware of when using the jobschcod command because the power schedule will be also affected, not only the schedule of Jobscheduler (except for HP-UX, AIX).

4.5 jobschsetproject Command

Synopsis

```
[To register/change]
  jobschsetproject -ent definition file [-sys subsystem-number]("1)
[To delete]
  jobschsetproject -del project name [-sys subsystem-number]("1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschsetproject command registers, changes and deletes the project information.

It also defines the users that can access (update/operate/reference) the job nets within a project.

Options

-ent

Based on the information in the definition file, registers and changes project information.

definition file

Specifies the name of the file storing the project information control statements.

For more information on the project information control statement, see "4.5.1 Project Information Control Statement".

-del

Deletes the project information.

project name

Specifies the name of the project to be deleted. The project cannot be deleted unless all definitions (group/job net) within the project have been deleted.

EE -sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- Projects can be registered/changed/deleted only by a system administrator (user belonging to the Administrator group or superuser). General users cannot use the jobschsetproject command.

4.5.1 Project Information Control Statement

This control statement is used to set project information.

Synopsis

```
PRJ                                                      ---------- (1)
projectname project name                                 ---------- (2)
(*)   (*1)
username owner(OS user)                                  ---------- (3)
(**)  (*2)
manage_user IDs of update rights users (OS users)         ---------- (4)      (*2)
change_user IDs of change rights users (OS users)         ---------- (5)      (*2)
operate_user IDs of operation rights users (OS users)     ---------- (6)      (*2)
monitor_user IDs of reference rights users (OS users)     ---------- (7)      (*2)
manage_com_user IDs of update rights users (Systemwalker user ID) ---------- (8)
change_com_user IDs of change rights users(Systemwalker user ID)   ---------- (9)
operate_com_user IDs of operation rights users (Systemwalker user ID) ------ (10)
monitor_com_user IDs of reference rights users (Systemwalker user ID) ------ (11)
;
```

*1:

The projectname operand should be specified immediately after the PRJ keyword.

*2:

No user principal name should be specified.

*:

This must be specified.

**:

This must be specified when registering.

About This Control Statement

(1) PRJ

This is the keyword which indicates the beginning of a project information control statement.

(2) projectname Project name

Specify the project name in 50 bytes or less. The following characters may not be used to specify a project name: /?*;,\"<>|:'., space, or any user-defined character. In addition, # cannot be used for the first character.

Note:

can be used for the first character if the attribute of all the job nets in the project is the following:

- Normal attribute

This is not case-sensitive. [Windows]

Note that, if the project name that has already been set up is specified, the setting information of already existing project is replaced by this new setting information.

This is case-sensitive. But, if project names with the same spelling but different case are registered separately, the schedule information cannot be collected properly with the policy extraction. [UNIX]

(3) username Owner (OS user)

Specify the user ID of the project owner.

If the impersonation option is specified with the Job Execution Control, this ID can be a user ID used to impersonate the job execution right. Specify the name in the "domain name\user ID" format. The maximum length of user ID is up to 36 bytes (15 bytes for domain name +1 byte for a backslash (\) + 20 bytes for user ID). If you omit the domain name, it is defaulted to the name of the computer where the Jobscheduler service resides. Group names (either for a local or global group) cannot be specified. Also, a user ID containing blank spaces cannot be specified. [Windows]

Specify the user ID of the project owner within 20 bytes. A user ID containing blank spaces cannot be specified. Even when the Extended User Management function [UNIX] or Systemwalker authentication repository is enabled, always specify the OS user for the owner. [UNIX]

(4) manage_user ID of user having update right (OS user)

Specify a user ID to whom permissions to update projects are given.

Specify the user ID in the "domain name\user ID" format. The domain name (including the backslash (\)) can be omitted. The ID must be specified in 36 bytes or less (if specifying a user ID only, it is up to 20 bytes). [Windows]

If the Extended User Management function is enabled, specify the Operation Manager user within 20 bytes in length. [UNIX]

If the Systemwalker authentication repository is enabled, specify the OS user to be given update rights when executing commands.

A group name can also be specified.

Multiple user IDs can be specified by separating them by a comma (,). Note that the ID cannot contain blank spaces.

When you want to delete all the registered users who have the permission to update, you can do it by omitting the user ID in manage_user.

(5) change_user ID of user having change rights (OS user)

Specify the ID of the user to be given change rights.

Specify the user ID using the following format: domain name\user ID. The domain name part (before the \ character) can be omitted. Specify no more than 36 bytes (or no more than 20 bytes if only the user ID is specified). [Windows]

If the Extended User Management function has been enabled, specify an Operation Manager user using no more than 20 bytes. [UNIX]

If the Systemwalker authentication repository is enabled, specify the OS user to be given change rights when executing commands.

It is also possible to specify group names. To specify more than one user ID, separate user ID by commas (,). User IDs that contain spaces cannot be specified.

To delete all change right users that have been registered, omit the user ID specified in the "change_user" control statement.

(6) operate_user ID of user having operation right (OS user)

Specify a user ID to whom permissions to operate projects are given.

Specify the ID in the "domain name\user ID" format. The domain name (including the backslash (\)) can be omitted. The name must be specified in 36 bytes or less (if specifying a user ID only, it is up to 20 bytes). [Windows]

If the Extended User Management function is enabled, specify the Operation Manager user within 20 bytes in length. [UNIX]

If the Systemwalker authentication repository is enabled, specify the OS user to be given operation rights when executing commands.

A group name can also be specified.

Multiple user IDs can be specified by separating them by a comma (,). Note that the ID cannot contain blank spaces.

When you want to delete all the registered users who have the permissions to operate, you can do it by omitting the user ID in operate_user.

(7) monitor_user ID of user having reference right (OS user)

Specify a user ID to whom permissions to monitor/view projects are given.

Specify the ID in the "domain name\user ID" format. The domain name (including the backslash (\)) can be omitted. The name must be specified in 36 bytes or less (if specifying a user ID only, it is up to 20 bytes). [Windows]

If the Extended User Management function is enabled, specify the Operation Manager user within 20 bytes in length. [UNIX]

If the Systemwalker authentication repository is enabled, specify the OS user to be given reference rights when executing commands.

A group name can also be specified. Multiple user IDs can be specified by separating them by a comma (,). Note that the ID cannot contain blank spaces

When you want to delete all the registered users who have the permissions to reference, you can do it by omitting the user ID in monitor_user.

(8) manage_com_user ID of user having update right (Systemwalker user ID)

Specify a Systemwalker user ID to whom permissions to update projects are given. This can only be specified when the Systemwalker authentication repository is enabled.

Specify no more than 32 bytes.

Delimit with commas when specifying more than one.

To delete all update right Systemwalker user IDs that have been registered, omit the Systemwalker user ID specified in manage_com_user.

(9) change_com_user ID of user having change rights (Systemwalker user ID)

Specify a Systemwalker user ID to whom permissions to change projects are given. This can only be specified when the Systemwalker authentication repository is enabled.

Specify no more than 32 bytes.

Delimit with commas when specifying more than one.

To delete all change right Systemwalker user IDs that have been registered, omit the Systemwalker user ID specified in change_com_user.

(10) operate_com_user ID of user having operation right (Systemwalker user ID)

Specify a Systemwalker user ID to whom permissions to operate projects are given. This can only be specified when the Systemwalker authentication repository is enabled.

Specify no more than 32 bytes.

Delimit with commas when specifying more than one.

To delete all operation right Systemwalker user IDs that have been registered, omit the Systemwalker user ID specified in operate_com_user.

(11) monitor_com_user ID of user having reference right (Systemwalker user ID)

Specify a Systemwalker user ID to whom permissions to reference projects are given. This can only be specified when the Systemwalker authentication repository is enabled.

Specify no more than 32 bytes.

Delimit with commas when specifying more than one.

To delete all reference right Systemwalker user IDs that have been registered, omit the Systemwalker user ID specified in monitor_com_user.

Cautions

- You cannot specify the same user ID or group name more than once in a single control statement. However the same user ID can be specified if they are for Systemwalker user ID and OS user.

- If you set another access right to the user to whom the access right has been given, his or her existing access right is overwritten.

Example of project information control statement

In the following example, the project name is set to "GeneralAffairsDept", the owner is set to "john", the ID of the update rights user is "james", the ID of the change rights user is "kerry", the ID of the operation rights user is "julie" and the IDs of the reference rights users are "anthony" and "mike".

```
PRJ
projectname GeneralAffairsDept
username john
manage_user james
change_user kerry
operate_user julie
monitor_user anthony,mike
;
```

EE

4.6 jobsch_start Command [UNIX]

Synopsis

```
jobsch_start [-sys {all | subsystem-number}]
```

Description

The jobsch_start command starts the Jobscheduler daemons. To use the jobsch_start command, write it in the following daemon start script.

Solaris:

/opt/FJSVJOBSC/etc/rc3.d/S99JOBSCH

HP-UX:

/opt/FHPJOBSCH/bin/S99JOBSCH

AIX:

/usr/FAIXJOBSC/bin/S99JOBSCH

Linux:

/opt/FJSVJOBSC/etc/rc3.d/S99JOBSCH

Options

-sys {all | subsystem-number}

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. If omitted, the default is subsystem 0.

all

Starts all the subsystems.

subsystem-number

Specifies the number of the subsystem to be started between 0 and 9.

Command Location

Solaris Linux	/opt/FJSVJOBSC/bin
HP-UX	/opt/FHPJOBSCH/bin
AIX	/usr/FAIXJOBSC/bin

Privilege Required

System administrator (superuser) privileges are required.

Cautions

- Execute the jobsch_start command by writing it in a daemon start script. If the command is directly executed, settings such as for environment variable performed within the daemon start script cannot be made. So it may cause problems where a different shell may be used at job execution.

- To start Jobscheduler on a desired subsystem manually, do the following steps.

    1. Copy a daemon start script.

    2. Edit the copied daemon start script.

    3. Execute the edited script manually.

- When writing the jobsch_start command in the daemon start script, it must be a full path.

- In Red Hat Enterprise Linux 7, the operating system may need to be stopped or restarted even if the daemons are configured to stop automatically when the operating system is stopped.
  In this case, manually execute the poperationmgr service/daemon termination command in advance to stop the Systemwalker Operation Manager daemons. [Linux]

EE 4.7 jobsch_end Command [UNIX]

Synopsis

```
jobsch_end [-sys {all | subsystem-number}]
```

Description

The jobsch_end command stops the Jobscheduler daemons. To use the jobsch_end command, write it in the following daemon stop script.

Solaris:

/opt/FJSVJOBSC/etc/rc0.d/K00JOBSCH

HP-UX:

/opt/FHPJOBSCH/bin/K00JOBSCH

AIX:

/usr/FAIXJOBSC/bin/K00JOBSCH

Linux:

/opt/FJSVJOBSC/etc/rc0.d/K00JOBSCH

Options

-sys {all | subsystem-number}

In the system with multiple subsystems, this option is used to specify the subsystem that you want to stop. If omitted, the default is subsystem 0.

all

Stops all the subsystems.

subsystem-number

Specifies the number of the subsystem to be stopped between 0 and 9.

Command Location

Solaris Linux	/opt/FJSVJOBSC/bin
HP-UX	/opt/FHPJOBSCH/bin
AIX	/usr/FAIXJOBSC/bin

Privilege Required

System administrator (superuser) privileges are required.

Cautions

- To stop the Jobscheduler on a desired subsystem manually, do the following steps.

    1. Copy a daemon stop script.

    2. Edit the copied daemon stop script.

    3. Execute the edited script manually.

- When writing the jobsch_end command in the daemon stop script, it must be a full path.

- In Red Hat Enterprise Linux 7, the operating system may need to be stopped or restarted even if the daemons are configured to stop automatically when the operating system is stopped.
  In this case, manually execute the poperationmgr service/daemon termination command in advance to stop the Systemwalker Operation Manager daemons. [Linux]

4.8 jobschgetoutput Command

Synopsis

```
jobschgetoutput -m message -f Job output acquisition file name [ -s Maximum file size]
```

Description

The jobschgetoutput command acquires the output of jobs in the job net indicated by the message specified as a parameter.

This command can be used only when Systemwalker Operation Manager is linked to Systemwalker Centric Manager. This command is used as an execution command in the mail notification settings for automatic actions.

This command can handle only messages that have been output as a result of selecting the **Completed**, **Abended**, **Canceled**, **Closed**, **Pseudo-normal** or **Automatic notification/handling** checkbox in the **Event output** tab of the **Define Jobscheduler Startup Parameters** window.

Options

-m message

Specifies a Systemwalker Operation Manager message that is to be monitored by Systemwalker Centric Manager. This command is used as an execution command for Systemwalker Centric Manager's mail notification automatic action, so specify %MSG.

-f Job output acquisition file name

Specify the full path to the file that stores output information for the jobs in the job net indicated by the message.

If this command is used as an execution command for mail notifications in Systemwalker Centric Manager automatic actions, information will be stored in the specified file.

-s Maximum file size

Specifies the maximum number of bytes for the job output acquisition file. Specify a value between 1000 and 10000000. If this option is not specified, the maximum number of bytes will be 20000 bytes

Return Values

Return value	Meaning
0	Indicates that processing has terminated normally.
11	Indicates that the message is in a format that cannot be handled by this command.
12	Indicates that reading job net information has failed.
13	Indicates that reading the job output has failed.
14	Indicates that writing the job output acquisition file has failed.
21	Indicates that the options have not been specified correctly.
22	Indicates that an error relating to the system or the user environment has occurred.
31	Indicates that an internal error relating to the Systemwalker Operation Manager environment has occurred.
35	Indicates that another type of internal error has occurred.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Privilege/Execution Environment Required

- System administrator (user belonging to the Administrator group or superuser) privileges are required.

- Systemwalker Operation Manager server function and one of the following products must be running on the same machine:

  - Systemwalker Centric Manager V13.3.0 (either an Operation Management Server, Section Management Server or Job Server)

Example

To acquire job output as an execution command for mail notification (an automatic action) and store the job output in the file "/var/tmp/jobschoutput.log" with the default maximum file size, make the following definitions (for the UNIX version).

```
/usr/bin/jobschgetoutput -m %MSG -f /var/tmp/jobschoutput.log
```

Execution Results/Output Format

The format of the job output acquisition file is as follows:

```
Subsystem Number: Subsystem number
Project Name: Project name
Job Net Name: Job net name
Job Net: Job net comment
Job Name:
Job number: Job name (Job status)
Job number: Job name (Job status)
.
.

Job Output (Job output percentage):
Job number Job output
Job number Job output
Job number Job output
.
.
```

Subsystem number:

For the Standard Edition, this will be 0. For the Enterprise Edition, this will be the number of the subsystem that the job net indicated by the message belongs to.

Project name:

This is the name of the project that the job net indicated by the message belongs to.

Job net name:

This is the name of the job net indicated by the message.

Job net comment:

This is the comment for the job net indicated by the message.

Job number:

This is a unique number for a job within the job net indicated by the message.

Job name:

This is the name of a job in the job net indicated by the message. Job names vary according to the type of job and the execution attribute of the job net, as follows:

- If the job is a batch framework job, then this will be the job name if a job name has been specified, or the batch job definition name otherwise.

- If the job is a child job net, this will be the name of the child job net.

- If the job is a linked job net, this will be the name of the linked job net.

- If the execution attribute for the job net is "Interstage", the job name will be the workunit name.

- If the execution attribute for the job net is "Normal", the job name will be the command name.

- In all other cases, a job name will be used as the job name if it has been set explicitly, and the command name will be used as the job name unless a job name has been set explicitly. If a parameter has been set, then a space can be inserted between the command name and the parameter only if the command name is used as the name of the job. Any job definition variables included in the command name or parameters will not be replaced.

Job status:

The job status is displayed as follows:

- "Completed"

- "Abended"

- "Canceled"

- "Warning"

- "Executing"

- "Waiting"

- "Paused"

- "Disabled"

- "Pseudo-normal"

- "Delayed"

Job output percentage:

This will be "all" if the entire job output is stored in the job output acquisition file. Otherwise, this will be the percentage of the job output that could be stored in the job output acquisition file, expressed as a percentage between 0% and 99%. The percentage is not completely accurate, because values are rounded down to the nearest integer. Use these percentages as an approximate indication of the proportion of the job output that is stored in the job output acquisition file.

Job output

This section is the job output, line by line. If a single job outputs multiple lines, each line will be stored in the job output acquisition file together with the job number as a set.

Cautions

- If job output acquisition files do not need to be kept, enable deletion of transmitted files when making the settings for the execution command for mail notifications in Systemwalker Centric Manager automatic actions.

- If the maximum file size is exceeded, data is stored in the job output acquisition file as follows:

If the part before the job output, such as the project name or job net name (the "header part") exceeds the maximum file size:

The first part of the job output (from the beginning up to the maximum file size) is stored in the job output acquisition file.

Job output is stored line by line. Incomplete lines are not stored.

If the total size of the header part and the job output exceeds the maximum file size:

Job output is stored after any excess data has been deleted from the beginning of the file.

Job output is stored line by line. Incomplete lines are not stored.

Job output is stored separately for each job net in the order that they terminated.

- There is a possibility that the header section and job output section of the job output acquisition file may not match (or this command may terminate with a return value of 13) if a job net is manipulated or edited before this command is executed after the job net has terminated.

- This command will terminate with return value "12" if a job net is deleted before this command is executed after the job net has terminated.

4.9 jobschsetnet Command

The jobschsetnet command is used when setting up (register/change/delete/copy) a job net, and setting (register/delete) startup days, message events and temporary changes information with the job net information.

Setting Up Job Nets

This section explains the jobschsetnet command in setting job nets along with the following items: Synopsis, Description, and Options.

Synopsis

```
[To register]
  jobschsetnet -nent [project name] definition file [-sys subsystem-number]⁽*¹⁾
```

```
[To change]
  jobschsetnet -nche [project name] definition file [-sys subsystem-number]⁽*¹⁾
[To delete]
  jobschsetnet -ndel [project name/]job net name [-sys subsystem-number]⁽*¹⁾
[To copy]
  jobschsetnet -ncpy [copy source project name/]copy source job net name
                     [copy destination project name/]copy destination job net name
                     [-sys subsystem-number]⁽*¹⁾
```

*1:

 [-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

Registers, changes, deletes and copies job nets.

When registering or changing a job net, you can set startup days information (including information on exception days) expanded by using the jobschprint -r command.

Can specify project names that you want to operate.

📝 Note

. . . .

About the number of job nets that can be registered with a project

The maximum number of job nets that can be registered with a project differs, depending on the edition (SE or EE):

 - In Systemwalker Operation Manager SE, up to 255 job nets can be registered.

 - In Systemwalker Operation Manager EE, there is no limit on the number of job nets that can be registered.

. . . .

Options

-nent

Based on the information in the definition file, registers job nets. Note that registration is not possible if the job net with the same name already exists.

project name

Specifies the name of the project that stores the job net information. When registering, if there is only one project for the user of executing the jobschsetnet command to access for updating, this option can be omitted. When changing/deleting/copying, if there is only one project having job nets registered among the projects accessible for update by the user of executing the jobschsetnet command, this option can be omitted.

definition file

Specifies the name of the file where the job net control statements and job control statements are stored.

To register multiple job nets, you need to write multiple job net control statements in the file.

For more details on the job net control statement, see "4.9.1 Job Net Control Statement".

For more details on the job control statement, see "4.9.2 Job Control Statement".

-nche

Changes existing job nets based on the information in the definition file. It is convenient to note the changes in the definition file used when registering the job net. For operands that have not been mentioned in the definition file, changes will not be made to them if it is a job net to be changed, but if a job is to be changed, care must be taken because the definitions will be deleted. List all operands in the definition file when changing a job.

-ndel

Deletes job nets. Once the job net has been deleted, the entire registration information of the job net and the jobs composing the job net is also deleted.

job net name

Specifies the name of the job net to be deleted.

-ncpy

Copies job nets. The status of the copy destination job net becomes "Disabled". However, copying cannot be performed if a job net with the same name already exists in the copy destination project.

If the job net with "base job net" specified is copied to a different project, the name of base job net is removed. However, the startup day information except the base job net name is the same as the copy source.

copy source project name

Specifies the name of the project where the source job net information is stored. If there is only one project having job nets registered among the projects that can be referenced by the user of executing the jobschsetnet command, this option can be omitted.

copy source job net name

Specifies the name of job net to be copied.

copy destination project name

Specifies the name of the project where the destination job net information is stored. If omitted, it is assumed that the same project name as the source project name is specified. Specify the project name that can be accessed for update by the user of executing the jobschsetnet command.

copy destination job net name

Specifies the name of job net to which the job net is copied.

EE **-sys subsystem-number**

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

📖 Information

. . . .

Job net copy option -ncop in V13.3.1 or earlier

Setting Startup Days/Message Events/Temporary Changes Information

This section explains the jobschsetnet command in setting startup days/message events/temporary changes information along with the following items: Synopsis, Description, Options, and Cautions.

Synopsis

```
[To register]
  jobschsetnet -sent [project name] definition file [-sys subsystem-number]⁽*¹⁾
[To delete]
  jobschsetnet -sdel [project name/] job net name [SCH|MSG|TMP]
                     [-sys subsystem-number]⁽*¹⁾
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

Registers/deletes the startup days, message events, and temporary changes information in the job net information.

Options

-sent

Based on the information in the definition file, registers the startup days, message events, and temporary changes information.

project name

Specifies the name of the project where the job net information is stored. If there is only one project having job nets registered among the projects accessible for update by the user of executing the jobschsetnet command, this option can be omitted.

definition file

Specifies the name of the file where the control statements for the startup days, message events and temporary changes information are stored.

For more details on the startup days control statement, see the "4.9.3 Startup Days Control Statement".

For more details on the message event control statement, see "4.9.4 Message Event Control Statement".

For more details on the temporary changes control statement, see the "4.9.5 Temporary Changes Control Statement".

-sdel

Deletes the startup days/message event/temporary changes information.

job net name [SCH|MSG|TMP]

Specifies the name of the job net on which the startup days/message event/temporary changes information is deleted.

SCH

Only deletes the startup days.

MSG

Only deletes the message events.

TMP

Only deletes the temporary changes information.

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Cautions

- For the job net with startup days/message event/temporary changes information that has been registered, when registering a new startup days/message event/temporary changes information, the new information in the definition file will override the previous one.

- When you want to delete all of the startup days/message event/temporary changes information, do not specify any SCH, MSG or TMP parameters.

- This command only requests the server to change the job net for its information, which means that completing the command does not guarantee that these changes have been applied to the job net (after executing this command, the changes may be reflected in the job net after an extended period.)

- When you change or delete the schedule pattern and the holiday calendar used in the job net to be registered using this command, it must be performed after this command is completed.

- If job nets are to be registered using this command with the files output by the jobschprint -r or jobschprint -R commands, job icon position information is initialized, so make the proper adjustments afterwards.

4.9.1 Job Net Control Statement

This is a control statement for setting registration information of job nets.

Synopsis

```
NET                                                      ---------- (1)
jobnetname Job net name                            ---------- (2)  (*)   (Note 1)
jobnetcomment Job net comment                       ---------- (3)  (*)    (Note
2)(Note 3)
noexecution No execution condition                  ---------  (4)  (**)  (Note
3)(Note 4)
starttime Single startup                           ---------- (5)  (**)  (Note 3)
multitime Multiple startups                        ---------- (6)  (**)  (Note 3)
intervalstart Fixed interval                       ---------- (7)  (**)  (Note 3)
iplstart At server startup                         ---------- (8)  (**)  (Note 3)
msgonly Message-driven                             ---------- (9)  (**)  (Note 3)
msgonlymode Action at message startup               ---------- (10)
msgstarttime Scheduled start time                  ---------- (11)       (Note 5)
stoptime Stop time                                      ---------- (12)
alarmtime Estimated end time                       ---------- (13)       (Note 6)
estimatetime Estimated processing time             ---------- (14)       (Note 7)
powermode Action at power off                       ---------- (15)
iplmode Action at server startup                    ---------- (16)
execattr Execution attribute                        ---------- (17)
carryovermode Instruction                           ---------- (18)
copystart Copy and start                           ---------- (19)       (Note 18)
jobnetsavenumber Number of saved copy destination job nets ---------- (20)     (Note 18)
(Note 19)
```

```
defaulthost Default host name                                       ---------- (21)
standnet Base job net declaration                                   ---------- (22)
execpattern Schedule pattern name                                   ---------- (23) (***)
execpatmode Method to combine the schedule patterns        -------- (24)       (Note 8)
basenet Base job net name                                           ---------- (25) (***)
execday Startup day setting                                         ---------- (26)
cancelday Startup day setting cancellation                 ---------- (27)
jobschprintr Startup day bit pattern                       ---------- (28)
annuallydate Date setting for every year                   ---------- (29) (***)
annuallyweek Day of the week setting for every year    ---------- (30) (***)
monthlydate Date setting for every month                   ---------- (31) (***)
monthlyweek Day of the week setting for every month   ---------- (32) (***)
bnsmode Criteria of business day basis                     ---------- (33) (***) (Note 9)
bnsbase Month's start date of business day basis      ---------- (34) (***) (Note 10)
bnsday Business days                                                 ---------- (35) (***)
bnslimit Set only business days of the month as startup days   ----- (36) (***)
offsetmode Offset                                                   ---------- (37)
offsetday Number of offset days                                     ---------- (38)
offsetholiday Sets the base day as a working day if it falls on a holiday  --- (39)
holidayshift Holiday shift                                 ---------- (40) (*)   (Note 3)
shiftlimit Number of maximum shifts                        ---------- (41)       (Note 12)
holidaycalendar Calendar name                              ---------- (42)
holidaymode Condition for combining calendars              ---------- (43)       (Note 11)
enablestart Start date of validity period                  ---------- (44)
enableend End date of validity period                      ---------- (44)
disablestart Start date of invalidity period               ---------- (45)
disableend End date of invalidity period                   ---------- (45)
tstarttime Single startup (temporary change)          ---------- (46) (****) (Note 13)
tmultitime Multiple startups (temporary change)            ---------- (47) (****) (Note
13)
tintervalstart Fixed interval (temporary change)           ---------- (48) (****) (Note
13)
tiplstart IPL startup (temporary change)                   ---------- (49) (****) (Note
13)
tmsgonly Message-driven (temporary change)                 ---------- (50) (****) (Note
13)
tmsgonlymode Action at message startup (temporary change)   ------ (51)
tmsgstarttime Scheduled start time (temporary change)  ---------- (52)       (Note 14)
tstoptime Stop time (temporary change)                     ---------- (53)
talarmtime Estimated end time (temporary change)       ---------- (54)       (Note 15)
tpowermode Action at power off (temporary change)          ---------- (55)
tiplmode Action when activated (temporary change)          ---------- (56)
temporarystart Start date of temporary change              ---------- (57)       (Note
13)
temporaryend End date of temporary change                  ---------- (57)
messagemode Message condition                              ---------- (58)
(Note 16)
carryover Carryover processing                             ---------- (59)
(Note 17)
messagename Message event name                             ---------- (60)
job Number of jobs within a job net                   ---------- (61) (*)    (Note 3)
operate Operation                                                    ---------- (62)
;
```

Note 1:

The jobnetname operand should be specified immediately after the NET keyword.

Note 2:

The jobnetcomment operand should be specified immediately after the jobnetname operand.

Note 3:

This operand can be omitted when the jobschsetnet -nche is specified.

Note 4:

If you have specified the noexecution operand, you cannot specify any of tstarttime, tmultitime, tintervalstart, tiplstart, tmsgonly, tstoptime, talarmtime, tpowermode, tiplmode, temporarystart, temporaryend, messagemode, messagename, msgonlymode, msgstarttime, tmsgonlymode, tmsgstarttime, carryover operands.

Note 5:

The msgstarttime operand can be specified if the msgonlymode operand is set to "ONCE". It should be specified immediately after the msgonlymode operand.

Note 6:

When specifying the alarmtime operand, you should also specify the starttime operand. The alarmtime operand cannot be specified with the stoptime operand, or the estimatetime operand simultaneously.

Note 7:

The estimatetime cannot be specified with the alarmtime operand simultaneously.

Note 8:

The execpatmode operand should be specified immediately after the execpattern operand.

Note 9:

The bnsmode operand cannot be specified with the offsetmode operand, the offsetday operand, or offsetholiday operand simultaneously. y.

Note 10:

This operand must be specified when the bnsmode operand is specified.

Note 11:

The holidaymode operand should be specified immediately after the holidaycalendar operand.

Note 12:

The shiftlimit operand should be specified immediately after the holidayshift operand.

Note 13:

If any operand for temporary changes is specified, this operand must be specified as well. It can be omitted if jobschsetnet -nche is specified.

Note 14:

The tmsgstarttime operand can be specified if "ONCE" is set to the tmsgonlymode operand. It should be specified immediately after the tmsgonlymode operand.

Note 15:

When specifying the talarmtime operand you should also specify the tstarttime operand. The talarmtime operand cannot be specified with the tstoptime operand, or the estimatetime operand simultaneously.

Note 16:

If the messagemode operand is specified, specify the messagename operand as well.

Note 17:

The carryover operand should be specified immediately after the messagemode operand.

Note 18:

This operand can be specified when the execution attribute is "Job Execution Control" (JES).

Note 19:

The jobnetsavenumber operand should be specified immediately after the copystart operand.

Remarks:

When changing the registration information of job net, you can omit all operands except for jobnetname. In that case, information of operands that have been omitted will not be changed.

*:

This must be specified.

**:

Make sure to select one of these.

***:

Specify one of the following. This can be omitted.

- execpattern

- basenet

- annuallydate and/or annuallyweek

- monthlydate and/or monthlyweek

- bnsmode, bnsbase, bnsday, bnslimit (The bnsmode operand and the bnsbase operand must be specified if these operands group is specified.)

****:

Select one of these if any is specified.

About This Control Statement

(1) NET

This is the keyword that indicates the beginning of a job net control statement.

(2) jobnetname Job net name

Specify a job net name within 50 bytes. It should be a unique job net name.

The following characters should not be used to specify a job net name: /?*;,\"<>|:'., space, or any user-defined character. In addition, # cannot be used for the first character.

Note:

can be used for the first character if the attribute of all the job nets in the project is the following:

- Normal attribute

(3) jobnetcomment Job net comment

Specify a comment on the job net within 100 bytes. The name should not contain a double quotation mark, a blank space, or a user-defined character.

(4) noexecution No execution condition

Specify "ON" when no execution condition is defined.

(5) starttime Single startup

When you execute the job net once a day, specify the execution time in the HHMM (hour and minute) format in four bytes.

Example of single startup setting

To execute the job net at 9:05 a.m.:

```
starttime 0905
```

To execute the job net at 9:00 p.m.:

```
starttime 2100
```

(6) multitime Multiple startups

When you start up the job net a number of times in the same day, specify the execution times in the HHMM (hour and minute) format in four bytes. You can specify up to six execution times. If only one execution time is specified, it is registered as a single startup. Multiple startups cannot be specified for the job net that belongs to a group.

Example of multiple startups setting

To start up at 8:40 a.m., 1:00 p.m. and 5:30 p.m.:

```
multitime 0840,1300,1730
```

(7) intervalstart Fixed interval

When you start up the job net at regular intervals, specify the execution time, end time, and execution interval. The execution interval can be specified between 1 and 240 (minutes). When specifying the execution time and end time, make sure the registered "day change time". The time span specified between the execution time and end time cannot cross over the day change time. Fixed intervals cannot be specified for the job net that belongs to a group.

Example of fixed interval setting

To execute the job net at an interval of 60 minutes between 10:00 a.m. and 5:30 p.m.:

```
intervalstart 1000,1730,60
```

In this example, the last job net execution time is 5:00 p.m.

(8) iplstart At server startup

Specify "ON" to start the job net when Jobscheduler service/daemon is activated. This operand cannot be specified for the job net that belongs to a group.

(9) msgonly Message-driven

Specify "ON" to start the job net only when a certain message event is generated. This operand cannot be specified for the job net that belongs to a group.

(10) msgonlymode Action at message startup

In case that the awaiting message event is generated multiple times in the same day, specifies whether or not the job net is executed every time the message event occurs. If omitted, "EVERYTIME" is defaulted. When the execution condition of job net is set to other than "Start only when message event has occurred", this option is ignored. When "ONCE" is specified, "ORD" and "BOT" cannot be specified with the messagemode option.

EVERYTIME

The job net is started every time the awaiting message event occurs.

ONCE

The job net is started only the first time the awaiting message occurs, even if the awaiting message is generated multiple times a day.

(11) msgstarttime Scheduled start time

This operand specifies the scheduled start time when the job net is the message-driven job net.

Specify this operand in the HHMM (hour minute) format and in four bytes. If omitted, no scheduled start time is set. When the execution condition of job net is set to other than "Start only when message event has occurred", this operand is ignored.

It can be specified only when "Action at message startup" is set to "ONCE".

(12) stoptime Stop time

Specifies the time to terminate the job net when the "Execution attribute" of the job net is set to "INTS". Specify this operand in the HHMM (hour minute) format, and in four bytes. If omitted, the stop time is not set.

Example of stop time setting

To stop the job net at 5:00 p.m.:

```
stoptime 1700
```

(13) alarmtime Estimated end time

Specifies the estimated end time of job net.

Specify this operand in the HHMM (hour minute) format and in four bytes. If omitted, no estimated end time is set. The time span specified between the execution time and the estimated end time cannot cross over the day change time. Note that it cannot be specified when the job net execution condition is not "single startup".

(14) estimatetime Estimated processing time

This operand specifies the estimated runtime of job net.

Specify it in a numeric string in the range of 1 and 9999 (minute). If omitted or "0" is specified, no estimated processing time is set. This operand cannot be specified when the execution attribute of job net is specified to "INTS".

(15) powermode Action at power off

Specify the action to be taken when the Jobscheduler service/daemon is not activated at the job net's scheduled execution time. If omitted, this operand defaults to "RESET". The value for "Action at power off" is ignored if an execution condition other than "single startup" is specified for the job net or if the job net belongs to a group.

RESET

The job net is not started even when the Jobscheduler service/daemon is activated.

SET

The job net is started when the Jobscheduler service/daemon is activated.

(16) iplmode Action at server startup

Specifies whether to make job net start each time the Jobscheduler service/daemon is activated at server startup. The default is "EVERYTIME".

EVERYTIME

The job net is started every time the Jobscheduler service/daemon is activated.

ONCE

The job net is started only the first time Jobscheduler service/daemon is activated, even if Jobscheduler services/daemons are started multiple times a day.

(17) execattr Execution attribute

Specifies a host name to which job net execution is requested as the execution attribute.

[Windows]

The following are a list of execution attributes. If omitted, the execution attribute is set to "JES," and execution is requested to the Job Execution Control.

JES

Specify this when requesting job net execution to the Job Execution Control.

INTS

Specify this when requesting job net execution to Interstage.

[UNIX]

The following are a list of execution attributes. If omitted, the execution attribute is set to "Normal attribute", and execution is requested to the Jobscheduler.

NORMAL

Specify this when requesting job net execution to Jobscheduler with the Normal attribute. This is the default when the execution attribute is omitted.

JES

Specify this when requesting job net execution to the Job Execution Control.

INTS

Specify this when requesting job net execution to Interstage.

PAIM

Specify this when requesting job net execution to PowerAIM.

(18) carryovermode Instruction

Specifies how to handle the carried-over job nets. This operand is available only when the Master Schedule Management environment is configured and daily schedule management is performed. If omitted, the default is "CANCEL".

CANCEL

Instructs the "Cut off" action. Automatically discontinues the operation on the job net when the job net is carried over.

OVER

Instructs the "Wait until exit" action. When the job net is carried over from the previous day, the same job net to be executed that day will be scheduled when the carried over job net is completed.

PARALLEL

Instructs the "Run concurrently" action. When the job net is carried over from the previous day, the same job net to be executed that day will be scheduled when the day change time is reached, regardless of the completion of the carried over jot net.

ACTION

Instructs the "Handle manually" action. The carried over job net becomes "Paused" status.

(19) copystart Copy and start

Specifies whether to perform copy and startup. The default is "OFF".

ON

Performs copy and start.

OFF

Does not perform copy and start.

(20) jobnetsavenumber Number of saved copy destination job nets

Specifies the number of saved copy destination job nets that were copied and started. Specify a number from 1 to 999. This operand cannot be omitted when "Copy and start" is ON.

(21) defaulthost Default host name

Specifies a default host name of the job whose execution is requested to the Job Execution Control, within 64 bytes. If the host name is omitted in the control statement of the job, the job is submitted to this default host.

When you need to request job execution to the second candidate execution server because the first candidate execution server goes down, the servers should be specified in the form "host name of first candidate, host name of second candidate". A local host name (connection destination host) cannot be specified as the host name of second candidate. The synopsis is shown below.

```
defaulthost host name of first candidate, host name of second candidate
```

 Note

....

Cautions when you specify the first candidate and second candidate execution servers

- A comma (,) is recognized as a separator used between the first candidate host and the second candidate host. Therefore, a host name should not contain a comma.

- The total number of bytes of host name is 64 bytes or less, which includes the host name of first candidate, the host name of second candidate, and a comma.

....

(22) standnet Base job net declaration

Specify this operand to declare that the job net is used as the base when startup days are generated. If omitted, the default is "RESET".

RESET

Specifies that the job net is not used as the base job net when generating startup days.

SET

Specifies that the job net is used as the base job net when generating startup days.

(23) execpattern Schedule pattern name

Specify the already registered schedule pattern name when using the schedule pattern. Up to four schedule pattern names can be specified for each job net. When multiple schedule patterns are specified, delimit with a space.

(24) execpatmode Method to combine the schedule patterns

Choose one of the following parameters to specify a method of combining schedule patterns when more than one schedule pattern is specified. If omitted, the default is "AND".

AND

Sets the days as startup days if they are set as startup days in all the schedule patterns.

OR

Sets the days as startup days if they are set as startup days in at least one of the schedule patterns.

(25) basenet Base job net name

When using the base job net, specify a job net name that is already registered as the base job net, which should be in the same project. This operand cannot be specified if the job net itself declares as the base job net in this statement.

(26) execday Startup day setting

Specify the scheduled startup days for the job net. When setting the startup day of MM (month) DD (day) in YYYY (year), the format should be "MM,DD,YYYY". More than one startup day can be set. The startup days can be set for three years: for the current year, the next year, and the year after next.

Example of startup day setting

To set October 1 and 10 in 2010 as startup days:

```
execday 10,01,10,2010
```

To set October 1 and December 10 in 2010 as startup days:

```
execday 10,01,2010
execday 12,10,2010
```

(27) cancelday Startup day setting cancellation

Specify the date for which you want to cancel the execution setting. Specify this operand in the same format as the one in the "Startup day setting" operand.

(28) jobschprintr Startup day bit pattern

Specify the bit pattern data for the Startup day setting expanded by using the jobschprint -r command. (If any value other than those expanded by the jobschprint command is specified, operation cannot be guaranteed. Also, operation is not guaranteed for any value other than those resulting from execution of different version of the jobschprint command. A different year from the one obtained by the jobschprint command cannot be specified in this operand.) If the bit pattern data is specified, you can make settings for three years in the same startup days as the job net expanded using the jobschprint -r command. Immediately after setting this operand, recalculation for startup days by using other startup days setting operands (such as monthly calendar, annual calendar, base job net name, schedule pattern, calendar name, holiday shift, offset, enable/ disable period, etc.) cannot be performed. Also note that even if the startup days are changed by this operand, the changes are not applied to the startup days of other job nets that use this job net as the base job net.

(29) annuallydate Date setting for every year

When you use an annual calendar, specify dates when the job net is executed for every year. When you set MM (month) DD (day) as a startup day, specify in the "MMDD" format. More than one startup day can be set.

(30) annuallyweek Day of the week setting for every year

When you use an annual calendar, specify the day(s) of the week on a monthly basis in which the job net is executed. It is only one month that can be specified with one operand. To set more than one month, specify this operand repeatedly. To specify this operant, enter a month as a number between 1 and 12 that is followed by the first three letters of the day of the week. The format should look similar to the following:12,SUN,MON,TUE,WED,THU,FRI,SAT.

(31) monthlydate Date setting for every month

When you use a monthly calendar, specify the date(s) in which the job net is executed for every month. To set DD (day) of every month as the startup day, specify "DD". More than one startup day can be set. To execute the job net at the end of every month, enter "LAST".

(32) monthlyweek Day of the week setting for every month

When you use a monthly calendar, specify the day(s) of the week on a weekly basis in which the job net is executed for every month. It is only one week that can be specified with one operand. To set more than one week, use this operand repeatedly. To specify this operand, enter the ordinal number of the week (where 1 - 5 represents the first through the fifth week, 0 indicates that every week is specified, and 6 represents the last week of the month) with a number between 0 and 6 that is followed by the first three letters of the day of the week. The format should look similar to the following: 5,SUN,MON,TUE,WED,THU,FRI,SAT.

(33) bnsmode Criteria of business day basis

When "Business days" is selected in the Basic information, specify either "Month's start date" or "Month's end date" as the selection criteria of business days.

START

Counts the business days from the Month's start date.

END

Counts the business days back from the Month's end date.

(34) bnsbase Month's start date of business day basis

When "Business days" is selected in the Basic information, enter a value of the "Month's start date". "0" is invalid.

1 to 28

Indicates 1st to 28th of month.

LAST

Indicates the end of month.

(35) bnsday Business days

When "Business days" is selected in the Basic information, specify business days. To specify more than one business days, use a comma (,) as a separator. "0" is invalid.

1 to 31

The business days that are specified by counting from the 1st business day (the first business day of the month) using a range from 1 to 31. This parameter cannot be specified if "END" is set to the bnsmode operand.

-1 to -31

The business days that are specified by calculating back from the -1st business day (the last business day of the month) using a range from -1 to -31. This parameter cannot be specified if "START" is set to the bnsmode operand.

(36) bnslimit Set only business days of the month as startup days

When "Business days" is specified in the Basic information, specify whether "Set only business days of the month as startup days" is set to ON or OFF. If the bnsmode operand is specified, the default is "ON".

ON

"Set only business days of the month as startup days" is selected.

OFF

"Set only business days of the month as startup days" is not selected.

(37) offsetmode Offset

Specify the method for counting the number of days shifted from the standard date by selecting any of the following parameters. If omitted, the default is "EXECDAY".

EXECDAY

Counts the number of days to be shifted based on working days on the calendar.

DAY

Counts the number of days to be shifted based on the absolute number of days, regardless of holidays.

(38) offsetday Number of offset days

Specify the number of days to be shifted from the standard date between -99 and 99. If omitted, the default is "0".

(39) offsetholiday Sets the base day as a working day if it falls on a holiday

Specify what happens if the base day is a holiday when counting the number of days shifted from the base day. If omitted, the default is "OFF".

This operand cannot be specified if "DAY" has been specified for offset.

ON

Sets the base day as a working day even if it falls on a holiday, and it is set as the base day for the number of days shifted.

OFF

Sets the base day as a non-working day if it falls on a holiday, and it is not set as the base day for the number of days shifted.

(40) holidayshift Holiday shift

If the startup days that are registered or set in the schedule pattern fall on holidays, select a method to handle them by selecting one of the following parameters:

BEF

Shifts the job net startup day to the preceding day.

AFT

Shifts the job net startup day to the next day.

NOT

Recognizes the day as a holiday, so does not execute the job net.

IGN

Recognizes the day as a startup day, so starts the job net.

(41) shiftlimit Number of maximum shifts

Set the number of days by specifying a number between "0" and "99", on which execution is suspended because the Holiday shift is performed on the consecutive holidays. If omitted, the default is "0".

(42) holidaycalendar Calendar name

Specify the name of the calendar to be used as the base when startup days are generated. You can specify up to five calendar names per one job net.

To specify more than one calendar, use a comma (,) as a separator.

(43) holidaymode Condition for combining calendars

When you specify a method of combining multiple calendars, select either of the following parameters: If omitted, the default is "AND".

AND

Sets the days as holidays if they are set as holidays in all calendars.

OR

Sets the days as holidays if they are set as holidays in at least one of the calendars.

(44) enablestart Start date of validity period/enableend End date of validity period

Specify the operating period of the job net in the "MM,DD,YYYY" format. If the start date of validity period is omitted, the job net operation is enabled from the current date until the end date of validity period. If the end date of validity period is omitted, the job net operation is enabled from the start date of validity period onward. If both the start date and end date of validity period are omitted, the job net operation is enabled on every day.

(45) disablestart Start date of invalidity period/disableend End date of invalidity period

Specify the non-operating period for the job net in the "MM,DD,YYYY" format. If the start date of invalidity period is omitted, the job net operation is disabled from the current day until the end date of invalidity period. If the end date of invalidity period is omitted, the job net operation is disabled from the start date onward. If both the start date and end date of invalidity period are omitted, the job net operation is enabled on every day.

(46) tstarttime Single startup (temporary change)

When you execute the job net once a day, specify the execution time in the HHMM format (hour and minute) in four bytes. This setting is valid from the start date of temporary change until the end date of temporary change.

(47) tmultitime Multiple startups (temporary change)

When you execute the job net a number of times in the same day, specify the execution times in the HHMM format (hour and minute) within four bytes. You can specify up to six execution times. This setting is valid from the start date of temporary change until the end date of temporary change. If only one execution time is specified, it is registered as a single startup (temporary change). Multiple startups (temporary change) cannot be specified for a job net that belongs to a group.

(48) tintervalstart Fixed interval (temporary change)

When you execute the job net at regular intervals, specify the execution time, termination time and execution interval. The execution interval can be specified between 1 and 240 (minutes). The time span specified between the execution time and termination time cannot cross over the day change time. This setting is valid from the start date of temporary change until the end date of temporary. Fixed intervals (temporary change) cannot be specified for a job net that belongs to a group.

(49) tiplstart IPL startup (temporary change)

Set this to "ON" to start the job net when the Jobscheduler service/daemon is started. This setting is valid from the start date of temporary change to the end date of temporary change. IPL startup (temporary change) cannot be specified for a job net that belongs to a group.

(50) tmsgonly Message-driven (temporary change)

Set this to "ON" to start the job net only when a particular message event is generated. This setting is valid from the start date of temporary change to the end date of temporary change. Message-driven (temporary change) cannot be specified for a job net that belongs to a group.

(51) tmsgonlymode Action at message startup (temporary change)

In case that the awaiting message event is generated multiple times a day, specify whether or not the job net should be executed every time the awaiting message event occurs. If omitted, the default is "EVERYTIME".

This specification is valid from the start date of temporary change to the end date of temporary change.

When the execution condition of job net is set to other than "Start only when message event has occurred", this specification is ignored.

EVERYTIME

The job net is started every time the awaiting message event occurs.

ONCE

The job net is started only the first time the awaiting message event occurs, even if the awaiting message is generated multiple times a day.

(52) tmsgstarttime Scheduled start time (temporary change)

Specifies the scheduled start time for the job net for message startup.

Specify this operand in the HHMM (hour minute) format and in four bytes. If omitted, no scheduled start time is set.

This specification is valid from the start date of temporary change to the end date of temporary change.

When the execution condition of job net is set to other than "Start only when message event has occurred", this operand is ignored.

It can be specified only when "Action at message startup" is set to "ONCE".

(53) tstoptime Stop time (temporary change)

Specifies the time to stop when the "Execution attribute" of the job net is "INTS". Specify this operand in the HHMM (hour minute) format, and in four bytes. If omitted, the stop time is not set. This setting is valid from the start date of temporary change to the end date of temporary change.

(54) talarmtime Estimated end time (temporary change)

Specify the estimated end time (temporary change) of job net.

Specify this operand in the HHMM (hour minute) format and in four bytes. If omitted, no estimated end time is set. The time span specified between the execution time and estimated end time cannot cross over the day change time. This setting is valid from the start date of temporary change to the end date of temporary change.

(55) tpowermode Action at power off (temporary change)

Specify the action to be taken when the Jobscheduler service/daemon is not activated when the job net's scheduled startup time is reached. If omitted, it defaults to "RESET". This setting is valid from the start date of temporary change to the end date of temporary change. The value for "Action at power off (temporary change)" is ignored if an execution condition other than "single startup (temporary change)" is specified for the job net or if the job net belongs to a group.

RESET

Does not start the job net even if the Jobscheduler daemon/service is activated.

SET

Starts the job net when the Jobscheduler service/daemon is activated.

(56) tiplmode Action when activated (temporary change)

Specifies whether to start the job net with IPL startup (temporary change) every time the Jobscheduler service/daemon is activated. If omitted, the default is "EVERYTIME". This setting is valid from the start date of temporary change to the end date of temporary change.

EVERYTIME

The job net is started every time the Jobscheduler service/daemon is activated.

ONCE

The job net is started only the first time, even if the Jobscheduler service/daemon is started more than once a day.

(57) temporarystart Start date of temporary change/temporaryend End date of temporary change

Specify the temporary change period of job net in the "MM,DD,YYYY" format. If only the start date of temporary change is specified, only this date can be the target of temporary change.

(58) messagemode Message condition

When you specify this operand, combine the message event waiting condition (AND/OR), execution condition (ORD/COG/BOT/UNC), and the condition with the execution time (ANDTIME/ORTIME). If this operand is omitted, it is defaulted to "AND,UNC,ORTIME" as its parameter. "Valid during job net execution" ("ORD" and "BOT") cannot be specified for the "execution condition" if "ANDTIME" is specified as the "condition with the execution time" or if the job net belongs to a group. When the job net belongs to a group, the value specified for the "condition with the execution time" is ignored.

AND

Starts the job net if all of the registered message events are generated.

OR

Starts the job net if at least one of the registered message events is generated.

ORD

"Valid during job net execution" is specified. Message events generated while the job net is being executed become valid as the execution condition. Therefore, as soon as the job net is terminated, it is restarted as many times as the message events were generated during execution. The message events generated during execution of the job net become invalid as the execution condition if this parameter is not specified.

COG

"Valid only on startup days" is specified. Starts the job net only if the specified message event is generated on a selected startup days.

BOT

"Valid during job net execution & Valid only on startup days" is specified. If the specified message event is generated on a selected startup days, the job net is executed as many times as the number of message events generated.

UNC

Specify this parameter when the execution condition is not set.

ANDTIME

Starts the job net when the scheduled startup time is reached and the awaiting message event occurs. This parameter is valid only when the startup condition of the job net is set to "single startup" ("single startup (temporary change)" during a temporary change period).

ORTIME

Starts the job net either when the scheduled startup time is reached or the awaiting message event is generated.

(59) carryover Carryover processing

Specifies that processing be carried over when the day change time is reached. This can be specified if the job net waits for the startup time to arrive ("ANDTIME" is specified in the messagemode operand), or if the startup conditions are for the job net to start only when a certain message event has occurred ("ON" is specified in the msgonly operand). The default value is "RESET".

RESET

Carrying over the job net process is not performed when the day change time is reached.

SET

Carrying over the job net process is performed when the day change time is reached.

(60) messagename Message event name

Specify a message event that triggers startup of the job net, using up to 12 bytes. The message event name cannot contain a comma (,), a colon (:), a blank, and a user-defined character. Up to 70 message event names per one job net can be registered.

Example of message event setting

To register three message events: MSG1, MSG2 and MSG3:

```
messagename MSG1,MSG2,MSG3
```

(61) job Number of jobs within a job net

Specify the number of jobs that make up the job net. This can be set between 1 and 255.

Note that you can specify only one job for the job net having the Interstage attribute. If omitted, the job control statement cannot be written.

(62) operate Operation

Indicates the status resulting from the operation as either paused or disabled. If this parameter is omitted, the status of the job net will be "NONE" when job net information is registered, or the status will not change when job net information is changed.

PAUSE

Paused

DISABLE

Disabled

NONE

When job net information is registered, the status of the job net will be "waiting". To change the status of the job net, either perform a [Continue] operation if the job net is paused or an [Enable] operation if the job net is disabled. The status of the job net cannot be changed unless it is "paused" or "disabled".

Cautions

If you change a job net by using a control statement that includes any of the following operands, the startup days will be recreated.

In response to this change, the startup days specified in the [Startup days] window, **Startup Days(Preview)** window or [Application Plan] window may be overwritten by the content of the control statement. Additionally, if the changed job net is the base job net, the startup days of other job nets that use the startup days of the base job net will also be recreated.

```
execpattern, basenet, execday, cancelday, jobschprintr, annualydate, annualyweek,
monthlydate, monthlyweek, bnsmode, bnsbase, bnsday, bnslimit, offsetmode, offsetday,
holidaycalendar, holidaymode, holidayshift, shiftlimit, enablestart, enableend,
disablestart, disableend
```

Example of Job Net Control Statement

The following provides a sample of job net and its control statements when it is configured as the following diagram.
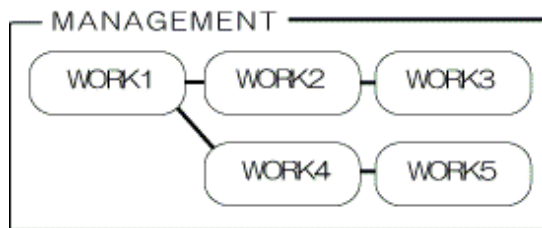

```
NET
jobnetname KANRI                        <- Job net name of MANAGEMENT
jobnetcomment Business Management       <- Job net comment of MANAGEMENT
starttime 0850                          <- Starts at 8:50
execpattern 20thEveryMonth              <- Schedule pattern name,"20th EverymMonth" is
                                           set as the job startup day.
holidaycalendar SYSTEM_CALENDAR         <- SYSTEM-CALENDAR is used as the calendar.
holidayshift BEF                        <- If the holiday in the calendar falls on job
                                           net startup day, the job net
                                           startup day shifts to one day before.
job 5                                   <- Five job control statements are defined.
;
JOB
jobname WORK1.bat
jobnumber 1
;
JOB
jobname WORK2.bat
jobnumber 2
pre_job 1
;
JOB
jobname WORK3.bat
jobnumber 3
pre_job 2
;
JOB
jobname WORK4.bat
jobnumber 4
pre_job 1
;
JOB
jobname WORK5.bat
jobnumber 5
pre_job 4
;
```

4.9.2 Job Control Statement

This is a control statement for setting registration information of jobs. When defining a job control statement, ensure that you specify a job net control statement immediately before the job control statement.

Synopsis

Job control statement for job nets with the Job Execution Control attribute

```
JOB                                                    ---------- (1)
jobname Job                                            ---------- (2)  (*)   (Note 1)
jobnumber Job number                                   ---------- (3)  (*)   (Note 2)
jobattr Job attribute                                  ---------- (4)        (Note 4)
jobname_jes Job name in Job Execution Control      ---------- (5)       (Note 8)
masterlinkjobnet Master linked job net name          ---------- (6)       (Note 17)
jobparam Parameter                                     ---------- (7)        (Note 8)
directory Directory                                    ---------- (8)        (Note 8)
jobcomment Comment                                     ---------- (9)        (Note 8)
req_host Request host name                             ---------- (10)       (Note 8)
queuename Queue name                                   ---------- (11)       (Note 8)
normallimit Upper limit of normal completion code    ---------- (12)       (Note 8)
endcodeignore Ignore completion code                  ---------- (13)       (Note 8)
nooutput Do not save output                           ---------- (14)       (Note 8)
restartpoint Restart point                            ---------- (15)       (Note 5)
pre_job Preceding job number                          ---------- (16)
pre_job_endcode Condition concerning completion
            code of preceding job                     -------- (17)        (Note3)(Note 4)
execdaymode Valid only on startup days         --- (18) (Note12)(Note13)(Note 14)(Note
15)
execuser Execution user name                          ---------- (19)       (Note 8)
resource Resource name                                ---------- (20)       (Note 8)
resource_attr Resource allocation                    ---------- (21)      (Note 7)(Note 8)
stdoutfile Standard output file name                  ---------- (22)       (Note 8)
stderrfile Standard error output file name            ---------- (23)       (Note 8)
errtostdfile Output standard error output to standard output file   --------- (24)
(Note8)(Note10)
jobfile_trans Transmission of execution command    ---------- (25)       (Note 8)
priority Execution priority                           ---------- (26)       (Note 8)
predictiontime Estimated processing time              ---------- (27)       (Note 8)
limittime Cut-off execution time                      ---------- (28)       (Note 8)
netvar_prefix Job net variable prefix                 ---------- (29)       (Note 8)
envname Environmental variable name                   ---------- (30)       (Note 8)
envvalue Environmental variable value               ---------- (31)      (Note 8)(Note 9)
joblogfile Joblog filename                            ---------- (32)       (Note 11)
stepname Start step name                              ---------- (33)       (Note 11)
codelaststep Use last executed stop completion code
          as job completion code                      ---------- (34) (Note 11)
operate Operation                                     ---------- (35)       (Note 8)
jobicon Icon type                                     ---------- (36)       (Note
6)iconposition Icon position                          ---------- (37)       (Note 16)
;
```

Note 1:

   The jobname operand should be specified immediately after the JOB keyword.

Note 2:

   The jobnumber operand should be specified immediately after the jobname operand.

Note 3:

   To specify the pre_job_encode operand, at least one preceding job number should be specified for

   the pre_job operand.

Note 4:

The pre_job_endcode cannot be specified with the jobattr operand simultaneously.

Note 5:

A restart point can be set for only one of the jobs making up a job net.

Note 6:

Specify a value other than "GLJO," "GLBO" or "INTS". [Windows]. Specify a value other than "INTS". [UNIX]

Note 7:

The resource_attr operand should be specified immediately after the resource operand.

Note 8:

This operand cannot be specified if the child job net "JNET" or the linked job net "LNET" is specified as the icon type.

Note 9:

The envvalue operand should be specified immediately after the envname operand.

Note 10:

This operand cannot be specified with the stderrfile operand simultaneously.

Note 11:

This operand can be specified only when the Batch framework job "BJOB is specified as the icon type.

Note 12:

This operand can be specified only when the child job net "JNET" or the linked job net "LNET" is specified as the icon type.

Note 13:

This operand cannot be specified if "RECOVERYONLY" or "RECOVERYRERUN" is specified with the jobattr operand.

Note 14:

This operand cannot be specified if "ON" is specified with the restartpoint operand.

Note 15:

This operand cannot be specified if any value is specified with the pre_job_endcode operand. Additionally, if the pre_job_endcode operand is specified for a subsequent job, this operand cannot be specified.

Note 16:

This operand must be specified for all jobs that make up the job net.

Note 17:

This operand must be specified when the linked job net "LNET" is specified as the icon type.

*:

This must be specified.

Job control statement for job nets with the Interstage execution attribute

```
JOB                                               ---------- (1)
jobnumber Job number                              ---------- (3)   (*)   (Note 2)
workunitname Workunit name                        ---------- (38)  (*)   (Note 1)
systemname System name                            ---------- (39)
jobcomment Comment                                ---------- (9)
req_host Request host name                        ---------- (10)
queuename Queue name                              ---------- (11)
resource Resource name                            ---------- (20)
resource_attr Resource allocation                 ---------- (21)       (Note 4)
normallimit Upper limit of normal completion code ---------- (12)
```

```
endcodeignore Ignore completion code                        ---------- (13)
operate Operation                                           ---------- (35)
jobicon Icon type                                           ---------- (36)  (*)  (Note 3)
iconposition Icon position                                  ---------- (37)      (Note 5)
;
```

Note 1:

The workunitname operand should be specified immediately after the JOB key word.

Note 2:

The jobnumber operand should be specified immediately after the workunitname operand.

Note 3:

In this operand, "INTS" must be specified. If an icon type other than "INTS" is specified for the job net having the Interstage attributes or if "INTS" is specified for a job net having the attributes other than Interstage, this command is terminated abnormally.

Note 4:

The resource_attr operand should be specified immediately after the resource operand.

Note 5:

This operand must be specified for all jobs that make up the job net.

*:

This must be specified.

Job control statement for job nets with the PowerAIM execution attribute [UNIX]

```
JOB                                                             ---------- (1)
jobnumber Job number                                           ---------- (3)  (*) (Note 2)
jobattr Job attribute                                          ---------- (4)      (Note 4)
systemname Business system name                                ---------- (40)  (*) (Note 1)
workunitname Workunit name                                     ---------- (38)  (*) (Note 1)
jobcomment Comment                                             ---------- (9)
normallimit Upper limit of normal completion code              ---------- (12)
endcodeignore Ignore completion code                           ---------- (13)
nooutput Do not save output                                    ---------- (14)
restartpoint Restart point                                     ---------- (15)      (Note 5)
pre_job Preceding job number                                   ---------- (16)
pre_job_endcode Condition concerning completion code of preceding job --- (17)  (Note 3)
(Note 4)
operate Operation                                              ---------- (35)
jobicon Icon type                                              ---------- (36)
iconposition Icon position                                     ---------- (37)      (Note 6)
;
```

Note 1:

The systemname operand and workunitname operand should be specified immediately after the JOB keyword in a row.

Note 2:

The jobnumber operand should be specified immediately after the workunitname operand.

Note 3:

To specify the pre_job_encode operand, at least one preceding job number should be specified for the pre_job operand.

Note 4:

The pre_job_endcode cannot be specified with the jobattr operand simultaneously.

Note 5:

A restart point can only be set for one of the jobs making up a job net.

Note 6:

This operand must be specified for all jobs that make up the job net.

*:

This must be specified.

Job control statement for job nets with the Normal execution attribute [UNIX]

```
JOB                                                          ---------- (1)
jobname Job                                                  ---------- (2)
(*)  (Note 1)
jobnumber Job number                                         ---------- (3)
(*)  (Note 2)
jobattr Job attribute                               ---------- (4)    (Note 4)
directory Directory                                          ---------- (8)
jobcomment Comment                                           ---------- (9)
normallimit Upper limit of normal completion code        ---------- (12)
endcodeignore Ignore completion code                      ---------- (13)
nooutput Do not save output                               ---------- (14)
restartpoint Restart point                       ---------- (15)    (Note 5)
pre_job Preceding job number                              ---------- (16)
pre_job_endcode Condition concerning completion code of preceding job ----- (17) (Note 3)
(Note 4)
operate Operation                                         ---------- (35)
jobicon Icon type                                         ---------- (36)
iconposition Icon position                          ---------- (37)
(Note 6)
;
```

Note 1:

The jobname operand should be specified immediately after the JOB keyword.

Note 2:

The jobnumber operand should be specified immediately after the jobname operand.

Note 3:

To specify the pre_job_endcode operand, at least one preceding job number is specified for the pre_job operand.

Note 4:

The pre_job_endcode operand cannot be specified in conjunction with the jobattr operand.

Note 5:

A restart point can be specified for only one of the jobs making up a job net.

Note 6:

This operand must be specified for all jobs that make up the job net.

*:

This must be specified.

About This Control Statement

(1) JOB

This is the keyword that indicates the beginning of a job control statement.

(2) jobname Job

The name of the batch program, command or program that will be executed as a job must be no longer than 260 bytes, including the pathname. The file name and parameters together must not exceed 300 bytes in length.

$ `

If a file name specified in a UFT-8 environment (or, in the case of network jobs, an environment where either the submitting server or the execution server uses UTF-8 codes) contains Japanese characters, be sure to specify a job name that does not include Japanese character codes.

If the child job net "JNET" is specified as the icon type, the name of a child job net must be specified.

If the batch framework job "BJOB" is specified as the icon type, specify the batch job definition within 64 bytes using lowercase letters (a-z), uppercase letters (A-Z), numbers (0-9), hyphens (-) and/or underscores (_). Also, the name must start with a lowercase letter (a-z) or uppercase letter (A-Z).

If the linked job net "LNET" is specified as the icon type, specify the job net name of the linked job net within 50 bytes. Ensure that the name does not duplicate the names of other job nets or linked job nets. The following characters cannot be used: /?*;,\"<>|:'., space, or any user-defined character.

Do not specify the qsub command of Job Execution Control or a batch file or shell script that calls the qsub command as a job.

(3) jobnumber Job number

Allocate numbers corresponding to the jobs by using numbers ranging from 1 to 255. Unique numbers should be assigned, according to the order in which the jobs are executed. They should be allocated in ascending order from the preceding to the subsequent jobs.

(4) jobattr Job attribute

Specifies the job as a recovery job. The preceding job numbers must be specified. Also, this operand specifies whether or not to rerun the preceding jobs of the recovery job.

RECOVERYONLY

Specifies this job as the recovery job. Even if this recovery job is completed normally, the preceding job is not rerun.

RECOVERYRERUN

Specifies this job as the recovery job. If this recovery job is completed normally, the preceding job is rerun.

(5) jobname_jes Job name in Job Execution Control

Specify job names within 64 bytes each so that they can be distinguished from each other. Job names are not allowed to contain spaces, any user-defined characters, or the following characters:

/ ? * ; , \ " < > | : ( ) ' . $ `

cannot be used for the first character.

An ampersand (&) cannot be used for the job name of network jobs and Distributed Execution jobs.

The job name defined here can be also used as the job name used in the Job Execution Control.

(6) Master linked job net name

If the linked job net "LNET" is specified as the icon type, specify the job net name of the master linked job net specified as the reference of the linked job net.

(7) jobparam Parameter

Specify the character string (parameter) as a command argument with 1024 bytes or less.

If the batch framework job "BJOB" is specified as the icon type, this cannot be specified.

(8) directory Directory

For the job having the Job Execution Control attribute, specify the directory name to which the qsub command of the Job Execution Control is to be executed, with 64 bytes or less. The name should not contain spaces or user-defined characters.

If the batch framework job "BJOB" is specified as the icon type, this operand cannot be specified.

If the attribute of job is the Normal attribute, specify the directory in which the job is to be executed. [UNIX]

(9) jobcomment Comment

Specify a comment on the job in 64 bytes or less. The comment cannot contain the user-defined characters.

(10) req_host Request host name

Specify the name of the host to which the job is submitted with 64 bytes or less. If omitted, the job is submitted to the default host specified in the job net.

If the batch framework job "BJOB" is specified as the icon type, this operand cannot be specified.

Specify a second candidate execution server which is used in case the first candidate execution server goes down. The servers should be specified in the form "host name of first candidate, host name of second candidate". A local host name (connection destination host) cannot be specified as the host name of second candidate. The synopsis is shown below.

```
req_host host name of first candidate, host name of second candidate
```

Note

. . . .

Cautions when you specify the first candidate and second candidate execution servers

- A comma (,) is recognized as a separator used between the first candidate host and the second candidate host. Therefore, a host name should not contain a comma.

- The total number of bytes of host name is 64 bytes or less, which includes the host name of first candidate, the host name of second candidate, and a comma.

. . . .

(11) queuename Queue name

Specify the name of the queue to which the job will be submitted within 15 bytes using alphanumeric characters. If omitted, the job is submitted to default queue.

If the batch framework job "BJOB" is specified as the icon type, specify the job queue name within 15 bytes using lowercase letters (a-z), uppercase letters (A-Z), numbers (0-9), hyphens (-) and/or underscores (_). Also, the name must start with a lowercase letter (a-z) or uppercase letter (A-Z). This operand can be omitted.

(12) normallimit Upper limit of normal completion code

Specify the value to determine whether the job has been completed normally or terminated abnormally.

If the value of the job completion code is "0" or more and less than the value in the upper limit of normal completion code, the job is considered to have completed normally. If the value of the job completion code is greater than the value in the upper limit of normal completion code, the job is considered to have terminated abnormally.

A value between "0" and "255" can be specified. If omitted, the default is "0".

Note that the job status is differentially displayed if the status option [Validate pseudo-normal] is selected in the [Define Jobscheduler Startup Parameters] window. Namely, for the job with completion code 0, the status is displayed as "Normal completion", and for the job with completion code more than 1 and less than the upper limit of normal completion code, the status is displayed as "Pseudo-normal".

(13) endcodeignore Ignore completion code

Specify "ON" to ignore the completion code of the job. Setting "ON" causes the job to be handled as normally completed regardless of the completion code, and the subsequent job is executed.

(14) nooutput Do not save output

Specify "ON" when the standard output/error output is not referenced. Setting "ON" disables saving of the standard output/error output of the job.

If the batch framework "BJOB" is specified as the icon type, and when you do not want to save the standard output/standard error output in a job log file, specify "ON".

(15) restartpoint Restart point

Set "0" with this operand to start from this job when restarting the job net. The restart point can be specified for only one of the jobs making up the job net. Note that it cannot be specified for a recovery job.

(16) pre_job Preceding job number

Specify job numbers of preceding jobs. Use a comma as a separator to specify more than one preceding job number, which should be in the ascending order.

To use the "OR" condition to wait for the preceding job to end, first specify "OR," ("OR" followed by a comma) followed by the preceding job number. The following restrictions apply to the "OR," specification:

- You cannot specify "OR," for a recovery job.

- You cannot specify "OR," for a job whose execution condition is the completion code of the preceding job.

- You cannot specify a job other than an event wait job as the preceding job.

- You cannot set the specified preceding job as the preceding job of another job.

The jobs specified as preceding jobs must be defined beforehand. If this parameter is omitted, the top of jobs is defaulted.

(17) pre_job_endcode Condition concerning completion code of preceding job

This is specified to narrow down the execution conditions of subsequent jobs, according to the completion code of preceding job. You need to choose the synopsis between two types depending on the conditions "By value" or "By range" that you select.

An error will occur if the specified execution condition is inconsistent with the condition of the preceding job that normally completes. Ensure that the value greater than the one that you will specify is set to the upper limit of normal completion code of the preceding job or "Ignore completion code" is set for the preceding job.

For the conditions "By value":

The synopsis is shown below.

```
pre_job_encode Judgment operator, comparison value
```

The following six operators for checking are available: Specify an integer from 0 to 255 as the comparison value that is compared with the preceding job completion code.

EQ (EQual):

Equal to "comparison value"

NE (Not Equal):

Not equal to "comparison value"

GT (Greater Than):

Greater than "comparison value"

GE (Greater than Equal to):

Greater than or equal to "comparison value"

LT (Less Than):

Less than "comparison value"

LE (Less than Equal to):

Less than or equal to "comparison value"

For the conditions "By range":

The synopsis is shown below.

```
pre_job_encode Judgment operator 1, Comparison value 1, Judgment operator 2, Comparison
value 2
```

Specify an integer between 0 and 255 for "Comparison value 1" and "Comparison value 2", and specify a value smaller than "Comparison value 2" for "Comparison value 1".

Specify one of the following to "Judgment operator 1."

GT (Greater Than):

Greater than "comparison value 1"

GE (Greater than Equal to):

Equal to or greater than "comparison value 1"

Specify one of the following to "Judgment operator 2".

LT (Less Than):

Smaller than "comparison value 2"

LE (Less than Equal to):

Equal to or smaller than "comparison value 2"

(18) execdaymode Valid only on startup days

Specify this option if you want the job (job net) to start only if the specified message is generated on a selected startup days. To do so, specify "ON" with the "Valid only on startup days". If omitted, "Valid only on startup days" is not set.

ON

Specifies "Valid only on startup days".

(19) execuser Execution user name

Specify the name of the user who will execute the job, within 64 bytes.

If the batch framework job "BJOB" is specified as the icon type, this operand cannot be specified.

(20) resource Resource name

Specify the name of resource to be used within 64 bytes. English alphanumeric characters, "_" (under bar) and "-" (hyphen) are available. If omitted, resources will not be allocated. If the batch framework job "BJOB" is specified as the icon type, this operand cannot be specified.

(21) resource_attr Resource allocation

Specify how to allocate the specified resource (SH: shared use, EX: exclusive use). If omitted, the default is SH.

If the batch framework job "BJOB" is specified as the icon type, this operand cannot be specified.

(22) stdoutfile Standard output file name

Specify the name of the file to which standard output for the job is output, in a full path name within 256 bytes.

If the batch framework job "BJOB" is specified as the icon type, this operand cannot be specified.

(23) stderrfile Standard error output file name

Specify a full path name to the file to which standard error output is output, within 256 bytes.

If the batch framework job "BJOB" is specified as the icon type, this operand cannot be specified.

(24) errtostdfile Output standard error output to standard output file

Specify "ON" when outputting the job standard error output to standard output file.

If the batch framework job "BJOB" is specified as the icon type, this operand cannot be specified.

(25) jobfile_trans Transmission of execution command

Specify "ON" when you transfer the execution file to be started as the job to the Request host. If omitted, the execution file existing on the Request host is executed.

If the batch framework job "BJOB" is specified as the icon type, this operand cannot be specified.

(26) priority Execution priority

If the connection destination is the server based on Window version, specify the execution priority of the jobs between 4 (the highest priority) and 0 (the lowest priority).

If the connection destination is the server based on UNIX version, specify the execution priority of the jobs between 39 (the highest priority) and 0 (the lowest priority).

If omitted, the default is priority 0.

If the batch framework job "BJOB" is specified as the icon type, specify the execution priority in job queue using range from 63 (the highest priority) and 0 (the lowest priority). This can be omitted. If omitted, the default is 32.

(27) predictiontime Estimated processing time

Specify an estimated elapsed time for the job. This value can be specified between 1 and 35791394 (minutes). If omitted, no estimated elapsed time is set.

This operand is valid only for the job within a job net with the "Job Execution Control" attribute.

(28) limittime Cut-off execution time

Specify a value between 1 and 1666666 in minutes to limit the processing time of the job.

Make sure to specify the value larger than the one specified as the estimated processing time.

(29) netvar_prefix Job net variable prefix

To import the "NAME=VALUE" format line from the previous log into the job net variable, specify a job net variable prefix up to 13 bytes.
Only alphanumeric characters can be used. If this operand is omitted, the format line is not imported into the job net variable.

(30) envname Environmental variable name

Specify the name of the environmental variable when you specify the environmental variable at job execution. Specify it within 64 bytes using only alphanumeric characters and underscore (_).

If the batch framework job "BJOB" is specified as the icon type, specify the replaceable parameters in the batch job definition within 64 bytes using lowercase letters (a-z), uppercase letters (A-Z), numbers (0-9), hyphens (-) and/or underscores (_). Also, the name must start with a lowercase letter (a-z) or uppercase letter (A-Z).

Up to 32 environmental variable names can be specified.

(31) envvalue Environmental variable value

Specify the value of the environment variable name that is specified with the envname operand immediately before this operand with up to 256 bytes. The value cannot contain user-defined characters.

If the batch framework job "BJOB" is specified as the icon type, specify the replaceable parameters in the batch job definitions with up to 256 bytes. The value cannot contain user-defined characters.

Example of environmental variable specification

[Windows]

```
envname TEMP_PATH
envvalue d:\temp
```

```
envname DATA_PATH
envvalue d:\work1\data
```

[UNIX]

```
envname LD_LIBRARY_PATH
envvalue /usr/openwin/lib
envname OPENWINHOME
envvalue /usr/openwin
```

(32) joblogfile Joblog filename

Specify a job log file name in a full path name within 256 bytes.

The name cannot contain spaces or user-defined characters.

(33) stepname Start step name

Specify the Start step name with 64 bytes or less using lowercase letters (a-z), uppercase letters (A-Z), numbers (0-9), hyphens (-) and/or underscores (_). Also, the name must start with a lowercase letter (a-z) or uppercase letter (A-Z). This operand can be omitted.

(34) codelaststep Use last executed step completion code as job completion code

Use this option when you want to use the completion code of the last-executed job step as the job completion code. To do so, specify "ON". If omitted, the completion code of the last-executed job step does not become the job completion code.

If this option is omitted, the completion code of this job becomes the completion code defined in the batch framework job (either the maximum completion code or the completion code of the last executed job).

(35) operate Operation

Indicates the status resulting from the operation as either paused or disabled. The default value is "NONE".

PAUSE

Paused

DISABLE

Disabled

NONE

When job information is registered or changed, the status of the job will be "Waiting".

(36) jobicon Icon type

Specify an icon type to be added by using parameters. The correspondence between icon types and parameters are shown in the table below. And an example of synopsis is described later in this article. If this operand is omitted, the default is Execute command "CMND". For a job with the Interstage attribute, this operand cannot be omitted.

Icon type	Parameter
Execute command	CMND
Send message	MESG
Start service	SSTR
Stop service	SEND
File wait	FWAT
Backup cooperation	BKUP
Application execution (client)	EXAC
Application execution (server)	EXAS

Icon type	Parameter
File compression/expansion (client)	FPRC
File compression/expansion (server)	FPRS
File control (client)	FCLC
File control (server)	FCLS
File transfer (client)	FCPC
File transfer (server)	FCPS
Power on (client)	PONC
Shutdown (client)	POFC
Remote access control	RASC
Interstage	INTS
Batch framework job	BJOB
Remote access control (Connect)	RASS
Remote access control (Disconnect)	RASE
Rename file (client)	RNFC
Rename file (server)	RNFS
Delete file (client)	DLFC
Delete file (server)	DLFS
Create directory (client)	MKDC
Create directory (server)	MKDS
Delete directory (client)	RMDC
Delete directory (server)	RMDS
Check log file	CHKL
Wait for time	WATT
Check service status	CHKS
Job net	JNET
Linked job net	LNET
Wait for message event	CMSG
Wait until specified time	SLEP
Check file existence	JGFL
Check job net variable	JGVR
User definition 1 (to user definition 100)	U001 (to U100)

Statement Example

```
jobicon CMND
```

(37) iconposition Icon position

This option specifies position information for the job icon using the following format. The numbers indicate the coordinates for the grid position, taking the top left corner of the job flow as the origin (1,1).

```
Horizontal position (1 to 255), vertical position (1 to 255)
```

If this option is omitted, the position of the icon will be determined automatically. If the icon position is changed using the **Change Group** window after the job has been registered without specifying this option, the new icon position will be stored, and this control statement will be output using the jobschprint -r command.

(38) workunitname Workunit name

Specify the job (workunit name) to be requested to Interstage for execution, with 36 bytes or less.

(39) systemname System name

Specify the system name within 8 bytes.

(40) systemname Business system name [UNIX]

Specify the business system name as the request destination within 7 bytes.

Example of Job Control Statement

The following provides a sample control statement when a job net has the job configuration as shown below:
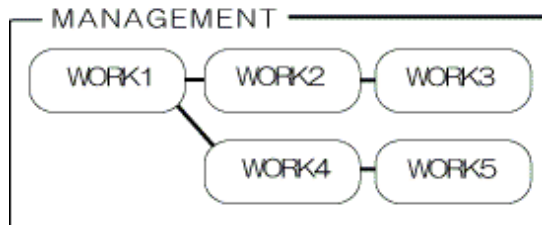

```
NET
jobnetname MANAGEMENT
jobnetcomment BusinessManagement
starttime 0850
execpattern 20thEveryMonth
holidaycalendar SYSTEM_CALENDAR      <- SYSTEM_CALENDAR is used as the calendar.
holidayshift BEF
job 5
;
JOB
jobname WORK1.bat
jobnumber 1
;
JOB
jobname WORK2.bat                <- Definition of WORK2 job
jobnumber 2                      <- Job number of WORK2
pre_job 1                        <- Job number of the preceding job of WORK2
;                                   (Preceding job number 1 is already defined as WORK1)
JOB
jobname WORK3.bat
jobnumber 3
pre_job 2
;
JOB
jobname WORK4.bat
jobnumber 4
pre_job 1
;
JOB
jobname WORK5.bat
```

```
jobnumber 5
pre_job 4
;
```

4.9.3 Startup Days Control Statement

This is a control statement for setting job net startup days.

Synopsis

```
SCH                                                    ---------- (1)
jobnetname Job net name                                ---------- (2)  (*)  (Note 1)
execpattern Schedule pattern name                      ---------- (3)  (**)
execpatmode Method to combine the schedule patterns    ---------- (4)       (Note 2)
basenet Base job net name                              ---------- (5)  (**)
execday Startup day setting                            ---------- (6)
cancelday Startup day setting cancellation             ---------- (7)
jobschprintr Startup day bit pattern                   ---------- (8)
annuallydate Date setting for every year               ---------- (9)  (**)
annuallyweek Day of the week setting for every year    ---------- (10) (**)
monthlydate Date setting for every month               ---------- (11) (**)
monthlyweek Day of the week setting for every month    ---------- (12) (**)
bnsmode Criteria of business day basis                 ---------- (13) (**)
(Note 5)
bnsbase Month's start date of business day basis       ---------- (14)(**)  (Note 6)
bnsday Business days                                   ---------- (15)(**)
bnslimit Set only business days of the month as startup days ---------- (16)(**)
offsetmode Offset                                      ---------- (17)
offsetday Number of offset days                        ---------- (18)
offsetholiday Sets the base day as a working day if it falls on a holiday -- (19)
holidayshift Holiday shift                             ---------- (20) (*)
shiftlimit Number of maximum shifts                    ---------- (21)     (Note 4)
holidaycalendar Calendar name                          ---------- (22)
holidaymode Condition for combining calendars          ---------- (23)     (Note 3)
enablestart Start date of validity period              ---------- (24)
enableend End date of validity period                  ---------- (24)
disablestart Start date of invalidity period           ---------- (25)
disableend End date of invalidity period               ---------- (25)
;
```

Note 1:

The jobnetname operand should be specified immediately after the SCH keyword.

Note 2:

The execpatmode operand should be specified immediately after the execpattern operand.

Note 3:

The holidaymode operand should be specified immediately after the holidaycalendar operand.

Note 4:

The shiftlimit operand should be specified immediately after the holidayshift operand.

Note 5:

The bnsmode operand cannot be specified with the offsetmode operand, the offsetday operand, or offsetholiday operand simultaneously.

Note 6:

This operand must be specified when the bnsmode operand is specified.

*:

This must be specified.

**:

Specify one of the following. This can be omitted

- execpattern

- basenet

- annuallydate and/or annuallyweek

- monthlydate and/or monthlyweek

- bnsmode, bnsbase, bnsday, bnslimit (The bnsmode operand and the bnsbase operand must be specified if these operands group is specified.)

About This Control Statement

(1) SCH

This is the keyword that indicates the beginning of a startup days control statement

(2) jobnetname Job net name

Specify the name of a job net for which startup days are to be set.

When you specify a message event control statement and a temporary changes control statement along with this statement, specify the same job net name across those statements.

(3) execpattern Schedule pattern name

When you apply a schedule pattern, specify the schedule pattern name that is already registered.

(4) execpatmode Method to combine the schedule patterns

Choose one of the following parameters to specify a method of combining schedule patterns when more than one schedule pattern is specified. If omitted, the default is "AND".

AND

Sets the days as startup days if they are set as startup days in all the schedule patterns.

OR

Sets the days as startup days if they are set as startup days in at least one of the schedule patterns.

(5) basenet Base job net name

When you use a base job net, specify the name of job net that is already registered as the base job net. This operand cannot be specified if the job net itself declares as the base job net in this statement.

(6) execday Startup day setting

Specify the scheduled startup days for the job net. When setting the startup day of MM (month) DD (day) in YYYY (year), the format should be "MM,DD,YYYY". More than one startup day can be set. If you set with the schedule pattern name in combination, you can set the startup days more easily.

Example of startup day setting

To set October 1 and 10 in 2010 as startup days:

```
execday 10,01,10,2010
```

To set October 1 and December 10 in 2010 as startup days:

```
execday 10,01,2010
execday 12,10,2010
```

(7) cancelday Startup day setting cancellation

Specify the date for which you want to cancel the execution setting. Specify this operand in the same format as the one in the "Startup day setting" operand.

(8) jobschprintr Startup day bit pattern

Specify the bit pattern data for the Startup day setting expanded by using the jobschprint -r command. (If any value other than those expanded by the jobschprint command is specified, operation cannot be guaranteed. Also, operation is not guaranteed for any value other than those resulting from execution of different version of the jobschprint command. A different year from the one obtained by the jobschprint command cannot be specified in this operand.) If the bit pattern data is specified, you can make settings for three years in the same startup days as the job net expanded using the jobschprint -r command. Immediately after setting this operand, recalculation for startup days by using other startup days setting operands (such as monthly calendar, annual calendar, base job net name, schedule pattern, calendar name, holiday shift, offset, enable/disable period, etc.) cannot be performed. Even if the startup days are changed by this operand, the changes are not applied to the startup days of other job nets that use this job net as the base job net.

(9) annuallydate Date setting for every year

When you use an annual calendar, specify dates when the job net is executed for every year. When you set MM (month) DD (day) as a startup day, specify in the "MMDD" format. More than one startup day can be set.

(10) annuallyweek Day of the week setting for every year

When you use an annual calendar, specify the day(s) of the week on a monthly basis in which the job net is executed. It is only one month that can be specified with one operand. To set more than one month, specify this operand repeatedly. To specify this operant, enter a month as a number between 1 and 12 that is followed by the first three letters of the day of the week. The format should look similar to the following:12,SUN,MON,TUE,WED,THU,FRI,SAT.

(11) monthlydate Date setting for every month

When you use a monthly calendar, specify the date(s) in which the job net is executed for every month. To set DD (day) of every month as the startup day, specify "DD". More than one startup day can be set. To execute the job net at the end of every month, enter "LAST".

(12) monthlyweek Day of the week setting for every month

When you use a monthly calendar, specify the day(s) of the week on a weekly basis in which the job net is executed for every month. It is only one week that can be specified with one operand. To set more than one week, use this operand repeatedly. To specify this operand, enter the ordinal number of the week (where 1 - 5 represents the first through the fifth week, 0 indicates that every week is specified, and 6 represents the last week of the month) with a number between 0 and 6 that is followed by the first three letters of the day of the week. The format should look similar to the following: 5,SUN,MON,TUE,WED,THU,FRI,SAT.

(13) bnsmode Criteria of business day basis

When "Business days" is selected in the Basic information, specify either "Month's start date" or "Month's end date" as the selection criteria of business days.

START

Counts the business days from the Month's start date.

END

Counts the business days back from the Month's end date.

(14) bnsbase Month's start date of business day basis

When "Business days" is selected in the Basic information, enter a value of the "Month's start date". "0" is invalid.

1 to 28

Indicates 1st to 28th of month.

LAST

Indicates the end of month.

(15) bnsday Business days

When "Business days" is selected in the Basic information, specify business days. To specify more than one business days, use a comma (,) as a separator. "0" is invalid.

1 to 31

The business days that are specified by counting from the 1st business day (the first business day of the month) using a range from 1 to 31. This parameter cannot be specified if "END" is set to the bnsmode operand.

-1 to -31

The business days that are specified by calculating back from the -1st business day (the last business day of the month) using a range from -1 to -31. This parameter cannot be specified if "START" is set to the bnsmode operand.

(16) bnslimit Set only business days of the month as startup days

When "Business days" is specified in the Basic information, specify whether "Set only business days of the month as startup days" is set to ON or OFF. If the bnsmode operand is specified, the default is "ON".

ON

"Set only business days of the month as startup days" is selected.

OFF

"Set only business days of the month as startup days" is not selected.

(17) offsetmode Offset

Specify the method for counting the number of days shifted from the standard date by selecting any of the following parameters. If omitted, the default is "EXECDAY".

EXECDAY

Counts the number of days to be shifted based on working days on the calendar.

DAY

Counts the number of days to be shifted based on the absolute number of days, regardless of holidays.

(18) offsetday Number of offset days

Specify the number of days to be shifted from the standard date between -99 and 99. If omitted, the default is "0".

(19) offsetholiday Sets the base day as a working day if it falls on a holiday

Specify what happens if the base day is a holiday when counting the number of days shifted from the base day. If omitted, the default is "OFF".

This operand cannot be specified if "DAY" has been specified for offset.

ON

Sets the base day as a working day even if it falls on a holiday, and it is set as the base day for the number of days shifted.

OFF

Sets the base day as a non-working day if it falls on a holiday, and it is not set as the base day for the number of days shifted.

(20) holidayshift Holiday shift

If the startup days that are registered or set in the schedule pattern fall on holidays, select a method to handle them by selecting one of the following parameters:

BEF

Shifts the job net startup day to the preceding day.

AFT

Shifts the job net startup day to the next day.

NOT

Recognizes the day as a holiday, so does not execute the job net.

IGN

Recognizes the day as a startup day, so starts the job net.

(21) shiftlimit Number of maximum shifts

Set the number of days by specifying a number between "0" and "99", on which execution is suspended because the Holiday shift is performed on the consecutive holidays. If omitted, the default is "0".

(22) holidaycalendar Calendar name

Specify the name of the calendar (for the SYSTEM_CALENDAR or some other) to be used as the base when startup days are generated. You can specify up to five calendar names per one job net. To specify more than one calendar, use a comma (,) as a separator.

(23) holidaymode Condition for combining calendars

When you specify a method of combining multiple calendars, select either of the following parameters. If omitted, the default is "AND".

AND

Sets the days as holidays if they are set as holidays in all calendars.

OR

Sets the days as holidays if they are set as holidays in at least one of the calendars.

(24) enablestart Start date of validity period/enableend End date of validity period

Specify the operating period of the job net in the "MM,DD,YYYY" format. If the start date of validity period is omitted, the job net operation is enabled from the current date until the end date of validity period. If the end date of validity period is omitted, the job net operation is enabled from the start date of validity period onward. If both the start date and end date of validity period are omitted, the job net operation is enabled on every day.

(25) disablestart Start date of invalidity period/disableend End date of invalidity period

Specify the non-operating period for the job net in the "MM,DD,YYYY" format. If the start date of invalidity period is omitted, the job net operation is disabled from the current day until the end date of invalidity period. If the end date of invalidity period is omitted, the job net operation is disabled from the start date onward. If both the start date and end date of invalidity period are omitted, the job net operation is enabled on every day.

4.9.4 Message Event Control Statement

This is a control statement for setting message events.

Synopsis

```
MSG                                ---------- (1)
jobnetname  Job net name           ---------- (2)  (*) (Note 1)
messagemode Message condition      ---------- (3)
carryover   Carryover processing   ---------- (4)      (Note 2)
messagename Message event name     ---------- (5)
;
```

Note1:

The jobnetname operand should be specified immediately after the MSG keyword.

Note2:

The carryover operand should be specified immediately after the messagemode operand.

*:

This must be specified.

About This Control Statement

(1) MSG

This is the keyword which indicates the beginning of a message event control statement.

(2) jobnetname Job net name

Specify the name of a job net for which message events are to be set.

When you specify a startup days control statement and a temporary changes control statement along with this statement, specify the same job net name across those statements.

(3) messagemode Message condition

When you specify this operand, combine the message event waiting condition (AND/OR), execution condition (ORD/COG/BOT/UNC), and the condition with the execution time (ANDTIME/ORTIME). If this operand is omitted, it is defaulted to "AND,UNC,ORTIME" as its parameter. "Valid during job net execution" ("ORD" and "BOT") cannot be specified for the "execution condition" if "ANDTIME" is specified as the "condition with the execution time" or if the job net belongs to a group. When the job net belongs to a group, the value specified for the "condition with the execution time" is ignored.

AND

Starts the job net if all of the registered message events are generated.

OR

Starts the job net if at least one of the registered message events is generated.

ORD

"Valid during job net execution" is specified. Message events generated while the job net is being executed become valid as the execution condition. Therefore, as soon as the job net is terminated, it is restarted as many times as the message events were generated during execution. The message events generated during execution of the job net become invalid as the execution condition if this parameter is not specified.

COG

"Valid only on startup days" is specified. Starts the job net only if the specified message event is generated on a selected startup days.

BOT

"Valid during job net execution & Valid only on startup days" is specified. If the specified message event is generated on a selected startup days, the job net is executed as many times as the number of message events generated.

UNC

Specify this parameter when the execution condition is not set.

ANDTIME

Starts the job net when the scheduled startup time is reached and the awaiting message event occurs. This parameter is valid only when the startup condition of the job net is set to "single startup" ("single startup (temporary change)" during a temporary change period).

ORTIME

Starts the job net either when the scheduled startup time is reached or the awaiting message event is generated.

(4) carryover Carryover processing

Specifies that processing be carried over when the day change time is reached. This can be specified if the job net waits for the startup time to arrive ("ANDTIME" is specified in the messagemode operand), or if the startup conditions are for the job net to start only when a certain message event has occurred ("ON" is specified in the msgonly operand). The default value is "RESET".

RESET

Carrying over the job net process is not performed when the day change time is reached.

SET

Carrying over the job net process is performed when the day change time is reached.

(5) messagename Message event name

Specify a message event that triggers startup of the job net, using up to 12 bytes. The message event name cannot contains a comma (,), a colon (:), a blank, and a user-defined character. Up to 70 message event names per one job net can be registered.

Example of message event setting

To register three message events, MSG1, MSG2 and MSG3:

```
messagename MSG1,MSG2,MSG3
```

4.9.5 Temporary Changes Control Statement

This control statement sets temporary change information on a job net.

Synopsis

```
TMP                                                          ---------- (1)
jobnetname Job net name                             ---------- (2)  (*)  (Note )
tstarttime Single startup (temporary change)          ---------- (3)  (**)
tmultitime Multiple startups(temporary change)        ---------- (4)  (**)
tintervalstart Fixed interval (temporary change)      ---------- (5)  (**)
tiplstart IPL startup (temporary change)              ---------- (6)  (**)
tmsgonly Message-driven (temporary change)            ---------- (7)  (**)
tpowermode Action at power off (temporary change)   ---------- (8)
tiplmode Action when activated (temporary change)   ---------- (9)
tstoptime Stop time (temporary change)                ---------- (10)
talarmtime Estimated end time (temporary change)    ---------- (11)
temporarystart Start date of temporary change        ---------- (12) (*)
temporaryend End date of temporary change            ---------- (12)
tmsgonlymode Action at message startup (temporary change)  ---------- (13)
tmsgstarttime Scheduled start time (temporary change)      ---------- (14)
;
```

Note1:

The jobnetname operand should be specified immediately after the TMP keyword.

Note2:

The tmsgstarttime operand can be specified when "ONCE" is specified with the tmsgonlymode. Specify this operand immediately after the tmsgonlymode operand.

*:

This must be specified.

**:

Make sure to select one of these.

About This Control Statement

(1) TMP

This is the keyword that indicates the beginning of a temporary change control statement.

(2) jobnetname Job net name

Specify the name of a job net for which temporary change information is to be set.

When you specify a startup days control statement and a message event control statement along with this statement, specify the same job net name across those statements.

(3) tstarttime Single startup (temporary change)

When you execute the job net once a day, specify the execution time in four bytes.

This setting is valid from the start date of temporary change to the end date of temporary change.

(4) tmultitime Multiple startups (temporary change)

When you start up the job net a number of times in the same day, specify the execution times in four bytes or less. You can specify up to six execution times. This setting is valid from the start date of temporary change to the end date of temporary change. If only one execution time is specified, it is registered as a single startup (temporary change). Multiple startups (temporary change) cannot be specified for a job net that belongs to a group.

(5) tintervalstart Fixed interval (temporary change)

When you start up the job net at regular intervals, specify the execution time, end time, and execution interval. The execution interval can be specified between 1 and 240 (minutes). The time span specified between the execution time and termination time cannot cross over the day change time. This setting is valid from the start date of temporary change to the end date of temporary change. Fixed intervals (temporary change) cannot be specified for a job that belongs to a group.

(6) tiplstart IPL startup (temporary change)

Set this to "ON" to start the job net when the Jobscheduler service/daemon is started. This setting is valid from the start date of temporary change to the end date of temporary change. IPL startup (temporary change) cannot be specified for a job net that belongs to a group.

(7) tmsgonly Message-driven (temporary change)

Set this to "ON" to start the job net only when a particular message event is generated. This setting is valid from the start date of temporary change to the end date of temporary change. Message-driven (temporary change) cannot be specified for a job net that belongs to a group.

(8) tpowermode Action at power off (temporary change)

Specify the action to be taken when the Jobscheduler service/daemon is not activated when the job net's scheduled startup time is reached. If omitted, it defaults to "RESET". This setting is valid from the start date of temporary change to the end date of temporary change. The value for "Action at power off (temporary change)" is ignored if an execution condition other than "single startup (temporary change)" is specified for the job net or if the job net belongs to a group.

RESET

The job net is not started even if the Jobscheduler daemon/service is activated.

SET

The job net is started when the Jobscheduler service/daemon is activated.

(9) tiplmode Action when activated (temporary change)

Specifies whether to start the job net with IPL startup every time the Jobscheduler service/daemon is activated. If omitted, the default is "EVERYTIME". This setting is valid from the start date of temporary change to the end date of temporary change.

EVERYTIME

The job net is started every time the Jobscheduler service/daemon is activated.

ONCE

The job net is started only the first time, even if the Jobscheduler service/daemon is started more than once a day.

(10) tstoptime Stop time (temporary change)

Specifies the time to terminate the job net when the "Execution attribute" of the job net is set to "INTS". Specify this operand in the HHMM (hour minute) format, and in four bytes. If omitted, the stop time is not set.

If the temporary change stop time is set earlier than the current time, the stop time is set to the next day.

This setting is valid from the start date of temporary change to the end date of temporary change.

(11) talarmtime Estimated end time (temporary change)

Specify the estimated end time (temporary change) of the job net.

Specify this operand in the HHMM (hour minute) format and in four bytes. If omitted, no estimated end time is set. The time span specified between the execution time and the estimated end time cannot cross over the day change time. This setting is valid from the start date of temporary change to the end date of temporary change.

(12) temporarystart Start date of temporary change/temporaryend End date of temporary change

Specify the temporary change period of job net in the "MM,DD,YYYY" format. If only the start date of temporary change is specified, only this date can be the target of temporary change.

(13) tmsgonlymode Action at message startup (temporary change)

In case that the awaiting message event is generated multiple times in the same day, specifies whether or not the job net is executed every time the message event occurs. If omitted, "EVERYTIME" is defaulted.

This operand is valid from the start date of temporary change until the end date of temporary change. When the execution condition (temporary change) of job net is set to other than "Start only when message event has occurred", this operand is ignored.

EVERYTIME

The job net is started every time the awaiting message event occurs.

ONCE

The job net is started only the first time the awaiting message occurs, even if the awaiting message is generated multiple times a day.

(14) tmsgstarttime Scheduled start time (temporary change)

This operand specifies the scheduled start time when the job net is the message-driven job net. It can be specified only when "Action at message startup" is set to "ONCE".

Specify this operand in the HHMM (hour minute) format and in four bytes. If omitted, no scheduled start time is set. This operand is valid from the start date of temporary change until the end date of temporary change. When the startup condition (temporary change) of job net is set to other than "Start only when message event has occurred", this operand is ignored.

4.10 jobschsetgrp Command

Synopsis

```
[To register]
  jobschsetgrp -ent [project name] definition file [-sys subsystem-number]⁽*¹⁾
[To change]
  jobschsetgrp -che [project name] definition file [-sys subsystem-number]⁽*¹⁾
```

```
[To delete]
   jobschsetgrp -del [project name/] group name [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschsetgrp command registers, changes and deletes a group.

Options

-ent

Based on the information in the definition file, registers a group. However, if the same group name already exists, it cannot be registered.

project name

Specifies the name of the project where the group information is stored. When registering, if there is only one project having job nets registered among the projects accessible for update by the user of executing the jobschsetgrp command, this option can be omitted. When changing/deleting, if there is only one project having groups registered among the projects accessible for update by the user of executing the jobschsetgrp command, this option can be omitted.

definition file

Specifies the file name where the group control statements and the control statement of job net in a group.

To register multiple groups, you need to write multiple group control statements.

For more information on the group control statement, see "4.10.1 Group Control Statement".

For more information on the group job net control statement, see "4.10.2 Group Job Net Control Statement".

-che

Changes a group based on the information in the definition file. When changing the group, it is convenient to use the definition file used in registering the group.

-del

Deletes a group. When a group is deleted, all the registered information of the group and the job nets in the group will be deleted.

group name

Specifies the name of the group to be deleted.

EE **-sys subsystem-number**

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Cautions

- This command only requests the server to change the group. This command does not guarantee the completion of requested processing. Thus, after executing this command, it may take time to reflect changes in the group on the server.

4.10.1 Group Control Statement

This is a control statement for setting the group registration information.

Synopsis

```
GRP                                  ---------- (1)
groupname Group name                 ---------- (2)  (*)
groupcomment Group comment           ---------- (3)  (*)
powermode Action at power off        ---------- (4)
jobnet Number of job nets in a group ---------- (5)  (*)
operate Operation                    ---------- (6)
;
```

*:

This must be specified.

About This Control Statement

(1) GRP

This is the keyword that indicates the beginning of a group control statement.

(2) groupname Group name

Specify the group name within 50 bytes. The name should be unique.

The name cannot contain the following signs, space, and user-defined characters:

[Windows]: /?*;,\"<>|:

[UNIX]: /?*;,

(3) groupcomment Group comment

Specify a comment on the group within 100 bytes. The comment should not contain spaces, or user-defined characters.

(4) powermode Action at power off

Specify the action to be taken when the Jobscheduler service/daemon is not activated when the scheduled startup time of the group is reached. If omitted, the default is "RESET".

RESET

The group is not started even when the Jobscheduler service/daemon is activated.

SET

The group is started when the Jobscheduler service/daemon is activated.

(5) jobnet Number of job nets in a group

Specify the number of job nets (using a number) making up the group.

In Systemwalker Operation Manager SE, you can specify any number between 1 and 50.

In Systemwalker Operation Manager EE, you can specify any number between 1 and 255.

(6) operate Operation

Indicates the status resulting from the operation as either paused or disabled. The default value is "NONE".

PAUSE

Paused

DISABLE

Disabled

NONE

When a group is registered or changed, the status of the group will be "Waiting".

4.10.2 Group Job Net Control Statement

This is a control statement for setting the registration information on a job net that belongs to a group.

<u>**Synopsis**</u>

```
GNT                                              ---------- (1)
jobnetname Job net name                          ---------- (2)  (*)  (Note 1)
jobnetnumber Job net number                      ---------- (3)  (*)  (Note 2)
pre_jobnet Preceding job net number          ---------- (4)
starttimeignore Ignore scheduled start time    ---------- (5)
iconposition Icon position                       ---------- (6)     (Note 3)
;
```

Note 1:

The jobnetname operand should be specified immediately after the GNT keyword.

Note 2:

The jobnetnumber operand should be specified immediately after the jobnetname operand.

Note 3:

This operand must be specified for all job nets that make up the group.

*:

This must be specified.

<u>**About This Control Statement**</u>

(1) GNT

This is the keyword that indicates the beginning of a group job net control statement.

(2) jobnetname Job net name

Specify a job net name that is already registered within 50 bytes.

"JSHEND" or "JSHFORCE" as a shutdown job net name, cannot be registered in a group.

It is not allowed to contain the following signs, blank space, or any user-defined characters.

[Windows]: /?*;,\"<>|:'.

[UNIX]: /?*;,'.

(3) jobnetnumber Job net number

Allocate numbers corresponding to each job net name. Unique numbers should be assigned, according to the job net execution order. They should be allocated in ascending order from the preceding job nets to the subsequent job nets.

In Systemwalker Operation Manager SE, you can specify any number between 1 and 50.

EE

In Systemwalker Operation Manager EE, you can specify any number between 1 and 255.

(4) pre_jobnet Preceding job net number

Specify job net numbers of preceding job nets. Use a comma as a separator to specify more than one preceding job net number, which should be in the ascending order. The job nets to be specified as preceding job nets must be defined beforehand.

(5) starttimeignore Ignore scheduled start time

Specify "ON" to execute the job net by ignoring the scheduled startup time.

(6) iconposition Icon position

This option specifies position information for the job icon using the following format. The numbers indicate the coordinates for the grid position, taking the top left corner of the job flow as the origin (1,1).

```
Horizontal position (1 to 255), vertical position (1 to 255)
```

If this option is omitted, the position of the icon will be determined automatically. If the icon position is changed using the Change Group window after the job has been registered without specifying this option, the new icon position will be stored, and this control statement will be output using the jobschprint -R command.

Cautions

The job net that satisfies the following conditions can be registered in a group:

- The job net must belong to a project to which the group is added.

- Only one execution time (once a day startup) is registered for the job net in the [Execute at] of [Execution condition].

- [Valid during job net execution] is not selected as the message event condition for the job net.

- There are no job nets with the same name as the target job net in the group.

- The job net does not belong to any other group.

4.11 jobschctljob Command

Synopsis

```
jobschctljob [project name/] job net name/job name control
[-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschctljob command controls (cancel/rerun/pause/continue/disable/enable) jobs.

Options

project name

Specifies the project name where the jobs to be controlled are stored. If there is only one project having jobs registered among the projects accessible for operation by the user of executing the jobschctljob command, this option can be omitted.

job net name

Specifies the job net name where the jobs to be controlled are stored.

job name

Specifies the job name to be controlled.

control

Specify one of the following six control items:

cancel

Cancels the job.

rerun

Restarts the job that was canceled or terminated abnormally.

pause

Pauses the job.

continue

Cancels the paused status of the job.

disable

Disables the job.

enable

Enables the disabled job.

`EE`

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Cautions

- Controlling jobs is only available for the jobs that belong to the job net with "Job Execution Control" attribute or "PowerAIM" ([UNIX]) attribute.

- When controlling the job, the job name must be specified and if the job name is omitted, the job cannot be controlled.

- Child job nets cannot be controlled.

- Linked job nets cannot be controlled.

- If there are multiple jobs with the same job name specified within the job net, this command fails.

- This command simply requests the server to operate on the job, and so the fact that the command has finished executing does not guarantee that job operation on the server has completed. For this reason, after this command has finished executing, check the status of the job by entering the "jobschprint -p" command.

4.12 jobschcontrol Command

Synopsis

```
jobschcontrol [project name/] job net name control
    [start option] [rerun option] [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschcontrol command controls (cancel/start/restart/pause/continue/disable/enable/cope/reinstate/revoke) job nets.

Options

project name

Specifies the project name to which the job net you want to control belongs. If there is only one project having job nets registered among the projects accessible for operation by the user of executing the jobschcontrol command, this option can be omitted.

job net name

Specifies the name of the job net to be controlled.

control

Specify one of the following ten control items:

cancel

Cancels the job net.

start

Starts the job net.

rerun

Restarts the job net that terminated abnormally, was canceled, or confirmed.

pause

Pauses the job net.

continue

Cancels the paused status of the job net.

disable

Disables the job net.

enable

Enables the disabled job net

cope

Changes the status of abnormally ended job net to Confirmed.

If **Check canceled** has been specified in the **Advanced Settings for Verification** window (which is displayed by clicking the **Details** button in the **Verification at error occurrence** section of the **Use function 1** tab of the **Define Jobscheduler Startup Parameters** window), then canceled job nets will also be subject to verification control.

reinstate

Clear the execution result of the job net that was already executed.

revoke

Releases the carried over status of the job net.

rerun options

Only when rerun is specified in the control option, the following two control items are available. When the rerun option is omitted, this command restarts jobs, starting with the job that abnormally terminated. If a restart point is set on the job prior to the abnormally terminated job, the job net is restarted from the job on which the restart point is set.

next

> Restarts the job net after setting the "Invalid at next restart" option for the abnormally terminated job. (The abnormally terminated job will be skipped and the subsequent job will be executed.) Canceled jobs will be handled in the same manner as abnormally terminated jobs.

top

> Restarts the job net from the top of the jobs.

Start option

These options can be specified only if "start" has been specified for the "control" option.

-p param

> Specify the variable parameter to be passed to each job in the job net to be started. If this option is specified, any "@.VPARAM@" variables in the job definitions will be replaced with the string specified for this option.
>
> Specify the variable parameter using a string that is no more than 200 bytes long. The variable parameter should not include user-defined characters.
>
> This option can be omitted. In this case, the "@.VPARAM@" variables in the job definitions will be replaced with an empty string before the jobs start.
>
> The variable parameter is also passed to job nets in the second and lower levels of nested job nets (child job nets).
>
> This option can only be specified once.
>
> Refer to "Using variable parameters to start a job net" in the S*ystemwalker Operation Manager User's Guide* for information about the "@.VPARAM@" variable.

EE

-sys subsystem-number

> In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Cautions

- After entering this command, issue the jobschprint -n command to confirm the status of the job net.

- This command only requests the server to operate the job net. Thus, it does not guarantee the completion of requested operation on the job net on the server side even if the command is complete. Therefore, after executing this command, it may take time to execute the requested processing on the server.

- Note the following point when specifying "-p param" with the start option.

    - For the variable parameter, specify a string that that does not exceed the maximum number of characters and does not contain any invalid characters for the definition item that replaces the @.VPARAM@ variable in the job definitions for the job net to be started. If the job definitions become invalid as a result of replacing the variable parameters, the job will not execute and will terminate abnormally with completion code 245 or 246.

- This command can only be used to pause, continue, disable, or enable master linked job nets or job nets nested therein.

4.13 jobschctlgrp Command

```
jobschctlgrp [project name/] group name control
            [-sys subsystem-number]('¹)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschctlgrp command controls (cancel /start/rerun/pause/continue/disable/enable/cope) a group.

Options

project name

Specifies the project name where group information is stored. If there is only one project having groups registered among the projects accessible for operation by the user of executing the jobschctlgrp command, this option can be omitted.

group name

Specifies the name of the group to be controlled.

control

Specify one of the following eight control items:

cancel

Cancels the group.

start

Starts the group.

rerun

Restarts the group that was canceled or confirmed, or terminated abnormally.

pause

Pauses the group.

continue

Cancels the paused status of the group.

disable

Disables the group.

enable

Enables the disabled group.

cope

Changes the status of abnormally ended group to Confirmed.

If **Check canceled** has been specified in the **Advanced Settings for Verification** window (which is displayed by clicking the **Details** button in the **Verification at error occurrence** section of the **Use function 1** tab of the **Define Jobscheduler Startup Parameters** window), then canceled job nets will also be subject to verification control.

EE **-sys subsystem-number**

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Cautions

- After entering this command, issue the jobschprint -x command to confirm the status of the job net.

- This command only requests the server to operate the group. Thus, it does not guarantee the completion of requested operation on the group on the server side even if the command is complete. Therefore, after executing this command, it may take time to execute the requested processing on the server.

4.14 jobschmove Command

Synopsis

```
jobschmove -group [project name/] group name [destination group name]
                    [-sys subsystem-number](*1)
jobschmove -gnet [project name/] group name [destination job net name]
                  [-sys subsystem-number](*1)
jobschmove -net [project name/] job net name [destination job net name]
                [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschmove command changes the addition order of the specified groups and/or job nets in a Gant Chart. If there is only one project having job nets registered among the projects accessible for update by the user of executing the jobschmove command, the project name option can be omitted.

Options

-group [project name/] group name [destination group name]

Changes the addition order of the specified groups.

Moves the group specified in the group name directly after the group specified in the destination group name. If the destination group name is omitted, the specified group is moved to the top of the groups in a Gant Chart.

-gnet [project name/] group name [destination job net name]

Moves the addition order of the job nets belonging to the specified group to directly after the other job net.

Moves the job nets belonging to the group specified in the group name to directly after the job net specified in the destination job net name. If the destination job net name is omitted, the job nets in the specified group are moved to the top of the job nets in a Gant Chart. A preceding job net takes precedence in the order of job nets within the group while they are moving.

-net [project name/] job net name [destination job net name]

Changes the addition order of the specified job net. Moves the job net specified in the job net name to directly after the job net specified in the destination job net name. If the destination job net name is omitted, the specified job net is moved to the top of the job nets in a Gant Chart.

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

4.15 jobschmsgclear Command

Synopsis

```
jobschmsgclear -all [project name] [-sys subsystem-number](*1)
jobschmsgclear -net [project name/] job net name
                    [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschmsgclear command invalidates generated message events.

When you need to prevent all the message events from occurring due to reasons such as date changed, initialize the already generated message events using the jobschmsgclear command.

Options

-all

Invalidates the already generated message events on all job nets in all projects that can be updatable by the user of executing this command.

project name

When specified with the -all option, this option invalidates the already generated message events on all job nets within the specified project.

-net

Invalidates the already generated message events on the specified job net.

project name

Specifies the project name hosting the job net for which generated message events are to be invalidated. If there is only one project having job nets registered among the projects accessible for update by the user of executing the jobschmsgclear command, this option can be omitted.

job net name

Specifies the name of the job net for which message events are to be invalidated. Specify this option with the -net parameter.

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Cautions

- The occurred time of message events continues to appear in the [Message Event List] window even after the generated message events were cleared by this command.

- When message events are issued while the job nets are paused, and then the job nets meet the startup conditions, the job nets cannot initialize the message events that have already occurred due to this command.

4.16 jobschmsgevent Command

Synopsis

[Windows]

```
jobschmsgevent message event name [:project name[/job net name]]
[host name|-f: file name] [-t:type] [-p param [-c suffix]]
[-sys {all | subsystem-number}](*1)
```

[UNIX]

```
jobschmsgevent message event name [:project name[/job net name]
[message text]] [-h host name] [-t:type] [-p param [-c suffix]]
[-sys {all | subsystem-number}](*1)
```

*1:

[-sys {all | subsystem-number}] is the option of Systemwalker Operation Manager EE.

Description

The jobschmsgevent command generates a message event.

If a message event is generated by this command, the message event of "nnnn message event has occurred" (message text, where nnnn is a message event name) is output in the event log, along with the source name "MpJobsch" and event ID "2301". A message event is generated to local host if both host name and file name are not specified. [Windows]

The message event generated by this command becomes valid for all job nets of all projects if the project name is omitted.

If both host name and file name are omitted, this command generates a message event on the local host. [Windows]

Options

message event name

Specify the name of message event that you want to generate within 12 bytes.

project name

Specify this option when you want to generate the message event only for the job nets in a particular project.

job net name

Specify this option when you want to generate the message event only for a particular job net.

message text [UNIX]

Specify any text string.

host name [Windows]
-h host name [UNIX]

Specifies the host name of the server where message events are generated. Specify it within 64 bytes using alphanumeric characters. If omitted, the message event is generated on a local host.

-f: file name [Windows]

Specifies the file name containing a list of host names of servers where message events are generated

-t: type [Windows]

Specifies the type of the message event to be generated, when is output to the event log.

If omitted, the default is "-t:INFO".

Specify one of the following:

INFO

"Information" is output.

WARN

"Warning" is output.

ERROR

"Error" is output.

-p *param*

Specify the variable parameter to be passed to each job in the job net started by the message event. If this option is specified, the "@.VPARAM@" variables in the job definitions will be replaced with the specified variable parameter when the job net starts as a result of receiving the message event.

Specify the variable parameter using a string that is no more than 200 bytes long. The variable parameter should not include user-defined characters.

The variable parameter can be omitted. In this case, the "@.VPARAM@" variables in the job definitions will be replaced with an empty string before the jobs start.

Only job nets that wait for a single message event and job nets that wait for only one of several message events can receive the variable parameter specified by this option.

The variable parameter is also passed to job nets in the second and lower levels of nested job nets (child job nets).

This option can only be specified once.

-c *suffix*

Use a string of up to 30 bytes long to specify the suffix that is attached to the names of job nets that are copied and started. If this option is specified, the target job nets that are started with a variable parameter specified will be copied to a job net whose name has the specified suffix appended, and the copied job net will be started. For nested job nets, job nets at the second and lower levels are also copied.

Alphanumeric characters and hyphens (-) can be used, but only alphanumeric characters can be used for the first character. Be sure to specify this option together with (but after) the "-p" option.

The job net names and job net comments of copy destination job nets become "*copy source job net name (or comment) + underscore (_) + string specified by the suffix option*".

Copying takes place when the job net starts (or, for nested job nets, when the parent job net starts). Copy destination job nets are overwritten the next time a job net with the same name is copied.

Note that this option is invalid for job nets in groups.

-sys {all|subsystem-number}

In the system with multiple subsystems, this option is used to specify a subsystem as operating target. If omitted, the default is subsystem 0.

However, if the job is executed as a Jobscheduler job in a subsystem environment, the number of this subsystem will be assumed.

all

All subsystems are targeted for operation.

subsystem-number

Specify the operation target subsystem number using a range of 0 to 9.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Cautions

- The user ID used to execute this command is also required for the host on which message events are generated.

- If the job net waiting for message event is in any of the following states, the message event cannot be recognized when it has occurred.

    - The job net is in a running state. (*1)

    - The job net is in a warning state. (*1) (*2)

    - The job net is in a disabled state.

    - The job net is in an abended state. (*3)

    - The job net is in a cancelled state. (*4)

    - The count of cumulative message occurrence exceeded 255 times. (*5)

    - The job net waiting for a message event falls on the unscheduled day. (*6)

    - The [Once in a day] option for [Start only when message event has occurred] is selected as the execution condition and the job net has already been started. (*7)

    *1:

    The message event can be recognized if the "Valid during job net execution" is selected in the **Message** sheet of the **Job Net Properties** window. The job net will be started the same number of times as the number of message events that have occurred while the job net is running. Note that the job nets are not started at the same time.

    *2:

    Only a job net with the Job Execution Control attribute can be in a warning state.

    *3:

    The message event can be recognized if the job net belongs to a group. Job nets that do not belong to a group are also recognized if the **Validate job net confirmation** checkbox in the **Advanced Settings for Verification** window has not been selected.

    *4:

    The message event can be recognized if the job net belongs to a group. Job nets that do not belong to a group are also recognized if the **Check canceled** checkbox in the **Advanced Settings for Verification** window has not been selected.

*5:

If the execution condition of job net is specified as [Start only when message event has occurred], it internally retains the count of message events occurrence. This occurrence count refers to the cumulative occurrence count. This occurrence count is incremented by one as a message event occurs, while it is decreased by one as the job net is executed when a message has occurred. Note that the cumulative occurrence count of message events should not exceed 255.

*6:

The message event can be recognized if the "Valid only on startup days" is not selected.

*7:

The message event can be recognized again after the [Reinstate] operation has been performed.

- When writing a list of host names, enter one host name within 64 bytes per line.[Windows]

- In the host name list, the first host name must be specified in column 1. No comments are allowed in the file. [Windows]

- The server function of Systemwalker Operation Manager for Windows must be installed on the host where message events are generated. [Windows]

- The Server service for Windows must be started in advance on the host where message events are generated. [Windows]

- The jobschmsgevent command functions to record events to the application event log of the specified host. Due to this, note the following: [Windows]

    - It is necessary that the host where message events are generated is able to authenticate the user ID used to execute the jobschmsgevent command.

    For example, when executing the command as a job of Systemwalker Operation Manager, confirm that the target server can authenticate the logon account of the Systemwalker MpMjes service.

    - If the event log reaches full, events cannot be written and the execution of the jobschmsgevent command may fail.

    Before starting system operation, increase the size of log on the properties of the application event log or specify "Overwrite Events as Needed" to avoid capacity shortage.

    - NetBIOS interfaces are used for recording the event log. Therefore, when you establish firewall among servers that have message link to restrict ports of NetBIOS interfaces, message event notification cannot be executed correctly. In this case, do not specify the host name with the jobschmsgevent command, and execute it as a network job.

- If you wish to generate message events to a different host, you must execute this command as an administrator user.

Using a network job eliminates the need to consider which user executes the command. Therefore, use a network job to generate message events on the request host. (However, this does not apply to cases where the user who executed the command belongs to the Guests group. Refer to the cautions below.) [Windows]

- If message events are generated on the local host and the user ID which issues the jobschmsgevent command belongs to the Guests group, the message events may fail to occur. Issue the jobschmsgevent command with a user ID that is not a member of the Guests group. [Windows]

- When you specify a host name by using the "-h host name" option, the UNIX version of the Systemwalker Operation Manager server function must be installed. However, if you do not want to generate a message event on the local host (namely, when you do not specify the "-h host name" option), you need not to install the Job Execution Control function on the local host.[UNIX]

- Execute the following script when generating a message event from Systemwalker Operation Manager EE to Systemwalker Operation Manager SE.

[Windows]

[jobschmsgevent.bat]

```
set JOBSCH_SUB_SYSTEM=
jobschmsgevent %*
setecode.exe %ERRORLEVEL%
```

The following example shows the command field for the job to execute the above script:

```
Drive name:\Path name\jobschmsgevent.bat MsgA HostName
```

[UNIX]

[jobschmsgevent.sh]

```
#!/bin/csh
unsetenv JOBSCH_SUB_SYSTEM
jobschmsgevent $*
exit $status
```

The following example shows the command field for the job to execute the above script.

```
/Path name/jobschmsgevent.sh MsgA -h HostName
```

- Note the following points if a variable parameter is specified using the -p option.

    - For the variable parameter, specify a string that does not exceed the maximum number of characters and does not contain any invalid characters for the definition item that replaces the @.VPARAM@ variable in the job definitions for the job net to be started. If the job definitions become invalid as a result of replacing the variable parameters, the job will not execute and will terminate abnormally with completion code 245 or 246.

    - Do not generate message events by specifying this option for job nets that cannot receive variable parameters.

    - This option cannot be specified if the host where the message event is generated is running a version prior to V13.2.0.

    - An underscore (_) and a suffix are appended to the name and comment of the destination job net. Make sure that the lengths of the source job net name, source job net comment, and suffix are such that the lengths of name and comment for the job net that is created by copying do not exceed the maximum lengths. If any of the maximum lengths is exceeded, the job net fails to be copied and started.

    - For the Enterprise Edition, there is no limit to the number of job nets that can be registered, but if a large number of job nets are copied then thorough performance tests should be conducted to check that job nets start on schedule without any problems. For details, refer to "Tuning of Performance" in the *Systemwalker Operation Manager User's Guide* and "Reference Information (Know-how, etc.)" in the *Systemwalker Operation Manager Technical Guide*.

4.17 jobschcancelnet Command [Windows]

Synopsis

```
jobschcancelnet project name job net name [-all]
                  [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is a command operand in Systemwalker Operation Manager EE.

Description

The jobschcancelnet command deletes the startup request for the job net in the job net execution queue. When a startup of job net is requested by using the job net startup API, if the same job net is already running, the job net is connected into the job net startup queue. At this time, if you want to delete the startup request for the job net in the queue, use this command.

This command deletes the startup request for the specified job net.

Options

project name

Specifies the name of the project to which the job net to be removed from the execution queue belongs.

job net name

Specifies the job net name to be removed.

-all

Removes all startup requests of the specified job net from the execution queue.

The first startup request is deleted if this option is omitted.

EE **-sys subsystem-number**

Specifies a subsystem as the operating target within the system having multiple subsystems. The target subsystem number can be specified from 0 to 9. If omitted, the default is subsystem 0.

However, if the job is executed as a Jobscheduler job in a subsystem environment, the number of this subsystem will be assumed.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin

4.18 jobschdeftocsv Command

Synopsis

```
jobschdeftocsv [-d] control statement file
```

Description

The jobschdeftocsv command converts the following control statement files into the CSV format file. The conversion result is output in standard output.

- Job net control statements and job control statements

- Group control statements and group job net control statements

- Holiday information control statements

- Schedule pattern control statements

Saving the conversion result in a file enables the control statement information to be managed in spreadsheet software. (Note)

Note:

Before importing the output result into the spreadsheet software, it is necessary to convert it into the character code supported by that spreadsheet software.

Options

-d

Items are enclosed in double quotes (") when they are output. The double quote character itself is output as two consecutive double quotes ("").

Specify this option if necessary when importing the output results into a spreadsheet program if a control statement includes double quotes.

Conversion example (for the following string: a"b)

Without the "-d" option: a"b

With the "-d" option: "a""b"

Control statement file

Specifies the control statement file to be converted. The files for the following control statement can be specified:

- Job net control and job control statements

- Group control and group job net control statements

- Holiday information control statements

- Schedule pattern control statements

To get the control statement file that is specified using this command, use the output result of the jobschprint -r/-R/-rh/-rsp command, or create it by following the example below.

- Example of creating a control statement file for job net control statement file and job control statements

An example of control statement file for the job net control statements and job control statements is shown below. For details of each control statement, see "4.9.1 Job Net Control Statement" and "4.9.2 Job Control Statement".

```
NET
jobnetname selling
jobnetcomment SalesManagement
starttime 2000
execpattern EveryFriday
holidayshift IGN
defaulthost king
job 4
operate PAUSE
;
JOB
jobname management1
jobnumber 1
jobcomment CustomerInquiry
restartpoint ON
iconposition 1,1
operate NONE
;
JOB
jobname management2
jobnumber 2
jobcomment CreditDetails
pre_job 1
iconposition 1,2
operate NONE
;
JOB
jobname management3
jobnumber 3
jobcomment DataSearch
pre_job 1
iconposition 2,2
operate DISABLE
;
JOB
jobname management4
jobnumber 4
jobcomment Backup
pre_job 2,3
iconposition 1,3
operate NONE
;
```

- Example of creating a control statement file for group control statements and group job net control statements

An example of control statement file for the group control statements and group job net control statements is shown below. For details of each control statement, see "4.10.1 Group Control Statement" and "4.10.2 Group Job Net Control Statement".

```
GRP
groupname weeklyreport
groupcomment WeeklyReport
jobnet 2
;
GNT
jobnetname sales
jobnetnumber 1
;
GNT
jobnetname selling
jobnetnumber 2
pre_jobnet 1
;
```

- Example of creating a control statement file for holiday information control statements

An example of a control statement file for holiday information control statements is shown below. Refer to "4.2.1 Holiday Information Control Statement" for details about the control statements.

```
HOL
calendarname MainOfficeCalendar
yearset 2010
monthset 11
dayset 1
yearset 2010
monthset 12
dayset 20
;
```

- Example of creating a control statement file for schedule pattern control statements

An example of a control statement file for schedule pattern control statements is shown below. Refer to "4.3.1 Schedule Pattern Control Statement" for details about the control statements.

```
CAL
execpatname Sample
monthset 1

monthset 1
dayset 1
monthset 2
dayset 10
;
```

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Cautions

When loading a CSV format file into spreadsheet software, the string that contains only numbers (no letters) may be converted as follows:

From: "0010" (Notation in CSV file)

To: "10" (Notation in spreadsheet software)

This problem occurs due to an error that the spreadsheet software recognizes the string with only numbers (no letters) as numerical data, rather than as the character string. In such a case, first change the format setting of the cell so that the relevant cell is treated as a character string, then correct the character string to match the notation in the CSV file.

Output Example

The following is an output example when you enter the control statement files shown in the above examples to create a control statement file.

- Output example of information on job net control statements and job control statements.

```
NET
jobnetname,jobnetcomment,starttime,execpattern,holidayshift,defaulthost,job,operate
selling,SalesManagement,2000,EveryFriday,IGN,king,4,PAUSE
JOB
jobname,jobnumber,jobcomment,restartpoint,pre_job,iconposition,operate
management1,1,CustomerInquiry,ON,1;1,NONE
management2,2,CreditDetails,,1,1;2, NONE
management3,3,DataSearch,,1,2;2,DISABLE
management4,4,Backup,,2;3,1;3,NONE
```

- Output example of information on group control statements and group job net control statements

```
GRP
groupname,groupcomment,jobnet
weeklyreport,WeeklyReport,2
GNT
jobnetname,jobnetnumber,pre_jobnet
sales,1,
selling,2,1
```

- Output example of the holiday information control statement information

```
HOL
calendarname,yearset,monthset,dayset,yearset,monthset,dayset
MainOfficeCalendar,2010,11,1,2010,12,20
```

- Output example of the schedule pattern control statement information

```
CAL
execpatname,monthset,dayset,monthset,dayset
Sample,1,1,2,10
```

The above contents will appear in the following format if you display them in the spreadsheet software:

- Display example of information on job net control statements and job control statements

NET							
jobnetname	jobnetcomment	starttime	execpattern	holidayshift	defaulthost	job	operate
selling	SalesManagement	2000	EveryFriday	IGN	king	4	PAUSE
JOB							
jobname	jobnumber	jobcomment	restartpoint	pre_job	iconposition	operate	

management 1	1	CustomerInquiry	ON		1;1	NONE	
management 2	2	CreditDetails		1	1;2	NONE	
management 3	3	DataSearch		1	2;2	DISABLE	
management 4	4	Backup		2;3	1;3	NONE	

- Display example of information on group control statements and group job net control statements

GRP		
groupname	groupcomment	jobnet
weeklyreport	WeeklyReport	2
GNT		
jobnetname	jobnetnumber	pre_jobnet
sales	1	
selling	2	1

- Display example of information on holiday information control statements

HOL						
calendarname	yearset	monthset	dayset	yearset	monthset	dayset
MainOffice Calendar	2010	11	1	2010	12	20

- Display example of information on schedule pattern control statements

CAL				
execpatname	monthset	dayset	monthset	dayset
Sample	1	1	2	10

4.19 jobschcsvtodef Command

Synopsis

```
jobschcsvtodef CSV file
```

Description

This command converts registration information on job nets, groups, holidays in the calendar, and schedule patterns from the CSV format into the control statement format. The conversion result is output in standard output.

The file used to save the conversion results can then be used as the definition file specified by the jobschsetnet, jobschsetgrp, jobschsethol, and jobschsetcal commands.

Options

CSV file

Specify the source CSV file to be converted. Specify one of the following CSV files:

- Job net definition CSV file

  CSV file with the job net control statements and job control statements entered.

- Group definition CSV file

  CSV file with the group control statements and control statements for job nets within groups entered.

- Calendar holiday definition CSV file

  CSV file with the calendar holiday control statements entered.

- Schedule pattern definition CSV file

  CSV file with the schedule pattern control statements entered.

The format for each CSV file is shown below.

- Format for job net definition CSV files

Create job net definition CSV files using the following format:

```
NET
control statement, control statement, ...
option value, option value, ...
JOB
control statement, control statement, ...
option value, option value, ...
option value, option value, ...
       .
```

Row 1:

Enter "NET". "NET" is the keyword that indicates the beginning of a job net control statement.

Row 2:

Enter the options of the job net control statement. Make sure to enter the mandatory options. See "4.9.1 Job Net Control Statement" or further information on options.

Row 3:

Enter the values of the options in the same order as those entered in Row 2. The values for these options can be omitted.

Row 4:

Enter "JOB". "JOB" is the keyword that indicates the beginning of a job control statement.

Row 5:

Enter the options of the job control statement. Make sure to enter the mandatory options. See "4.9.2 Job Control Statement" or further information on options.

Row 6:

Enter the values of the options in the same order as those entered in Row 5. The values for these options can be omitted.

Row 7 and on:

The synopsis is the same as for Row 6. Add as many lines as the number of jobs to be registered.

 Note

If there are multiple option values for a single control statement, use a semicolon (;) to separate option values. If there are multiple schedule patterns, however, delimit them with spaces.

- Example of how to create job net definition CSV files

```
NET
jobnetname,jobnetcomment,starttime,execpattern,holidayshift,defaulthost,job,operate
hanbai,Sales Management,2000,Every Friday,IGN,king,4,PAUSE
JOB
jobname,jobnumber,jobcomment,restartpoint,pre_job,iconposition,operate
kanri1,1,Customer Inquiry,ON,1;1,NONE
kanri2,2,Payment Details,,1,1;2, NONE
kanri3,3,Data Search,,1,2;2,DISABLE
kanri4,4,Backup,,2;3,1;3,NONE
```

- Format for group definition CSV files

```
Create group definition CSV files using the following format:
GRP
control statement, control statement, ...
option value, option value, ...
GNT
control statement, control statement, ...
option value, option value, ...
option value, option value, ...
    .
```

Row 1:

Enter "GRP". "GRP" is the keyword that indicates the beginning of a group control statement.

Row 2:

Enter the options of the group control statement. Make sure to enter the mandatory options. See "4.10.1 Group Control Statement" for further information on options.

Row 3:

Enter the values of the options in the same order as those entered in Row 2. The values for these options can be omitted.

Row 4:

Enter "GNT". "GNT" is the keyword that indicates the beginning of a group job net control statement.

Row 5:

Enter the options of the group job net control statement. Make sure to enter the mandatory options. See "4.10.2 Group Job Net Control Statement" for further information on options.

Row 6:

Enter the values of the options in the same order as those entered in Row 5. The values for these options can be omitted.

Row 7 and on:

The synopsis is the same as for Row 6. Add as many lines as the number of job nets in a group to be registered.

Note

If there are multiple option values for a single control statement, use a semicolon (;) to separate option values.

- Example of how to create group definition CSV files

```
GRP
groupname,groupcomment,jobnet
shuuhou,Weekly Report,2
GNT
jobnetname,jobnetnumber,pre_jobnet
eigyou,1,
hanbai,2,1
```

- Format for calendar holiday definition CSV files

Create calendar holiday definition CSV files in the following format:

```
HOL
control statement,control statement,...
option value,option value,...
```

Row 1:

Enter "HOL". "HOL" is the keyword that indicates the beginning of a holiday information control statement.

Row 2:

Enter the options of the holiday information control statement. Make sure that mandatory options are entered. Refer to "Holiday Information Control Statement" for details about options.

Row 3:

Enter the option values, matching the order of the options in row 2. Option values can be omitted.

📎 **Note**

. . . .

If there are multiple options values for a single control statement, delimit them with semicolons (;).

. . . .

- Example of how to create calendar holiday definition CSV files

```
HOL
calendarname,yearset,monthset,dayset,yearset,monthset,dayset
MainOfficeCalendar,2010,11,1,2010,12,20
```

- Format for schedule pattern definition CSV files

Create schedule pattern definition CSV files in the following format:

```
CAL
control statement,control statement,...
option value,option value,...
```

Row 1:

Enter "CAL". "CAL" is the keyword that indicates the beginning of a schedule pattern control statement.

Row 2:

Enter the options of the schedule pattern control statement. Make sure that mandatory options are entered. Refer to "Schedule Pattern Control Statement" for details about options.

Row 3:

Enter the option values, matching the order of the options in row 2. Option values can be omitted.

📎 **Note**

. . . .

If there are multiple options values for a single control statement, delimit them with semicolons (;).

. . . .

- Example of how to create schedule pattern definition CSV files

```
CAL
execpatname,monthset,dayset,monthset,dayset
Sample,1,1,2,10
```

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM \bin
Solaris HP-UX AIX Linux	/usr/bin

Output Example

The following example shows the output results for this command if the CSV files shown in the examples of how to create CSV files are input.

- Output example of information on job net control statements and job control statements.

```
NET
jobnetname selling
jobnetcomment SalesManagement
holidayshift IGN
execpattern EveryFriday
starttime 2000
defaulthost king
job 4
operate PAUSE
;
JOB
jobname management1
jobnumber 1
jobcomment CustomerInquiry
restartpoint ON
iconposition 1,1
operate NONE
;
JOB
jobname management2
jobnumber 2
jobcomment CreditDetails
pre_job 1
iconposition 1,2
operate NONE
;
JOB
jobname management3
jobnumber 3
jobcomment DataSearch
pre_job 1
iconposition 2,2
operate DISABLE
;
JOB
jobname management4
jobnumber 4
jobcomment Backup
pre_job 2,3
iconposition 1,3
```

```
operate NONE
;
```

- Output example of information on group control statements and group job net control statements

```
GRP
groupname weeklyreport
groupcomment WeeklyReport
jobnet 2
;
GNT
jobnetname sales
jobnetnumber 1
;
GNT
jobnetname selling
jobnetnumber 2
pre_jobnet 1
;
```

- Output example of information on the holiday information control statement

```
HOL
calendarname MainOfficeCalendar
yearset 2010
monthset 11
dayset 1
yearset 2010
monthset 12
dayset 20
;
```

- Output example of information on the schedule pattern control statement

```
CAL
execpatname Sample
monthset 1
dayset 1
monthset 2
dayset 10
;
```

4.20 jobschchecktime Command

Synopsis

```
jobschchecktime -wt waittime [-lt maxwaittime] [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschchecktime command waits until the wait time specified by the option comes. If the wait time has already passed, this command (job) stops immediately. If the maximum wait time is specified by the option, a return value varies depending on whether the job was executed before or after the maximum wait time.

Wait times and maximum wait times are based on day change times. Network jobs use the day change time on the execution server.

This command cannot be used when the server is not running.

Option

-wt *waittime*

This option is used to specify the wait time by setting the hour and minute in two digits each between 0000 and 2359. Also consider the day change time, for example, if the day change time is set to 10 o'clock, the value 0900 represents 9:00 of the next day.

-lt *maxwaittime*

This option is used to specify the maximum wait time by setting the hour and minute in two digits each between 0000 and 2359. Also consider the day change time, for example, if the day change time is 10 o'clock and the value 0930 represents 9:30 of the next day. If the maxwaittime is earlier than the waittime, it fails.

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Wait times (-wt) based on day change times and command start times

Wait times (-wt) are based on day change times.

Day change times and wait times specified with -wt, as well as the results of starting the jobschchecktime command, have their relationships shown in the following diagram:
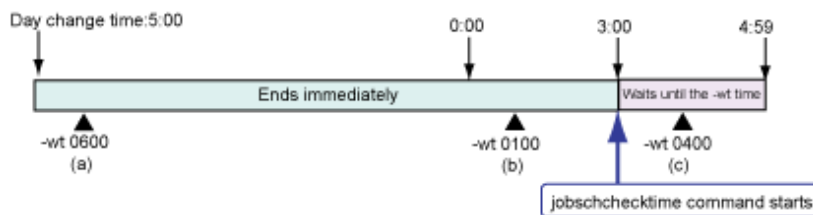
If the time when the jobschchecktime command starts is after the wait time specified by the -wt option, the command will end immediately. Also, if the time when the jobschchecktime command starts is before the wait time specified by the -wt option, the command will not be executed until the wait time.

So if the day change time is 5 o'clock, and the command start time is 3 o'clock, the behavior will be as follows.

   a. If the wait time is 6 o'clock (-wt 0600)

      The wait time has already passed, so the command will end immediately.

   b. If the wait time is 1 o'clock (-wt 0100)

      The wait time has already passed, so the command will end immediately.

   c. If the wait time is 4 o'clock (-wt 0400)

      It waits until 4 o'clock.

Wait times (-wt) and maximum wait times (-lt)

The following figure shows the relationship between the wait time specified by the -wt option and the maximum wait time specified by the -lt option:
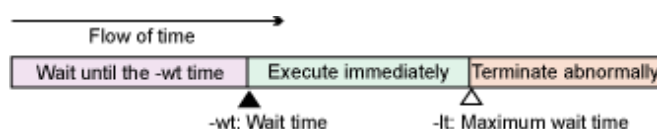
If the time when the jobschchecktime command starts is before the wait time specified by the -wt option, the command will not be executed until the wait time.

If the time when the jobschchecktime command starts is after the wait time specified by the "-wt" option but before the maximum wait time specified by the "-lt" option, the command will be executed immediately.

If the time when the jobschchecktime command starts is after the maximum wait time specified by the "-lt" option, the command will terminate abnormally with a return value of 1.

The maximum wait time (specified by the "-lt" option) can be used as follows:

JOB 1: Job that starts at 9:00 and takes several hours to complete
JOB 2: jobschchecktime -wt 1200 -lt 15:00
JOB 3: Job that must start between 12:00 and 15:00
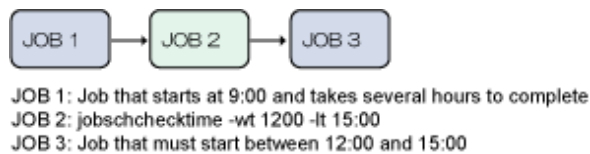
As can be seen from the diagram above, JOB3 needs to start sometime between 12:00 and 15:00. If the preceding job, JOB1, starts at 9:00 but takes several hours to complete, then a jobschchecktime command specifying the maximum wait time can be specified as JOB2 between JOB1 and JOB3.

If JOB1 finishes before 12:00, the command for JOB2 will not be executed until 12:00, so JOB3 will not start before 12:00. If JOB1 finishes between 12:00 and 15:00, the command for JOB2 will be immediately executed and terminated, so that JOB3 starts between 12:00 and 15:00. If JOB1 does not finish until after 15:00, then JOB2 will terminate abnormally and so JOB3 will not execute. This makes it possible to ensure that JOB3 will only ever start between 12:00 and 15:00.

Specifying wait times (-wt), maximum wait times (-lt) and day change times

Specify day change times, wait times (-wt), and maximum wait times (-lt) as shown in the following diagram:
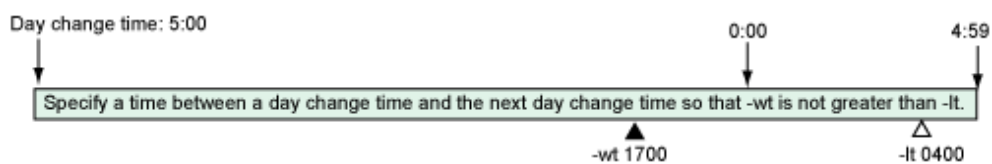
Values between 0000 and 2359 can be specified for -wt and -lt, but use the day change time as the basis for these specifications. Specify the wait time (the "-wt" option) to be before the maximum wait time (the "-lt" option), within the range determined by the day change time.

For example, if the day change time is 5:00 and "-wt 1700 -lt 0400" is specified, then 0400 will mean 4:00 the next day. If the "-lt" time comes first, as in "-wt 1700 -lt 1500", the command will produce an error and the job will terminate abnormally.

Return Values

0:

Indicates that any of the following has occurred.

- Started before the wait time and has waited until the wait time.

- Started after the wait time and the maximum wait time has not been specified.

- Started between the wait time and the maximum wait time.

1:

Indicates that a job has started after the maximum wait time.

21:

Indicates that there is an error in specification of options. If invalid option, incorrect values or improper combination of options are specified, this value is returned.

22:

Indicates that an error related to the system or the user environment occurred.

31:

Indicates that a Systemwalker Operation Manager environment related internal error occurred.

35:

Indicates other internal error occurred.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris Linux	/opt/FJSVJOBSC/bin
HP-UX	/opt/FHPJOBSCH/bin
AIX	/usr/FAIXJOBSC/bin

Example

To make the command wait until 10:00 and start no later than 11:00:

```
jobschchecktime -wt 1000 -lt 1100
```

Execution Result/Output Format

```
CT_WTIME=1000
CT_LTIME=1100
```

4.21 jobschchecklog Command

Synopsis

(1) To parse a file based on the end pointer:

```
jobschchecklog -f filename -e regexp [-k key] [-s] [-v] [-z] [-sys subsystem-number]⁽*¹⁾
```

(2) To delete the end pointer:

```
jobschchecklog -h -f filename [-k key] [-sys subsystem-number]⁽*¹⁾
```

(3) To parse the whole file not based on the end pointer:

```
jobschchecklog -a -f filename -e regexp [-s] [-v] [-z] [-sys subsystem-number]⁽*¹⁾
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschchecklog command parses a file in text format line by line to examine whether regular expression is contained or not.

You can specify from where to start checking by using the option: at either the beginning of the file, or the point immediately after the end point of the last parsing.

The end of parsing can be saved internally, as the end pointer based on the file name with the user name and the key. This end pointer can be deleted by using the option.

If a file contains lines in 2048 bytes or more, the lines after 2049 bytes will be ignored.

You cannot use this command when the server is not running.

Options

-a

Parses the entire file.

-e *regexp*

Specifies the regular expression to be used for parsing. The following special characters can be used:

^$.*+?[]|()

-f *filename*

Specifies the file to be parsed.

-h

Deletes the end pointer. The next parsing will start from the beginning of the file specified with *filename*.

-k *key*

Specifies a key to remember the end point. If omitted, *filename* will be used as the key.

-s

If the file does not exist, returns the same value as when the regular expression is not found.

-v

Outputs all the lines that contain the regular expression to standard output.

-z

Reverses the return values between when the regular expression was found and when it was not found.

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Return Values

0:

Under the synopsis (1) or (3), if the -z option is not specified it indicates that the regular expression has been found and if the -z option is specified it indicates that the regular expression has not been found. Under the synopsis (2), it indicates that the process has terminated successfully.

1:

Under the synopsis (1) or (3), if the -z option is not specified it indicates that the regular expression has not been found and if the -z option is specified it indicates that the regular expression has been found.

11:

Under the synopsis (1) or (3), it indicates that the file has not been found. This value will not be returned if the -s option is specified.

12:

Under the synopsis (1) or (3), it indicates that the regular expression is inappropriate. This value will be returned in such cases as when ([) and (]) are not corresponding (not by a matched pair of parentheses).

13:

Under the synopsis (1) or (3), it indicates that loading file has failed. This value will be returned if there are no access rights or the file is not an ordinary file.

21:

Indicates that there is an error in specification of options. When invalid option, incorrect values or improper combination of options are specified, this status value is returned.

22:

Indicates that an error regarding the system or user environment occurred.

31:

Indicates that an inter error regarding the Systemwalker Operation Manager environment occurred.

32:

Indicates that an internal error regarding saving the end point has occurred.

35:

Indicates other internal error occurred.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris Linux	/opt/FJSVJOBSC/bin
HP-UX	/opt/FHPJOBSCH/bin
AIX	/usr/FAIXJOBSC/bin

Example

To check if a line starting with the string "ABC" was written to C:\TEMP\file.txt:

```
jobschchecklog -f C:\TEMP\file.txt -e "^ABC.*" -k key1
```

Execution Result/Output Format

- Output example when the -v option is not specified:

```
CL_FILENAME=C:\TEMP\file.txt
CL_LOGCOUNT=1
CL_LOGTEXT=ABCDEF: logfile text
```

- Output example when the -v option is specified:

```
ABCDEF: logfile text
```

Cautions

- If any of the following conditions is met, it is determined that the file was replaced. So parsing (checking) must be performed from the beginning of the file after deleting the end point:

    - The size of the entire file is smaller than that of up to the end point in the file.

    - The character at the end point is not a line feed character.

- When the last line in the file does not contain a line feed character, parsing will end at one line before the last line if the -a operand is not specified. And if the -a option is specified, parsing will be performed until the end of the last line.

- If parsing will no longer be performed on the file, delete the end point by using the -h option.

4.22 chkfile / jobschchkfile Command

The chkfile / jobschchkfile command checks the existence of a file, and waits for the file to be created/updated/deleted/ resized.

Synopsis

[Windows]

```
chkfile [-fwait [-sw]] [-fdelete [-sd]] [-fsize|-fupdate] [-i <monitoring interval>] [-a]
drive:file name

chkfile [-wait|-update[-d:yyyymmdd][-t:hhmm]|-delete] [-i <monitoring interval>]
{drive:file name|-file:drive:list file name [-AND|-OR]}

chkfile [-z] drive:file name
```

[UNIX]

```
jobschchkfile [-fwait [-sw]] [-fdelete [-sd]] [-fsize|-fupdate] [-i <monitoring
interval>] file name

jobschchkfile [-wait|-update[-d:yyyymmdd] [-t:hhmm]|-delete] [-i <monitoring
interval>] {file name|-file: list file name [-AND|-OR]}

jobschchkfile [-z] file name
```

Options

-fwait

If the file does not exist, the command waits for it to be created. Nothing is returned if the file exists when the command starts. If the file is confirmed as having been deleted, a value is returned once the file is recreated.

You can combine this option with -fdelete, -fupdate, and -fsize to make the command wait. In this case, a value is returned when any of the conditions is satisfied.

-sw

Specify this option together with -fwait.

If you specify this option and the file exists when the command starts, a value is returned immediately.

-fdelete

If the file exists, the command waits for the file to be deleted. If the file does not exist when the command starts, nothing is returned. If the file is confirmed as having been created, a value is returned once the file is again deleted.

You can combine this option with -fwait, -fupdate, and -fsize to make the command wait. In this case, a value is returned when any of the conditions is satisfied.

-sd

Specify this option together with -fdelete.

If you specify this option and the file does not exist when the command starts, a value is returned immediately.

-fupdate

If the file exists, the command waits for the update date to be changed. If the file does not exist when the command starts, nothing is returned even if the file is created. If the file is confirmed as having been created, a value is returned once the update date is subsequently confirmed as having been changed.

You can combine this option with -fwait and -fdelete to make the command wait. In this case, a value is returned when any of the conditions is satisfied. You cannot combine this option with -fsize.

-fsize

The command waits for the file size to be changed. Nothing is returned if the file is created or deleted. If the file is updated but there is no change in size, nothing is returned. To make the command wait for the file to be updated without any change in size, use -fupdate.

You can combine this option with -fwait and -fdelete to make the command wait. In this case, a value is returned when any of the conditions is satisfied. You cannot combine this option with -fupdate.

-wait

If the file does not exist, the command waits for it to be created. If the file exists when the command starts, a value is returned immediately.

You cannot combine this option with -delete, -update, -fwait, -fdelete, -fupdate, or -fsize to make the command wait.

-delete

If the file exists, the command waits for the file to be deleted. If the file does not exist when the command starts, a value is returned immediately.

You cannot combine this option with -wait, -update, -fwait, -fdelete, -fupdate, or -fsize to make the command wait.

-update

If the file does not exist, the command waits for it to be created.

If the file exists, the command waits for the update date to be changed.

You cannot combine this option with -wait, -delete, -fwait, -fdelete, -fupdate, or -fsize to make the command wait.

-d:yyyymmdd

Specify this option together with -update.

Specify the reference date to be used to determine if the file was updated.

For example, for June 5 2014, specify -d:20140605.

If this option is omitted, the current date or later is assumed.

-t:hhmm

Specify this option together with -update.

Specify the reference time to be used to determine if the file was updated.

For example, for 13:35, specify -t:1335.

If this option is omitted, the current time or later is assumed.

-i interval

Specify the interval at which the status of the file will be checked, from 1 to 3600 seconds. If this option is omitted, the file status is checked at intervals of 1 second.

-a [Windows]

Use this option together with -fwait, -fupdate, or -fsize.

If the file status is checked, this command waits for access to the target file from other programs to cease. For example, if you specify -fwait to make the command wait for the file to be created and another program is creating (accessing) the file, the command does not terminate. If the file has been created when the status is checked after the next monitoring interval and other programs have ceased accessing the file, the command terminates.

-z

Checks the existence of the file. If the file does not exist, the command will terminate with return value 0. If you check the existence of the file without specifying this option and the file exists, the command will terminate with return value 0.

drive:file name [Windows]

Specify the full path name of the target file.

If none of the options -fwait, -fdelete, -fupdate, -fsize, -wait, -delete, or -update is set, specify the name of the file whose existence is to be checked.

You can specify the wildcard character "*" for all or part of the file name when you specify -fwait, -fdelete, -fupdate, or -fsize. Wildcard characters can be used only for the file name. They cannot be used to specify the drive name or directory name.

file name [UNIX]

Specify the full path name of the target file.

If none of the options -fwait, -fdelete, -fupdate, -fsize, -wait, -delete, or -update is set, specify the name of the file whose existence is to be checked.

You can specify the wildcard character "*" for all or part of the file name when you specify -fwait, -fdelete, -fupdate, or -fsize. But since the wildcard character "*" has a special meaning for the shell script, it must be escaped with a backslash (\). Wildcard characters can be used only for the file name. They cannot be used to specify the directory name.

-file:drive:list file name [Windows]

Use this option together with -wait, -delete, or -update.

Specify the name of a file (list file) that defines the target files.

If none of the options -wait, -delete, or -update is set, specify the name of the file that defines the files whose existence is to be checked.

You can list multiple file names in a list file.

You cannot specify wildcard characters in the list file name. Wildcard characters also cannot be specified in the file names defined in the list file.

-file:list file name [UNIX]

Use this option together with -wait, -delete, or -update.

Specify the name of a file (list file) that defines the target files.

If none of the options -wait, -delete, or -update is set, specify the name of the file that defines the files whose existence is to be checked.

You can list multiple file names in a list file.

You cannot specify wildcard characters in the list file name. Wildcard characters also cannot be specified in the file names defined in the list file.

-AND|OR

If a list file name is specified with the -file option, specify whether to target all files within the list file or to target a specific file.

If this option is omitted, the command targets all files within the list file. The following describes the behavior of the -wait, -delete, and -update options.

- -wait option:

-AND	Waits for all files within the list file to be created. If all files already exist, a value is returned immediately.
-OR	Waits for any one file within the list file to be created. If any file within the list file already exists, a value is returned immediately.

- -delete option:

-AND	Waits for all files within the list file to be deleted. If no files exist, the command will terminate abnormally with return value 1.
-OR	Waits for any file within the list file to be deleted. If there are already no files, a value is returned immediately.

- -update option:

-AND	Waits for all files within the list file to be updated or created.
-OR	Waits for any file within the list file to be updated or created.

Return Values

Return value	Meaning
0	The file was created/updated/deleted. Alternatively, the -z option was not specified and the file exists. Alternatively, the -z option was specified and the file does not exist.
1	The -z option was not specified and the file does not exist. Alternatively, the -z option was specified and the file exists.
10	An internal error occurred.

Command Location

Windows	Systemwalker Operation Manager installation directory \MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Example 1

- Example of waiting for a file with a name starting with "file" in the d:\temp directory to be created or deleted [Windows]

```
chkfile -fwait -fdelete d:\temp\file*
```

- Example of waiting for a file with a name starting with "file" in the /var/tmp directory to be created or deleted [UNIX]

```
jobschchkfile -fwait -fdelete /var/tmp/file\*
```

Example 2

Example of waiting for all files (c:\a.txt, d:\b.txt, d:\c.txt) defined in d:\temp\list_file to be updated at or after 13:30 on December 5, 2014 [Windows]

```
chkfile -update -d:20141205 -t:1330 -file:d:\temp\list_file -AND
```

[Example of d:\temp\list_file]

```
c:\a.txt
d:\b.txt
d:\c.txt
```

Example of waiting for all files (/home/a.dat, /home/b.dat, /home/c.dat) defined in /var/tmp/list_file to be updated at or after 13:30 on December 5, 2014 [UNIX]

```
jobschchkfile -update -d:20141205 -t:1330 -file:/var/tmp/list_file -AND
```

[Example of /var/tmp/list_file]

```
/home/a.dat
/home/b.dat
/home/c.dat
```

Information

• •

Registering commands as jobs

To automatically start job nets when a file is created/updated/deleted/resized, schedule the chkfile and jobschchkfile commands as jobs. By registering a command as the first job in a job net, you can make later processes wait for a file to be created/updated/deleted/resized.

• •

Execution Result/Output Format

- Output example of chkfile [Windows]

```
CF_FILENAME=d:\temp\file1130
CF_CHKTYPE=fwait
CF_FILESIZE=1024
CF_FILEYEAR=2014
CF_FILEDATE=1130
CF_FILETIME=2103
```

- Output example of jobschchkfile [UNIX]

```
CF_FILENAME=/var/tmp/file1130
CF_CHKTYPE=fwait
CF_FILESIZE=1024
CF_FILEYEAR=2014
CF_FILEDATE=1130
CF_FILETIME=2103
```

Cautions

- If you register this command with the -fwait, -fdelete, -fupdate, -fsize, -wait, -delete, or -update option as a job with Jobscheduler, the command continues executing until the desired file is created/updated/deleted/resized. In the submission queue, check the limitations on the job elapsed time and the number of jobs that can be executed simultaneously, and change the settings for the queue if there is an issue.

- You can use the -a option with this command to wait until access to a file is complete. However, it cannot be guaranteed that file transfer will be completed with transfer programs that temporarily cease access during file transfer. In this type of situation, send a small file reporting transfer completion after the target file has been sent. Then wait for the update of this file to confirm that the target file has finished transferring. [Windows]

- This command checks the status of files at the monitoring interval specified by the -i option. It detects changes in the file by comparing its state when it is monitored one time to the next. Changes will not be detected properly if files are created, deleted, updated, or resized at intervals shorter than the monitoring interval. If a file is created and then immediately deleted in the interval between one monitoring and the next, neither the creation nor the deletion of the file can be detected.

- If changes in multiple files are detected within one monitoring interval, the information on the file detected first will be output.

- Specify one file name per line in the list file. A file name cannot include spaces or tabs.

- On Solaris, HP-UX, AIX, and Linux x86, ensure that the target file does not exceed 2 GB.

4.23 jobschnetmemo Command

Synopsis

(1) To register

```
jobschnetmemo -ent [project name/] job net name memo contents [-sys subsystem-number](*1)
```

(2) To delete

```
jobschnetmemo -del [project name/] job net name [-sys subsystem-number](*1)
```

(3) To output contents

```
jobschnetmemo -out [project name/] job net name [-sys subsystem-number](*1)
```

(4) To search

```
jobschnetmemo -find [-p project name] -e regexp [-m] [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschnetmemo command registers/deletes a memo to/from the job net. It also outputs the contents of the memo.

When a search is performed, a list of job nets is output if they have memos registered containing the regular expressions.

Options

project name (to register or delete)

Specify the project name where the job net information is to be stored. This can be omitted if there is only one project that has job nets registered and that the execution user can operate or update.

project name (to output contents)

Specify the project name where the job net information is to be stored. This can be omitted if there is only one project that has job nets registered and that the execution user can refer to.

-p project name (to search)

Specify the project name where the job net information is to be stored. If this is omitted, searches are performed on all job net information in the projects that can be referenced by the user executing the jobschnetmemo command.

job net name

Specify a job net that you want to register, delete or output its contents.

memo contents

Specify character strings to be registered as a memo within 128 bytes. Any user-defined characters cannot be contained.

-e regexp

Specify no more than 256 bytes. The special characters ^$.*+?[]|() can be used.

-m

Memo content is output after the job net name, delimited by spaces. If this option is omitted, only the project name and job net name are output.

<kbd>EE</kbd>

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Return Values

0:

Indicates that the process (register/delete/output content) ended normally. For searches, this indicates that one or more job nets that have memos registered containing the regular expressions were found.

1:

For searches, this indicates that no job nets that have memos registered containing the regular expressions were found.

11:

Indicates that the project does not exist or that access permissions are inadequate. If the project was omitted, it means that there are no projects with access permissions, or that there are multiple projects with access permissions.

12:

Indicates that the specified job nets were not found with register/delete/output content.

13:

Indicates that memo content was invalid with register. This return value occurs if the memo content is too long or includes invalid characters.

14:

For searches, this indicates that the regular expression was invalid. This return value occurs if the regular expression is too long or is invalid.

21:

Indicates that an invalid option was specified. This return value occurs if an invalid option was specified or the option combination is invalid.

22:

Indicates that a system error or error with the user's environment occurred.

31:

Indicates that a Systemwalker Operation Manager environment error occurred.

32:

Indicates that reading or writing of job net information failed.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Cautions

- If the regular expression includes spaces or special characters, enclose the regular expression with double quotation marks. When including double quotations in the regular expression, put the escape character "\" before the double quotation mark so that the double quotations can be treated as a character. In this way, double quotation marks escaped with the "\" will not be treated as one of the quotation marks enclosing the regular expression.

Example

An example of registering is shown below:

```
jobschnetmemo -ent GeneralAffairs/payroll PayrollAndBonusCalc -sys 1
```

An example of deleting is shown below:

```
jobschnetmemo -del GeneralAffairs/payroll -sys 1
```

An example of outputting content is shown below:

```
jobschnetmemo -out GeneralAffairs/payroll -sys 1
```

An example of searching is shown below:

```
jobschnetmemo -find -p GeneralAffairs -e "Data.*Management" -m -sys 1
```

Output Example

An example of outputting content is shown below.

```
PayrollAndBonusCalc
```

An example of searching performed with the -m option specified is shown below:

```
GeneralAffairs cashier CashierDataManagement
GeneralAffairs sales SalesDataManagement
(snip)
```

4.24 jobschconvert Command

Synopsis

```
jobschconvert [-sys {all | subsystem-number}](*1)
```

*1:

[-sys {all | subsystem-number}] is the option of Systemwalker Operation Manager EE.

Description

The jobschconvert command converts the Jobscheduler database of the earlier version into the database that can be used in the current version.

Options

-sys {all|subsystem-number}

In the system with multiple subsystems, this option is used to specify a subsystem as operating target. If omitted, the default is subsystem 0.

all

All subsystems are specified as operation target.

subsystem-number

Specify operation target subsystem number using a range of 0 to 9.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Cautions

- System administrator (user belonging to the Administrator group or superuser) privileges are required. General users cannot use the jobschconvert command.

- This command can be activated only when the following Systemwalker Operation Manager components are in paused status:

    - Jobscheduler service/daemon

    - Calendar service/daemon

- Calendars with the name containing special characters such as /?*;,\"<>|: cannot be migrated.

- Prior to conversion, a character code is automatically determined.[UNIX]

- If you use this command for the database that is available in this version, only the character codes contained in the database will be converted.

4.25 jobschchkmsgevt Command

Synopsis

```
jobschchkmsgevt message event name [-i <monitoring interval>]
```

Description

The jobschchkmsgevt command is called by Wait for Message Event jobs.

It waits for message events (message events generated without a project name or job net name specified, and message events generated with a project and job net specified) generated for job nets where this command is executed as a job.

This command waits for messages generated after the command execution starts, and it ends when a message event being waited for is generated.

Options

message event name

Specify, using up to 12 bytes, the name of the message event that you want to wait for.

The name cannot contain a comma (,), space, or user-defined character.

-i interval

Specify the interval, from 1 to 3600 seconds, for checking the generation of message events. If this option is omitted, this check will be performed at an interval of 60 seconds.

Return Values

Return value	Meaning
0	Terminated normally
Other than 0	Terminated abnormally

Command Location

Windows	Systemwalker Operation Manager installation directory \MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Cautions

- This command can be executed only if it is registered as a Systemwalker Operation Manager job (Wait for Message Event job). It cannot be executed directly from the command prompt or by any means other than as a Systemwalker Operation Manager job.

- This command cannot wait for message events generated for anything other than the job net where this command is executed.

- This command waits only for message events generated in schedule servers. It cannot be executed in network jobs or Distributed Execution jobs.

- There is no limit on the number of simultaneous executions of this command, but if a large number of instances of this command (Wait for Message Event job) is executed, there may be a delay before the command (job) ends. This is because access is concentrated on the message event occurrence history file.

- If a message event occurs multiple times in the same monitoring interval, the message event that occurred first is output in the execution result.

Example

An example job definition that waits for message event "ABC" is shown below.

```
jobschchkmsgevt ABC
```

Execution Result/Output Format

```
CM_MSGEVENT=ABC
```

4.26 jobschchknetvar Command

Synopsis

```
jobschchknetvar varname [-num operator,value[,operator2,value2] | -char comp
method,char string] [-z]
```

Description

The jobschchknetvar command is called by jobs that check the values of job net variables.

Specify a condition against which the value of the specified job net variable is compared. Values can be compared as a numeric value or a string.

If you do not specify a condition, this command will check if the variable exists and information on the job net variable will be displayed in the standard output. If a variable with the target variable name does not exist, the condition will not be matched.

Information about the job net variable being checked is displayed in standard output in the format "varname=value". If the variable being checked does not exist, nothing is displayed.

Options

varname

Specify the name of the job net variable to be evaluated using up to 64 bytes.

The format of the job net variable name is <job net variable prefix>.<name output to job output information ("NAME" in the "NAME=VALUE" format)>.

If neither -num nor -char is specified, specify the name of the variable for checking the existence of the variable. If it exists, the condition is met. If the specified variable exists and the value is an empty string, the variable will be treated as if it exists.

-num operator,value[,operator2,value2]

Treat the value of the job net variable as a numeric value for comparison.

You can compare the job net variable against a single value or a range of values.

To compare against a single value, specify only "operator" and "value".

To compare against a range of values, also specify "operator2" and "value2". The job net variable is compared against the range from "value" to "value2" inclusive.

- For numeric comparison against a single value:

    The following six "operators" are available:

    - EQ (EQual)

        Equal to "value"

    - NE (Not Equal)

        Not equal to "value"

    - GT (Greater Than)

        Greater than "value"

    - GE (Greater than or Equal to)

        Greater than or equal to "value"

- LT (Less Than)

   Less than "value"

- LE (Less than or Equal to)

   Less than or equal to "value"

Specify an integer from -2147483648 to 2147483647 as the value that is compared with the variable value.

- For numeric comparison against a range of values

   Specify one of the following for "operator1":

   - GT (Greater Than)

      Greater than "value1"

   - GE (Greater than Equal to)

      Greater than or equal to "value1"

   Specify one of the following for "operator2":

   - LT (Less Than)

      Less than "value2"

   - LE (Less than Equal to)

      Less than or equal to "value2"

Specify an integer from -2147483648 to 2147483647 for "value1" and "value2". Ensure that "value1" is less than "value2".

-char char comp method,char string

Treat the value of the job net variable as a string for comparison.

Specify one of the following comparison conditions:

- EQ (EQual)

   The value of the variable is equal to the comparison string

- NE (Not Equal)

   The value of the variable is not equal to the comparison string

- IN (INclude)

   The value of the variable includes the comparison string

- NI (Not Include)

   The value of the variable does not include the comparison string

Specify a string of up to 512 bytes excluding user-defined characters.

If nothing is entered for the comparison string, behavior will be as follows:

- When the comparison condition is EQ:

   The condition is met if the value of the variable is an empty string.

- When the comparison condition is NE:

   The condition is met if the value of the variable is other than an empty string.

- When the comparison condition is IN:

   The condition is regarded as not being met.

- When the comparison condition is NI:

   The condition is always met.

-z

Reverses the return values of the comparison.

- When the -z option is not specified

  If the condition is met, 0 is returned. If the condition is not met, 1 is returned.

- When the -z option is specified

  If the condition is met, 1 is returned. If the condition is not met, 0 is returned.

Return Values

Return value	Meaning
0	Terminated normally

The -z option was not specified and the condition was met.

The -z option was specified and the condition was not met. |
| 1 | Terminated normally

The -z option was not specified and the condition was not met.

The -z option was specified and the condition was met. |
| 2 or more | Terminated abnormally |

Command Location

Windows	Systemwalker Operation Manager installation directory \MpWalker.JM\bin
Solaris
HP-UX
AIX
Linux	/usr/bin

Cautions

- This command can be executed only if it is registered as a Systemwalker Operation Manager job. It cannot be executed directly from the command prompt or as anything other than a Systemwalker Operation Manager job.

- This command uses the job net variable set on the schedule server in the comparison. It cannot be executed in network jobs or Distributed Execution jobs.

- Job net variables automatically set by the system cannot be specified.

Example

An example job definition that checks if the value of job net variable "JOB.ABC" is equal to string "xyz" is shown below:

```
jobschchknetvar JOB.ABC -char EQ,xyz
```

Execution Result/Output Format

```
JOB.ABC=xyz
```

4.27 jobschsleep Command

```
jobschsleep wait_time
```

Description

The jobschsleep command waits for the time period specified in wait_time.

Options

wait_time

Specify the time to wait between 0 (seconds) and 86400 (seconds). If 0 is specified, a value is returned immediately.

Return Values

Return value	Meaning
0	Terminated normally
Other than 0	Terminated abnormally

Command Location

Windows	Systemwalker Operation Manager installation directory \MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Example

An example of waiting for 600 seconds is shown below.

```
jobschsleep 600
```

Execution Result/Output Format

```
SL_SLEEPSEC=600
```

Chapter 5 Jobscheduler Commands (For Printing)

This chapter describes the Jobscheduler commands (for printing) provided by Systemwalker Operation Manager. Note that the commands must be executed on the server where the Systemwalker Operation Manager server function is installed.

Note

If commands are executed in an environment where the UAC function for Windows Server 2008 or later has been enabled, then a dialog box asking a user's privileges to be promoted may be displayed depending on the privileges of the user that logged in to the operating system, and action may be required.

Refer to "Notes on Executing Commands in Windows [Windows version]" in the *Systemwalker Operation Manager User's Guide* for more information.

Note

Notes when you execute Jobscheduler commands

When executing a command, make sure the following conditions are satisfied:

- The standard output and standard error output should not be closed.

The commands can be executed by a user who is already logged on to the Windows server/UNIX server.

If you execute a command to be registered as a job in the Jobscheduler, the default project name of this command will be the name of project that contains job nets on which the command is registered.

5.1 List of Commands

The following are the commands used in the Jobscheduler function (for printing):

jobschprint command

Outputs the information managed in the Jobscheduler to the standard output (stdout). The jobschprint command has the following options.

For options having an upper-case character in parentheses added, the processing content will be the same in either case. However, note that the command process will differ between the -R option and -r option.

-j(-J):

Outputs a list of job nets and jobs.

-n(-N):

Outputs the current state of job nets.

-a(-A):

Outputs the property information of job nets.

-e(-E):

Outputs a list of running jobs.

-f(-F):

Outputs the history information of job nets.

-g(-G):

Outputs the history information of job nets by date.

-i(-I):

Outputs the registration information of jobs.

-l(-L):

Outputs an application plan list of job nets. In this list, comments of job nets are also output.

-m(-M):

Outputs an application plan list of job nets. In this list, comments of job nets are not output.

-p(-P):

Outputs the current state of job nets.

-q(-Q):

Outputs a list of job nets connecting to job net startup queue. [Windows]

-R:

Outputs the group information in the control statement format.

-r:

Outputs the job net information in the control statement format.

-rh:

Outputs the calendar holiday information in the control statement format.

-rsp:

Outputs the schedule pattern information in the control statement format.

-s(-S):

Outputs a list of startup days (for three years) of job nets.

-w(-W):

Outputs a list of waiting jobs.

-x(-X):

Outputs the current state of groups.

-y(-Y):

Outputs a list of job nets in a group.

-b:

Outputs a list of schedule pattern names.

-c:

Outputs startup days in the schedule pattern set in the specific dates.

-d:

Outputs startup days in the schedule pattern set for days of the specific week.

-h:

Outputs the calendar information.

-u:

Outputs the information of users who use Jobscheduler

-v:

Outputs a list of calendar names.

-o(-O):

Outputs a list of projects.

> ## 📝 Note
> ....
>
> The format of date and time in the output result using the jobschprint command is:
>
> [Windows]
>
> Follows the specification value in the control panel on the Windows-based server.
>
> [UNIX]
>
> mm/dd/yyyy
> ....

jobschprintcsv command

Outputs the information managed with Jobscheduler to standard output (stdout) in CSV format.

The jobschprintcsv command has the following options:

-n:

Outputs the current state of job nets.

-g:

Outputs the history information of job nets by date.

-l:

Outputs an application plan list of job nets.

-f:

Outputs the history information of job nets by relative date.

5.2 jobschprint -j(-J) Command

Synopsis

```
jobschprint {-j|-J} [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -j(-J) command outputs a list of registered job nets and jobs.

Option

-j|-J

Outputs a list of job nets and jobs which can be referenced by the user of executing the command.

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX	/usr/bin

| AIX
Linux	

<u>**Output Items**</u>

The following explains the items to be output.

User Name

Name of the user who executes the jobschprint command

Project Name

Project name

Job Net

Job net name

Job Name

Command (command name, batch file name/shell script name, program name, or JCL name of Job Execution Control) that is registered as a job

The job net name of the child job net is output if a child job net is output as a job.

The job net name of the linked job net is output if a linked job net is output as a job.

The batch job definition name is output if a batch framework job is output as a job.

This is output if the execution attribute is "Job Execution Control" or "Normal attribute". [UNIX]

Job Param

Parameter to be passed to command

This is not output if a batch framework job is output as a job.

This is output if the execution attribute is "Job Execution Control".

Job Name JES

Job name of the Job Execution Control

This is not output if the execution attribute is "Interstage". [UNIX]

Job Comment

Job comment

The job net comment of the child job net is output if a child job net is output as a job.

The job net comment of the linked job net is output if a linked job net is output as a job.

Previous Job Name

Name of preceding job

If a preceding job is the child job net, a job net name of the child job net that is the preceding job is output.

If a preceding job is the linked job net, a job net name of the linked job net that is the preceding job is output.

This is output if the execution attribute is "Job Execution Control" or "Normal attribute". [UNIX]

Previous Job End Code

Displays the execution condition according to the completion code of preceding job. The output format varies depending on the execution conditions, which are "By value" or "By range". For information on the judgment operators, see "4.9.2 Job Control Statement."

When the execution condition is "By value":

Judgment operator Comparison value

```
Previous Job End Code: GT 10
```

When the execution condition is "By range":

Judgment operator 1 Comparison value 1 and Judgment operator 2 Comparison value 2

```
Previous Job End Code: GT 10 and LT 20
```

Job Attr

Mode of recovery job

This is output only when a job is the recovery job. [Windows]

Recovery Only:

Recovery job

Recovery Rerun:

Preceding job is rerun when this recovery job is complete.

Work Unit Name

Workunit name that is registered as a job

This is output if the execution attribute is the "PowerAIM" or "Interstage". [UNIX]

Previous System Name

Name of preceding business system

This is output if the execution attribute is the "PowerAIM". [UNIX]

System Name

Business system or system name

This is output if the execution attribute is the "PowerAIM" or "Interstage" attribute. [UNIX]

Previous Work Unit Name [UNIX]

Name of preceding workunit

This is output if the execution attribute is the "PowerAIM".

Execution Day Mode

Indicates whether or not (ON/OFF) the "Valid only on startup days" on job (child job net or linked job net) is specified.

It is output if the icon type is the child job net or linked job net.

ON

"Valid only on startup days" is specified.

OFF

"Valid only on startup days" is not specified.

Start Job

Indicates whether or not the job (child job net or linked job net) is targeted to start as a result of startup operation or when execution conditions are met.

It is output if the icon type is the child job net or linked job net and "Valid only on startup days" is set to "ON".

ON

Targeted to execute

OFF

Not targeted to execute

Rerun Job

Indicates whether or not the job (child job net or linked job net) is targeted to rerun when re-execution is performed.

It is output if the icon type is the child job net or linked job net and "Valid only on startup days" is set to "ON".

ON

Targeted to rerun

OFF

Not targeted to rerun

Or Mode

Indicates whether or not (ON/OFF) "Wait for preceding job using OR condition" is specified.

ON

"Wait for preceding job using OR condition" is specified.

OFF

"Wait for preceding job using OR condition" is not specified.

This is output if the execution attribute is "Job Execution Control".

Output Example

An example "Job List" which is an output result of this command, is shown below.

```
Job List                         date: 02/25/2015 time: 15:20

User Name: johnson


Project Name: GeneralAffairsDept


Job Net: selling


Job Name: management1.bat
Job Param: -file data1.dat
Job Name JES: management1
Job Comment: CustomerInquiry
Job Name: management2.bat
Job Param: -file data2.dat
Job Name JES: j123
Job Comment: CreditDetails
Previous Job Name: management1.bat
Previous Job End Code: GT 10

```

```
Project Name: GeneralAffairsDept

Job Net: customer

Job Name: management.bat
Job Param: -file data2.dat
Job Name JES: j124
Job Comment: CustomerManagement


Job Net: accounting
```

The configuration of the job net in the above example is shown below.
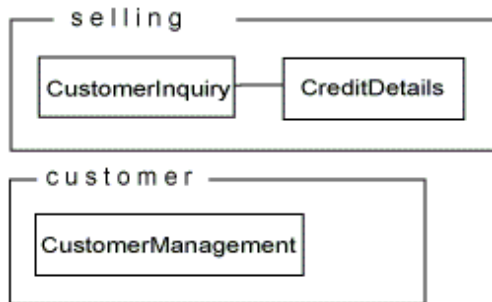
5.3 jobschprint -n(-N) Command

Synopsis

```
jobschprint {-n|-N} [-toplevel] [-long] [-sys subsystem-number](*1)
```

*1:

   [-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -n(-N) command outputs the current state of registered job nets.

Options

-n|-N

Outputs the current state of the job net that can be referenced by the user of executing the command.

-toplevel

This option outputs only single-level job nets or the first level of multi-level job nets. It does not output child job nets or linked job nets.

-long

This option outputs the full value of the "Job Net" and "Job Net Comment" output items.

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin

Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the items to be output.

User Name

Name of the user who executes the jobschprint command

Project Name

Project name

Job Net

Job net name

The first 8 bytes of the job net name are displayed. If the "-long" option is specified, all 50 bytes of the job net name will be displayed.

Job Net Comment

Job net comment

The first 24 bytes of the job net comment are displayed. If the "-long" option is specified, all 100 bytes of the job net comment will be displayed.

Status

Status of job net

The relevant status is selected from the following thirteen and displayed:

Normal:

Completed

Pseudo:

Pseudo-normal

Abend:

Abended

Forced:

Canceled

Warning:

Warning

Execute:

Executing

Wait:

Waiting

Pause:

Paused

Disable:

    Disabled

Coped:

    Confirmed

CarryOver:

    Carried over

StartDelay:

    Start delayed.

EndDelay:

    End delayed.

Note that the pseudo-normal is displayed as the completed if pseudo-normal is disabled in the [Define Jobscheduler Startup Parameters] window.

Scheduled Time

Scheduled start time

Start Time

Start date and time of the job net's previous execution

End Time

End date and time of the job net's previous execution

Completion Code

Completion code of the job net

<u>Cautions</u>

- When the Master Schedule Management function is enabled, use the stemprint -n(-N) command to output the current state of the job net. Using the stemprint -n(-N) command enables you to narrow down the process dates or projects that you want.

  For more information on the stemprint -n(-N) command, see "6.13 stemprint -n(-N) Command".

<u>Example</u>

```
jobschprint -n -toplevel -sys 2
```

<u>Output Example</u>

An example "Job Net Status List" which is an output result of this command, is shown below.

```
Job Net Status List       date: 10/25/2015   time: 16:28

User Name: johnson

Project Name: Sales

Job Net    Job Net Comment          Status    Scheduled Time    Start Time           End Time            Completion Code
selling    SalesManagement          Abend                       10/25/2015 10:22    10/25/2015 10:38    10
customer   CustomerSalesManagement  Wait      10/25/2015 18:00
sales      CreateWeeklySalesReport  Disable   10/26/2015         10/18/2015 23:00   10/18/2015 23:12


Project Name: GeneralAffairs

Job Net    Job Net Comment          Status    Scheduled Time    Start Time           End Time            Completion Code
cashier    CashierDataManagement    Execute   10/28/2015        10/25/2015 16:00
payroll    PayrollAndBonusCalc      Normal    11/10/2015        10/10/2015 23:00    10/10/2015 23:50    0
```

5.4 jobschprint -a(-A) Command

Synopsis

```
jobschprint {-a|-A} [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -a(-A) command outputs the property information of registered job nets.

Options

-a|-A

Outputs the property information of job net that can be referenced by the user of executing the command.

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Cautions

Information regarding monthly calendar, annual calendar, temporary changes will not be output.

When you want to collect detailed information, use the jobschprint -r command.

Output Items

The following explains the items to be output.

User Name

Name of the user who executes the jobschprint command.

Project Name

Project name

Job Net

Job net name

Job Net Comment

Job net comment

<No Execution>

Indicates whether or not (ON/OFF) the execution condition of "None" is specified.

ON:

"None" is specified.

OFF:

"None" is not specified.

<Single Time>

Indicates whether or not (ON/OFF) the execution condition of "Single Time" is specified.

ON:

The "Single Time" is specified.

In this case, the following items are displayed:

Day Change Time:

A time in which dates are updated

Start Time:

The time that was specified as the job net execution time

Stop Time:

The time that was specified as the job net stop time

This is output if the execution attribution is Interstage.

Alarm Time:

The time that was specified as the estimated end time for the job net

OFF:

The "Single Time" is not specified.

<Multi Time>

Indicates whether or not (ON/OFF) the execution condition of "Multiple Times" is specified.

ON:

The "Multiple Times" is specified.

In this case, the following items are displayed:

Day Change Time:

A time in which dates are updated

Multi Time1 - 6:

The times (up to six different points in time) at which the job net is to be executed.

OFF:

The "Multiple Times" is not specified.

\<Interval\>

Indicates whether or not (ON/OFF) the execution condition of "Fixed Interval" is specified.

ON:

The "Fixed Interval" is specified.

In this case, the following items are displayed:

Day Change Time:

A time in which dates are updated

Start Time:

The time that was specified as the start time of job net execution

End Time:

The time that was specified as the end time of job net execution

Interval:

Displays a startup interval.

OFF:

"Fixed Interval" is not specified.

\<IPL\>

Indicates whether or not (ON/OFF) the execution condition of "at the time of server startup" is specified.

ON:

The "at the time of server startup" is specified.

In this case, the following item is displayed:

Day Change Time:

A time in which dates are updated

OFF:

The "at the time of server startup" is not specified.

\<MSG\>

Indicates whether or not (ON/OFF) the execution condition of "Message" is specified.

ON:

"Message" is specified.

In this case, the following items are displayed:

Day Change Time:

A time in which dates are updated

Wait Condition:

The wait condition for a message event. "AND" or "OR" appears on the display. The AND condition starts the job net if all the registered message events are generated. The OR condition starts the job net if at least one registered message event is generated.

Mode (order):

"ON" appears if "Valid during job net execution" is specified, and "OFF" appears if not. If ON is specified, the job net is executed as many times as the message events were generated during execution.

Mode (Execution Day):

"ON" appears if "Valid only on startup days" is specified, and "OFF" appears if not. If a message event occurs on the scheduled startup day of the job net, the job net having the execution condition of "Valid only on startup days" selected can be started.

Mode (Time):

"condition with the startup time" is displayed. For the job net that starts when the startup time is reached and the relevant message event is generated, "AND" is displayed. For the job net that starts when either of the execution conditions of "start time is reached" or "the awaiting message event has occurred" is met, "OR" is displayed.

Carry Over:

If "Carry over the process regardless of the day change time" is selected, "ON" is displayed.

Message:

The registered message events are displayed. An asterisk (*) preceding the message indicates that the message already occurred. And the asterisk disappears when the job net has been executed.

OFF:

"Message" (as the execution condition) is not specified.

<Exec_Attr>

"execution attribution" specified when the job net was registered.

Normal:

The Jobscheduler executes the job net. [UNIX]

JES:

The Job Execution Control executes job net.

Interstage:

Interstage executes the job net.

PAIM:

PowerAIM executes the job net. [UNIX]

In addition, if the execution attribution is "JES" or "Interstage", the following item is displayed.

Host:

Default host name [UNIX]

<Copy Start >

Specify to execute 'Copy Start' or not when job net start.

ON:

Execute 'Copy Start'.

OFF:

Not execute 'Copy Start'.

Save Number:

Display save number of copy destination job net.

<Default_Host>

Default host name [Windows]

\<Power Mode\>

The job net's action to be applied when power is off

Indicates the job net's action in case that the Jobscheduler service is activated after the startup time of the job net is overdue under the condition where the execution condition is set to "Single".

ON:

Job net is started when the Jobscheduler service is activated.

OFF:

Job net is not started even when the Jobscheduler service is activated.

\<IPL Mode\>

The job net's action to be applied at server startup

Indicates the action when the Jobscheduler service is activated more than once a day under the condition where the execution condition is set to "at the time of server startup".

Everytime:

The job net is started every time the Jobscheduler service is started.

Once:

Even if the Jobscheduler service is started more than once, the job net is started only once when the service is activated for the first time.

\<MSG Mode\>

Indicates the action when the awaited message event is generated more than once a day under the condition where the execution condition of "Message" is set.

Everytime:

The job net is started every time the awaited message events are generated.

Once:

Even if the awaited message events are generated more than once a day, the job net is started only once when the message event is generated for the first time.

In case of "ONCE", the following item is displayed.

Msg Start Time:

Time specified as the scheduled start time of job net

If omitted, a blank will be displayed.

\<Estimate Time\>

Estimated processing time (in minutes) of job net

\<Standard Job Net\>

"ON" appears if the job net is declared as the base job net.

\<Schedule Pattern\>

Indicates whether or not (ON/OFF) the "Schedule pattern" is specified in the Basic information.

ON:

"Schedule pattern" is specified.

In this case, the following item is displayed:

Pattern Name:

>Schedule pattern name that is specified as the basic information

OFF:

"Schedule pattern" is not specified.

<Schedule Pattern Mode>:

Indicates a method of combining schedule patterns when more than one schedule pattern is specified. The contents displayed mean as follows:

AND:

Sets the days as startup days if they are set as startup days in all the schedule patterns.

OR:

Sets the days as startup days if they are set as startup days in at least one of the schedule patterns.

<Base Job Net>

Indicates whether or not (ON/OFF) "Another job net" is specified in the Basic information.

ON:

"Another job net" is specified.

In this case, the following item is displayed:

Base Job Net Name:

>Base job net name when "Another job net" is specified in the Basic information.

OFF:

"Another job net" is not specified.

<Annually Calendar>

Indicates whether or not (ON/OFF) "Yearly" is specified in the Basic information.

ON:

"Yearly" is specified.

OFF:

"Yearly" is not specified.

<Monthly Calendar>

Indicates whether or not (ON/OFF) "Monthly" is specified in the Basic information.

ON:

"Monthly" is specified.

OFF:

"Monthly" is not specified.

<Offset Mode>

A method for counting the number of days shifted from the standard date.

Execday:

Counts the number of offset days based on working days.

Day:

Counts the number of offset days on the calendar basis.

Offset Day:

The number of offset days shifted from the standard date.

Offset Holiday:

Indicates the treatment of the base day when it falls on a holiday (ON/OFF)

ON:

Sets the base day as a working day if it falls on a holiday.

OFF:

Sets the base day as a holiday if it falls on a holiday.

\<Holiday Calendar\>

Indicates whether or not (ON/OFF) "Calendar" is specified for the holiday information.

ON:

"Calendar" is specified.

OFF:

"Calendar" is not specified.

When "Calendar" is specified, the following item is displayed:

Calendar Name:

The name of the calendar that was specified when registering the job net

\<Holiday Mode\>

Indicates a method of combining calendars when more than one calendar is specified. The contents displayed mean as follow:

AND:

Sets the days as holidays if they are set as holidays in all calendars.

OR:

Sets the days as holidays if they are set as holidays at least one of the calendars.

\<Holiday Shift\>

Indicates the handling and the operation period of holidays. The contents displayed mean as follow:

Before Shift:

Shifts the job net startup day to the preceding day.

After Shift:

Shifts the job net startup day to the next day.

Cancel:

The current day is recognized as a holiday, so the job net is not executed.

Ignore:

The current day is recognized as a startup day, so the job net is executed.

Shift Limit:

Displays the number of days, on which execution is suspended because the Holiday shift is performed on the consecutive holidays.

Enable Term:

The validity period for the specified startup day

Disable Term

The invalidity period for the specified startup day

<Business Pattern>

Indicates whether or not (ON/OFF) "Business days" is specified in the Basic information.

ON:

The "Business days" is specified.

In this case, the following items are displayed:

Business Base:

Month's start date that you specified

The contents displayed mean as follow:

1 to 28:

Indicates 1st to 28th of month.

LAST:

Indicates the end of month.

Business Day:

Business days that you specified

If more than one business day is specified, the business days as many as you specified are displayed. No business day is specified, nothing is displayed.

1 to 31:

The business days that are specified by counting from the 1st_business_day (the first business day of the month)

-1 to -31:

The business days that are specified by calculating back from the -1st_business_day (the last business day of the month)

Business Limit:

Indicates whether or not (ON/OFF) "Set only business days of the month as startup days" is specified.

ON:

"Set only business days of the month as startup days" is selected.

OFF:

"Set only business days of the month as startup days" is not selected.

OFF:

The "Business days" is not specified.

<Carryover Mode>

Shows the instructions defined for the carried over job net. This can be displayed only when daily schedule management is performed in the Master Schedule Management environment.

CANCEL:

"Cut off" is specified.

OVER:

"Wait until exit" is specified.

PARALLEL:

"Run concurrently" is specified.

ACTION:

"Handle manually" is specified.

Output Example

An example "Information of Job Net" which is an output result of this command, is shown below.

```
Information of Job Net           date: 02/25/2015 time: 18:58


User Name: johnson


Project Name: SalesDepartment


Job Net: sales
Job Net Comment: SalesManagement
<No Execution>: OFF
<Single Time>: ON
  Day Change Time: 00:00
  Start Time: 08:00
  Alarm Time: 10:00
<Multi Time>: OFF
<Interval>: OFF
<IPL>: OFF
<MSG Mode>: OFF
<Estimate Time>: 60
<MSG>: OFF
<Exec_Attr>: JES
<Copy Start>: ON
Save Number:10
<Default_Host>: king
<Power Mode>: OFF
<IPL Mode>: Everytime
<Schedule Pattern>: ON
  Pattern Name: Everyday
<Schedule Pattern Mode>: AND
<Base Job Net>: OFF
<Annually Calendar>: OFF
<Monthly Calendar>: OFF
<Offset Mode>: Execday
  Offset Day: 0
  Offset Holiday: OFF
<Holiday Calendar>: ON
  Calendar Name: WestSalesOffice
<Holiday Mode>: AND
<Holiday Shift>: Cancel
  Shift Limit: 0
  Enable Term: 03/01/2015,03/31/2015
  Disable Term:
<Business Pattern>: ON
Business Base: 20
  Business Day: 1
  Business Day: 3
  Business Day: 5
  Business Limit: ON
<Carryover Mode>: CANCEL
```

5.5 jobschprint -e(-E) Command

Synopsis

```
jobschprint {-e|-E} [-long] [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -e(-E) command outputs a list of the jobs that are currently running.

Options

-e|-E

Among the jobs that can be referenced by the user executing this command, outputs a list of the jobs that are currently running.

-long

Outputs the full value of the "Project Name", "Job Net" and "Job Net Comment" output items.

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the items to be output.

User Name

Name of the user who executes the jobschprint command

Project Name

Project name

The first 20 bytes of the project name are displayed. If the "-long" option is specified, all 50 bytes of the project name will be displayed.

Job Net

Job net name

The first 8 bytes of the job net name are displayed. If the "-long" option is specified, all 50 bytes of the job net name will be displayed.

Job Net Comment

Job net comment

The first 24 bytes of the job net comment are displayed. If the "-long" option is specified, all 100 bytes of the job net comment will be displayed.

Job Name

The currently running job

The command of the job being executed (command name, batch file name/shell script name, program name, or JCL name of Job Execution Control)

Displays the first 24 bytes. If the "-long" option is specified, all 300 bytes of the job name will be displayed.

Displays the workunit name if the execution attribution is "PowerAIM" or "Interstage"

If a child job net is output as a job, the job net name of the child job net is output. If a linked job net is output as a job, the job net name of the linked job net is output.

Start Time

Start time of the job execution

Displayed in "month/day/calendar-year hour:minute:second" format.

If a child job net is output as a job, the time the child job net started is output.

If a linked job net is output as a job, the time the linked job net started is output.

Example

```
jobschprint -e -sys 2
```

Output Example

An example "Job Net Execution List" which is an output result of this command, is shown below.

5.6 jobschprint -f(-F) Command

Synopsis

```
jobschprint {-f|-F}P [-vparam] [-long] [-sys subsystem-number]⁽*¹⁾
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -f(-F) command outputs the execution result history of registered job nets. To output the history information that is older than five days, see "5.7 jobschprint -g(-G) Command" that outputs job net execution information by date.

Options

-f|-F

Outputs the history information of job nets that can be referenced by the user of executing the command.

P

Specifies how far back (days) to output the history information, with a numeric value ranging from 0 to 5.

-vparam

Displays the variable parameters for job nets that were started by specifying a variable parameter.

-long

This option outputs the full value of the "Job Net" and "Job Net Comment" output items.

`EE` **-sys subsystem-number**

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the items to be output.

User Name

Name of the user who executes the jobschprint command.

Day Change Time

A time in which dates are updated

Project Name

Project name

Job Net

Job net name

The first 8 bytes of the job net name are displayed. If the "-long" option is specified, all 50 bytes of the job net name will be displayed.

Job Net Comment

Job net comment

The first 24 bytes of the job net comment are displayed. If the "-long" option is specified, all 100 bytes of the job net comment will be displayed.

Status

Status of the job net

The relevant status is selected from the following eight and displayed:

Normal:

Completed (completion code: 0)

Pseudo:

Pseudo-normal (completion code: from 1 up to the upper limit of completion code)

Abend:

  Abended (completion code: the upper limit of completion code +1 up to 255)

Forced:

  Canceled (completion code: 256)

Execute:

  Executing

E_Cancel:

  Refused

Closed:

  Interrupted (completion code: 239).

Skipped:

  Skipped.

Start Time

  The date and time the job net started

  It is displayed in "month/day/calendar-year hour:minute:second" format.

End Time

  The date and time the job net was completed

  It is displayed in "month/day/calendar-year hour:minute:second" format.

End Code

  Completion code of the job net

Variable Parameter

  This is the variable parameter for job nets that were started by specifying a variable parameter. The variable parameter is only displayed if the "-vparam" option was specified.

<u>**Caution**</u>

  - When the Master Schedule Management function is enabled, use the stemprint -f(-F) command to output history information. This command does not output the history information on the process date that has been expired because the generation exceeded the number set in the [Store schedule information for] using the Master Schedule Management function. For more information on the stemprint -f(-F) command, see "6.12 stemprint -f(-F)/-g(-G) Command".

<u>**Example**</u>

```
jobschprint -f 0 -vparam -sys 2
```

<u>**Output Example**</u>

  An example "History Information of Job Net" which is an output result of this command, is shown below.

```
History Information of Job Net          date: 08/13/2015 time: 16:45

User Name: johnson

Day Change Time: 00:00

Project Name: GeneralAffairsDept

Job Net       Job Net Comment       Status    Start Time             End Time               End Code   Variable Parameter
payroll       PayrollAndBonusCalc   Normal    08/13/2015 10:00:01    08/13/2015 11:33:32    0          "201507"
backup        BackupJob             Normal    08/13/2015 10:30:01    08/13/2015 11:49:26    0
sales         SalesDataManagement   Normal    08/13/2015 12:48:42    08/13/2015 12:50:56    0          "sales01.dat"
management    CustomerManagemen     Abend     08/13/2015 13:32:45    08/13/2015 13:42:37    2
data          CashierDataManagement Forced    08/13/2015 16:27:21    08/13/2015 16:31:21    256
data          CashierDataManagement Execute   08/13/2015 16:44:30
```

5.7 jobschprint -g(-G) Command

Synopsis

```
jobschprint {-g|-G} MM DD YYYY [-vparam] [-long] [-sys subsystem-number](*1)
```

*1:

    [-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -g(-G) command outputs the execution result history of registered job nets on a daily basis. The day on which the history information you want can be specified in a space-separated date format.

Options

-g|-G

Outputs the history information on the job net that can be referenced by the user executing this command.

MM

Specify the month in which the date you want to output appears, in one or two digits.

DD

Specify the date you want to output in one or two digits.

YYYY

Specify the year in four digits.

-vparam

Displays the variable parameters for job nets that were started by specifying a variable parameter.

-long

This option outputs the full value of the "Job Net" and "Job Net Comment" output items.

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the items to be output.

User Name

Name of the user who executes the jobschprint command.

Day Change Time

A time in which dates are updated

Project Name

Project name

Job Net

Job net name

The first 8 bytes of the job net name are displayed. If the "-long" option is specified, all 50 bytes of the job net name will be displayed.

Job Net Comment

Job net comment

The first 24 bytes of the job net comment are displayed. If the "-long" option is specified, all 100 bytes of the job net comment will be displayed.

Status

Status of the job net

The relevant status is selected from the following eight and displayed:

Normal:

Completed

Pseudo:

Pseudo-normal

Abend:

Abended

Forced:

Canceled (completion code: 256).

Execute:

Executing

E_Cancel:

Refused

Closed:

    Interrupted (completion code: 239)

Skipped:

    Skipped

Start Time

The date and time the job net started

It is displayed in "month/day/calendar-year hour:minute:second" format.

End Time

The date and time the job net was completed

It is displayed in "month/day/calendar-year hour:minute:second" format.

End Code

Completion code of the job net

Variable Parameter

This is the variable parameter for job nets that were started by specifying a variable parameter. The variable parameter is only displayed if the "-vparam" option was specified.

Caution

- From among the log files of the Jobscheduler (jobdb1.log/jobdb2.log/jobdb3.log) that are managed cyclically, this command outputs history information of the current log file and the log file one generation before.

- When the Master Schedule Management function is enabled, use the stemprint -g(-G) command to output history information. This command does not output the history information on the process date that has been expired because the generation exceeded the number set in the [Store schedule information for] using the Master Management function. For more information on the stemprint -g(-G) command, see "6.12 stemprint -f(-F)/-g(-G) Command."

Example

```
jobschprint -g 2007 10 24 -vparam -sys 2
```

Output Example

An example "History Information of Job Net" which is an output result of this command, is shown below.

History Information of Job Net   date: 08/13/2015   time: 16:45

User Name: johnson

Day Change Time: 00:00

Project Name: GeneralAffairsDept

Job Net	Job Net Comment	Status	Start Time	End Time	End Code	Variable Parameter
payroll	PayrollAndBonusCalc	Normal	08/13/2015 10:00:01	08/13/2015 11:33:32	0	"201507"
backup	BackupJob	Normal	08/13/2015 10:30:01	08/13/2015 11:49:26	0	
sales	SalesDataManagement	Normal	08/13/2015 12:48:42	08/13/2015 12:50:56	0	"sales01.dat"
management	CustomerManagement	Abend	08/13/2015 13:32:45	08/13/2015 13:42:37	2	
data	CashierDataManagement	Forced	08/13/2015 16:27:21	08/13/2015 16:31:21	256	
data	CashierDataManagement	Execute	08/13/2015 16:44:30			

5.8 jobschprint -i(-I) Command

Synopsis

```
jobschprint {-i|-I} [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -i(-I) command outputs the information (job's registration information) configured when the job was registered. If the value in "Do not save standard/error output," "Restart point," "Ignore completion code," or "Invalid at next restart" is updated on the monitoring window for jobs, the updated value overrides the value that has been defined at registration.

Options

-i|-I

Outputs the registration information about jobs that can be referenced by the user executing this command.

-sys subsystem-number

EE

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the items to be output.

User Name

Name of the user who executes the jobschprint command.

Project Name

Project name

Job Net

Job net name

Job Name

The command that is registered as a job (command name, batch file name/shell script name, program name, or JCL name of Job Execution Control)

If a child job net is output as the job, the job net name of the child job net is output.

If a linked job net is output as the job, the job net name of the linked job net is output.

If a batch framework job is output as the job, the batch job definition name is output.

This is output if the execution attribute is "Job Execution Control" or "Normal attribute". [UNIX]

Job Param

Outputs the command parameters.

This is not output if a batch framework job is output as the job.

Job Name JES

Job name that is used in the Job Execution Control

This is not output if a child job net is output as the job.

This is not output if a linked job net is output as the job.

This is not output if the execution attribute is "Interstage". [UNIX]

Job Comment

Job comment

If a child job net is output as the job, the job net comment of the child job net is output.

If a linked job net is output as the job, the job net comment of the linked job net is output.

Directory

Name of the directory used in executing qsub (job submission) when the execution attribute is "Job Execution Control."

This is not output if a child job net is output as the job.

This is not output if a linked job net is output as the job.

This is not output if a batch framework job is output as the job.

This is not output if the execution attribute is "PowerAIM" or "Interstage." [UNIX]

This is the directory name where the command is executed if the execution attribute is "Normal attribute." [UNIX]

System Name

Business system name/system name

This is output if the execution attribute is "PowerAIM" or "Interstage". [UNIX]

Work Unit Name

Workunit name that is registered as the job

This is output if the execution attribute is the "PowerAIM" or "Interstage". [UNIX]

Host

Name of the host to which the job is submitted

This is not output if a child job net is output as the job.

This is not output if a linked job net is output as the job.

This is not output if a batch framework job is output as the job.

This is output if the execution attribute is "Job Execution Control" or "Interstage." [UNIX]

Jobfile Transfer

Displays whether or not (ON/OFF) "Transmission of execution command" is specified. This is output if the execution attribute is "Job Execution Control."

This is not output if a child job net is output as the job.

This is not output if a linked job net is output as the job.

This is not output if a batch framework job is output as the job.

ON:

"Transmission of execution command" is specified.

OFF:

"Transmission of execution command" is not specified.

Queue

Name of the queue to which the job is submitted

This is not output if a child job net is output as the job.

This is not output if a linked job net is output as the job.

The job queue name is output if a batch framework job is output as the job.

This is output if the execution attribute is "Job Execution Control" or "Interstage." [UNIX]

Resource

Name of the resource to be used by the job

This is not output if a child job net is output as the job.

This is not output if a linked job net is output as the job.

This is not output if a batch framework job is output as the job.

This is output if the execution attribute is "Job Execution Control" or "Interstage." [UNIX]

Resource_Attr

Mode of use of resource that is used by the job

This is not output if a child job net is output as the job.

This is not output if a linked job net is output as the job.

This is not output if a batch framework job is output as the job.

This is output if the execution attribute is "Job Execution Control" or "Interstage." [UNIX]

Sh:

Shared use is specified.

Ex:

Exclusive use is specified.

Ignore Complete Code

Displays whether or not (ON/OFF) "Ignore completion code" is specified.

This is not output if a child job net is output as the job.

This is not output if a linked job net is output as the job.

ON:

"Ignore completion code" is specified. At this time, the subsequent jobs are executed even if the completion code for this job has a value exceeding the "Upper limit of completion code."

OFF:

"Ignore completion code" is not specified.

No Output

Displays whether or not (ON/OFF) "Do not save standard/error output" is specified. This is not output if a child job net is output as the job.

This is not output if a linked job net is output as the job.

This is output if the execution attribute is "Job Execution Control." [Windows]

This is not output if the execution attribute is "Interstage." [UNIX]

ON:

"Do not save standard/error output" is specified. Neither standard output nor error output of this job is saved.

OFF:

"Do not save standard/error output" is not specified.

When a batch framework job is output as the job, outputs whether or not (ON/OFF) "Do not save standard/error output to job log file" is specified.

ON:

"Do not save standard/error output to job log file" is specified.

OFF:

"Do not save standard/error output to job log file" is not specified.

Restart Point

Displays whether or not (ON/OFF) "Restart point" is specified.

This is not output if the execution attribute is "Interstage." [UNIX]

ON:

"Restart point" is specified. When the job net is restarted, the job net is started from this job.

OFF:

"Restart point" is not specified.

Normal Limit

Upper limit of completion code

This value is used to determine if the job was normally completed (which includes a case of pseudo-normal) or abended.

This is not output if a child job net is output as the job.

This is not output if a linked job net is output as the job.

Prediction Time

Displays the estimated processing time from starting to completing a job.

This is not output if a child job net is output as the job.

This is not output if a linked job net is output as the job.

This is output if the execution attribute is "Job Execution Control." [UNIX]

Job Icon

Displays the job icon type. The following table lists the contents displayed and each meaning.

Contents displayed	Meaning (Icon type)
Command	Execute command
Message	Send message

Contents displayed	Meaning (Icon type)
SrvStart	Start service
SrvStop	Stop service
FileWait	File wait
Backup	Backup cooperation
CL_Appl	Application execution (client)
SV_Appl	Application execution (server)
CL_Press	File compression/expansion (client)
SV_Press	File compression/expansion (server)
CL_Control	File control (client)
SV_Control	File control (server)
CL_Copy	File transfer (client)
SV_Copy	File transfer (server)
CL_Pon	Power on (client)
CL_Poff	Shutdown (client)
Interstage	Interstage
BatchJob	Batch framework job
CL_RenameFile	Rename file (client)
SV_RenameFile	Rename file (server)
CL_DelFile	Delete file (client)
SV_DelFile	Delete file (server)
CL_MakeDir	Create directory (client)
SV_MakeDir	Create directory (server)
CL_RemoveDir	Delete directory (client)
SV_RemoveDir	Delete directory (server)
CheckLog	Check log file
WaitTime	Wait for time
CheckSrv	Check service status
JobNet	Job net
LinkJobNet	Linked job net
WaitMessage	Wait for Message Event
Sleep	Wait until Specified Time
FileExist	Determine File Existence
CheckNetVar	Determine Job Net Variable
User001(~User100)	User definition 1 (~ User definition 100)

Exec User

Outputs the name of the job execution user.

This is not output if a batch framework job is output as the job.

Job Env

Outputs the environmental variable when the job is executed. The output format is shown below:

```
Job EnvX: Environmental variable name = Environmental variable value
    X: An integer between 1 and 32
```

Outputs the replaceable parameters in batch job definitions if a batch framework job is output as the job. The output format is shown below:

```
Job EnvX: Parameter name = Parameter value
    X: An integer between 1 and 32
```

Next Time Disable

Displays whether or not (ON/OFF) "Invalid at next restart" is specified.

This is not output if the execution attribute is "Interstage." [UNIX]

ON:

"Invalid at next restart" is specified.

OFF:

"Invalid at next restart" is not specified.

Previous Job Name

Name of preceding job

If the preceding job is a child job net, the job net name of the child job net that is the preceding job is output.

If the preceding job is a linked job net, the job net name of the linked job net that is the preceding job is output.

This is not output if the execution attribute is "PowerAIM" or "Interstage." [UNIX]

Previous Job End Code

Displays the execution condition according to the completion code of the preceding job. The output format varies depending on the execution conditions: "By value" or "By range". For information on the judgment operators, see "4.9.2 Job Control Statement."

When the execution condition is "By value":

Judgment operator Comparison value

```
Previous Job End Code: GT 10
```

When the execution condition is "By range":

Judgment operator 1 Comparison value 1 and Judgment operator 2 Comparison value 2

```
Previous Job End Code: GT 10 and LT 20
```

Stdout File

Outputs the name of file to which standard output on the job is output.

This is not output if a batch framework job is output as the job.

Stderr File

Outputs the name of file to which standard error output on the job is output.

This is not output if a batch framework job is output as the job.

Stderr To Stdout File

Outputs whether or not (ON/OFF) "Redirect default error log to the default log file" is specified.

This is not output if a batch framework job is output as the job.

ON:

"Redirect default error log to the default log file" is specified.

OFF:

"Redirect default error log to the default log file" is not specified.

Limit Time

Outputs the cut-off execution time of the job in minutes.

Exec Priority

Outputs the execution priority of the jobs.

If a batch framework job is output as the job, the execution priority in the job queue is output.

Job Attr

Displays the mode of the recovery job.

This is output only when the job is a recovery job. [Windows]

Recovery Only:

Indicates the job is a recovery job. The preceding job is not rerun.

Recovery Rerun:

The preceding job is rerun after this recovery job is complete.

Previous System Name [UNIX]

Name of preceding business system

This is output if the execution attribute is the "PowerAIM".

Previous Work Unit Name [UNIX]

Name of preceding workunit name

This is output if the execution attribute is the "PowerAIM".

Job Log File

Outputs the job log file name.

This is output only when a batch framework job is output as the job. If not specified, this is not output.

Beginning Step Name

Outputs the start step name.

This is output only when a batch framework job is output as the job. If not specified, this is not output.

End code of last step

Outputs whether or not (ON/OFF) "Use the exit code of the last job step as the job's exit code" is specified.

This is output only when a batch framework job is output as the job.

ON:

"Use the exit code of the last job step as the job's exit code" is specified.

OFF

> "Use the exit code of the last job step as the job's exit code" is not specified.

Execution Day Mode

Displays whether or not (ON/OFF) the "Valid only on startup days" on job (child job net or linked job net) is specified if the icon type is the child job net or linked job net.

ON

> "Valid only on startup days" is specified.

OFF

> "Valid only on startup days" is not specified.

Netvar Prefix

Displays the prefix of the job net variable name.

This is not output if a batch framework job is output as a job.

This is output if the execution attribute is "Job Execution Control".

Or Mode

Indicates whether or not (ON/OFF) "Wait for preceding job using OR condition" is specified.

ON

> "Wait for preceding job using OR condition" is specified.

OFF

> "Wait for preceding job using OR condition" is not specified.

This is output if the execution attribute is "Job Execution Control".

Output Example

An example "Information of Job" which is an output result of this command, is shown below.

```
Information of Job                 date: 03/26/2015 time: 10:00


User Name: Johnson


Project Name:GeneralAffairsDept


Job Net: management


Job Name: management1.bat
Job Param: -file data1.dat
Job Name JES: management1
Job Comment: CustomerInquiry
Directory: c:\tmp
Host: queen
Jobfile Transfer: ON
Queue: queue3
Resource: resource1
Resource_Attr: Sh
Ignore Complete Code: ON
No Output: ON
Restart Point: OFF
Normal Limit: 10
Prediction Time: 10
Job Icon: Command
Exec User: administrator
Job Env1: TEMP_PATH = d:\temp
```

```
Job Env2: DATA_PATH = d:\management\data
Next Time Disable: OFF
Stdout File: c:\tmp\outfile.log
Stderr File: c:\tmp\errfile.log
Stderr To Stdout File: OFF
Limit Time: 30
Exec Priority: 0


Project Name: GeneralAffairsDept


Job Net: selling


Job Name: selling1.bat
Job Param: -file data2.dat
Job Name JES: selling1
Job Comment: CollectedBillReflection
Directory: c:\tmp
Host: atlas
Jobfile Transfer: ON
Queue: queue2
Resource: resource2
Resource_Attr: ExIgnore Complete Code: OFF
No Output: ON
Restart Point: OFF
Normal Limit: 1
Prediction Time: 20
Job Icon: Command
Exec User: administrator
Job Env1: DATA_PATH = d:\selling\data
Next Time Disable: OFF
Stdout File: c:\tmp\err.log
Stderr File:
Stderr To Stdout File: ON
Limit Time: 5
Exec Priority: 0
Job Name: cldnet
Job Comment: cldnet
Restart Point: OFF
Job Icon: JobNet
Next Time Disable: OFF
Execution Day Mode: OFF
```

5.9 jobschprint -l(-L) Command

Synopsis

```
jobschprint {-l|-L} MM YYYY [-long] [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -l(-L) command outputs a list of application plan for the registered job nets by specifying the year and month.

Options

-l|-L

Outputs a list of application plan specified in the year and month, regarding the job nets that can be browsed by the user executing this command. Also outputs the job net comment.

Job nets that have been copied and started will not be output.

_MM_

Specify the month in which the date you want to output appears, in one or two digits.

YYYY

Specify the year in four digits.

-long

This option outputs the full value of the "Job Net" and "Job Net Comment" output items.

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the items to be output.

User Name

Name of the user who executes the jobschprint command

Project Name

Project name

Job Net

Job net name

The first 8 bytes of the job net name are displayed. If the "-long" option is specified, all 50 bytes of the job net name will be displayed.

Job Net Comment

Job net comment

The first 24 bytes of the job net comment are displayed. If the "-long" option is specified, all 100 bytes of the job net comment will be displayed.

Example

```
jobschprint -l 2007 10 -sys 2
```

Output Example

An example "Job Net Execution Day Matrix" which is an output result of this command, is shown below.

Job net startup days are marked with an asterisk (*). The dates marked with "@" are Sundays.

```
   Job Net Execution Day Matrix      date: 10/25/2015   time: 16:52

User Name: johnson

Project Name: SalesDept

11/2015


-----------------------------------------------------------------------------------------------
           |                    |   @          @          @          @
           |                    | 0         10         20         30
Job Net    | Job Net Comment    | 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0
-----------------------------------------------------------------------------------------------
selling    | SalesManagement    | *         *          *          *          *
customer   | CostomerSalesManagement| *     *          *          *          *
sales      | CreateWeeklySalesReport |         *          *          *          *
-----------------------------------------------------------------------------------------------
```

5.10 jobschprint -m(-M) Command

Synopsis

```
jobschprint {-m|-M} MM YYYY [-long] [-sys subsystem-number](*1)
```

*1:

    [-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -m(-M) command outputs a list of application plan for registered job nets by specifying the year and month.

Options

-m|-M

Outputs a list of application plan specified in the year and month, regarding the job nets that can be browsed by the user executing this command.

Job nets that have been copied and started will not be output.

MM

Specify the month in which the date you want to output appears, in one or two digits.

YYYY

Specify the year in four digits.

-long

Outputs the full value of the "Job Net" output item.

EE -sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the items to be output.

User Name

Name of the user who executes the jobschprint command

Project Name

Project name

Job Net

Job net name

The first 8 bytes of the job net name are displayed. If the "-long" option is specified, all 50 bytes of the job net name will be displayed.

Example

```
jobschprint -m 2007 10 -sys 2
```

Output Example

An example "Job Net Execution Day Matrix" which is an output result of this command, is shown below.

Job net startup days are marked with an asterisk (*). The dates marked with "@" are Sundays.

```
   Job Net Execution Day Matrix      date: 10/25/2015   time: 16:52

User Name: johnson

Project Name: SalesDept

11/2015

----------------------------------------------------------------
         |    @            @            @            @
         | 0           10           20           30
Job Net  | 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0
----------------------------------------------------------------
selling  | *           *           *           *           *
customer | *           *           *           *           *
sales    |     *           *           *           *
----------------------------------------------------------------
```

5.11 jobschprint -p(-P) Command

Synopsis

```
jobschprint {-p|-P} [-long] [-sys subsystem-number]("1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -p(-P) command outputs the current status of registered jobs.

Options

-p|-P

Outputs the current status of jobs that can be browsed by the user executing this command.

-long

Outputs the full value of the "Job Net" output item.

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the items to be output.

User Name

Name of the user who executes the jobschprint command

Project Name

Project name

Job Net

Job net name

The first 8 bytes of the job net name are displayed. If the "-long" option is specified, all 50 bytes of the job net name will be displayed.

Job Name

Registered job

The command registered as a job (command name, batch file name/shell script name, program name, or JCL name of Job Execution Control)

This is a workunit name if the execution attribute is "Interstage".

If a child job net is output as the job, the job net name of the child job net is output.

If a linked job net is output as the job, the job net name of the linked job net is output.

Displays the first 24 bytes of the job name. If the "-long" option is specified, all 300 bytes of the job name will be displayed.

Status

Status of the job

If a child job net is displayed as the job, the status of the child job net is displayed.

If a linked job net is displayed as the job, the status of the linked job net is displayed.

The relevant status is selected from the following thirteen and displayed:

Normal:

Completed

Pseudo:

Pseudo-normal

Abend:

Abended

Forced:

Canceled

Execute:

Executing (When a state of the child job net is end delayed, it is displayed as under executing.)

Delay:

Execution delayed (If the job is the child job net, "Delay" is excluded.)

Warning:

Warning (Only for child job net)

Wait:

Waiting

Pause:

Paused

Disable:

Disabled

Pass & W:

The job in the "Waiting" state has been passed.

Pass & P:

The job in the "Paused" state has been passed.

Pass & D:

The job in the "Disabled" state has been passed.

Note that the "pseudo-normal" state is displayed as the "Completed" if the pseudo-normal option is not selected in the [Define Jobscheduler Startup Parameters] window.

Start Time

Start time of the job's execution

If a child job net is output as the job, the start time of the child job net is output.

If a linked job net is output as the job, the start time of the linked job net is output.

End Time

End time of the job's execution

If a child job net is output as the job, the end time of the child job net is output.

If a linked job net is output as the job, the end time of the linked job net is output.

Completion Code

Completion code of the job

If a child job net is output as the job, the completion code of the child job net is output.

If a linked job net is output as the job, the completion code of the linked job net is output.

Example

```
jobschprint -p - sys 2
```

Output Example

An example "Job Status List" which is an output result of this command, is shown below.

```
Job Status List                 date: 10/25/2015   time: 17:31

User Name: johnson

Project Name: GeneralAffairsDept

Job Net       Job Name        Status    Start Time         End Time           Completion Code
selling       management1.bat Normal    10/25/2015 17:20   10/25/2015 17:25   0
              management2.bat Execute   10/25/2015 17:30
              management3.bat Pass & W
              management4.bat Wait
customer      c_management.bat Normal   10/21/2015 20:24   10/21/2015 20:56   0
sales         weeklyreport.bat Wait
payroll       payroll.bat     Normal    10/10/2015 22:00   10/10/2015 22:48   0
data          cashier.bat     Execute   10/25/2015 17:29
management    customer.bat    Abend     10/25/2015 15:00   10/25/2015 15:14   2
```

5.12 jobschprint -q(-Q) Command [Windows]

Synopsis

```
jobschprint {-q|-Q} [-long] [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -q(-Q) command displays a list of job nets that are currently connected into the job net execution queue when the job net execution was requested by using the job net execution API.

Options

-q|-Q

For the job nets that can be browsed by the user executing this command, outputs a list of the job nets that are connected into the job net execution queue.

-long

Outputs the full value of the "Project Name" and "Job Net" output items.

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the items to be output.

User Name

Name of the user who executes the jobschprint command

Project Name

Project name

The first 20 bytes of the project name are displayed. If the "-long" option is specified, all 50 bytes of the project name will be displayed.

Job Net

Job net name

The first 8 bytes of the job net name are displayed. If the "-long" option is specified, all 50 bytes of the job net name will be displayed.

Parameter

Parameters (first 30 bytes) to be passed to jobs

Example

```
jobschprint -q -sys 2
```

Output Example

An example "Queuing Job Net List" which is an output result of this command, is shown below.

```
    Queuing Job Net List              date: 10/25/2015  time: 17:12

User Name : johnson


Project Name          Job Net              Parameter
GeneralAffairsDept    management
SalesDept             backup               d:\users\default
SalesDept             backup               e:\users\default
GeneralAffairsDept    workhour
```

5.13 jobschprint -R Command

```
jobschprint -R [project name/] group name [-operate] [-sys subsystem-number]⁽*¹⁾
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -R command outputs the information defined in the group control statements or group job net control statements.

Options

-R

Outputs the information on the registered group in the format of the group control statement specified by using the jobschesetgrp command.

project name

Specifies the name of project that contains the group information. Among the projects that can be browsed by the user executing the jobschprint command, if there is only one project where a group is registered, this option can be omitted.

group name

Specifies the name of the group whose control statement is to be expanded.

-operate

Outputs "operate" control statements ("paused" or "disabled" states resulting from operations) to the group control statement. If this option is omitted, "operate" control statements will not be output.

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

For details on the output items, see "4.10.1 Group Control Statement" and "4.10.2 Group Job Net Control Statement."

Output Example

An example output result of this command is shown below.

```
GRP
groupname weeklyreport
groupcomment WeeklyReport
jobnet 2
operate PAUSE
;
GNT
jobnetname sales
```

```
jobnetnumber 1
iconposition 1,1
;
GNT
jobnetname selling
jobnetnumber 2
pre_jobnet 1
iconposition 2,1
;
```

5.14 jobschprint -r Command

Synopsis

```
jobschprint -r [project name/] job net name [-detail]
               [-operate] [-sys subsystem-number](*1)
```

*1:

    [-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -r command outputs the information defined in the job net control statements and job control statements.

Options

-r

Outputs the information of registered job nets in the format of the job net control statement specified by using the jobschsetnet command.

project name

Specifies the name of the project that contains the job net information. Among the projects that can be browsed by the user executing the jobschprint command, if there is only one project where a job net is registered, this option can be omitted.

job net name

Specifies the name of the job net whose information is to be output.

-detail

Outputs "jobschprintr" control statements (bit pattern for three years). Specify this option when you want to register the job net start days including exception days by using the jobschsetnet command.

-operate

Outputs "operate" control statements ("paused" or "disabled" states resulting from operations). If this option is omitted, "operate" control statements will not be output.

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX	/usr/bin

AIX Linux	

Output Items

For details on the output items, see "4.9.1 Job Net Control Statement" and "4.9.2 Job Control Statement."

Output Example

An example output result of this command is shown below.

```
NET
jobnetname selling
jobnetcomment SalesManagement
starttime 2000
holidayshift IGN
execattr JES
copystart ON
jobnetsavenumber 10
defaulthost queen
operate NONE
job 2
;
JOB
jobname management1.bat
jobnumber 1
jobparam -file data2.dat
jobcomment CreditDetails
pre_job 1
jobicon FWAT
execuser administrator
stdoutfile c:/tmp/exec.log
stdoutfile c:/tmp/error.log
priority -10
envname INPUT
envvalue FILE
envname OUTPUT
envvalue DAT
iconposition 1,1
operate NONE
;
JOB
jobname child1
jobnumber 2
pre_job 1
jobicon JNET
iconposition 1,2
;
```

5.15 jobschprint -s(-S) Command

Synopsis

```
jobschprint {-s|-S} [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -s(-S) command outputs a list of startup days for the three years (the current year, next year, and year after next) for the registered job nets.

Options

-s|-S

Outputs the startup days for three years for the job nets that can be browsed by the user executing this command. Job nets that have been copied and started will not be output.

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the items to be output.

User Name

Name of the user who executes the jobschprint command

Project Name

Project name

Job Net

Job net name

Job Net Comment

Job net comment

Assortment

Schedule pattern name

When multiple schedule patterns are selected, delimit with a space.

Assortment Mode

Indicates the method of combining schedule patterns. Output when multiple schedule patterns are selected.

AND:

Sets the days as startup days if they are set as startup days in all the schedule patterns.

OR:

Sets the days as startup days if they are set as startup days in at least one of the schedule patterns.

Scheduled Time

Scheduled start time

Output Example

An example "Job Net Execution Day List" which is an output result of this command, is shown below.

Job net startup days are marked with an asterisk (*).

5.16 jobschprint -w(-W) Command

Synopsis

```
jobschprint {-w|-W} [-long] [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -w(-W) command outputs a list of jobs in a waiting state. If a job in the waiting state has been passed, it is excluded from the list.

If a job net is in a waiting state or start delayed state, where jobs are not in a waiting state (e.g., completed), this job net is included in the waiting list.

Options

-w|-W

Among the jobs that can be browsed by the user executing this command, outputs a list of the jobs in the waiting state.

-long

Outputs the full value of the "Project Name", "Job Net" and "Job Net Comment" output items.

`EE`

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the items to be output.

User Name

Name of the user who executes the jobschprint command

Project Name

Project name

The first 20 bytes of the project name are displayed. If the "-long" option is specified, all 50 bytes of the project name will be displayed.

Job Net

Job net name

The first 8 bytes of the job net name are displayed. If the "-long" option is specified, all 50 bytes of the job net name will be displayed.

Job Net Comment

Job net comment

The first 24 bytes of the job net comment are displayed. If the "-long" option is specified, all 100 bytes of the job net comment will be displayed.

Job Name

A job in a waiting state

The command of the job in the execution queue (command name, batch file name/shell script name, program name, or JCL name of Job Execution Control)

If a child job net is specified as the job, the job net name of the child job net is output.

If a linked job net is specified as the job, the job net name of the linked job net is output.

Displays the first 24 bytes of the job name. If the "-long" option is specified, all 300 bytes of the job name will be displayed.

If the execution attribute is "Interstage" or "PowerAIM," a workunit name is output.

Schedule Time

Scheduled startup times of job nets that contain the jobs in a waiting state. This is displayed in "month/day/calendar-year hour:minute:second" format.

Example

```
jobschprint -w -sys 2
```

Output Example

An example "Job Net Wait List" which is an output result of this command, is shown below. Blanks are displayed when no startup is scheduled for the job net.

```
Job Net Wait List                    date: 10/25/2015 time: 17:02

User Name : johnson

Project Name        Job Net    Job Net Comment          Job Name            Scheduled Time
GeneralAffairsDept  selling    SalesManagement          management1.bat
                                                        management2.bat
                                                        management3.bat
                    customer   CostomerSalesManagement  c_management.bat    10/25/2015 18:00
                    sales      CreateWeeklySalesReport   weeklyreport.bat    10/25/2015 18:00
                    payroll    PayrollAndBonusCalc       payroll.bat         11/10/2015 22:00
```

5.17 jobschprint -x(-X) Command

Synopsis

```
jobschprint {-x|-X} [-long] [-sys subsystem-number]
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -x(-X) command outputs the current status of the registered groups.

Options

-x|-X

Outputs the current status of the group that can be browsed by the user executing this command.

-long

Outputs the full value of the "Group" and "Group Comment" output items.

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX	/usr/bin

AIX Linux	

Output Items

The following explains the items to be output.

User Name

Name of the user who executes the jobschprint command

Project Name

Project name

Group

Group name

The first 8 bytes of the group name are displayed. If the "-long" option is specified, all 50 bytes of the group name will be displayed.

Group Comment

Group comment

The first 24 bytes of the group comment are displayed. If the "-long" option is specified, all 100 bytes of the group comment will be displayed.

Status

Status of the group

The relevant status is selected from the following ten and displayed:

Normal:

Normal completion

Pseudo:

Pseudo-normal

Abend:

Abended

Forced:

Canceled

Execute:

Executing

Wait:

Waiting

Pause:

Paused

Disable:

Disabled

Coped:

Confirmed

Warning:

Warning

Note that the "pseudo-normal" state is displayed as the "Completed" if the "pseudo-normal" option is not selected in the [Define Jobscheduler Startup Parameters] window

Scheduled Time

Scheduled time for the next start of group

Start Time

Start date and time of the group's previous execution

End Time

End date and time of the group's previous execution

Completion Code

Completion code of the group's previous execution

The group's completion code is equal to the maximum value of the completion code of the job net in the group.

Example

```
jobschprint -x -sys 2
```

Output Example

An example "Group Status List" which is an output result of this command, is shown below.

```
Group Status List        date: 10/25/2015  time: 17:27

User Name: johnson

Project Name: GeneralAffairsDept

Group          Group Comment  Status  Scheduled Time   Start Time         End Time           Completion Code
accounting     Accounting     Normal  11/10/2015       10/10/2015 18:00   10/10/2015 20:27   0
weeklyreport   WeeklyReport   Wait    10/25/2015 20:00
sales          Sales          Wait    10/25/2015 23:00
```

5.18 jobschprint -y(-Y) Command

Synopsis

```
jobschprint {-y|-Y} [-long] [-sys subsystem-number][*1]
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -y(-Y) command outputs a list of job nets in the registered group.

Options

-y|-Y

Outputs a list of job nets in the group that can be browsed by the user executing this command.

-long

Outputs the full value of the "Group" and "Job Net" output items.

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the items to be output.

User Name

Name of the user who executes the jobschprint command

Project Name

Project name

Group

Group name

The first 8 bytes of the group name are displayed. If the "-long" option is specified, all 50 bytes of the group name will be displayed.

Job Net

Job net name

The first 8 bytes of the job net name are displayed. If the "-long" option is specified, all 50 bytes of the job net name will be displayed.

Schedule

Displays whether or not (ON/OFF) execution for the current day is scheduled.

YES:

Execution is scheduled for the current day.

NO:

Execution is not scheduled for the current day.

Status

Status of the job net

The relevant status is selected from the following eleven and displayed:

Normal:

Normal completion

Pseudo:

    Pseudo-normal

Abend:

    Abended

Forced:

    Canceled

Warning:

    Warning

Execute:

    Executing

Wait:

    Waiting

Pause:

    Paused

Disable:

    Disabled

StartDelay:

    Start delayed

EndDelay:

    End delayed

Note that the "pseudo-normal" state is displayed as the "Completed" if the pseudo-normal option is not selected in the [Define Jobscheduler Startup Parameters] window.

Start Time

Start date and time of the job net's previous execution

End Time

End date and time of the job net's previous execution

Completion Code

Completion code of the job net's previous execution

<u>Example</u>

```
jobschprint -y -sys 2
```

<u>Output Example</u>

An example "Group List" which is an output result of this command, is shown below.

```
Group List              date: 10/25/2015  time: 17:28

User Name: johnson

Project Name: GeneralAffairsDept

Group        Job Net    Schedule  Status  Start Time         End Time           Completion Code
management   payroll    Yes       Normal  10/25/2015 15:23   10/25/2015 15:40   0
             customer   Yes       Normal  10/25/2015 15:40   10/25/2015 16:26   0
             backup     Yes       Normal  10/25/2015 16:26   10/25/2015 16:37   0

weeklyreport selling    No        Wait
             sales      No        Wait
```

5.19 jobschprint -b Command

Synopsis

```
jobschprint -b [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -b command outputs a list of registered schedule pattern names.

Options

-b

Outputs a list of registered schedule pattern names.

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the item to be output.

Schedule Model

Schedule pattern name

Output Example

An example "Schedule Model List" which is an output result of this command, is shown below.

```
Schedule Model List          date: 10/25/2015   time: 17:30

Schedule Model: Everyday
Schedule Model: Weekday
Schedule Model: EveryMonday
Schedule Model: EveryFriday
Schedule Model: 10thOfEveryMonth
```

5.20 jobschprint -c Command

Synopsis

```
jobschprint -c schedule pattern name [-sys subsystem-number]⁽*1⁾
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -c command outputs the startup days that are specified by date among the startup days registered as the schedule pattern.

Options

-c

Outputs the startup days that are specified by date among those registered as the schedule pattern.

schedule pattern name

Specifies the name of schedule pattern to be output.

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the item to be output.

Schedule Model

Displays the name of the schedule pattern.

Output Example

An example "Schedule Model Matrix" which is an output result of this command, is shown below.

In the table, the months (from January through December) are shown along the vertical axis, and the days of a month are shown along the horizontal axis. Job net startup days registered in a schedule pattern are marked with an asterisk (*).

```
 Schedule Model Matrix          date: 10/25/2015 time: 17:06

Schedule Model: 10thOfEveryMonth


----------------------------------------------------------------
      | 0          10          20          30
 MON  | 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
----------------------------------------------------------------
  1   |                       *
  2   |                       *
  3   |                       *
  4   |                       *
  5   |                       *
  6   |                       *
  7   |                       *
  8   |                       *
  9   |                       *
 10   |                       *
 11   |                       *
 12   |                       *

----------------------------------------------------------------
```

5.21 jobschprint -d Command

Synopsis

```
jobschprint -d schedule pattern name [-sys subsystem-number]⁽*¹⁾
```

*1:

   [-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -d command outputs the startup days of the specific week among the startup days registered as the schedule pattern.

Options

-d

Outputs the startup days of the specific week among the startup days registered as the schedule pattern.

schedule pattern name

Specifies the name of the schedule pattern that you want to output.

EE **-sys subsystem-number**

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the item to be output.

Schedule Model

Displays the name of the schedule pattern.

Output Example

An example "Schedule Model Matrix" which is an output result of this command, is shown below.

In the table, months (from January through December) are shown along the vertical axis, and the days of the week are shown along the horizontal axis.

The days of the week marked with an asterisk (*) are job net startup days registered in a schedule pattern.

```
Schedule Model Matrix          date: 10/25/2015      time: 17:08

Schedule Model: EveryFriday


--------------------------------------------------------
MON | Sun   Mon   Tue   Wed   Thu   Fri   Sat
--------------------------------------------------------
  1 |                                 *
  2 |                                 *
  3 |                                 *
  4 |                                 *
  5 |                                 *
  6 |                                 *
  7 |                                 *
  8 |                                 *
  9 |                                 *
 10 |                                 *
 11 |                                 *
 12 |                                 *

--------------------------------------------------------
```

5.22 jobschprint -h Command

Synopsis

```
jobschprint -h calendar name [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -h command outputs a list of holidays on the registered calendars, which are for three years (the current year, next year, and year after next).

Options

-h

Outputs a list of registered holidays on the calendars (where holidays for three years: the current year, next year, and year after next, are displayed).

calendar name

Specifies the name of the calendar that you want to output.

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. The calendar information is commonly used in all Jobschedulers. Therefore, even if this option is specified or not, the same calendar information is output.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the items to be output.

Calendar Name

Calendar name

Day Change Time

A time in which dates are updated

Output Example

An example "Job Net Holiday List" which is an output result of this command, is shown below.

Holidays are marked with an asterisk (*).

```
Job Net Holiday List              date: 10/25/2015  time: 17:09

Calendar Name: Sunday

Day Change Time: 00:00

01/2015

 Sun    Mon    Tue    Wed   Thu    Fri    Sat
                              1      2      3

  4      5      6      7      8      9     10
  *
 11     12     13     14     15     16     17
  *
 18     19     20     21     22     23     24
  *
 25     26     27     28     29     30     31
  *

02/2015

 Sun    Mon    Tue    Wed   Thu    Fri    Sat
  1      2      3      4      5      6      7
  *
  8      9     10     11     12     13     14
  *
```

5.23 jobschprint -u Command

<u>**Synopsis**</u>

```
jobschprint -u [-long] [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

<u>**Description**</u>

The jobschprint -u command outputs the user name and the client name (IP address) as the user information if you want to identify who is currently accessing the Jobscheduler from which terminal.

The user information also contains the connection information on the Multi-server Monitoring clients.

<u>**Options**</u>

-u

Outputs the Jobscheduler user information (user name, client name (IP address)).

-long

Outputs the full value of the "User Name" output item.

`EE`

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

<u>**Command Location**</u>

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

<u>**Output Items**</u>

The following explains the items to be output.

User Name

User name who is using the Jobscheduler

[Windows]

All 20 bytes will be displayed regardless of whether the -long option has been specified.

In environments where the Systemwalker User Management function is enabled, all 32 bytes will be displayed regardless of whether the -long option has been specified.

[UNIX]

The first 12 bytes will be displayed.

All 20 bytes will be displayed if the -long option has been specified.

In environments where the Systemwalker User Management function is enabled, all 32 bytes will be displayed regardless of whether the -long option has been specified.

Client Name

Client name and IP address of the terminal

"???" is displayed if no client name is found.

<u>**Example**</u>

```
jobschprint -u -sys 2
```

<u>**Output Example**</u>

An example "Jobsch User List" which is an output result of this command, is shown below.

```
   Jobsch User List              date: 10/25/2015  time: 17:12

User Name      Client Name
johnson        king(123.123.123.123)
brown          jack(123.123.123.125)
jones          ???(123.123.123.124)
```

5.24 jobschprint -v Command

<u>**Synopsis**</u>

```
jobschprint -v [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

<u>**Description**</u>

The jobschprint -v command outputs a list of the registered calendar names.

<u>**Options**</u>

-v

Outputs a list of registered calendar names.

EE   **-sys subsystem-number**

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. The calendar information is commonly used in all Jobschedulers. Therefore, even if this option is specified or not, the same calendar information is output.

<u>**Command Location**</u>

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

<u>**Output Items**</u>

The following explains the item to be output.

Calendar Name

Calendar name

Output Example

An example "Holiday Calendar List" which is an output result of this command, is shown below.

```
   Holiday Calendar List          date: 10/25/2015  time: 17:08

Calendar Name:  Monday
Calendar Name:  Tuesday
Calendar Name:  WestSalesOffice
Calendar Name:  SouthBranch
```

5.25 jobschprint -o(-O) Command

Synopsis

```
jobschprint {-o|-O} [-long] [-sys subsystem-number]⁽*¹⁾
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprint -o(-O) command outputs a list of registered project names.

Options

-o|-O

Outputs a list of projects for which the user executing this command has the reference rights or higher.

-long

Outputs the full value of the "Project Name" output item.

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the items to be output.

User Name

Name of the user who executes the jobschprint command

Project Name

Project name

The first 20 bytes of the project name are displayed. If the "-long" option is specified, all 50 bytes of the project name will be displayed.

Access

Displays the access rights for the project assigned to the user executing this command. The relevant status is selected from the following four and displayed:

all:

Update rights

(Otherwise displayed when the user has all of change right, operation right, and reference right)

change:

Change rights

operate:

Operation rights

monitor:

Reference rights

Owner Name

Owner name of the project

Example

```
jobschprint -o -sys 2
```

Output Example

An example "Jobsch Project List" which is an output result of this command, is shown below.

```
 Jobsch Project List          date: 10/25/2015  time: 17:08

User Name: johnson

Project Name          Access   Owner Name
GeneralAffairsDept    all      smith
DevelopmentDept       operate  williams
```

5.26 jobschprint -rh Command

Synopsis

```
jobschprint -rh calendar name
```

Description

Outputs the calendar holiday information in the control statement format.

Options

-rh

Outputs registered calendars in the holiday control statement format specified by the jobschsethol command.

Calendar Name

Specify the name of the calendar to be output.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

Refer to "Holiday Information Control Statement" for details on the output items.

Output Example

An example output result of this command is shown below.

```
HOL
calendarname MainOfficeCalendar
yearset 2010
monthset 11
dayset 1
yearset 2010
monthset 12
dayset 20
;
```

5.27 jobschprint -rsp Command

Synopsis

jobschprint -rsp schedule pattern name [-sys subsystem-number][(*1)]

*1)

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

Outputs the schedule pattern information in the control statement format.

Options

-rsp

Outputs registered schedule patterns in the schedule pattern control statement format specified by the jobschsetcal command.

schedule pattern name

Specify the name of the schedule pattern to be output.

-sys subsystem-number

Specify the subsystem to be operated in systems that are running multiple subsystems. Specify the number of the subsystem to be operated using a numeral between 0 and 9. Subsystem number 0 is specified if this option is omitted.

Command Location

Windows	<*Systemwalker Operation Manager installation directory*>\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

Refer to "Schedule Pattern Control Statement" for details on the output items.

Output Example

An example output result of this command is shown below.

```
CAL

execpatname Sample

monthset 1

dayset 1,5,10,15,20

weekset MON,FRI

monthset 2

dayset 1,4,8,12,16

weekset MON,THU

monthset 5

dayset 1,10,20,30

weekset TUE,THU

monthset 11

dayset 1,11,22,30

weekset WED,THU

;
```

5.28 jobschprintcsv -n Command

Synopsis

```
jobschprintcsv -n [-toplevel] [-title] [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprintcsv -n command outputs the job net's current status to standard output in CSV format.

Saving the output result in a file enables the job net current status to be managed in spreadsheet software. (Note)

Note:

Before importing the output result into the spreadsheet software, it is necessary to convert it into the character code supported by that spreadsheet software.

Options

-n

Outputs the current status of the job nets that can be browsed by the user executing this command.

-toplevel

This option outputs only single-level job nets or the first level of multi-level job nets. It does not output child job nets or linked job nets.

-title

If this option is specified, the title is output on the first line.

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the items to be output.

Host Name

Host name of the server that executed this command

Sub System

Subsystem number which is corresponding to the Jobscheduler to be operated

Project Name

Name of the project where the job net is registered

Job Net

Name of job net that is registered in the Jobscheduler

Job Net Comment

Job net comment that is registered in the Jobscheduler

Status

Status of the job net

The relevant status is selected from the following thirteen and displayed:

Normal:

Completed

Pseudo:

Pseudo-normal

Abend:

Abended

Forced:

Canceled

Warning:

Warning

Execute:

Executing

Wait:

Waiting

Pause:

Paused

Disable:

Disabled

Coped:

Confirmed

CarryOver:

Carried over

StartDelay:

Start delayed

EndDelay:

End delayed

Note that the "pseudo-normal" state is displayed as the "Completed" if the pseudo-normal option is not selected in the [Define Jobscheduler Startup Parameters] window.

Scheduled Time

Scheduled startup time

Start Time

Start date and time of the job net's previous execution

End Time

End date and time of the job net's previous execution

Completion Code

Completion code of the job net

Caution

- When the Master Schedule Management function is enabled, use the stemprint -n(-N) command to output the current state of the job net. Using the stemprint -n(-N) command enables you to narrow down the process dates or projects that you want to output.

For more information on the stemprint -n(-N) command, see "6.13 stemprint -n(-N) Command."

Output Example

An example output result of this command is shown below.

```
Host Name,Sub System,Project Name,Job Net,Job Net Comment,Status,Scheduled Time,Start
Time,End Time,Completion Code
host1,3,Sales,selling,SalesManagement,Abend,,03/01/2014 10:22,10/25/2014 10:38,10
host1,3,Sales,customer,CustomerSalesManagement,Wait,03/01/2014 18:00,,,
host1,3,Sales,sales,CreateWeeklySalesReport,Disable,03/02/2014,02/26/2015
23:00,02/26/2015 23:12,
host1,3,GeneralAffairst,cashier,CashierDataManagement,Execute,03/05/2014,03/01/2015
14:00,,
host1,3,GeneralAffairs,payroll,PayrollAndBonusCalc,Normal,03/20/2014,02/20/2015
23:00,02/20/2015 23:50,0
```

When you display the above file in spreadsheet software, you will see a listing similar to:

Host Name	Sub System	Project Name	Job Net	Job Net Comment	Status	Scheduled Time
host1	3	Sales	selling	SalesManagement	Abend	
host1	3	Sales	customer	CustomerSalesManagement	Wait	03/01/2015 18:00
host1	3	Sales	sales	CreateWeeklySalesReport	Disable	03/02/2015
host1	3	GeneralAffairs	cashier	CashierDataManagement	Execute	03/05/2015
host1	3	GeneralAffairs	payroll	PayrollAndBonusCalc	Normal	03/20/2015

5.29 jobschprintcsv -g Command

Synopsis

```
jobschprintcsv -g MM DD YYYY [-title] [-vparam] [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprintcsv -g command outputs the execution result history of job nets to standard output in CSV format.

Saving the output result in a file enables the execution result history to be managed with spreadsheet software. (Note)

Note:

Before importing the output result into the spreadsheet software, it is necessary to convert it into the character code supported by that spreadsheet software.

Options

-g

Outputs the history information on the job net that can be browsed by the user executing this command.

MM

Specify the month in which the date you want to output appears, in one or two digits.

DD

Specify the date that you want to output, in one or two digits.

YYYY

Specify the year in four digits.

-title

If this option is specified, the title is output on the first line.

-vparam

Displays the variable parameters for job nets that were started by specifying a variable parameter.

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the items to be output.

Host Name

Host name of the server that executed this command

Sub System

Subsystem number which is corresponding to the Jobscheduler to be operated

Project Name

Name of the project where the job net is registered

Job Net

Name of the job net that is registered in the Jobscheduler

Job Net Comment

Job net comment that is registered in the Jobscheduler

Status

Status of job net

The relevant status is selected from the following eight and displayed:

Normal:

   Normal completion

Pseudo:

   Pseudo-normal

Abend:

   Abended (completion code: from the upper limit of completion code +1 up to 255)

Forced:

   Canceled (completion code: 256)

Execute:

   Executing

E_Cancel:

   Refused

Closed:

   Interrupted (completion code: 239)

Skipped:

   Skipped

Start Time

The date and time the job net started

It is displayed in "month/day/calendar-year hour:minute:second" format.

End Time

The date and time the job net was completed

It is displayed in "month/day/calendar-year hour:minute:second" format.

End Code

Completion code of the job net

Variable Parameter

This is the variable parameter for job nets that were started by specifying a variable parameter. The variable parameter is only displayed if the "-vparam" option was specified.

Caution

- From among the log files of the Jobscheduler (jobdb1.log/jobdb2.log/jobdb3.log) that are managed cyclically, this command outputs history information of the current log file and the log file one generation before.

- When the Master Schedule Management function is enabled, use the stemprint -g(-G) command to output history information. This command does not output the history information on the process date that has been expired because the generation exceeded the number set in the [Store schedule information for] using the Master Schedule Management function.

   For more information on the stemprint -g(-G) command, see "6.12 stemprint -f(-F)/-g(-G) Command."

Output Example

An example output result of this command is shown below.

```
Host Name,Sub System,Project Name,Job Net,Job Net Comment,Status,Start Time,End Time,End
Code, Variable Parameter
```

```
host1,3,GeneralAffairs,payroll, PayrollAndBonusCalc, Normal,03/01/2014
10:00:01,03/01/2014 11:33:32,0,"200707"
host1,3,Sales,backup,BackupJob,Normal,03/01/2014 10:30:01,03/01/2014 11:49:26,0,
host1,3,Sales,sales,SalesDataManagement,Normal,03/01/2014 12:48:42,03/01/2014
12:50:56,0,"sales01.dat"
host1,3,Sales,management,CustomerManagement,Abend,03/01/2014 13:32:45,03/01/2014
13:42:37,2,
host1,3,GeneralAffairs,data,CashierDataManagement, Forced,03/01/2014 16:27:21,03/01/2014
16:31:21,256,
host1,3,GeneralAffairs,data,CashierDataManagement, Execute,03/01/2014 16:44:30,,,
```

When you display the above file in spreadsheet software, you will see a listing similar to:

Host Name	Sub System	Project Name	Job Net	Job Net Comment	Status	Start Time
host1	3	GeneralAffairs	payroll	PayrollAndBonusCalc	Normal	03/01/2015 10:
host1	3	Sales	backup	BackupJob	Normal	03/01/2015 10:3
host1	3	Sales	sales	S alesDataManagement	Normal	03/01/2015 12:48
host1	3	Sales	management	CustomerManagement	Abend	03/01/2015 13:32:
host1	3	GeneralAffairs	data	CashierDataManagement	Forced	03/01/2015 16:27:2
host1	3	GeneralAffairs	data	CashierDataManagement	Execute	03/01/2015 16:44:

5.30 jobschprintcsv -l Command

Synopsis

```
jobschprintcsv -l MM YYYY  [-title] [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprintcsv -l command outputs a list of application plan of job nets to standard output in CSV format.

Saving the output results in a file enables the job net application plan list to be managed with spreadsheet software. (Note)

Note:

Before importing the output result into the spreadsheet software, it is necessary to convert it into the character code supported by that spreadsheet software.

Options

-l

Outputs a list of application plan for the job net that can be browsed by the user executing this command, by specifying the desired month of the year.

Job nets that have been copied and started will not be output.

MM

Specify the month in which the date you want to output appears, in one or two digits.

YYYY

Specify the year in four digits.

-title

If this option is specified, the title is output on the first line.

EE | -sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the items to be output.

The job net startup days are indicated by an asterisk (*), and the non-startup days are indicated by a single space. Each asterisk or space is separated by a comma (,). Note that after the first day of the specified month/year there is a delimiting space, and after this comes either the asterisk (*) indicating the job net startup day or the space indicating the non-startup day.

Host Name

Host name of the server that executed this command

Sub System

Subsystem number which is corresponding to the Jobscheduler to be operated

Project Name

Name of the project where the job net is registered

Month/Year

Date specified by the -l option

Job Net

Name of the job net that is registered in the Jobscheduler

Job Net Comment

Job net comment that is registered in the Jobscheduler

Output Example

An example output result of this command is shown below.

The numbers indicate dates. Job net startup days are marked with an asterisk (*).

```
Host Name,Sub System,Project Name,Month/Year,Job Net,Job Net Comment,1,2,3,4,5,6,7,8,9,
10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31
host1,3,Sales,
3/2014,selling,SalesManagement, , , ,*, , , , , , , ,*, , , , , , , ,*, , , , , , , ,*, , , , ,
```

```
host1,3,Sales,
3/2014,customer,CustomerSalesManagement,  , ,*, , , , , , , , ,*, , , , , , , ,*, , , , , ,
,*, , , , ,
host1,3,Sales,3/2014,sales,CreateWeeklySalesReport,
*, , , , , , ,*, , , , , , ,*, , , , , , , ,*, , , , , ,*, ,
```

When you display the above file in spreadsheet software, you will see a listing similar to:

Host Name	Sub System	Project Name	Month/Year	Job Net	Job Net Comment	1	2	3	4
host1	3	Sales	3/2015	selling	SalesManagement				*
host1	3	Sales	3/2015	customer	CustomerSalesManagement			*	
host1	3	Sales	3/2015	sales	CreateWeeklySalesReport	*			

5.31 jobschprintcsv -f Command

Synopsis

```
jobschprintcsv - f P [-title] [-vparam] [-sys subsystem-number]⁽*¹⁾
```

*1:

[-sys subsystem-number] is the option of Systemwalker Operation Manager EE.

Description

The jobschprintcsv -f command outputs the execution result history of job nets to standard output in CSV format.

Saving the output result in a file enables the execution result history of job nets to be managed with spreadsheet software. (Note)

Note:

Before importing the output result into the spreadsheet software, it is necessary to convert it into the character code supported by that spreadsheet software.

Options

-f

Outputs the execution result history of the job net that can be browsed by the user executing this command.

-title

If this option is specified, the title is output on the first line.

-vparam

Displays the variable parameters for job nets that were started by specifying a variable parameter.

P

Specifies how far back (days) to output the history information, with a numeric value ranging from 0 to 5.

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/bin

Output Items

The following explains the items to be output.

Host Name

Host name of the server that executed this command

Sub System

Subsystem number which is corresponding to the Jobscheduler to be operated

Project Name

Name of the project where the job net is registered

Job Net

Name of the job net that is registered in the Jobscheduler

Job Net Comment

Job net comment that is registered in the Jobscheduler

Status

Status of job net

The relevant status is selected from the following eight and displayed:

Normal:

Completed

Pseudo:

Pseudo-normal

Abend:

Abended (completion code: from the upper limit of completion code +1 up to 255)

Forced:

Canceled (completion code: 256)

Execute:

Executing

E_Cancel:

Refused

Closed:

Interrupted (completion code: 239)

Skipped:

Skipped

Start Time

The date and time the job net started

It is displayed in "month/day/calendar-year hour:minute:second" format.

End Time

The date and time the job net was completed

It is displayed in "month/day/calendar-year hour:minute:second" format.

End Code

Completion code of the job net

Variable Parameter

This is the variable parameter for job nets that were started by specifying a variable parameter. The variable parameter is only displayed if the "-vparam" option was specified.

Caution

- When the Master Schedule Management function is enabled, use the stemprint -f(-F) command to output history information.

  This command does not output the history information on the process date that has been expired because the generation exceeded the number set in the [Store schedule information for] using the Master Schedule Management function. For more information on the stemprint -f(-F) command, see "6.12 stemprint -f(-F)/-g(-G) Command."

Output Example

An example output result of this command is shown below.

```
Host Name,Sub System,Project Name,Job Net,Job Net Comment,Status,Start Time,End Time,End
Code, Variable Parameter
host1,3,GeneralAffairs,payroll,PayrollAndBonusCalc, Normal,03/01/2014 10:00:01,03/01/2014
11:33:32,0,"200707"
host1,3,Sales,backup,BackupJob,Normal,03/01/2014 10:30:01,03/01/2014 11:49:26,0,
host1,3,Sales,sales,SalesDataManagement,Normal,03/01/2014 12:48:42,2014/03/01
12:50:56,0,"sales01.dat"
host1,3,Sales, management,CustomerManagement,Abend,03/01/2014 13:32:45,03/01/2014
13:42:37,2,
host1,3,GeneralAffairs,data,CashierDataManagement,Forced,03/01/2014 16:27:21,03/01/2014
16:31:21,256,
host1,3,General;Affairs,data,CashierDataManagement, Execute,03/01/2014 16:44:30,,,
```

When displayed in spreadsheet software, the above file appears in the following format:

Host Name	Sub System	Project Name	Job Net	Job Net Comment	Status	Start Time
host1	3	GeneralAffairs	payroll	PayrollAndBonusCalc	Normal	03/01/2015 10:
host1	3	Sales	backup	BackupJob	Normal	03/01/2015 10:3
host1	3	Sales	sales	S alesDataManagement	Normal	03/01/2015 12:48
host1	3	Sales	management	CustomerManagement	Abend	03/01/2015 13:32:
host1	3	GeneralAffairs	data	CashierDataManagement	Forced	03/01/2015 16:27:2
host1	3	GeneralAffairs	data	CashierDataManagement	Execute	03/01/2015 16:44:

Chapter 6 Master Schedule Management Commands

This chapter describes the commands used in the Systemwalker Operation Manager Master Schedule Management function.

📖 Note

If commands are executed in an environment where the UAC function for Windows Server 2008 or later has been enabled, then a dialog box asking a user's privileges to be promoted may be displayed depending on the privileges of the user that logged in to the operating system, and action may be required.

Refer to "Notes on Executing Commands in Windows [Windows version]" in the *Systemwalker Operation Manager User's Guide* for more information.

6.1 List of Commands

The following are the commands used for the Master Schedule Management function:

stemenvset environment setup command

Configures and removes the Master Schedule Management function environment.

stemManager schedule management command

Executes and manages the daily schedule management.

stemSetPrjMaster project master setting command

Sets the registered projects as the master definition of the daily schedule management.

stemSchDelete project delete command

Deletes a project including the information in the project.

stemCopyProject operation change preset command

Creates a project to be used in performing operation changes. (Copying the project)

stemMoveProject operation change release day change command

Changes a release day of project that was created for operation changes. (Moving the project)

stemBackProject operation change cancellation command

Returns the master information to the state before the release day that the operation change was implemented.

jobschmsgeventex message event generation command

Generates a message event based on process date.

stemprintnet save information confirmation command

Displays the saved schedule information.

stemConfig Master Schedule Management environment setup command [UNIX]

Changes the IP address of the management server or schedule server under the cluster environment.

stemprint -f(-F)/-g(-G) command

Outputs the job net history information if the Master Schedule Management function is enabled.

stemprint -n(-N) command

Outputs a job net's current status if the Master Schedule Management function is enabled.

stemschnet job net submission command

Submits (extract/register) job nets individually from the master information to the already created Schedule by dates project.

stemsetdbpath DB directory change command for cluster [Windows]

Changes a database directory of the Master Schedule Management function.

stemregservice service registration command for cluster [Windows]

Registers/deletes the service for cluster in the Master Schedule Management function. This command is used to use the Master Schedule Management function in a cluster environment.

 🅿 Point

・・

When the Extended User Management function is enabled, the access right authentication depends on the OS user that executes the jobschmsgeventex and stemprint commands:

- If the OS user is a system administrator, the commands are executed with the Operation Manager administrator privileges (has update rights for all projects).

- If the OS user is a project owner, the commands are executed with the assumption that the owner has update rights over their projects.

- If the OS user is neither a system administrator nor a project owner, the OS user that executed the commands are included in the OS users corresponding to all Operation Manager users that have access to the project, and the commands are executed if the user has the privileges. If the OS user that executed the commands corresponds to more than one Operation Manager user, the OS user will be given the most powerful privileges (update rights > change and operation rights > reference rights) among the privileges that the corresponding Operation Manager users have. If the corresponding Operation Manager user has both change rights and operation rights, then the OS user will also have both privileges.

When the Systemwalker authentication repository is enabled, the access right authentication depends on the OS user that executes the jobschmsgeventex and stemprint commands:

- If the OS user is a system administrator, the commands are executed with the Operation Manager administrator privileges (has update rights for all projects).

- If the OS user is a project owner, the commands are executed with update rights over the project they are the owner of.

- If the OS user is neither a system administrator nor a project owner, the commands are executed if the OS user has access rights over the project (update rights, change rights, operation rights, and reference rights).

・・

6.2 stemenvset Environment Setup Command

Synopsis

[Windows]

```
stemenvset.exe [-set | -u]
```

[UNIX]

```
stemenvset.sh [-set | -u]
```

Description

The stemenvset command configures and removes the environment of the Systemwalker Operation Manager Master Schedule Management function.

Options

When the options below are omitted, this command displays whether the Master Schedule Management function is enabled or not.

-set

Configures the environment of the Systemwalker Operation Manager Master Schedule Management function. Once this command is executed with this option, the Master Schedule Management function becomes available.

-u

Removes the environment of the Master Schedule Management function. In the system with multiple subsystems, the Master Schedule Management environment is removed from all of the subsystems.

Return Values

0:

Completed

A value other than 0:

Abended

Command Location

Windows	Systemwalker Operation Manager installation directory\mpwalker.jm\bin
Solaris HP-UX AIX Linux	/opt/FJSVstem/bin

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- You can't remove the environment of the Master Schedule Management function as long as the schedule distribution is taking place.

- When the environment of the Master Schedule Management function is removed, all information on the schedule state is discarded.

Execution Results/Output Format

Completed: None

Abended: Error contents are output to the standard output

Example

Refer to the *Systemwalker Operation Manager User's Guide - Master Schedule Management* for the examples of use of this command.

6.3 stemManager Schedule Management Command

Synopsis

[Windows]

```
stemManager [[-d yyyymmdd] [-force | -del | -chg] | -n] [-sys subsystem number]
```

[UNIX]

```
stemManager [[-d yyyymmdd] [[-force] [-h schedule_server]| -del | -chg] | -n] [-sys
subsystem number]
```

Description

The stemManager command applies the schedule information after extracting it from the master information. This command also performs schedule cancellation, rescheduling, future scheduling, and operation changes.

Options

-d *yyyymmdd*

Implements scheduling on the date specified by *yyyymmdd*. This option is used in the following cases:

- To recover the processing when the schedule fails to start automatically. Or,

- To register a future schedule in advance. Or,

- To register a schedule for the process date that has passed without performing Start schedule.

-n

Registers the schedule information as the Jobscheduler's job, and it creates the next day's schedule every time.

When the -d and -n options are omitted, the schedule for the current day is executed.

-force

Replaces extracted and applied master information with newly extracted information (rescheduling). This option is used in the following cases:

- When master information is changed after a schedule is executed on a future day. Or,

- When master information is changed when the schedule status for the current day is in "No information" status.

This option is not available for the current day whose schedule information is already created or a date in the past.

-h schedule_server [UNIX]

Recovers the schedule or performs rescheduling for the specified schedule server. This option is not available to register a new schedule or cancel an existing schedule.

-del

Cancels registered schedules.

-chg

Performs operation changes only.

-sys subsystem number

Specify a value between 0 and 9 for a subsystem that you want to schedule.

If omitted, the default is subsystem 0. When this command is activated as a job of the Jobscheduler under the subsystem operation environment, the number of the subsystem that activates this command is specified.

This command creates a schedule for the current day if all of the options are omitted.

Return Values

0:

Completed

A value other than 0:

Abended

Command Location

Windows	Systemwalker Operation Manager installation directory\mpwalker.jm\bin

Solaris HP-UX AIX Linux	/opt/FJSVstem/bin

Privileges Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- When the target schedule is already running, creation/change/deletion cannot be made on that schedule.

- The past date for executing a schedule can be set by going back only up to the same number of days as the number of days the schedule status is managed.

- For the job nets that have been registered with a past schedule, note the following:

    - If the maximum number (1000) of projects is exceeded, the projects are targeted for backup regardless of the status of job nets. The job nets that have been backed up are no longer operated by Systemwalker Operation Manager clients. You need to quickly execute and finish the job nets that you have registered.

    - Carry over control is not performed on these job nets.

    - The time-based scheduling is not executed, so you need to start the job nets manually if the job nets are configured to start at a specific time on the execution condition.

    - It is possible to link the job nets with message events. However, you need to generate a message event based on process date using the jobschmsgeventex command with the -d, -dx, -s and -n options.

      (If a message event that is not based on process date is generated, a job net may be started by the unexpected message.)

    - A job net that remains not executed is treated as completed, and even if the option of [Do not delete incomplete job nets] is selected in the **Master Schedule Management Environment Setup** dialog box, schedule information of the job net is deleted at the time it has been expired because the generation exceeded the number of stored generations.

- If schedule information was distributed over old versions of the schedule server, some functions may be unavailable. Refer to "Supportable Range in Management Server-Schedule Server Connection" in the *Systemwalker Operation Manager User's Guide - Master Schedule Management* for details.

Execution Results/Output Format

Completed:

None

Abended:

Error contents are output to the standard output and event log/SYSLOG. When the -force or -del option is specified, the error contents are output only to the standard output.

6.4 stemSetPrjMaster Project Master Setting Command

Synopsis

[Windows]

```
stemSetPrjMaster.exe {set | del} -p project name [-sys subsystem number]
```

[UNIX]

```
stemSetPrjMaster.sh {set | del} -p project name [-sys subsystem number]
```

Description

The stemSetPrjMaster command makes settings to treat the project registered in Systemwalker Operation Manager as master information of the daily schedule management. The project specified in this command is used as the master definition to be extracted and applied by the daily schedule management.

Since the project specified in this command is treated as the master definition, it is not scheduled on the master database.

Options

set

Performs master setting on a designated project. Scheduling the project on which the master setting has been performed is suppressed.

del

Deletes the master setting of the designated project.

-p project name

Specifies the project on which master setting is performed, or specifies the project of which master setting is deleted. The project name must be specified with 45 bytes or less.

-sys subsystem number

Specify a subsystem number between 0 and 9.

Only the projects in the subsystem specified in this option can be target of master setting. If omitted, the default is subsystem 0.

Return Values

0:

Completed

A value other than 0:

Abended

Command Location

Windows	Systemwalker Operation Manager installation directory\mpwalker.jm\bin
Solaris HP-UX AIX Linux	/opt/FJSVstem/bin

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- If the project name is suffixed by five bytes of "+" or "=" and four digits, the master setting is disabled.

- The projects that have been set in the master are no longer scheduled on the master database at either of the following points:

  - When the day change time is reached after the master settings were made

  - When the Jobscheduler daemon/service of the subsystem to which the project belongs has been restarted after the master settings were made

- The projects on which the master settings have been removed can be scheduled at either of the following points:

    - When the day change time is reached after the master settings were removed

    - When the Jobscheduler daemon/service of the subsystem to which the project belongs has been restarted after the master settings were removed

   Note that the projects stored in the master information folder of client are set as the master and are no longer scheduled.

Execution Results/Output Format

Completed:

None

Abended:

Error contents are output to the standard output.

Example

[UNIX]

When setting "SalesPRJ" as the master:

```
# stemSetPrjMaster.sh set -p SalesPRJ
```

6.5 stemSchDelete Project Delete Command

Synopsis

```
stemSchDelete project name [-mod mode] [-sys subsystem number]
```

Description

Deletes a specified project (any of master project, schedule by dates projects, or project for operation changes can be deleted). The specified project is deleted regardless of whether it contains job net information. However, it is not deleted when there is a job net in the state specified by the -mod option.

Options

project name

Specify the name of the project to be deleted with 50 bytes or less.

Specify the project name that is registered in the Jobscheduler. For example, specify "project name" for master information; "project name + MMDD" for schedule information; and "project name = MMDD" for the project for operation changes.

-mod *mode*

Specify the delete mode with 0 or 1. The criteria of job net's state that determines whether the project is deleted or not varies depending on the mode you specify:

0:

The project is not deleted if a job net or group that is currently being executed exists in the project.

1:

The project is not deleted if an incomplete job net or group is existent in the project.

"Incomplete" denotes the following status.

    - A job net or group is in Executing, Warning or Paused state.

    - Start date is set but the processing has not taken place yet. (Not completed)

    - In Abended or Canceled state. (Confirm operation is enabled but the operation is actually not executed.)

The default is 0.

-sys subsystem number

Specify the number of the subsystem that contains the project that you want to delete, in the range of 0 to 9. If omitted, the default is subsystem 0.

Return Values

Code	Meaning	Action
0	Completed	-
1	Usage display	Confirm how to specify the option.
2	The user has no privileges for executing this command.	System administrator (user belonging to the Administrator group or superuser) privileges are required.
3	A system error occurred	Check the system environment.
4	Execution unavailable because of the differences in the code systems.	The code system used for Systemwalker Operation Manager is not identical with that for the command execution environment. Use the same code system as that for Systemwalker Operation Manager.
5	DB access failed	Failed on opening, or reading from or writing to DB. Confirm the following. - Use event log/SYSLOG to check an MpJobsch error. - Use the anti-virus software or backup software to confirm whether another process accesses the database.
6	The Jobscheduler service/daemon is not running, or it is during startup processing.	Activate the service/daemon. If the service/daemon is already activated, it is during startup processing or it is not running properly. Wait a few moments, or reactivate it.
7	The specified project is not found.	Confirm whether the specified project name is correct.
8	The project cannot be deleted because a group or job net is currently executing.	After the currently executing group or job net terminates, execute the command again.
11	This command has failed in reading ACL manager library.	Confirm whether the following file exists. If the file is found, it may be damaged. Re-install the file. **[Windows]:** Systemwalker Installation Directory\mpwalker\bin\f3crsapi.dll **[Solaris/AIX/Linux]:** /opt/systemwalker/lib/libmpfwsecjm.so **[HP-UX]:** /opt/systemwalker/lib/libmpfwsecjm.sl
12	The project cannot be deleted because a carried over job net exists.	After addressing the problem by, for example, disabling the carried over job net, execute the command again.
13	The project cannot be deleted because a group or job net is currently operating or being stored	After the currently operating or stored group or job net terminates, execute the command again.
14	Timeout	Confirm the load of Systemwalker Operation Manager, then execute the command again.

Code	Meaning	Action
15	Unable to delete because a shutdown job net (JSHEND or JSHFORCE) has executed.	Restart the Jobscheduler, then execute the command again.

Command Location

Windows	Systemwalker Operation Manager installation directory\mpwalker.jm\bin
Solaris HP-UX AIX Linux	/opt/FJSVstem/bin

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Execution Results/Output Format

Completed:

None

Abended:

Error contents are output to the standard output.

6.6 stemCopyProject Operation Change Preset Command

Synopsis

```
stemCopyProject project name  -d release-day [-sys subsystem number]
```

Description

To change the master information in advance, the stemCopyProject command copies a project to be pre-registered in the master information to the same subsystem. Using this command, the system creates a project for operation changes (containing definition information on the same job net and job) in order for users to change that project. Contents of the project for operation changes are replaced with the master information on the release day.

Options

project name

Specify the target project of operation changes. The project name must be specified with 45 bytes or less.

-d release-day

Specify the release day in the mmdd format. The allowable range of release days is arbitrary. The operation change takes place during schedule distribution on the specified date.

-sys subsystem number

Specify the number of the subsystem between 0 and 9 that contains the project on which operation change is performed. If omitted, the default is subsystem 0.

Return Values

0:

Completed

A value other than 0:

Abended

Command Location

Windows	Systemwalker Operation Manager installation directory\mpwalker.jm\bin
Solaris HP-UX AIX Linux	/opt/FJSVstem/bin

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- Projects on which operation change settings are performed beforehand are restricted to master project.

- You cannot perform operation changes beforehand for the project on which the operation change has ended unsuccessfully.

Execution Results/Output Format

Completed:

None

Abended:

Error contents are output to the standard output.

6.7 stemMoveProject Operation Change Release Day Change Command

Synopsis

```
stemMoveProject project name -fd unchanged-release-day -d changed-release-day [-sys
subsystem number
```

Description

The stemMoveProject command changes the release day of the project for operation changes.

Options

project name

Specify the target project of operation changes.

-fd unchanged-release-day

Specify the unchanged release day in the mmdd format.

-d changed-release-day

Specify the changed release day in the mmdd format.

-sys subsystem number

Specify the number of the subsystem between 0 and 9 that contains the project on which the release day is to be changed. If omitted, the default is subsystem 0.

Return Values

0:

Normal termination

A value other than 0:

Abended

Command Location

Windows	Systemwalker Operation Manager installation directory\mpwalker.jm\bin
Solaris HP-UX AIX Linux	/opt/FJSVstem/bin

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

When the operation change has ended unsuccessfully on a process date (release day), you cannot change the release day of the project for operation changes to the failed process date.

Execution Results/Output Format

Completed:

None

Abended:

Error contents are output to the standard output.

6.8 stemBackProject Operation Change Delete Command

Synopsis

```
stemBackProject project name -d target-release-day  [-sys subsystem number]
```

Description

The stemBackProject command returns master information (in which operations are changed) to the definition information set before the release day.

Options

project name

Specify the master project whose operation change is to be cancelled.

-d target-release-day

Specify the Release day, on which the operation change has been conducted, in the mmdd format.

-sys subsystem number

Specify the number of the subsystem between 0 and 9 that contains the project in which the operation change is to be cancelled. If omitted, the default is subsystem 0.

Return Values

0:

Normal termination

A value other than 0:

Abended

Command Location

Windows	Systemwalker Operation Manager installation directory\mpwalker.jm\bin
Solaris HP-UX AIX Linux	/opt/FJSVstem/bin

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- When a subsystem contains a project that is experiencing errors due to operation change, you cannot cancel the operation change.

Execution Results/Output Format

Completed:

None

Abended:

Error contents are output to the standard output.

6.9 jobschmsgeventex Message Event Generation Command

Synopsis

[Windows]

```
jobschmsgeventex message-event-name[:project name[/job-net-name]]
    [-h host-name|-f file name] [-p param [-c suffix]] [-sys {all | subsystem number}]
    [-d mmdd | -dx| -n| -s]
```

[UNIX]

```
jobschmsgeventex message-event-name[:project name[/job-net-name]] [-h host-name]
    [-p param [-c suffix]] [-sys {all | subsystem number}] [-d mmdd | -dx| -n| -s]
```

Description

The jobschmsgeventex command generates a message event based on the process date in addition to the specifications of the conventional message event generation command (jobschmsgevent).

Options

message-event-name

Specify the message event name to be generated with 12 or fewer bytes.

project name

Specify a project name to generate message events for job nets in a specific project only.

If this option is used with the -dx, -d, -n or -s option, message events will be reported to the project specified for the operating day obtained with the -dx, -d, -n or -s option. If a project name for the operating day is specified, the operating day obtained with the -dx, -d, -n or -s option will be ignored and message events will be reported to the specified project.

job-net-name

Specify a target job net if you generate a message event only for a particular job net.

-h host-name

Specify the name of the host where a message event is to be generated. The host name must be specified with 64 or less alphanumeric characters.

-f file name [Windows]

Specify the file name that contains a list of hosts of servers where a message event is to be generated.

-p *param*

Specify the variable parameter to be passed to each job in the job net started by the message event. If this option is specified, the "@.VPARAM@" variables in the job definitions will be replaced with the specified variable parameter when the job net starts as a result of receiving the message event.

Specify the variable parameter using a string that is no more than 200 bytes long. The variable parameter should not include user-defined characters.

The variable parameter can be omitted. In this case, the "@.VPARAM@" variables in the job definitions will be replaced with an empty string before the jobs start.

Only job nets that wait for a single message event can receive the variable parameter specified with this option.

Variable parameters are not passed to job nets (child job nets) in the second or lower levels of nested job nets.

This option can be specified only once.

-c *suffix*

Use a string of up to 30 bytes long to specify the suffix that is attached to the names of job nets that are copied and started. Alphanumeric characters and hyphens (-) can be used, but only alphanumeric characters can be used for the first character. Be sure to specify this option together with (but after) the "-p" option.

Job nets that are started by specifying a variable parameter will be copied to a job net whose name has the specified suffix appended, and the copied job net will be started. For nested job nets, job nets in the second and lower levels will also be copied. This enables multiple job nets with different parameters to be executed in parallel and also allows the execution results of job nets with different parameters to be held without the execution results of one job net overwriting the execution results of another.

The names and comments of job nets that are copied become "*source job net name (or comment)* + *underscore* (_) + *string specified by the suffix* option".

Copying takes place when the job net starts (or, for nested job nets, when the parent job net starts). Job nets that have been copied are overwritten the next time a job net with the same name is copied.

Note that this option is invalid for job nets in groups.

-sys {all | subsystem number}

Specify the subsystem number using "all" or a value between 0 and 9 where a message event is to be generated. When "all" is specified, the message event is generated in all subsystems. If omitted, the message event is generated in subsystem 0. When, however, activated as a job for the Jobscheduler, the message event is generated in the subsystem having the same number as that of the activated-from subsystem.

-dx

This option is specified when defining this command as a job in a job net. The command cannot be executed with this option specified from the jobs that have not been registered with a schedule by dates project, or from the network jobs and distributed execution jobs that have been registered with a schedule by dates project.

If this option is specified, message events will be generated for the schedule information of the same operating day as the operating day of the job net with which this command is registered. Unlike the -s option, messages are not enabled for unapplied job nets for which **Valid only on startup days** has been set.

To generate message events on another host, use the -dx option together with the -h option and run as a local job.

-d *mmdd*

This option generates message events for schedule information of the specified operating day. Messages are not enabled for unapplied job nets for which **Valid only on startup days** has been set.

-n

This option is specified when defining this command as a job in a job net. The command cannot be executed with this option specified from the jobs that have not been registered with a schedule by dates project, or from the network jobs and distributed execution jobs that have been registered with a schedule by dates project.

If this option is specified, message events will be generated for schedule information of the operating day following the operating day of the job net with which this command has been registered. Messages will also be enabled for unapplied job nets for which **Valid only on startup days** has been set.

To generate message events on another host, use the -n option together with the -h option and run as a local job.

-s

This option is specified when defining this command as a job in a job net. The command cannot be executed with this option specified from the jobs that have not been registered with a schedule by dates project, or from the network jobs and distributed execution jobs that have been registered with a schedule by dates project.

If this option is specified, message events will be generated for the schedule information of the same operating day as the operating day of the job net with which this command is registered. Unlike the -dx option, messages are also enabled for unapplied job nets for which **Valid only on startup days** has been set.

To generate message events on another host, use the -s option together with the -h option and run as local job.

Return Values

0:

Normal termination

A value other than 0:

Abended

Command Location

Windows	Systemwalker Operation Manager Installation Directory\mpwalker.jm\bin
Solaris HP-UX	/opt/FJSVstem/bin

AIX	
Linux	

Privilege Required

This command is executable with the general users' privileges.

Cautions

- See "Cautions" in the 4.16 jobschmsgevent Command .

- It is not possible to use the -n or -s option to report message events to servers running Systemwalker Operation Manager V13.1.0 or earlier.
  If this is attempted, the following error message will be output and the message event generation process will terminate abnormally:

  > MpStem: ERROR: 10376: The option is not supported on the specified host. (*specified host name*)

Execution Results/Output Format

Completed:

None

Abended:

Error contents are output to the standard output.

6.10 stemprintnet Saved Information Confirmation Command

Synopsis

```
stemprintnet [-f file_name] [-title] [-sys subsystem number]
```

Description

The stemprintnet command outputs schedule information, which has been saved because the project exceeds its maximum value when a schedule is registered, in the csv format.

Options

-f *file_name*

Specify the output file name with a full path. If omitted, the contents are output to the standard output.

-title

Specify when the title line is output.

-sys subsystem number

Specify the number of the subsystem between 0 and 9, from which schedule information is obtained. If omitted, the default value is 0.

Return Values

0:

Completed

A value other than 0:

Abended

Command Location

Windows	Systemwalker Operation Manager Installation Directory\mpwalker.jm\bin
Solaris HP-UX AIX Linux	/opt/FJSVstem/bin

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Execution Results/Output Format

Completed:

When the -title option is specified, the execution result is output in the following format:

"Host Name, Sub System, Project Name, Job Net, Job Net Comment, Status, Scheduled Time, Start Time, End Time, and Completion Code."

```
Host Name,Sub System,Project Name,Job Net,Job Net Comment,Status,Scheduled Time,Start
Time,End Time,Completion Code
host1,9,over+1018,test007,test007,Pseudo,,10/18/2004 13:27:16,10/18/2004 13:27:16,2
host1,9,over+1018,test008,test008,Abend,,10/18/2004 13:26:21,10/18/2004 13:26:21,2
host1,9,action+1018,test010,test010,Disable,,,,
```

Abended:

Error contents are output to the standard output.

Output Items

The following explains the items to be output.

Host Name

The host name of the server on which stemprintnet command has been executed

Sub System

The target subsystem number of operation

Project Name

Project name

Job Net

Job net name

Job Net Comment

Job net comment

Status

Status of the job net

The relevant status is selected from the following thirteen and displayed.

Normal:

Completed (Completion code:0)

Pseudo:

Pseudo-normal (Completion code: from 1 to the upper limit of the normal completion code)

Abend:

Abended (Completion code: From upper limit of normal completion code+1 to 255)

Forced:

Cancel (Completion code: 256)

Warning:

Warning

Execute:

Executing

Wait:

Waiting

Pause:

Paused

Disable:

Disabled

Coped:

Confirmed

CarryOver:

Carried over

StartDelay:

Start delayed

EndDelay:

End delayed

Note that the "pseudo-normal" state is displayed as the "Completed" if the pseudo-normal option is not selected in the [Define Jobscheduler Startup Parameters] window.

Scheduled Time

Scheduled start time

It is displayed in "MM/DD/YYYY hh:mm:ss" format.

Start Time

Start date and time of the job net's previous execution

It is displayed in "MM/DD/YYYY hh:mm:ss" format.

End Time

End date and time of the job net's previous execution

It is displayed in "MM/DD/YYYY hh:mm:ss" format.

Completion Code

Completion code of the job net

6.11 stemConfig Master Schedule Management Environment Setup Command [UNIX]

Synopsis

```
stemConfig set {-m | -h schedule_server} {-ip ip_address}
-sys {all | subsystem number}
```

Description

The stemConfig command changes the IP addresses of the management server and schedule server when setting a logical IP address in a cluster environment. To execute this command, the Master Schedule Management environment must be configured and the daily schedule management must be set for the target subsystem.

Options

set

Use this option to set an IP address in the Master Schedule Management Environment Setup.

-m

Use this option when an IP address for the management server is set. If this option is specified, you need to change the setting on all schedule servers.

-h *schedule_server*

Use this option when the IP address of a schedule server is changed. Specify the host name of the IP address to be changed, which should be selected from the host names added in the [Schedule server] field in the **Master Schedule Management Environment Setup** dialog box - [Subsystem].

-ip *ip_address*

Specify the IP address that you want to set. This option must not be omitted.

-sys {all | subsystem number}

Specify the number of the subsystem with "all" or a value between 0 and 9 for the IP address to be set. When "all" is specified, this option is available for all subsystems where the daily schedule management is performed. If omitted, the default is subsystem 0.

Return Values

0:

Completed

A value other than 0:

Abended

Command Location

Solaris HP-UX AIX Linux	/opt/FJSVstem/bin

Privilege Required

System administrator (superuser privileges) are required.

Completed:

None

Abended:

Error contents are output to the standard output.

6.12 stemprint -f(-F)/-g(-G) Command

Synopsis

```
stemprint {{-f|-F} P | {-g|-G} YYYY MM DD } {[-long]} [ -csv [ -title ] ]}
          [-vparam] [-sys subsystem number]
```

Description

The stemprint -f(-F)/-g(-G) command outputs the history information of job nets.

Options

{-f|-F} *P*

Outputs the history information of job nets that can be browsed by the user executing this command, on a daily basis.

When the management server does not have the right to access the Schedule by dates project or the Schedule by dates project has been deleted, it outputs the history information depending on the access rights of the master project. The schedule server outputs the history information for the number of generations being specified from the [Store schedule Information for] in the **Master Schedule Management Environment Setup** dialog box - [Schedule].

Specify how far back (days) to output the history information with a numeric value ranging from 0 to 5.

{-g|-G} *YYYY MM DD*

Outputs the history information of job nets that can be browsed by the user executing this command.

When the management server does not have the right to access the Schedule by dates project or the Schedule by dates project has been deleted, it outputs the history information depending on the access right of the master project. The schedule server outputs the history information for the number of generations being specified from [Store schedule information for] of **Master Schedule Management Environment Setup**.

Specify the dates on which you want to output the history information in the format "YYYY MM DD" by using a space to separate the parts of the date.

-long

Outputs the full value of the "Job Net" and "Job Net Comment" output items.

-csv

Outputs the job net's history information in CSV format.

You can save the output in a file and manage the job net's history information with spreadsheet software. (Note)

Note:

Before loading the above output to the spreadsheet software, it must be translated to the character code supportable by the software.

-title

The title line is output in the first line. It is specifiable only when the -csv option is specified.

-vparam

Displays the variable parameters for job nets that were started by specifying a variable parameter.

-sys subsystem number

In systems with multiple subsystems, this option is used to specify a subsystem as operating target. Specify the operating target subsystem number using a range of 0 to 9. If omitted, the subsystem number 0 is specified.

Return Values

0:

Completed

A value other than 0:

Abended

Command Location

Windows	Systemwalker Operation Manager Installation Directory\mpwalker.jm\bin
Solaris HP-UX AIX Linux	/opt/FJSVstem/bin

Privilege Required

This command is executable with the general users' privileges.

Cautions

- Among the Jobscheduler's log files (jobdb1.log/jobdb2.log/jobdb3.log) placed under cyclic management, this command outputs the log files currently used and also log files of one generation ago.

- The information that is to be output on a daily basis is the job net's history information whose start information is present on the target date.

Example

```
stemprint -f 0 -vparam -sys 2
```

Execution Results/Output Format

Completed:

-csv option is not specified

```
   History Information of Job Net  date: 08/13/2015  time: 15:54

User Name: johnson

Day Change Time: 00:00

Project Name: GeneralAffairsDept

Date  Job Net        Job Net Comment       Status    Start Time            End Time              End Code  Variable Parameter
0620  payment        PayrollAndBonusCalc   Normal    08/13/2015 10:00:01   08/13/2015 11:33:32   0         "201507"
0620  backup         BackupJob             Normal    08/13/2015 10:30:01   08/13/2015 11:49:26   0
0620  sales          SalesDataManagement   Normal    08/13/2015 12:48:42   08/13/2015 12:50:56   0         "sales01.dat"
0620  management     CustomerManagement    Abend     08/13/2015 13:32:45   08/13/2015 13:42:37   2
0620  data           CashierDataManagement Forced    08/13/2015 16:27:21   08/13/2015 16:31:21   256
0620  data           CashierDataManagement Execute   08/13/2015 16:44:30
```

Completed:

-csv option is specified

```
Host Name,Sub System,Project Name,Date,Job Net,Job Net Comment,Status,Start Time,End Time,End Code,Variable Parameter
host1,3,GeneralAffairsDept,0813,payroll,PayrollAndBonusCalc,Normal,08/13/2015 10:00:01,08/13/2015 11:33:32,0,"201507"
host1,3,GeneralAffairsDept,0813,backup,BackupJob,Normal,08/13/2015 10:30:01,08/13/2015 11:49:26,0
host1,3,GeneralAffairsDept,0813,sales,SalesDataManagement,Normal,08/13/2015 12:48:42,08/13/2015 12:50:56,0,"sales01.dat"
host1,3,GeneralAffairsDept,0813,management,CustomerManagement,Abend,08/13/2015 13:32:45,08/13/2015 13:42:37,2
host1,3,GeneralAffairsDept,0813,data,CashierDataManagement,Forced,08/13/2015 16:27:21,08/13/2015 16:31:21,256
host1,3,GeneralAffairsDept,0813,data,CashierDataManagement,Execute,08/13/2015 16:44:30,,
```

Abended:

Error contents are output to the standard output.

Output Items

The following explains the items to be output.

User Name

Name of the user who has executed the stemprint command

It is not displayed when the -csv option is specified.

Day Change Time

Day change time

It is not displayed when the -csv option is specified.

Host Name

Host name of the server that has executed stemprint command

It is displayed only when the -csv option is specified.

Sub System

Subsystem number to be operated

It is displayed only when the -csv option is specified.

Project Name

Name of the master project

Date

Job net process date from the Master Schedule Management function.

It is left blank for the non-Schedule by dates projects.

Job Net

Job net name

The first 8 bytes of the job net name are displayed. If the "-long" option is specified, all 50 bytes of the job net name will be displayed.

Job Net Comment

Job net comment.

The first 24 bytes of the job net comment are displayed. If the "-long" option is specified, all 100 bytes of the job net comment will be displayed.

Status

Status of the job net

The relevant status is selected from the following eight and displayed.

Normal:

Completed (Completion code: 0)

Pseudo:

Pseudo-normal (Completion code: From 1 to the upper limit of the normal completion code)

Abend:

Abended (Completion code: From upper limit of normal completion code+1 to 255)

Forced:

Cancel (Completion code: 256)

Execute:

Executing

E_Cancel:

Refused

Closed:

Interrupted (Completion code: 239)

Skipped:

Skipped

Start Time

The date and time the job net started

It is displayed in "month/day/calendar-year hour:minute:second" format.

End Time

The date and time the job net was completed

It is displayed in "month/day/calendar-year hour:minute:second" format.

End Code

Completion code of the job net

Variable Parameter

This is the variable parameter for job nets that were started by specifying a variable parameter. The variable parameter is displayed only if the "-vparam" option was specified.

6.13 stemprint -n(-N) Command

Synopsis

```
stemprint {-n|-N} [-d MMDD | -p project name] [-toplevel]
      {[-long]|[ -csv [ -title ]]} [-sys subsystem number]
```

Description

The stemprint -n(-N) command outputs the current status of a job net.

Options

-n|-N

Outputs the current status of a job net that can be browsed by the user executing the command.

When the -d option or -p option is not specified, current status of the job net being registered to the Schedule by dates project on that day is output.

-d *MMDD*

Outputs status of the job net that is registered to all the Schedule by dates projects on the specified process date.

A process date is specified in the mmdd format.

-p project name

Outputs the current status of the job net being registered to the specified project.

-toplevel

This option outputs only single-level job nets or the first level of multi-level job nets. It does not output child job nets or linked job nets.

-long

Outputs the "Job Net" and "Job Net Comment" output items up to the maximum length.

-csv

Outputs the current status of the job net in CSV format.

You can save the output in a file and manage the current status of job net with spreadsheet software. (Note)

Note:

Before loading the above output to the spreadsheet software, it must be translated to the character code supportable by the software.

-title

The tile line is output in the first line. It is specifiable only when -csv option is specified.

-sys subsystem number

In systems with multiple subsystems, this option is used to specify a subsystem as operating target. Specify operating target subsystem number using a range of 0 to 9. If this option is omitted, the subsystem number 0 is specified.

Return Values

0:

Completed

A value other than 0:

Abended

Command Location

Windows	Systemwalker Operation Manager Installation Directory\mpwalker.jm\bin
Solaris HP-UX AIX Linux	/opt/FJSVstem/bin

Privilege Required

This command is executable with the general users' privileges.

Execution Results/Output Format

Completed:

-csv option is not specified

```
 Job Net Status List        date: 08/13/2015  time: 16:29
 User Name: johnson
 Project Name: Sales
 Job Net   Job Net Comment            Status    Scheduled Time    Start Time          End Time            Completion Code
 selling   SalesManagement            Abend                       08/12/2015 10:22    08/12/2015 10:38    10
 customer  CustomerSalesManagement    Wait      08/13/2015 18:00
 sales     CreateWeeklySalesReport    Disable   08/20/2015        08/10/2015 23:00    08/10/2015 23:12
 08/10/2015 23:12

 Project Name: GeneralAffairs
 Job Net   Job Net Comment            Status    Scheduled Time    Start Time          End Time            Completion Code
 cashier   CashierDataManagement      Execute   08/14/2015        08/13/2015 16:00
 payroll   PaymrollAndBonusCalc       Normal    09/10/2015        08/10/2015 23:00    08/10/2015 23:50    0
```

Completed:

-csv option is specified

```
Host Name,Sub System,Project Name,Job Net,Job Net Comment,Status,Scheduled Time,Start Time,End Time,Completion Code
host1,3,Sales,selling,SalesManagement,Abend,,08/12/2015 10:22,08/12/2015 10:38,10
host1,3,Sales,customer,CustomerSalesManagement,Wait,08/13/2015 18:00,,,
host1,3,Sales,sales,CreateWeeklySalesReport,Disable,08/20/2015,08/10/2015 23:00,08/10/2015 23:12,
host1,3,GeneralAffairs,cashier,CashierDataManagement,Execute,08/14/2015,08/13/2015 16:00,,
host1,3,GeneralAffairs,payroll,PayrollAndBonusCalc,Normal,09/10/2015,08/10/2015 23:00,08/10/2015 23:50,0
```

Abended:

Error contents are output to the standard output.

Output Items

The following explains the items to be output.

User Name

Name of the user who has executed stemprint command

It is displayed when the -csv option is not specified.

Host Name

Host name of the server that has executed stemprint command

It is displayed only when the -csv option is specified.

Sub System

Subsystem number to be operated

It is displayed only when the -csv option is specified.

Project Name

Project name

Job Net

Job net name

The first 8 bytes will be displayed. If the -long option is specified, the entire 50 bytes will be displayed.

Job Net Comment

Job net comment

The first 24 bytes will be displayed. If the -long option is specified, the entire 100 bytes will be displayed.

Status

Status of the job net

The relevant status is selected from the following thirteen and displayed.

Normal:

Completed (Completion code: 0)

Pseudo:

Pseudo-normal (Completion code: From 1 to the upper limit of the normal completion code)

Abend:

Abended (Completion code: From upper limit of normal completion code+1 to 255)

Forced:

Cancel (Completion code: 256)

Warning:

Warning

Execute:

Executing

Wait:

Waiting

Pause:

Paused

Disable:

Disabled

Coped:

Confirmed

CarryOver:

Carried over

StartDelay:

Start delayed

EndDelay:

End delayed

Note that the "pseudo-normal" state is displayed as the "Completed" if the pseudo-normal option is not selected in the [Define Jobscheduler Startup Parameters] window.

Scheduled Time

Scheduled startup time

It is displayed in "MM/DD/YYYY hh:mm" format.

Start Time

Start date and time of the job net's previous execution

It is displayed in "MM/DD/YYYY hh:mm" format.

End Time

End date and time of the job net's previous execution

It is displayed in "MM/DD/YYYYhh:mm" format.

Completion Code

Completion code of the job net

6.14 stemschnet Job Net Submission Command

Synopsis

```
stemschnet -ent project name/job net name -d yyyymmdd
      [-correct] [ -sys subsystem number ]
```

Description

The stemschnet command submits (extract/register) job nets individually from the master information to the already created Schedule by dates project. If a schedule server is already registered and a default host name is changed, job nets are submitted to the specified schedule server.

Options

-ent

Submits the specified job net to Schedule by dates project.

project name

Specify a project name in the master information.

Projects that can be specified are a master project or project for operation changes. The project name must be no longer than 50 bytes in length.

Under the following conditions, the project specified in this option cannot be registered.

- The Schedule by dates project on the desired process date cannot be found, to which you want to submit the project. And, the number of Schedule by dates projects has already reached the limits (256).

job net name

Specify a job net name in the master information. The job net name must be no longer than 50 bytes in length.

The following job nets cannot be specified:

- Layered (child) job net

- Linked job net

- Master linked job net

- Job net in a group

When submitting job nets, specify the job net with the process date whose start date is already set.

When submitting job nets, if the job net with the same name already exists in the Schedule by dates project on the desired process date, you cannot submit this job net.

-d *yyyymmdd*

Specify the process date on the destination to which job nets will be submitted in the yyyymmdd format.

The process date that can be specified is limited to the current day or future day. For the future day, up to 30 days ahead can be specified.

Only the process dates for which the schedule information has been distributed can be specified.

-correct

When a job net is submitted and the execution time specified in the master information has passed, the job net will be submitted with the execution time corrected to five minutes after the current time. A time correction will also be performed if the execution time is within five minutes of the current time. This option will be ignored for job nets whose execution condition is not **Execute at** (run once per day).

-sys subsystem number

In systems with multiple subsystems, this option is used to specify a subsystem as operating target. Specify operating target subsystem number using a range of 0 to 9. If omitted, the subsystem number 0 is specified.

Return Values

0:

Completed

A value other than 0:

Abended

Command Location

Windows	Systemwalker Operation Manager Installation Directory\mpwalker.jm\bin
Solaris HP-UX AIX Linux	/opt/FJSVstem/bin

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- Even if a job net or job in the master information is in a disabled state or stopped state, the status cannot be inherited after it has been submitted. And the status becomes "in a waiting state".

To retain the status in the master information after submission, follow the procedure below:

For job nets

1. Output the current status of registered job nets using the jobschprint -n command.

2. If a job net is in a disabled state or stopped state as the results of 1), change it to disabled state/stopped state by using the jobschcontrol command.

For jobs

1. Output the current status of registered jobs using the jobschprint -p command.

2. If a job is in a disabled state or stopped state as the results of 1), change it to disabled state/stopped state by using the jobjobschctljob command. (Note)

   Note: For the jobs that are to be controlled using the jobschctljob command, "job name" must be specified in addition to "job net name".

   - A job net with the -correct option specified cannot be submitted to a schedule server that is running Systemwalker Operation Manager V13.1.0 or earlier. If this is attempted, the job net submission command will terminate with an error.

Example

In the following example, the job net "NET001" registered in the master project "PRJ" is submitted to the Schedule by dates project "PRJ+0523".

```
# stemschnet -ent PRJ/NET001 -d 20050523
```

Execution Results/Output Format

Completed: None

Abended: Error contents are output to the standard output

[UNIX]

To submit job nets to a schedule server, regardless of whether or not it is "Completed" or "Abended", the schedule server name is output to the standard output in the following format:

If it is "Abended", after the schedule server name output, error contents are output.

```
Schedule Server : XXXX
```

XXXX: schedule server name

6.15 stemsetdbpath DB Directory Change Command for Cluster [Windows]

Synopsis

```
stemsetdbpath {-p | dbpath [-n] [-r] [-sys subsystem number]}
```

Description

The stemsetdbpath command changes a database directory of the Master Schedule Management function. Use this command when you relocate the database directory of the Master Schedule Management function on a shared disk in a cluster environment.

When executing the command on the active node, change the registration information of the database directory of the Master Schedule Management function, and then copy it, without using the -n option. For the standby node, use the -n option to change the registration information only.

Options

-p

Displays a present database directory.

dbpath

Specify a full path to the database directory to which you want to move. To move the database directory of the Master Schedule Management function, append "stemDBn (n: subsystem number) to the desired directory.

-n

Specify this option when changing only the registration information of the database directory. If omitted, the database is relocated to the destination directory.

-r

Backs up the database before the database directory is changed. Create a stemDB*n*.local (*n*: subsystem number) in the same directory as the previous one to save the database. If omitted, the database is not saved.

-sys subsystem number

Specify the subsystem number whose database directory is changed, using a range of 0 to 9. If omitted, the default is subsystem 0.

Return Values

0:

Completed

A value other than 0:

Abended

Command Location

Windows	Systemwalker Operation Manager Installation Directory\mpwalker.jm\bin

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

This command must be executed in an environment where the Jobscheduler service is inactive.

Example

- When you deploy a database directory on a shared disk on the active node while creating a cluster environment in a 1:1 active/standby configuration:

```
stemsetdbpath.exe y:\disk -sys 0
```

The database directory information of the subsystem 0 is changed into "y:\disk\stemDB0" to which the database will be moved.

- When you deploy a database directory on a shared disk on the standby node while creating a cluster environment in a 1:1 active/standby configuration:

```
stemsetdbpath.exe y:\disk -n -sys 0
```

The database directory information of the subsystem 0 is changed into "y:\disk\stemDB0". The database is not moved.

- When you deploy the database directory of the subsystem 1 on a shared disk on the active node while creating a cluster environment in a 2-node standby configuration:

```
stemsetdbpath.exe y:\disk -sys 1
```

The database directory information of the subsystem 1 is changed into "y:\disk\stemDB1" to which the database to be moved.

- When you display the current database directory:

```
stemsetdbpath.exe -p
```

The database directory currently configured is displayed.

Execution Results/Output Format

Completed:

- When the database directory is changed:

```
MpStem: INFO: stemsetdbpath.exe exited normally.
```

- When the database is displayed:

```
No. path
-----------------------------------------------
  1  z:\omgr1\mpstemsv\stemDB1
  2  y:\omgr2\mpstemsv\stemDB2
```

Abended:

Error contents are output to the standard output.

6.16 stemregservice Service Registration Command for Cluster [Windows]

Synopsis

```
stemregservice {-reg service logon account name | -del} [-sys subsystem number]
```

Description

The stemregservice command registers/deletes the service for cluster ("Systemwalker MpStem*n*" service) in the Master Schedule Management function. This command is used to register the Master Schedule Management function in a cluster environment.

Options

-reg

Registers the service for cluster of the Master Schedule Management function.

-del

Deletes the service for cluster of the Master Schedule Management function.

-sys subsystem number

Specify the number of the subsystem between 0 and 9, to/from which the service is registered/deleted. If omitted, the default value is 0.

Return Values

0:

Completed

A value other than 0:

Abended

Command Location

| Windows | Systemwalker Operation Manager Installation Directory\mpwalker.jm\bin |

Privilege Required

The user is a member of Administrators group and has the right "Act as part of the operating system".

Example

- When adding the service for cluster of the Master Schedule Management function on the subsystem 0:

```
stemregservice.exe -reg systemwalker -sys 0
```

- When deleting the service for cluster of the Master Schedule Management function on the subsystem 0:

```
stemregservice.exe -del -sys 0
```

Execution Results/Output Format

Completed:

```
MpStem: INFO: stemregservice.exe exited normally.
```

Abended:

Error contents are output to the standard output.

Chapter 7 Job Execution Control Commands

This chapter describes the Job Execution Control commands provided by Systemwalker Operation Manager. Note that the commands must be executed on the server where the Systemwalker Operation Manager server function is installed.

Note

. . . .

If commands are executed in an environment where the UAC function for Windows Server 2008 or later has been enabled, then a dialog box asking a user's privileges to be promoted may be displayed depending on the privileges of the user that logged in to the operating system, and action may be required.

Refer to "Notes on Executing Commands in Windows [Windows version]" in the *Systemwalker Operation Manager User's Guide* for more information.

. . . .

7.1 List of Commands

The following are the commands used in the Job Execution Control function.

Commands executed with system administrator privileges

The commands provided by Job Execution Control and executed using the system administrator privileges are discussed in the sections from 7.2 to 7.13 in this chapter.

mjstart start command

Starts the operation of the Job Execution Control.

mjstop stop command

Stops the operation of the Job Execution Control.

mjsview log file output command

Displays the job execution history and operation information of the Job Execution Control service/daemon.

qalter queue attribute alter command

Alters queue attributes.

qjalter job attribute alter command

Alters job attributes.

qstart queue start command

Starts the inactive queue.

qstop queue stop command

Stops the running queue.

mjdjadd distributed execution host add command

With distributed job execution, adds a destination host to the host group during its operation.

mjdjdel distributed execution host delete command

With distributed job execution, deletes a destination host from the host group during its operation.

mjdjalt distributed execution setup alter command

With distributed job execution, alters the multiplicity of jobs on the destination host during its operation.

mjsdaemon job execution control daemon start command [UNIX]

Starts the Job Execution Control daemons.

mjshut job execution control daemon stop command [UNIX]

Stops the Job Execution Control daemons.

mjsmdset startup mode specification command

Specifies the startup mode of the Job Execution Control.

Commands executed with the general users privileges

The commands provided by Job Execution Control and executed using the general users privileges are discussed in the sections from 7.14 to 7.21.

qdel job delete command

Deletes jobs.

qhold job hold command

Holds jobs.

qjstat job information display command

Displays the job information (job name/job number/queue name/job status etc.).

qmove inter-queue job transfer command

Moves a job to another queue.

qrls held job release command

Releases the held job.

qstat queue status display command

Displays the queue status.

qsub job submit command

Submits a job.

mjdjstat distributed execution status display command

Displays the destination host status with distributed job execution.

mjptree job process information display command [Windows]

Displays processes running under jobs. The user who submitted the job can also execute this command.

Cautions

If you request other servers to execute a network job, the job should be operated on the server that submitted it, not on the execution server that executes the job. If you operate it on the execution server, it is not certified to work correctly.

7.2 mjstart Start Command

Synopsis

```
mjstart [-sys subsystem-number](*1)
```

*1:

    [-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The Job Execution Control function starts when the following two processes are completed in sequence.

    a. Startup of the Job Execution Control service

    b. Startup of the Job Execution Control operation

The process (a) is automatically started when the Systemwalker Operation Manager server starts, and then the process (b) is automatically started.

The mjstart command starts the operation of Job Execution Control (process (b)).

Option

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin
Solaris HP-UX AIX Linux	 /usr/lib/mjes

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

7.3 mjstop Stop Command

Synopsis

```
mjstop [-sys subsystem-number](*1) [-c | -w] [-i]
```

*1:

[-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The Job Execution Control function stops when the following two processes are completed:

    a. Termination of the Job Execution Control operation

    b. Termination of the Job Execution Control service/daemon

The process (a) is automatically terminated when Systemwalker Operation Manager server is stopped, and then the process (b) automatically terminates.

The mjstop command stops the operation of Job Execution Control (process (a)).

Additionally, this command can declare the startup mode used when the Job Execution Control operation or service/daemon is started next time.

If the option is omitted:

  - The startup mode is not changed. [Windows]

  - The -w option is assumed to be specified.[UNIX]

For the startup modes, see the *Systemwalker Operation Manager User's Guide*.

Options

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

-c:

Sets the next startup mode to the initialization mode.

-w:

Sets the next startup mode to the continuation mode.

-i:

Stops only the service/daemon without terminating Workunits when stopping the service/daemon with this command.

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/lib/mjes

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- Consider the following when you restart Systemwalker Operation Manager after executing this command with the -i option:

    - Because the Workunit is not stopped, you will fail to restart the Workunit from Systemwalker Operation Manager. So be sure to stop the Workunit beforehand.

    In addition, you will fail to restart the Workunit after failover in a cluster environment. In such cases, you also need to stop the Workunit beforehand.

- Do not start this command when running a jobscheduler daemon. If this command is started when the jobscheduler daemon is running, the job net may be remained in the running status. Execute this command after stopping the jobscheduler daemon. [UNIX]

7.4 mjsview Log File Output Command

Synopsis

```
mjsview  [-sys subsystem-number](*1) [-c | -csv] [log file name]
```

*1:

[-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The log file output by the Job Execution Control contains binary attributes, which you cannot read directly. This is why you need to use the mjsview command to edit the log file so that it can be formatted to the standard output to be read.

See the *Systemwalker Operation Manager Installation Guide* for how to make setting for outputting a log file.

Also note that if "Save job execution history information" (log parameter) is not selected in the [Define Operating Information] window, the log file will be output to the following location.

[Windows]

Job Execution Control spool path\hist\date.log

date: mmddyyyy format

[UNIX]

/var/spool/mjes/hist/date.log

date: mmddyyyy format

Options

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

-c

This option allows to continue monitoring the log file to continuously format and output logs even after outputting the formatted log contents up to the time of command issuance. If a log file is not found, the command quits.

If this option is omitted, the command terminates after outputting the formatted contents of the log file up to the time of command issuance.

-csv

Specifying this option outputs the contents of the log file up to the time of command issuance to a csv format file, and then the command terminates.

log file name

Specifies the log file name.

If omitted, the last log file that has output job execution history information will be targeted for this command. You must specify a full path name to the log file such as when you process the log file contents on the specific day.

Return Values

Return Value	Meaning
0	The command process has terminated normally.
Other than 0	The command process has failed. An error message is output.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/lib/mjes

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- If Systemwalker Operation Manager is restarted or the server is down during job queuing, the log information may not be stored in the log file.

- For a job for recovery, since the job is restarted when you restart Systemwalker Operation Manager, the same log could be output again.

- If all of the following conditions are satisfied, the job purge log will be output to the log file at once.

    - On the execution server that executes network jobs.

    - You failed to connect from the execution server to the job submitting server when the job is completed. (The job submitting server encountered a system failure or line disconnection.)

    - After the connection failed, the connection has succeeded. (This may occur when the job submitting server is restarted or line connection has been reestablished.)

📖 Information
. . . .

About a trigger to switch to another log file

A log file is not automatically switched to another log file when the date has changed.

If the Job Execution Control service/daemon is running since the previous day and if no job is executed since the date has changed, it means that no log files for today are output. Therefore, a log file switch does not happen in this situation. The log file will be switched to another one when a job is executed after the date has changed.
. . . .

Output Example

The following is an example output in the mjsview log file output command.

```
11/11/2005 09:24:24 MJS901I MpMjes service start.
11/11/2005 14:33:00 MJS007I job sleep(13.general) accept
11/11/2005 14:33:00 MJS001I job sleep(13.general) start at queue1
11/11/2005 14:35:00 MJS002I job sleep(13.general) stop RC(0000)
11/11/2005 14:35:00 MJS006I job sleep(13.general) purge
11/11/2005 19:44:11 MJS007I job sleep(14.general) accept
11/11/2005 19:44:11 MJS001I job sleep(14.general) start at queue1
11/11/2005 19:44:21 MJS005I job sleep(14.general) cancel
11/11/2005 19:44:21 MJS002I job sleep(14.general) stop EC(40000222)
11/11/2005 19:44:21 MJS006I job sleep(14.general) purge
```

Remark:

Job Execution Control service activation
Completed job (jobnumber = 13)
Canceled job (job number = 14)

The internal code displayed with "EC" is the string of hexadecimal numbers.
The internal code displayed with "RC" is the string of decimal numbers.

The following is the example when the "-csv" option is specified.

```
11/11/2005,09:24:24,MJS901I,MpMjes service start.
11/11/2005,14:33:00,MJS007I,job sleep(13.general) accept
11/11/2005,14:33:00,MJS001I,job sleep(13.general) start on queue1
11/11/2005,14:35:00,MJS002I,job sleep(13.general) stop,RC(0000)
11/11/2005,14:35:00,MJS006I,job sleep(13.general) purge
11/11/2005,19:44:11,MJS007I,job sleep(14.general) accept
```

```
11/11/2005,19:44:11,MJS001I,job sleep(14.general) start on queue1
11/11/2005,19:44:21,MJS005I,job sleep(14.general) cancel
11/11/2005,19:44:21,MJS002I,job sleep(14.general) stop,EC(40000222)
11/11/2005,19:44:21,MJS006I,job sleep(14.general) purge
```

Remark:

Job Execution Control service activation
Completed job (jobnumber = 13)
Canceled job (job number = 14)

The internal code displayed with "EC" is the string of hexadecimal numbers.
The internal code displayed with "RC" is the string of decimal numbers.

Execution Results/Output Format

```
1) mm/dd/yyyy hh:mm:ss MJS001I job jobname(jobno.hostname) start at queuename
2) mm/dd/yyyy hh:mm:ss MJS002I job jobname(jobno.hostname) stop RC(rc)|EC(ec)
3) mm/dd/yyyy hh:mm:ss MJS003I job jobname(jobno.hostname) step stepname start
4) mm/dd/yyyy hh:mm:ss MJS004I job jobname(jobno.hostname) step stepname stop RC(rc)|EC(ec)
5) mm/dd/yyyy hh:mm:ss MJS005I job jobname(jobno.hostname) cancel
6) mm/dd/yyyy hh:mm:ss MJS006I job jobname(jobno.hostname) purge
7) mm/dd/yyyy hh:mm:ss MJS007I job jobname(jobno.hostname) accept from host-name
8) mm/dd/yyyy hh:mm:ss MJS901I MpMjes service start.
9) mm/dd/yyyy hh:mm:ss MJS902I MpMjes service stop.
```

The following describes the meaning of each log:

1) The job has started.

2) The job has finished.

3) The job step has started.

4) The job step has finished.

5) The cancel request has been accepted.

6) The job has been removed from a spool.

7) The job has been accepted.

8) The Job Execution Control service/daemon has started.

9) The Job Execution Control service/daemon has finished.

The following explains the contents of the variable parts to be output.

jobname:

Job name is output.

jobno:

The job number assigned by the Job Execution Control is output.

hostname:

The host name that submitted the job is output.

queuename:

The queue name from which the job is started.

stepname:

Job step name is output.

RC(rc)|EC(ec):

Either RC(rc) or EC(ec) is output. If the job processing is aborted due to cancel or network problems, EC(ec) is output.
Otherwise, RC(rc) is output.

rc:

Job completion code is output in decimal.

ec:

Completion code of error is output in hexadecimal. If a job is canceled, "40000222" is output.

host-name:

The host name of the server that requested the network jobs/Distributed Execution jobs is output. If the execution server accepted network jobs or Distributed Execution jobs that were requested from other remote server, "from host-name" is additionally output on the execution server.

7.5 qalter Queue Attribute Alter Command

Synopsis

```
qalter [-sys subsystem-number](*1)  queue-name parameter
```

*1:

[-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The qalter command alters the queue attributes.

The following attributes can be altered:

- Max submit jobs

- Max execution jobs

- Time lapse limit

- Execution priority

- Dispatch priority

Option

`EE`

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Operands

queue-name

Specifies the name of the queue whose attributes are to be altered.

parameter

Specifies the attribute and its value to be altered.

This operand can be specified more than one at the same time.

The attributes and values that can be altered are:

maxjob

Max submit jobs

The range of possible values: 1-999

maxexec

Max execution jobs (number of simultaneously executable jobs)

The range of possible values: 1-99 [Windows], 1-999 [UNIX]

elapse

Time lapse limit

The range of possible values: 1-99999999

qprty

Execution priority

The range of possible values: 0-63

dprty

Dispatch priority

The range of possible values: 0-4 [Windows], -20-19 [UNIX]

To specify those parameters, use the same format as for the Job Execution Control initialization parameters.

The following is an example where to specify "Max submit jobs" as five.

```
maxjob=5
```

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/lib/mjes

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- The altered attributes are retained until the Job Execution Control is activated in the initialization mode. Activating the Job Execution Control in the initialization mode resets the queue attributes to what is defined in the initialization file.

7.6 qjalter Job Attribute Alter Command

Synopsis

```
qjalter [-sys subsystem-number]⁽*¹⁾ -dp dispatch-priority job
qjalter [-sys subsystem-number]⁽*¹⁾ -p priority job
qjalter [-sys subsystem-number]⁽*¹⁾ -dp dispatch-priority -p priority job
```

*1:

[-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The qjalter command alters the job attributes.

The following attributes can be altered:

- Dispatch priority

- Execution priority

Options

EE

<div style="background-color:#fadadd">

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

</div>

-dp *dispatch-priority*

[Windows]

Specify the priority of the threads of the program that are to be executed from the job by specifying an integer value between 0 and 4. The greater the value is, the higher the priority is. This option is not available for the job in a "run" or "oph" state.

[UNIX]

Specify the priority of the processes executed from the job by specifying an integer value between -20 and 19. Dispatch priority shows the CPU allocation priority. 19 is the lowest and -20 the highest. This option is not available for the job in a "run" or "oph" state.

-p *priority*

Specify the priority of jobs in a queue by specifying an integer value between 0 and 63. 0 is the lowest priority, and 63 the highest. This option is not available for the job in a "run" or "oph" state.

Operand

job

Specifies a job whose attributes are to be altered.

Use the following format to specify a job:

```
job-name(jobno.hostname)
```

"job-name" is the job name specified when the job is submitted.

"jobno" is the acceptance number that the Job Execution Control allocates.

"hostname" is the name of the host that submitted the job. If omitted, the default is a local host name. .

"jobno.hostname" can be omitted if there is not more than one job with the same name.

Wildcards cannot be specified for the job name.

If the job is specified as follows, the command will end normally:

- job-name

- job-name(jobno)

- job-name(jobno.hostname)

- jobno

- jobno.hostname

If specified as follows, the command will result in error:

- job-name(hostname)

- job-name(.hostname)

- (hostname)

- hostname

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin
Solaris HP-UX AIX Linux	 /usr/lib/mjes

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

7.7 qstart Queue Start Command

Synopsis

```
qstart [-sys subsystem-number]⁽*¹⁾ queue-name ...
```

*1:

[-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The qstart command restarts a queue that is currently stopping.

Option

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Operand

queue-name

Specifies the name of the queue that you want to start.

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Windows	*\<Systemwalker Operation Manager installation directory\>*\MpWalker.JM\bin
Solaris HP-UX AIX Linux	 /usr/lib/mjes

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

7.8 qstop Queue Stop Command

Synopsis

```
qstop [-sys subsystem-number]⁽*¹⁾ queue-name ...
```

*1:

   [-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The qstop command stops a queue that is currently running.

Option

EE

-sys subsystem-number

   In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Operand

queue-name

Specifies the name of the queue that you want to stop.

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Windows	*\<Systemwalker Operation Manager installation directory\>*\MpWalker.JM\bin
Solaris HP-UX AIX Linux	 /usr/lib/mjes

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

7.9 mjdjadd Distributed Execution Host Add Command

Synopsis

```
mjdjadd [-sys subsystem-number](*1) hostgroup-name host-name [parameter]
```

*1:

[-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The mjdjadd command adds a configuration host to the destination host group with the distributed job execution during its operation.

Option

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Operands

hostgroup-name

Specifies the name of host group to which you want to add a configuration host.

host-name

Specifies the name of host to be added. The already registered host name cannot be added. Up to 100 hosts can be set in a single host group. The local host should not be specified for the host name.

parameter

Specifies the attribute and its values of the host to be added. This operand is optional.

The attribute, its values, and default value that can be specified are:

maxexec

Max execution jobs (number of simultaneously executable jobs)

The range of possible values for Windows: 1-99

The range of possible values for UNIX: 1-999

Default: 10

The following is an example where to specify "Max Execution Jobs" as five.

```
maxexec=5
```

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Windows	<*Systemwalker Operation Manager installation directory*>\MpWalker.JM\bin

Solaris HP-UX AIX Linux	/usr/lib/mjes

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- This command is executable only on the Systemwalker Operation Manager server (source server of the distributed job execution).

- The option of "Enable distributed execution" cannot be changed during its operation.

- The definition can be changed only when:

  - The queue is working as the distributed execution queue (that is, when the [Enable distributed execution] is selected in the [Create Queue] or [Edit Queue] window).

  - The Job Execution Control service/daemon is running.

- The changes made on the definition are retained until the Job Execution Control function is activated in the initialization mode. Activating the Job Execution Control function in the initialization mode resets the settings for the distributed job execution to what are defined in the initialization file.

- When you add a local host to the host group, do it from the [Define Operating Information] window, not by using this command. Otherwise, the distributed execution job may terminate abnormally with completion code 0x40000999, 1, or others.

Example

In the following example, the host "djhost2" is added to the host group "djgroup".

```
C:\>mjdjadd djgroup djhost2 maxexec=5
hostgroup = djgroup
  hostname = djhost0
    maxexec = 20  state = ACT    run = 2
  hostname = djhost1
    maxexec = 7   state = ACT    run = 0
  hostname = djhost2
    maxexec = 5   state = ACT    run = 0
```

7.10 mjdjdel Distributed Execution Host Delete Command

Synopsis

```
mjdjdel [-sys subsystem-number](*1) hostgroup-name host-name
```

*1:

[-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The mjdjdel command deletes a configuration host from the destination host group with the distributed job execution during its operation.

Option

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Operands

hostgroup-name

Specifies the name of host group from which you want to delete a host.

host-name

Specifies the name of host that you want to delete. A local host should not be specified for this operand.

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/lib/mjes

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- This command is executable only on the Systemwalker Operation Manager server (source server of the distributed job execution).

- The option of "Enable distributed execution" cannot be changed during its operation.

- The definition can be changed only when:

    - The queue is working as the distributed execution queue (that is, when the [Enable distributed execution] is selected in the [Create Queue] or [Edit Queue] window).

    - The Job Execution Control service/daemon is running.

- The changes made on the definition are retained until the Job Execution Control is activated in the initialization mode. Activating the Job Execution Control in the initialization mode resets the settings for the distributed job execution to what is defined in the initialization file.

- You can use this command if there are more than one host in the host group. Because at least one host must be defined in the host group.

- If you attempt to delete the host where a job is left running, the host is deleted by definition. And the running job continues processing, but jobs are no longer submitted to this host from that point.

- When you delete the local host that was defined as the configuration host, do it from the [Define Operating Information] window, not by using this command. Otherwise, the distributed execution job may terminate abnormally with completion code 0x40000999, 1, or others.

Example

In the following example, the host "djhost1" is deleted from the host group "djgroup".

```
C:\>mjdjdel djgroup djhost1
hostgroup = djgroup
  hostname = djhost0
    maxexec = 20  state = ACT    run = 2
  hostname = djhost2
    maxexec = 5   state = ACT    run = 0
```

7.11 mjdjalt Distributed Execution Setup Alter Command

Synopsis

```
mjdjalt [-sys subsystem-number]⁽*¹⁾ hostgroup-name host-name parameter
```

*1:

[-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The mjdjalt command alters the settings with the distributed job execution during its operation.

Option

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Operands

hostgroup-name

Specifies the name of the host group whose settings you want to alter.

host-name

Specifies the name of the destination host whose settings you want to alter.

parameter

Specifies the attribute and its values of the host to be altered.

The attribute and its values that can be specified are:

maxexec

Max execution jobs (number of simultaneously executable jobs)

The range of possible values for Windows: 1-99

The range of possible values for UNIX: 1-999

If the value of the modified multiplicity is smaller than that of the number of running jobs, the number of running jobs temporarily exceeds the multiplicity. However, it can be finally limited to the new multiplicity to which you altered as the running jobs are complete.

The following gives an example to set the max execution jobs to three.

```
maxexec=3
```

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/lib/mjes

Privilege Required

System administrator (user belonging to the Administrator group or superuser) privileges are required.

Cautions

- This command is executable only on the Systemwalker Operation Manager server (source server of the distributed job execution).

- The option of "Enable distributed execution" cannot be changed during its operation.

- The definition can be changed only when:

    - The queue is working as the distributed execution queue (that is, when the [Enable distributed execution] is selected in the [Create Queue] or [Edit Queue] window).

    - The Job Execution Control service/daemon is running.

- The changes made on the definition are retained until the Job Execution Control is activated in the initialization mode. Activating the Job Execution Control in the initialization mode resets the settings for the distributed job execution to what is defined in the initialization file.

Example

In the following example, the multiplicity of jobs set to the host "djhost1" in the host group "djgroup" is altered to three.

```
C:\>mjdjalt djgroup djhost1 maxexec=3
hostgroup = djgroup
  hostname = djhost0
    maxexec = 20  state = ACT    run = 2
  hostname = djhost1
    maxexec = 3   state = ACT    run = 0
  hostname = djhost2
    maxexec = 5   state = ACT    run = 0
```

7.12 mjsdaemon Job Execution Control Daemon Start Command [UNIX]

Synopsis

```
mjsdaemon [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The Job Execution Control starts when the following two processes are completed in sequence.

 a. Startup of the Job Execution Control daemon

 b. Startup of the Job Execution Control operations

The process (a) is automatically activated when the Systemwalker Operation Manager server starts, and then the process (b) is automatically activated.

The mjsdaemon command starts the Job Execution Control daemon and the operation of the Job Execution Control. Use this command from the following job execution demon startup scripts:

Solaris Linux	/etc/opt/FJSVMJS/etc/rc3.d/S99MJS
HP-UX	/opt/FHPMJS/usr/lib/mjes/S99MJS
AIX	/opt/FAIXMJS/usr/lib/mjes/S99MJS

Option

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Solaris HP-UX AIX Linux	/usr/lib/mjes

Privilege Required

System administrator (superuser) privileges are required.

Cautions

- Be sure to use this command from the job execution control daemon startup script. If you execute the command directly, the settings such as environment variable configured within the job execution control daemon startup script are not set, which causes the problem of inconsistent character codes.

- In Red Hat Enterprise Linux 7, the operating system may need to be stopped or restarted even if the daemons are configured to stop automatically when the operating system is stopped.
  In this case, manually execute the poperationmgr service/daemon termination command in advance to stop the Systemwalker Operation Manager daemons. [Linux]

7.13 mjshut Job Execution Control Daemon Stop Command [UNIX]

Synopsis

```
mjshut [-sys subsystem-number](*1)
```

*1:

[-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The Job Execution Control function is terminated when the following two processes are completed:

    a. Termination of the Job Execution Control operations (mjstop command)

    b. Termination of the Job Execution Control daemon

The process (a) is automatically terminated when Systemwalker Operation Manager server is stopped, and then the process (b) is automatically terminated.

The mjshut command stops the operation of the Job Execution Control, and stops the Job Execution Control daemon.

If this command is executed without executing the mjstop command, the next startup mode is the continuation mode (same as the mode specified in the mjstop with -w option).

Option

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Solaris HP-UX AIX Linux	/usr/lib/mjes

Privilege Required

System administrator (superuser) privileges are required.

Cautions

- Do not start this command when running a jobscheduler daemon. If this command is started when the jobscheduler daemon is running, the job net may be remained in the running status. Execute this command after stopping the jobscheduler daemon.

- In Red Hat Enterprise Linux 7, the operating system may need to be stopped or restarted even if the daemons are configured to stop automatically when the operating system is stopped.
In this case, manually execute the poperationmgr service/daemon termination command in advance to stop the Systemwalker Operation Manager daemons. [Linux]

7.14 qdel Job Delete Command

Synopsis

```
qdel [-sys subsystem-number]("1) [-u user-name] job ...
```

*1:

> [-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The qdel command deletes the specified job.

The following users can use this command to delete jobs.

- The user who submitted the job

- System administrator

Options

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

-u *user-name*

When the system administrator deletes other user's job, the user name of the job owner must be specified in this option.

If blanks are contained in the user name, enclose the entire user name in double quotation marks. [Windows]

The user of Operation Manager cannot be specified. [UNIX]

Operand

job

Specifies a job that to be deleted.

Use the following format to specify the job:

For more details on each attribute, see "7.6 qjalter Job Attribute Alter Command."

```
job-name(jobno.hostname)
```

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin
Solaris Linux	/opt/FJSVMJS/usr/bin
HP-UX	/opt/FHPMJS/usr/bin
AIX	/opt/FAIXMJS/usr/bin

Cautions

- The job name "job-name (jobno.hostname)" cannot be omitted.

7.15 qhold Job Hold Command

Synopsis

```
qhold [-sys subsystem-number]⁽*¹⁾ [-u user-name] job ...
```

*1:

[-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The qhold command holds the specified job.

The following users can hold jobs:

- The user who submitted the job

- System administrator

To release the job in a holding state, use the qrls command.

Options

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

-u *user-name*

When the system administrator holds other user's job, the user name of the job owner must be specified in this option. If the job owner holds his/her own job, this option can be omitted.

The user of Operation Manager cannot be specified. [UNIX]

The job in a holding state made by the system administrator cannot be released even by the owner of this job.

If blanks are contained in the user name, enclose the entire user name in double quotation marks. [Windows]

Operand

job

Specifies the job to be held.

To specify the job, use the following format:

For more details on each attribute, see "7.6 qjalter Job Attribute Alter Command".

```
job-name(jobno.hostname)
```

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin
Solaris Linux	/opt/FJSVMJS/usr/bin

HP-UX	/opt/FHPMJS/usr/bin
AIX	/opt/FAIXMJS/usr/bin

Cautions

The job name "job-name (jobno.hostname)" cannot be omitted.

Only jobs in a "que" state can be held. Then, the job enters a "hld" state.

Jobs in any other state than the "que" cannot be held.

7.16 qjstat Job Information Display Command

Synopsis

```
qjstat [-sys subsystem-number](*1) [-u user-name] [-l] [-m] [-x] [job ...]
```

*1:

[-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The qjstat command displays the following job information:

- Job name

- Job number

- Queue name

- Job status

If this command is executed by the system administrator, the following information is displayed as well:

- Client host name

- Client user name

Depending on the privileges that the user executing this command has, the jobs that can be displayed differ as follow:

- Privileges other than the system administrator privileges:

  Only the submitted jobs can be displayed.

- If the system administrator executes this command without specifying the -u option:

  All jobs can be displayed.

If all the options are omitted, the status of the job that was submitted by the user executing this command is displayed.

If the appropriate job does not exist, nothing is displayed.

Options

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

-m

Displays the following job information if the user executing this command is the system administrator:

- Job process number

  Process number of the process that was generated when a local job is executed. The process number of network jobs, Workunit jobs, JCL, and the distributed execution jobs (when job execution is performed on a remote host) is not displayed.

- Request host name

  Request host name of network jobs, JCL, and the distributed execution jobs. If the execution server is configured in the mirrored configuration, the host name to which job execution is requested is displayed.

-u *user-name*

When the system administrator displays the information of a particular user's job, the user name of the job owner must be specified in this option.

If blanks are contained in the user name, enclose the entire user name in double quotation marks. [Windows]

The user of Operation Manager cannot be specified. [UNIX]

-l

Displays the detailed job information.

- Execution priority

- Dispatch priority

- The time that has elapsed since the job execution start time (only for jobs currently being executed)

If the system administrator specifies this option, the command also displays the following information:

- The date and time that a job was submitted

- The date and time that a job execution started (only for jobs currently being executed)

-x

Displays the information of jobs that are running (in a run state).

Operand

job

Specifies the job that whose information is to be displayed.

Use the following format to specify this operand:

For more details on each attribute, see "7.6 qjalter Job Attribute Alter Command".

```
job-name(jobno.hostname)
```

Job status

A job will enter various statuses after being submitted until it completes.

Those statues are the following:

hld

Indicates that the job status is on hold, and cannot enter another status.

que

Indicates that the job is waiting for the pre-execution process and its turn to enter the "run" state.

run

Indicates that the job is currently executing.

oph

Indicates that outputting is being suspended.

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin
Solaris Linux	/opt/FJSVMJS/usr/bin
HP-UX	/opt/FHPMJS/usr/bin
AIX	/opt/FAIXMJS/usr/bin

Output Example

The following is an example of the display when the system administrator executed the command with the "-l" and "-m" options.

```
(1)    jobname = updatejob
(2)     jobno = 00300
(8)     clienthost = client1        clientuser = user1              (9)
(3)     queuename = queue1@host1
(4)     status   = run
(5)     qprty = 31                  input date = 11.18.2015 input time = 20:03:56  (10)
(6)     dprty =  2 elapsed =     1231 start date = 11.18.2015 start time = 20:03:57  (11)
(12)    jobprocid =
(13)    exechost  =
                        ↑ (7)
```

The following is an example of the display when the general users executed the command with the "-l" option.

```
jobname            jobno  queuename          status qprty dprty  elapsed
updatejob          00300  queue1@host1          run    31     2     1231
(1)                (2)    (3)                   (4)   (5)   (6)      (7)
```

The following lists the items that are displayed in the above examples:

(1) Job name (The job name length is limited to 20 bytes if the job is executed by a general user.)

(2) Job number

(3) Queue name

(4) Job status [run: executing, que: waiting, hld: holding, oph: holding output]

(5) Execution priority

(6) Dispatch priority

(7) The time that has elapsed since the job execution start time (only for jobs currently executing)

(8) Client host name

(9) Client user name

(10) The date and time that the job was submitted

(11) The date and time that the job execution started (only for jobs currently running)

(12) Job process number

(13) Request host name

Cautions (the order of information displayed)

- Jobs are sorted by status. If more than one job with the same status exists, they are sorted by job number for each status.

7.17 qmove Inter-queue Job Transfer Command

Synopsis

```
qmove [-sys subsystem-number]⁽*¹⁾ [-d] [-u user-name] [-z] job [queue]...
```

*1:

[-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The qmove command moves the specified job on the local server from the current queue to the specified queue. (The job is deleted from its current queue and placed in the queue specified with the queue operand). Only when the job is in a "que" state, it can be moved.

The following users can move a job:

- The user who submitted the job

- System administrator

The following cannot be performed using this command:

- Moving the jobs submitted in the distributed execution queue

- Moving jobs into the distributed execution queue.

Options

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

-d

Displays the result every time you attempt to move a job to the queue specified with the queue operand.

-u *user-name*

When the system administrator moves other user's job, the user name of the job owner must be specified in this option.

If blanks are contained in the user name, enclose the entire user name in double quotation marks. [Windows]

The user of Operation Manager cannot be specified. [UNIX]

-z

If you specify this option, no message to report that the specified job has been successfully moved is displayed.

If this option is omitted, the job name and destination queue name are displayed when the job has been moved successfully. If the -u option is specified with this operand, the name of the job owner is also displayed.

If the move failed, or an error occurs during the move, an error message will be generated regardless of this option.

Operands

job

Specifies the job to be moved.

Use the following format to specify this operand:

For more details on each attribute, see "7.6 qjalter Job Attribute Alter Command".

```
job-name(jobno.hostname)
```

queue

Specify the destination queue in "queue"

Wildcards cannot be used in queue names.

If this operand is omitted, the move will be attempted to the default queue specified in "Specify the default queue to submit jobs to" (the dfltqueue operand in the system parameter) in the [Define Operating Information] window.

You can specify more than one queue name. The Job Execution Control function attempts to move a job to the queues in the specified order, and moves the job to the first available queue it finds. Even if there are specified queues left behind after moving the job, it cannot be moved to those queues.

If none of the specified queues is found or if the queue currently containing the job is specified as the destination, the Job Execution Control performs no action.

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin
Solaris Linux	/opt/FJSVMJS/usr/bin
HP-UX	/opt/FHPMJS/usr/bin
AIX	/opt/FAIXMJS/usr/bin

Cautions

- Job name "job-name (jobno.hostname)" cannot be omitted.

Example

For example, the following command shows how the system administrator moves the job (job number 1) of "user1" into "queue3" (due to some reason, if moving to "queue3" failed, the job 1 will be moved into "queue4").

```
qmove -d -u user1 1 queue3 queue4
```

7.18 qrls Held Job Release Command

Synopsis

```
qrls [-sys subsystem-number]⁽*¹⁾ [-u user-name] job ...
```

*1:

[-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The qrls command releases the specified job that is in a holding (hld) state.

The following users can release a job.

- The user who submitted the job

- System administrator

The job released with this command becomes the status that existed prior to being held (that is, to the "que" state).

Options

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

-u *user-name*

When the system administrator releases a job, the name of the user who submitted the job must be specified in this option.

If blanks are contained in the user name, enclose the entire user name in double quotation marks. [Windows]

The user of Operation Manager cannot be specified. [UNIX]

Operand

job

Specifies the job to be released.

Use the following format to specify this operand:

For more details on each attribute, see "7.6 qjalter Job Attribute Alter Command".

```
job-name(jobno.hostname)
```

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin
Solaris Linux	/opt/FJSVMJS/usr/bin
HP-UX	/opt/FHPMJS/usr/bin
AIX	/opt/FAIXMJS/usr/bin

Cautions

The job name "job-name (jobno.hostname)" cannot be omitted.

For the job that was changed into "holding state" by the system administrator, even the owner of this job cannot release the job using command. If the job could be easily released by the job owner when the system administrator wants to hold the job for the Job Execution Control operation, it causes a problem during operating the Job Execution Control. This is why to prevent the held job from being released by the job owner.

7.19 qstat Queue Status Display Command

Synopsis

```
qstat [-a] [-sys subsystem-number]⁽*¹⁾ [<queuename(s)>...]
```

*1:

[-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The qstat command displays the following queue information:

- Queue name

- Queue status

- Execution priority

- Dispatch priority

- Time lapse limit

- Max execution jobs (multiplicity)

- Max submit jobs (number of jobs that can be submitted)

- Executing jobs (number of jobs in executing state)

- Number of jobs in queuing state

- Number of jobs in holding state

- Default queue

- System multiplicity

- Default priority

Options

EE **-sys subsystem-number**

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

-a

Displays the following queue information:

- Default queue

- System multiplicity

- Default priority

Operand

<queuename(s)>

Specify the queue name whose information that you want to display, using up to 15 bytes.

When you want to display the information for a particular queue, use this operand.

If this operand is omitted, information on all the queues is displayed.

Wildcards cannot be used in this operand.

Queue status

There are two statuses for queues:

ACT

A queue is active. Jobs can be accepted and executed.

INACT

A queue is inactive. Jobs can be accepted but not be executed.

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin
Solaris Linux	/opt/FJSVMJS/usr/bin
HP-UX	/opt/FHPMJS/usr/bin
AIX	/opt/FAIXMJS/usr/bin

Output Example

The following is a display example of resulting from executing the qstat queue command.

```
queuename      status  qprty  dprty  elapsed  maxexec  maxjobs  run  que  hold
queue1@host1   ACT       31     0        0       10      999     0    0    0
queue2@host1   ACT       31     0        0       10      999     0    0    0
queue3@host1   ACT       31     0        0       10      999     0    0    0
queue4@host1   ACT       31     0        0       10      999     0    0    0
queue5@host1   ACT       31     0        0       10      999     0    0    0
online1@host1  ACT       31     0        0       30      999     0    0    0
```

The following is an example of the display when executing the command with the "-a" option.

```
        (11)                              (12)           (13)

 defqueue:queue1@host1    maxexec:30      defprty:31
 queuename                status qprty dprty  elapsed maxexec maxjobs  run que hold
 queue1@host1             ACT      31    0        0      -1      -1      0   0   0

        (1)                      (2)   (3)  (4)      (5)     (6)     (7)  (8) (9) (10)
```

The following lists the items that are displayed in the above example:

(1) Queue name

(2) status [ACT: active, INACT: inactive]

(3) Execution priority

(4) Dispatch priority

(5) Time lapse limit

(6) Max execution jobs (multiplicity) [-1: Not limited]

(7) Max submit jobs (number of jobs that can be submitted) [-1: Not limited]

(8) Executing jobs (number of jobs in executing state)

(9) Number of jobs in queuing state

(10) Number of jobs in holding state

(11) Default queue name [not specify: Not specified]

(12) System multiplicity [-1: Not limited]

(13) Default priority [-1: Not specified]

7.20 qsub Job Submit Command

Synopsis

```
qsub [options] [job_file] [parameters]
```

Description

The qsub command submits a job.

Options

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

-C *comment*

Adds a comment to the job. Comments can be specified using up to 511 bytes or less. If blanks are contained in the comment, enclose the entire comment in double quotation marks. An example is given below.

```
-C "This is a comment."
```

If no comment is specified in this option, the Job Execution Control searches for the environment variable "JC_COMMENT". If this environment variable is found, the string in JC_COMMENT is recognized as the comment.

If the command contains double quotation marks as characters, put a backslash before each quotation mark so that the double quotation marks are treated as characters. By putting a backslash before each quotation mark, the double quotation marks are treated differently from those for enclosing the entire comment. An example is given below.

```
-C "This is a \"client listing\""
```

-ch *client-hostname*

Specifies the client host name in order to identify the host who requested the job execution.

The client host name must be specified using up to 64 bytes.

When the JCL file is submitted, this option will be valid and the JCL host statement that was specified will be invalid.

-cu client-username

Specifies the client user name in order to identify the user who requested the job execution. The client user name must be specified using up to 64 bytes.

When the JCL file is submitted, this option will be valid and the JCL user statement that was specified will be invalid.

Only the system administrator (user belonging to the Administrator group or superuser) can specify this option.

-dp dispatch-priority

[Windows]

Specify the priority of the threads of the program that are to be executed from the job by specifying an integer value between 0 and 4. The greater the value is, the higher the priority is.

When the JCL file is submitted, this option will be valid and the JCL dprty statement that was specified will be invalid.

If this option is omitted, the default value defined in the "Specify default job priority" (also, can be defined using the dfltdprty operand with queue parameter in the Job Execution Control initialization file) in the [Create Queue] or [Edit Queue] window is set. However, if the dprty statement was specified in the JCL file, this dprty statement will be valid.

[UNIX]

Specify the priority of the processes executed from the job by specifying an integer value between -20 and 19. The priority means the CPU allocation priority during the job execution, and the lowest dispatch priority is 19 and the highest is -20.

When the JCL file is submitted, this option will be valid and the JCL dprty statement that was specified will be invalid.

If this option is omitted, the default value defined in the "Specify default job priority" (also, can be set using the dfltdprty operand with queue parameter in the Job Execution Control initialization file) in the [Create Queue] or [Edit Queue] window is set. However, if the dprty statement was specified in the JCL file, this dprty statement will be valid.

-e [*drive-letter:*] [*path-name*] *stderr-file* [Windows]
-e [*path-name*] *stderr-file* [UNIX]

Outputs the standard error output, which is generated during the job execution, as a file named stderr-file to the path name specified at "path-name".

If the path-name is omitted, the current working directory used in submitting the job is assumed as the path name.

If failed to output to the specified path name, it is output to the current working directory.

If this option is omitted, the file name of the standard error output is "job name.ejob number". For example, if the job name is "ACCOUNT", and its job number is "20", the file name of this standard error output is "ACCOUNT.e20".

This option cannot be specified together with the -eo option.

-env *environment variable configuration file name* [-jo *position number*]

Specifies an environment variable for a job when submitting a job with the qsub command.

This cannot be used in the job submission API (Mp_SubmitJob).

Environment variables passed to jobs are first specified in a file in the format described in "7.20.1 Environment Variable Configuration File for the Job Execution Control", then the name of the environment variable configuration file is specified when a job is submitted.

It is also possible to specify more than one pattern of environment variable within one environment variable configuration file, and it is possible to specify which pattern of environment variable is to be used with a number.

  - Up to 32 environment variables can be specified per job.

  - Up to 64 bytes can be specified for the environment variable names for each environment variable.

  - Up to 256 bytes can be specified for the environment variable values for each environment variable.

  - Up to 255 patterns can be specified within each environment variable configuration file.

environment variable configuration file name

Specify when giving a job an environment variable.

If the -env option is specified, it is necessary to specify the environment variable configuration file name immediately after this option. If the environment variable configuration file name is not specified or there is an error in the file name, the job will not be submitted.

The environment variable configuration file stores the environment variable names and environment variable values given to a job. Refer to "7.20.1 Environment Variable Configuration File for the Job Execution Control" for the format.

position number

Specify which of the patterns in the environment variable configuration file specified by the -env option to use. The first pattern will be used if this option is omitted.

-eo

Outputs the standard error output to a standard output file.

Cannot be specified together with the -e option.

-ld *HHMM*

Specifies the cut-off time of the job. The job on which the cut-off time is specified is cancelled with the completion code "0x40000322" when the specified cut-off time is reached.

In "HH", specify any value between 00 and 23 (hour), and in "MM", specify any value between 00 and 59 (minute).

This option cannot be specified with the -lt option or when the [Estimated processing time] box in the [Detail information] sheet of the [Add/Change - Job] window is selected.

-j *job-name*

Specifies the job name. There is no limit to the length of job name but only the first 64 bytes are valid.

The job name is not allowed to contain spaces, user-defined characters, or the following characters:

/ ? * ; , \ " < > | : ( ) ' . $ `

'#' cannot be specified for the first character.

An ampersand (&) cannot be used for the job name for network jobs and distribute execution jobs.

When the JCL file is submitted, this option will be valid and the job name that was specified in the JCL jobstart statement will be invalid.

If this option is omitted, the job file name (excluding the path name and file extension) is used as the job name. However, if the job name was specified in the jobstart statement in the JCL file, this jobstart statement will be valid.

If the job file name is also omitted, "STDIN" is used as the job name.

If "job-name" is prefixed by a number, the Job Execution Control prefixes the job name with "J".

-lt *limittime*

Specifies the time limit of job execution.

The value (in seconds) must be specified in a range from 1 to 99999999.

If this option is omitted, the value defined in the "Specify the time limit of job execution" (limittime operand in the queue parameter) in the [Create Queue] or [Edit Queue] window is set.

This option cannot be specified together with the -ld option.

-no

If this option is specified, job results files (job standard output file or error output file) are not generated.

If this option is specified, -o, -e and -eo options cannot be specified to output the standard output and standard error output to a specified file.

-nt [Windows]

Executes the job in no-trace mode. Normally, the Job Execution Control manages the child process or grandchild process by tracing the execution status of the job process. However, under no-trace mode, the management by tracing cannot be performed. Therefore, if the job started with the Job Execution Control failed with an exception code, the error due to the exception code could be cleared because it does not appears for the child/grandchild process.

However, when the job is executed with this option, the child process and grandchild process are not managed. Therefore, the job is recognized as completed when the parent process has completed, even if the child or grandchild process is left incomplete. Also, when the job is cancelled, only the parent process can be stopped forcibly.

This option is only valid when trace mode is specified as the job startup mode. It is ignored in job object mode.

Furthermore, this option cannot be used for the network jobs or the distributed execution jobs. That is, this job (distributed execution job) is executed with no-trace mode on the job submission source server, but the job is executed in normal trace mode on the destination server where it is submitted by remote operation, because the no-trace mode becomes invalid.

Also, this option cannot be written to a batch file. If specified, it is recognized as an invalid option and this specification is ignored.

-nr

Prohibits re-executing the job.

If a system failure occurred while the job executed with this option is running, it is not re-executed automatically when you reboot the Job Execution Control.

If a system failure occurred on the execution server, network jobs and distributed execution jobs are not re-executed when the Job Execution Control is rebooted, regardless of the status of the jobs and whether this option is specified.

-o [*drive-letter*:] [*path-name*] *stdout-file* [Windows]
-o [*path-name*] *stdout-file* [UNIX]

Outputs the standard output file, which is generated during the job execution, as a file named "stdout-file" to the path name specified at "path-name".

If the path-name is omitted, the current working directory used in submitting the job is assumed as the path name.

If failed to output to the specified path name, it is output to the current working directory.

If this option is omitted, the file name of the standard output is "job name.ejob number". For example, if the job name is "ACCOUNT", and its job number is "20", the name of this standard output file is "ACCOUNT.e20".

-ofe

Specify this option when the job is to be treated as ending with an error (0x40000008 is set as the completion code) if the output file specified by the MJ_OUTPUTFILES environment variable does not exist on the execution server.

If the -ofe option is not specified, 0x40000008 is not set as the job completion code even if the output file specified by the MJ_OUTPUTFILES environment variable does not exist on the execution server. The job completion code on the execution server is used.

Refer to "7.20.2 Specifying the I/O Files to Transfer" for information on the MJ_OUTPUTFILES environment variable.

-p priority

Specifies the priority of jobs in a queue by specifying an integer value between 0 and 63.

0 is the lowest priority value and 63 is the highest.

The priority specified in this option is to determine the order of job execution in a queue, but not decide the execution priority.

When placing a new job in a queue, the Job Execution Control examines the priority of jobs existing in the queue. When a new job is queued, it is placed in a proper position according to its priority value.

If the job with the same priority value as the new job already exists in the queue, the new job is placed after this job in the queue.

When the JCL file is submitted, this option will be valid and the JCL qprty statement that was specified will be invalid.

If this option is omitted, the default value defined in "Specify default job priority" in the [Define Operating Information] window (also, can be defined using the dfltdprty operand with queue parameter in the Job Execution Control initialization file) is set. If this value has not been defined, "-l" will be displayed. However, if the qprty statement was specified in the JCL file, this qprty statement will be valid.

-q *queue-name*

Specifies the name of the queue in which the job is queued.

If a queue name is not specified in this option, the Job Execution Control searches for the environment variable "QSUB_QUEUE". If this environment variable is found, the string in QSUB_QUEUE is recognized as the queue.

If this option is not specified and if the environment variable QSUB_QUEUE is not found, the job is queued in the default queue set by the system administrator.

If this option is not specified and the environment variable QSUB_QUEUE is not found, and if the default queue is not set up, "not specify" is displayed, and the job is not queued.

-R *resource-name*[*,resource-type*]

Specifies the resource name and the exclusivity attribute used in the job. "resource-name" must be specified using up to 64 characters. Specify either of the following for "resource-type".

sh

Shares resources with other jobs

This is the default for resource-type.

ex

Exclusively occupies resources

A comma (,) is used as a separator between resource-name and resource-type.

The following is an example when the resource name "RDB" and its exclusivity attribute is set to "ex".

```
-R RDB,ex
```

When the JCL file is submitted, this option will be valid and the JCL resource statement that was specified will be invalid.

-rh *hostname*

If you specify this option for a particular job, the job is executed on the specified server. However, the standard output and standard error output files of the job are generated on the server from which the job was submitted.

Consider the following when specifying this option:

- If the job is JCL, this option is ignored.

- If submitting the job with this option into the distributed execution queue, submitting is rejected.

The queue to which the job is submitted is determined in the following sequence:

1. The queue having the same name as the execution queue on the server that submitted the job (which is the queue specified at submitting the job, or default queue) can be set.

2. If the condition of the above 1. is not met, the default queue defined in "Specify default queue to submit jobs to" in the [Define Operating Information] window (also, can be defined using the dfltqueue operand with system parameter in the Job Execution Control initialization file) can be set.

3. If the above 2. default queue name has not been specified, "not specify" is displayed, and queuing the job failed.

If the job is requested to another server as the network job by specifying this option, the network job cannot be executed under the following condition a., b., or c.:

a. The network job that meets all of the following conditions cannot be executed:

- Both the job submission server and the job execution server (the server specified in this option) are the Windows-based servers

- The "Execute jobs under the respective job owner's authority" in the [Define Operating Information] window on the execution server is selected.(also, this can be defined using the execasuser parameter in the Job Execution Control initialization file)

- The job has been submitted by the user having the name "root".

- The password for the user who submitted the job (the above user named "root") and the password for the user who was logged in the "Systemwalker MpMjes" service on the execution server do not match.

b. The network job that meets all of the following conditions cannot be executed:

- It is the network job that is submitted from the UNIX-based server and is to be executed on the Windows-based server.

- The "Execute jobs under the respective job owner's authority" in the [Define Operating Information] window on the execution server is selected.(also, this can be defined using the execasuser parameter in the Job Execution Control initialization file)

c. The job queued in the distributed execution queue as the network job cannot be executed.

Still, the user who has submitted the job (the above (a)) is the user name "root" and not necessarily the system administrator.

The user who has submitted the job refers to the following:

- Owner of the project executed from the Jobscheduler

- Client user name specified in the qsub command with the "-cu" option (if the option is omitted, the user who executed the qsub command is defaulted as the client user)

- User who executed the application which uses a job submission API

-rsys execution destination subsystem number

Specifies the subsystem number on the server where the network job is to be executed. The job will be submitted to the subsystem with the specified subsystem number on the execution destination server. The subsystem number must be a number between 0 and 9.

Specify the "-rh" option together with the "-rsys" option when requesting a network job to a subsystem on a host other than the local host. The "-rh" option does not need to be specified when requesting a network job to a subsystem on the local host.

For example, to request that a job file named "job.bat" be passed to subsystem 1 on the request host "Ahost", specify as follows:

```
qsub -rh Ahost -jft -rsys 1 job.bat
```

Jobs for which this option is specified cannot be submitted to distributed execution queues or compatibility mode load distribution queues, or else an error will occur.

-jft

Transfers the job file to the server specified with the -rh option or the distributed job execution server, and executes it on the transfer destination server. If this option is not specified, the job file existing on the server specified with the -rh option or the distributed job execution server is executed.

This option is only valid under the following conditions:

- The job file is not a JCL file.

- The -rh option is specified. Or, the job is submitted into the distributed execution queue.

-x

Makes all the existing environment variables available in the job execution environment when the job is submitted.

Do not specify this option together with the -rh option.

To pass the environment variables to the execution server, use the -env option instead of this option and set the required environment variables.

-z

Prevents a message from displaying when the job is submitted.

If this option is specified, a message reporting the successful submission does not appear even when the job is successfully submitted.

If, however, the job submission failed, an error message is always displayed regardless of this option.

If this option is omitted, a message similar to the following format is displayed when the job is submitted successfully:

```
Job job submitted to queue: queue-name
```

The "job" is displayed in the following format.

For more details on each attribute, see "7.6 qjalter Job Attribute Alter Command".

```
job-name(jobno.hostname)
```

"queue-name" is the name of the queue in which the job is queued.

-r *retrycount*

Performs retries if a line problem occurs while execution request of the network job is being made. The number of retries must be specified with any value between 1 and 20.

By default, the retry interval is set to 300 seconds. If the **Change the connection retry settings for network jobs** option in the **Network** sheet of the **Define Operating Information** window is enabled, the retry interval that you specified will be applied.

Also note that, no retries are performed with this option if a line problem occurs during waiting for completion of the job after its execution request was completed.

Operands

job_file

Specifies the job file to be executed. The files that can be specified are as follow:

- Batch file/shell script

  A file with the extension "BAT" or "CMD" [Windows]

- Execution program

  A file with the extension "EXE" or "COM" [Windows]

- JCL file

  A file with the extension "JCL", which is created in the Job Control Language (JCL) provided by Job Execution Control

- Systemwalker script file

  A file with the extension "SWT", which is created in the Systemwalker script language

parameters

Specifies parameters to be passed to the batch file/shell script, execution program, Systemwalker script, or JCL.

The following is an example if the parameters "0822" and "1000" are passed to the job file "job.bat":

```
qsub job.bat 0822 1000
```

If the string specified for the parameter contains the characters such as (*), (?), ([), or (]) that have special meanings depending on shells, enclose each string in single or double quotes. The following is a sample shell scrip where /usr/bin/ls is set as the job file, and -l, /home/guest/*, and /home/guest/work/??? are set as the parameters [UNIX]:

```
qsub /usr/bin/ls -l '/home/guest/*' '/home/guest/work/???'
```

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM \bin
Solaris Linux	/opt/FJSVMJS/usr/bin
HP-UX	/opt/FHPMJS/usr/bin
AIX	/opt/FAIXMJS/usr/bin

Cautions

- If the job_file operand is omitted, a command is loaded from the standard input file (STDIN) as the job to be executed. If the standard input is performed from a console, press the following keys to end the command entry.

  [Windows]

  ```
  [Ctrl+Z], [Enter]
  ```

  [UNIX]

  ```
  [Ctrl+D]
  ```

- The file specified in the job_file operand could be copied to the spool for the Job Execution Control as a work file when the qsub command is submitted. (This is performed when selecting the "Copy the scripts specified when registering jobs before execution" option in the [Backward compatibility] sheet of the [Define Operating Information] window.) In this case, the Job Execution Control executes the job by referencing the work file, so the changes made to the job_file after queuing cannot be applied to the job execution.

- All the options that can be specified in the command lines can be also written in the first comment block of the batch file. [Windows]

  If you specify the same option entered in the batch file to the command lines, the option specified on the command lines and its arguments take priority.

- Do not specify the qsub command of Job Execution Control or batch files or shell scripts that are calling the qsub command of the Job Execution Control to the job_file operand.

- Do not execute the qsub command as a user that has a space in the user name. Also, do not specify a user name in the -cu option that has a space.

- Do not use spaces in the following directory and file names:

  - Job file name specified in the job_file operand

  - Path and file names specified in the -e option

- Path and file names specified in the -o option

Do not use spaces in the path names in PATH environment variables if these are to be read.

If this command is executed from a Systemwalker Operation Manager job, the behavior of Systemwalker Operation Manager and Systemwalker Centric Manager is not guaranteed. Refer to "Commands that Should Not Be Executed as Jobs" in the *Systemwalker Operation Manager User's Guide* for details.

About Batch File [Windows]

Inputting options in a batch file gets rid of the need for specifying those options when submitting the qsub command.

However, when the "Copy the scripts specified when registering jobs before execution" option is not selected in the [Backward compatibility] sheet of the [Define Operating Information] window, the options cannot be identified.

Algorithms for Identifying Options

1. One line is read from a batch file. If it is the end of the file, the process proceeds to step 8.

2. If it is an empty line, or if the first character encountered is ":", the process proceeds to step 1.

3. If the first word encountered is not "rem", the process proceeds to step 8.

4. If the character encountered after "rem" is not "@$", the process proceeds to step 1.

5. If the characters "@$" is not followed by "-", the process proceeds to step 8.

6. Option parsing is performed.

7. Search is performed for the next "-" character. The process proceeds to step 6 when "-" is encountered. If the end of the line is reached without encountering "-", the process proceeds to step 1.

8. End. All the options that follow are ignored.

Example of Option Statement

```
rem batch file example
rem
rem CLASSA is specified for the queue to which the job is submitted.
rem @$-q CLASSA
rem
rem All the options that follow are ignored.
rem @$
```

7.20.1 Environment Variable Configuration File for the Job Execution Control

File name

Any

Description

This is called by the -env environment variable configuration file name option of the qsub command and specifies environment variables to be given to a job.

File Location

Any

File Format

Multiple patterns of environment variables can be defined within each environment variable configuration file.

Delimit the patterns by putting [JOBn] starting at the first character of the first line. n is a number between 1 and 255.

It is recommended to start from [JOB1] and add in ascending order to reduce the chance of mistakes and to make maintenance easier.

Environment variable definitions are made up of a line with the environment variable name and a line with the environment variable value.

An example is shown below:

```
[JOB1]
ENVNAME1=DATA1                            -+                        -+
ENVVALUE1=/export/200401/data001-+2 lines make the set     |
ENVNAME2=DATA2                                              |Pattern 1
ENVVALUE2=/export/200401/data002                           -+
[JOB2]
ENVNAME1=DATA1
ENVVALUE1=/export/200402/data001
ENVNAME2=DATA2
ENVVALUE2=/export/200402/data002
```

Parameters

[JOB*n*]

Must be included at the beginning of each pattern.

- Start at the first character.

- *n* is a number between 1 and 255. There is no restriction on the order.

- If the pattern number specified by the -jo option of the qsub command cannot be found, the -jo option is omitted, and if [JOB1] does not exist, an error occurs.

- If two patterns have the same number, the environment variable information in the pattern found first when searching from the beginning of the file is used.

- Not all the formats in the environment variable configuration file are checked when the qsub command is submitted. Only the necessary environment variable information with the specified pattern number is loaded. So if the *n* in [JOB*n*] is not a number between 1 and 255, or not a number at all, or omitted, an error will not occur.

ENVNAME*m*= environmental variable name

Specify the name for the environment variable.

- ENVNAME must be used as is, with the *m* being a number, starting from 1 and increasing. The maximum value for *m* is 32.

  The number inserted in place of *m* is searched, starting at 1. The search stops at the point where the next consecutive number cannot be found, and environment variable information found up until that point is effective. For example, if the numbers 1, 2, 3, and 5 are in place of *m*, the information in 1, 2, and 3 are effective, but the information in 5 is not.

- An error occurs if "ENVNAME1" cannot be found.

- Environment variable names and environment variable values are merged by matching the m for ENVVALUE, mentioned below.

- Up to 64 bytes can be specified for environment variable names.

- Any characters may be used for environment variable names.

- Make sure that the environment variable names do not duplicate the environment variables set on the operating system or any other "ENVNAMEm" values.

ENVVALUE*m*= environment variable value

Specify the environment variable values set in the environment variables specified with ENVNAME*m*.

- ENVVALUE must be used as is, with the *m* being a number matching the m in ENVNAME*m*. The maximum value for *m* is 32.

- An error occurs if an ENVVALUE*m* matching the ENVNAME*m* cannot be found.

- Up to 256 bytes can be specified for environment variable values.

- Any characters may be used for environment variable values.

Example

To set the environment variable value /export/200401/data001 in the environment variable DATA1, write as follows:

```
ENVNAME1=DATA1
ENVVALUE1=/export/200401/data001
```

7.20.2 Specifying the I/O Files to Transfer

The I/O files to transfer are specified by including the names of the input and output files in the environment variable of the user executing the qsub command. This section explains how to specify the environment variables to transfer I/O files.

I/O files are transferred by specifying the environment variables described below for the user executing the qsub command.

Up to 32 input files or 32 output files can be specified to be transferred for each job.

Each file name can be a maximum of 64 bytes in length.

Files in the current directory when qsub is executed can be sent and received. Files that require the full path to be specified cannot be transferred.

File transfer cannot be used in the job submission API (Mp_SubmitJob).

Environment variable name

Environment variable	Meaning
MJ_INPUTFILES	Specify input files to be sent to the execution server.
MJ_OUTPUTFILES	Specify output files to be received from the execution server.

Format for specifying environment variables

Specify the name of the file to be transferred in the environment variable. The full path cannot be specified.

A maximum of 32 file names can be specified, each to be delimited with a comma.

Each file name can be a maximum of 64 bytes in length.

Other than characters specifically disallowed by the operating system file system, file names can not include spaces, or the following characters:

? * , \ " < > | : ( ) & =

Specify environment variables as follows:

```
file name 1, file name 2, file name 3,...
```

Input files

Prepare the input file necessary to execute the applications on the execution server.

Input files must be put in the following directory:

```
Current directory at the time of execution of the qsub command
```

Files that do not reside in the above directory cannot be transferred to the execution server.

Files specified in MJ_INPUTFILES will be transferred to the "job working directory" on the execution server.

When the job finishes, input files transferred to the "job working directory" on the execution server will be deleted.

Output files

If the files output by the executed application on the execution server are to be transferred to the schedule server, care must be taken to ensure that the applications output the files to the "job working directory".

Files that do not reside in the above directory cannot be transferred from the execution server to the schedule server.

Files specified in MJ_OUTPUTFILES will be transferred to the current directory of the relevant job when the qsub command is executed. Files will be overwritten if they already exist in this directory.

When the job finishes, output files output to the "job working directory" on the execution server will be deleted.

The -ofe option can specify if 0x40000008 is to be set as an error completion code when the output file specified by the MJ_OUTPUTFILES environment variable does not exist on the execution server. Refer to "7.20 qsub Job Submit Command" for details on the -ofe option.

Job working directory

The job working directory is set in the following environment variable set in the job process.

```
JC_TRANSFILEDIR
```

The above environment variable does not need to be set if I/O file transfer is not to be performed.

7.21 mjdjstat Distributed Execution Status Display Command

Synopsis

```
mjdjstat [-sys subsystem-number](*1) [hostgroup-name...]
```

*1:

[-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The mjdjstat command displays the following distributed job execution information.

- Host group name

- Host name

- Max execution jobs (multiplicity) of host

- Host status

- Executing jobs (number of jobs in executing state) of host

Option

EE **-sys subsystem-number**

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Operand

hostgroup-name

Specifies the name of host group whose status is to be displayed. When specifying multiple names, separate them by a space. No wildcard can be used.

When you want to display the information for a particular host group, use this operand. If hostgroup-name is omitted, the information on all the host groups is displayed.

Host Status

The host can be in one of the following states.

ACT

The host in this state can be a destination server to which the job for distributed execution is submitted. The host will be in any of the following states:

- No job has ever been submitted to this server as the execution destination since the Job Execution Control operation was started.

- A previous job submission was successfully performed.

- The execution server was in DOWN status when the job submission was attempted last time, but approximately 10 minutes have passed already since the DOWN status was detected.

DOWN

The host in this state cannot be a destination server to which the job for distributed execution is submitted. If the destination server of the distributed job execution is detected as DOWN state, this status continues for ten minutes at maximum.

NA

The Job Execution Control function on the server that executed this command is stopping, so cannot get the status of the destination host.

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin
Solaris Linux	/opt/FJSVMJS/usr/bin
HP-UX	/opt/FHPMJS/usr/bin
AIX	/opt/FAIXMJS/usr/bin

Output Example

The following gives an example to display the information of "djgroup1" and "djgroup2" host groups.

```
C:\>mjdjstat djgroup1 djgroup2
hostgroup = djgroup1
  hostname = djhost11
    maxexec = 20  state = ACT    run = 2
  hostname = djhost12
    maxexec = 3   state = ACT    run = 0
  hostname = djhost13
    maxexec = 5   state = ACT    run = 0
hostgroup = djgroup2
  hostname = djhost21
    maxexec = 10  state = ACT    run = 1
  hostname = djhost22
    maxexec = 10  state = ACT    run = 1
```

Cautions

- This command is available only on the Systemwalker Operation Manager server (source server of the distributed job execution).

- The definition status at that time can be referenced even if the Job Execution Control service/daemon on the server that executed this command is stopping. However, the host status is "NA", and the number of executing jobs on the host is "0".

7.22 mjptree Job Process Information Display Command [Windows]

Synopsis

```
mjptree [-sys subsystem-number](*1) job
```

*1:

[-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

The mjptree command displays the processes that are running under a job.

Depending on the privileges that the user executing this command has, the jobs that can be displayed differ as follow:

- Privileges other than the system administrator privileges:

    The jobs submitted by the user executing this command can be displayed.

- System administrator privileges

    All jobs can be displayed.

Option

EE

-sys subsystem-number

In the system with multiple subsystems, this option is used to specify the subsystem that you want to operate. Specify the target subsystem number using a range of 0 to 9. If omitted, the default is subsystem 0.

Operand

job

Specifies the job for which a process list is to be displayed.

Jobs are specified using the following format. See "7.6 qjalter Job Attribute Alter Command" for an explanation of the various parameters that can be specified and other important information.

```
job-name(jobno.hostname)
```

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

| Windows | *<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin |

Output Example

The following example displays process information relating to job1:

```
C:\> mjptree job1
1744 CMD.EXE
  4656 ModuleA.exe
  4304 ModuleB.exe
  5228 ModuleC.exe
  (1)      (2)
```

The following lists the items that are displayed in the above examples:

(1) Process number

(2) File name of executed module

Cautions

- This command can only be used on jobs started in trace mode.

- This command can only display active processes.

- If this command is executed on a job that is not running (a job whose status is "Waiting", "Holding" or "Output held"), an error will result.

- Run this command on an execution server if it is to be used on a network job. An error will occur if it is executed on the server from which the job is submitted.

7.23 mjsmdset Startup Mode Specification Command

Synopsis

```
mjsmdset [-sys subsystem-number] [-c|-w](*1)
```

*1:

[-sys subsystem-number] is an option of Systemwalker Operation Manager EE.

Description

Specify the startup mode used when the Job Execution Control operation or the service/daemon is started next time.

The -w option is assumed to be specified if this option is omitted.

Refer to the *Systemwalker Operation Manager User's Guide* for more information on startup modes.

Options

EE

-sys subsystem-number

Specify the subsystem to be operated in systems that are running multiple subsystems. Specify the number of the subsystem to be operated using a numeral between 0 and 9. Subsystem number 0 is specified if this option is omitted.

-c

Sets the next service/daemon startup mode to the initialization mode.

-w

Sets the next service/daemon startup mode to the continuation mode.

Return Values

Return Value	Meaning
0	The command processing has terminated normally.
Other than 0	The command processing has failed. An error message is output.

Command Location

Windows	*<Systemwalker Operation Manager installation directory>*\MpWalker.JM\bin
Solaris HP-UX AIX Linux	/usr/lib/mjes/bin

Privilege Required

System administrator (user belonging to the Administrators group) privileges are required.

Chapter 8 Action Control Commands [Windows]

This chapter describes the Action Control commands provided by Systemwalker Operation Manager. Accordingly, you must run the commands on servers in which Systemwalker Operation Manager server functions are stalled.

Note

If commands are executed in an environment where the UAC function for Windows Server 2008 or later has been enabled, then a dialog box asking a user's privileges to be promoted may be displayed depending on the privileges of the user that logged in to the operating system, and action may be required.

Refer to "Notes on Executing Commands in Windows [Windows version]" in the *Systemwalker Operation Manager User's Guide* for more information.

8.1 Mp_SpAct Action Stop Command

Synopsis

```
Mp_SpAct -s
```

Description

This command stops audio notification (message reading and WAV file playback).

Return Values

0	Normally stopped.
4	Audio notification is not in running.
8	The Action Control service is not running
32	Parameter error
64	Other error

Privilege Required/Execution Environment

When executing this command on Windows Vista(R), Windows(R) 7, or Windows(R) 8 or later, do it on a Command Prompt opened with system administrator privileges.

8.2 Mp_SpSnd Sound Stop Command

Synopsis

```
Mp_SpSnd
```

Description

The sound stop command stops audio notification (message reading and WAV file playback). This command is executed on the client supporting audio notification.

Return Values

0	Normally stopped
4	Audio notification is not in running.
8	The Action Control service is not running

32	Parameter error
64	Other error

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker\bin

Privilege Required/Execution Environment

- General user command privileges are required.

- When executing this command on Windows Vista(R), Windows(R) 7, or Windows(R) 8 or later, do it on a Command Prompt opened with system administrator privileges.

Example

```
Mp_SpSnd
```

Chapter 9 Task Link Commands [Windows]

This chapter describes the Task Link commands [Windows] provided by Systemwalker Operation Manager. It is organized into the following sections. Note that the commands must be executed on the server where the Systemwalker Operation Manager server function is installed.

Note
....

If commands are executed in an environment where the UAC function for Windows Server 2008 or later has been enabled, then a dialog box asking a user's privileges to be promoted may be displayed depending on the privileges of the user that logged in to the operating system, and action may be required.

Refer to "Notes on Executing Commands in Windows [Windows version]" in the *Systemwalker Operation Manager User's Guide* for more information.
....

9.1 Server Task Link Commands

The Server Task Link commands support operation with data and job linkage, streamlining server business system configurations.

Information
....

The Server Task Link commands can create a batch file and register it.
....

9.1.1 EXECASVR.EXE Application Startup Command

Synopsis

```
EXECASVR.EXE [/UN:UserName] [/LG:LoginGroup]
                [/RC:nn /RI:nnnn] /AE:ExecuteAppli
              AppliParam
```

Description

The EXECASVR.EXE command activates any application installed on a server. The activated application runs in the background and is not displayed on-screen.

This command will end without waiting for the application to complete.

Parameters

/UN:*UserName*

Specifies the right to be used to start an application by specifying the user name.

And the user name specified to *UserName* must be set in the password management book in advance. If *UserName* contains a space character, enclose it in double quotation marks.

To specify *UserName*, use either one of the following formats: "domain name\user name" or "server name\user name". The domain name or server name must be no longer than 15 alphanumeric characters, and the user name must be no longer than 20 alphanumeric characters.

If this parameter is omitted, the application is started with the system rights.

/LG:*LoginGroup*

Specifies a login definition name used to manage the password management book.

Specify *LoginGroup* within 256 characters, and enclose it in double quotation marks if a space character is contained.

If the /UN parameter is specified, be sure to specify this parameter.

/RC:*nn*

Specifies the retry count in case an error occurs in application startup.

Specify an integer value between 0 and 99 for *nn*. Setting any other value results in an error. If omitted, no retry is performed.

/RI:*nnnn*

Specifies the retry interval in case an error occurs.

Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Setting any other value results in an error. The default value is 0.

This parameter is valid when 1 or a greater value is specified to *nn* in the /RC parameter.

/AE:*ExecuteAppli*

Specifies an application that you want to activate using a local drive path. Specify the application name in *ExecuteAppli*. If the application name contains a space character, enclose it in double quotation marks. The total number of characters in *ExecuteAppli* including *AppliParam* must be no longer than 511 characters.

AppliParam

Specifies parameters for the application specified in *ExecuteAppli*. If more than one parameter is specified, put a space character between parameters. If a space is contained in any parameter, enclose it in double quotation marks.

Return Values

Return value	Meaning
0	Completed
Value other than 0	Abended

Command Location

Systemwalker Operation Manager installation directory\MpWalker.JM\bin

Example

In the following example, the application "c:\Program files\app\App.exe" is started using the user rights of "foo\User1". The user's password used in server logon is obtained from the login definition name "grp1" in the password management book.

```
EXECASVR.EXE /UN:foo\Usr1 /LG:grp1 /AE:"c:\Program files\app\App.exe"
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

The contents to be output are shown below.

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Application] Name of the application activated
[Retry count] Number of retries performed
[Error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

If an error occurs before starting the application, a message is output in the following format:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Command error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- The Task Link service must be already working on the server side when you use this command.

- Prior to running this command, you need to configure the password management book by using the password management command. See "9.3.1 LGMANAGE.EXE Password Management Command" for further details.

- This command does not monitor the application process that the command activated. To review the process results, the application itself needs to have the feature to report a completion code.

- The application started by this command always operates behind the screen, so applications launch on a specific screen cannot be activated by this command. Also, this command cannot start an application that requires user registry.

- When specifying the /UN:UserName and /LG:LoginGroup parameters, you need to set "Logon as a service" beforehand for the user specified to *UserName*. To do it on the Windows Server 2008 or later based computer, use the "Local Security Policy Setting".

- The application started by this command cannot be canceled, even if you cancel this command.

- If this command is canceled, it is carried out at the point when the application has completely started. Therefore, the command takes a longer time to quit as the application takes more to start.

- This command cannot start the application that requires keyboard entry.

9.1.2 FCOPYSVR.EXE File Transfer Command

Synopsis

```
FCOPYSVR.EXE /S:SourceName [/SU:sourceUserName]
             /D:DestName [/DU:DestUserName]
             [/LG:LoginGroup] [/A | /B]
             [/RC:nn /RI:nnnn]
```

Description

The FCOPYSVR.EXE command transfers files between servers.

It allows you to perform file transfer by using the Windows shared directories (Windows network share method) or file transfers through FTP services (FTP method). If you wish only to perform file transfer, you can transfer files even to the destination server without having Systemwalker Operation Manager installed.

Parameters

/S:*SourceName*

Specify a path name of the transfer source file, including the file name. You can use a wildcard for this file name.

Use one of the following methods to specify *SourceName* in the URL format. The parts of path name must be no longer than 255 characters. If the path name contains a space character, enclose it in double quotation marks.

A local drive path:

Specify a path including a drive name.

Specify the path name of desired file directly after "file://" string.

Example:

file://C:\DATABASE\DATABASE.DB

UNC method (Windows network share method):

Specify a path in the format "\\computer name\share name". For a computer name, specify a Windows-based computer name within 15 alphanumeric characters. Specify a share name within 80 bytes.

Specify the path name of desired file directly after "file://" string.

Example:

file://\\foo\SHARE\DATABASE\DATABASE.DB

A path on the FTP server (FTP method):

Specify a path name to the FTP server.

The path name can be specified with a full path name or a relative path name from the current directory in logging in with FTP.

Specify the path name in the format of "server name/path name of desired file" directly after "ftp://" string.

Example) To specify using the full path name:

ftp://foo.bar.com//Usr/DataBase/DataBase.DB

Example) To specify using the relative path name:

ftp://foo.bar.com/DataBase/DataBase.DB

/SU:*SourceUserName*

Specifies a user name to be used to access the file specified in *SourceName*.

The user name to be specified to *SourceUserName* must be set in the password management book in advance. If a space character is contained in *SourceUserName*, enclose it in double quotation marks.

If *SourceName* is specified using a local drive path, this parameter's specification is ignored.

However, if the UNC method or a path on the FTP server is specified, be sure to specify this parameter.

If the UNC method is used, specify *SourceUserName* in either "domain name\user name" or "server name\user name" format. In this case, the domain name or server name must be no longer than 15 alphanumeric characters, and the user name must be no longer than 20 characters.

When specifying a path on the FTP server, "domain name\user name" cannot be used. Specify the user registered as "user name@domain name" in the password management book or the local account.

If a path on the FTP server is used, *SourceUserName* must be specified using up to 20 alphanumeric characters.

/D:*DestName*

Specify a path name of the destination file, including the file name. You can use a wildcard for this file name.

You can use the same specification methods as those described in the above "/S:SourceName" parameter. See the "/S:SourceName" parameter.

If the file name that already exists is specified, it will be overwritten.

Note that the same file path name as the one (*SourceName*) specified with /S should not be specified.

/DU:*DestUserName*

Specifies a user name to be used to access the file specified in *DestName*.

To specify the user name in *DestUserName*, it must be set in the password management book in advance. If *DestUserName* contains a space character, enclose the name in double quotation marks.

If a local drive path is set to *DestName*, this parameter's specification is ignored.

However, if *DestName* is specified using the UNC method or path on the FTP server, be sure to specify this parameter.

If the UNC method is used, specify *DestUserName* in either "domain name\user name" or "server name\user name" format. The domain name or server name must be no longer than 15 alphanumeric characters, and the user name must be no longer than 20 alphanumeric characters.

When specifying a path on the FTP server, "domain name\user name" cannot be used. Specify the user registered as "user name@domain name" in the password management book or the local account.

If a path on the FTP server is specified, *DestUserName* must be specified using up to 20 alphanumeric characters.

/LG:*LoginGroup*

Specifies a login definition name used to manage the password management book.

LoginGroup must be no longer than 256 characters, and enclose it in double quotation marks if a space character is contained.

If the /SU or /DU parameter is specified, be sure to specify this parameter.

/A

Transfers a file in the ASCII text mode.

This parameter is valid only when you transfer files on the FTP server (FTP method).

/B

Transfers a file in the binary mode.

This parameter is valid only when you transfer files on the FTP server (FTP method).

If neither /A nor /B is specified, the file is transferred in the binary mode.

/RC:*nn*

Specifies the retry count in case an error occurs in file transfer.

Specify an integer value between 0 and 99 for *nn*. Setting any other value results in an error. If omitted, no retry is performed.

If a wildcard is used to specify the transfer source file or transfer destination file, retry is performed only for the failed file.

/RI:*nnnn*

Specifies the retry interval in case an error occurs.

Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Setting any other value results in an error. The default value is 0.

This parameter is valid if 1 or a greater value is specified to *nn* in the /RC parameter.

Return Values

Return value	Meaning
0	Completed
Value other than 0	Abended

Command Location

Systemwalker Operation Manager installation directory\MpWalker.JM\bin

Example

In the following example, the local file "c:\database\master\database.db" on the server is transferred to the local file "d:\database\database.db". If the transfer fails, retry is performed up to three times at an interval of 60 seconds.

```
FCOPYSVR.EXE /S:file://c:\database\master\database.db
              /D:file://d:\database\database.db /RC:3 /RI:60
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

The contents to be output are shown below.

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Server host/User name] Server host name/user name (*)
[Retry count] Number of retries performed (*)
[Error] Error message (*)
[File transfer] Transfer source file name Transfer destination file name (**)
[Retry count] Number of retries performed (**)
[Error] Error message (**)
[Process end time] MM-DD-YYYY HH:MM:SS
```

*:

This information is always output if the /SU or /DU parameter is specified.

**:

If a wildcard is used, the information of each file is output by file basis.

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the S*ystemwalker Operation Manager Message Guide*.

If an error occurs before performing file transfer, a message is output in the following format:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Command error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- The Task Link service must be working already on the server side when you use this command.

- Prior to running this command, you need to configure the password management book by using the password management command. See "9.3.1 LGMANAGE.EXE Password Management Command" or further details.

- If you use this command by specifying the UNC method, you must use the Windows network share method. Therefore, be sure to use the same information (user name/password) when you try to connect from/to the same server simultaneously, using the Task Link command or other application (such as Explorer).

  If the command is executed using the different authentication information by different users, it results in a connection error.

- When transferring files by specifying a path on the FTP server, it can be performed only when the path on the FTP server is specified for either the transfer source or the destination. The valid combinations of file path format at source and destination are listed below. Select one of the combinations from the list.

	Specifying source file	Specifying destination file
1	Local path method	Local path method
2	Local path method	UNC method
3	Local path method	FTP server path method
4	UNC method	Local path method
5	UNC method	UNC method
6	UNC method	FTP server path method

	Specifying source file	**Specifying destination file**
7	FTP server path method	Local path method
8	FTP server path method	UNC method

- The FTP servers available for file transfer by specifying a path on it are limited to those on which the execution results of "DIR" command are displayed in the following format (with permission, number of links, owner, group, size, date, time, and file name are listed in this order, in alphanumerical notation).

```
-rw-rw-r--  1 xxxxx  xxxxxx     605877 Dec 19 18:44 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxxx     560661 Dec 19 18:59 xxxxx.xxx
drwx------  4 xxxxx  xxxxx          96 Jan 20  1998 xxxxx
-rw-rw-r--  1 xxxxx  xxxxx        6042 Jan 25 13:38 xxxxxxxxxxx
-rw-rw-r--  1 xxxxx  xxxxx        5433 Feb 3  17:11 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxx        2155 Feb 3  17:35 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxx        3090 Feb 1  20:13 xxxxx.xxx
```

To verify whether or not file transfer can be performed by using the path on the FTP server, use the following method.

[Windows]

- Execute the ftp command at a command prompt.

[UNIX]

- Execute the ftp command on a server console or a telnet-connected console.

Follow the steps below for checking on Windows.

```
C:\ ftp XXXXX                        <-ftp command(XXXXX:ftp server name)
Connected to XXXXX.
220 XXXXX FTP server (UNIX(r) System V Release 4.0) ready.
Name (XXXXX:YYYYY): YYYYY      <-User name setting for the ftp server(YYYYY:user name)
331 Password required for YYYYY.
Password:                            <-Password entry
230 User YYYYY logged in.
ftp> dir                             <-dir command entry
200 PORT command successful.

The dir result is output.
```

- If this command is canceled, it is carried out at the point when one file has been completely transferred. Therefore, the command takes a longer time to quit if it is cancelled while a large file is being transferred.

- When using a wildcard for the file name specified with the /S parameter, the following considerations are needed:

The number of files to be processed at a time should be 500 or less, including the files to be processed simultaneously by other Server Task Link commands.

- The size per file should be 2GB or less.

9.1.3 FCTRLSVR.EXE File Operation Command

Synopsis

```
FCTRLSVR.EXE {/DELETE | /RENAME | /MKDIR | /RMDIR} /F:FilePathName
             [/FN:FileName] [/UN:UserName]
             [/LG:LoginGroup] [/RC:nn /RI:nnnn]
```

Description

The FCTRLSVR.EXE command operates files or directories on a server. For file operations, the command deletes or renames files. For directory operations, the command creates or deletes directories.

File operations with this command are performed by using the Windows shared directories (Windows network share method) or file operation through FTP services (FTP method).

If you wish only to operate remote files, you can do it even on the remote server without having Systemwalker Operation Manager installed.

Parameters

/DELETE

Deletes a file.

/RENAME

Renames a file.

/MKDIR

Creates a directory.

/RMDIR

Deletes a directory.

/F:*FilePathName*

Specifies a path name of the file or directory to be operated. You can use a wildcard for this file name only when using with the /DELETE or /RENAME parameter.

Use one of the following methods to specify *FilePathName* in the URL format. *FilePathName* must be no longer than 255 characters, and enclose it in double quotation marks if a space character is contained.

A local drive path:

Specify a path including the drive name.

Specify the path name of desired file directly after "file://" string.

Example:

file://C:\DATABASE\DATABASE.DB

UNC method (Windows network share method):

Specify a path in the format "\\computer name\share name". For a computer name, specify a Windows-based computer name within 15 alphanumeric characters. Specify a share name within 80 bytes.

Specify the path name of desired file directly after "file://" string.

Example:

file://\\foo\SHARE\DATABASE\DATABASE.DB

A path on the FTP server (FTP method):

Specify a path name to the FTP server.

The path name can be specified with a full path name or relative path name from the current directory in logging in with FTP.

Specify the path name in the format of "server name/path name of desired file" directly after "ftp://" string.

Example) To specify using the full path name:

ftp://foo.bar.com//Usr/DataBase/DataBase.DB

Example) To specify using the relative path name:

ftp://foo.bar.com/DataBase/DataBase.DB

/FN:*FileName*

Specifies the new file name you change to. You can use a wildcard for this parameter. Specify a file name in *FileName* within 255 characters, and enclose it in double quotation marks if a space character is contained.

This parameter is valid only when the /RENAME parameter is specified.

/UN:*UserName*

Specifies a user name to be used to access the file or directory specified with *FilePathName*.

The user name in *UserName* must be set in the password management book in advance. If a space character is contained in the user name, enclose it in double quotation marks.

If *FilePathName* is specified using a local drive path, this parameter's specification is ignored.

However, if the UNC method or path on the FTP server is specified, be sure to specify this parameter.

If the UNC method is used, specify *UserName* in either "domain name\user name" or "server name\user name" format. The domain name or server name must be no longer than 15 alphanumeric characters, and the user name must be no longer than 20 characters.

If a path on the FTP server is specified, *UserName* must be specified using up to 20 alphanumeric characters.

/LG:*LoginGroup*

Specifies a login definition name used to manage the password management book. *LoginGroup* must be no longer than 256 characters, and enclose it in double quotation marks if a space character is contained.

If the /UN parameter is specified, be sure to specify this parameter.

/RC:*nn*

Specifies the retry count in case an error occurs in file operation.

Specify an integer value between 0 and 99 for *nn*. Setting any other value results in an error. If omitted, no retry is performed.

/RI:*nnnn*

Specifies the retry interval in case an error occurs.

Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Setting any other value results in an error. The default value is 0.

This parameter is valid if 1 or a greater value is specified to *nn* in the /RC parameter.

Return Values

Return value	Meaning
0	Completed
Value other than 0	Abended

Command Location

Systemwalker Operation Manager installation directory\MpWalker.JM\bin

Example

In this example, the local file "c:\database\master\database.db" on the server is renamed to "database_backup.db". If renaming fails, retry is performed up to three times at an interval of 60 seconds.

```
FCTRLSVR.EXE /RENAME /F:file://c:\database\master\database.db
                /FN:database_backup.db /RC:3 /RI:60
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

The contents to be output are shown below.

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Server host/User name] Server host name/user name (*)
[Retry count] Number of retries performed (*)
[Error] Error message (*)
[File operation type] File operations/Directory name New file name (**)(***)
[Retry count] Number of retries performed (**)
[Error] Error message (**)
[Process end time] MM-DD-YYYY HH:MM:SS
```

*:

This information is output if the /UN parameter is specified.

**:

If a wildcard is used, the information of each file is output by file basis.

***:

The new file name you changed to is output only when the /RENAME parameter is specified.

In [Error], "None" is output when the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

In [File operation type], any of "Delete file", "Rename file", "Create directory", or "Delete directory" is output.

If an error occurs before performing file operations, a message is output in the following format:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Command error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- The Task Link service must be already working on the server side when you use this command.

- Prior to running this command, you need to configure the password management book by using the password management command. See "9.3.1 LGMANAGE.EXE Password Management Command" or further details.

- If you use this command by specifying the UNC method, you must use the Windows network share method. Therefore, be sure to use the same information (user name/password) when you try to connect from/to the same server simultaneously, using the Task Link command or other application (such as Explorer).

  If the command is executed using the different authentication information by different users, it results in a connection error.

- The FTP servers available for file operations by specifying a path on it are limited to those on which the execution results of "DIR" command are displayed in the following format (with permission, number of links, owner, group, size, date, time, and file name are listed in this order, in alphanumerical notation).

```
-rw-rw-r--  1 xxxxx  xxxxxx     605877 Dec 19 18:44 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxxx     560661 Dec 19 18:59 xxxxx.xxx
drwx------  4 xxxxx  xxxxx          96 Jan 20  1998 xxxxx
-rw-rw-r--  1 xxxxx  xxxxx        6042 Jan 25 13:38 xxxxxxxxxxx
-rw-rw-r--  1 xxxxx  xxxxx        5433 Feb 3  17:11 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxx        2155 Feb 3  17:35 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxx        3090 Feb 1  20:13 xxxxx.xxx
```

See "9.1.2 FCOPYSVR.EXE File Transfer Command" or how to verify whether or not the path on the FTP server can be specified.

- When using a wildcard for the file name specified to the /F parameter, note the following:

    The number of files to be processed at a time should be 500 or less, including the files to be processed by other Server Task Link commands.

- The size per file should be 2GB or less.

9.1.4 FPRSSVR.EXE File Compression/Expansion Command

Synopsis

```
FPRSSVR.EXE {/P | /E} /S:SourceName /D:DestName
            [/RC:nn /RI:nnnn]
```

Description

The FPRSSVR.EXE command compresses or expands local files on a server (including directories on the NFS server).

This command allows you to compress files on the transfer source and expand them on the transfer destination. This capability reduces the file transfer time as well as the load on the network.

Parameters

/P

Compresses files.

/E

Expands files.

/S:*SourceName*

Specifies a path name of the compression source file/expansion source file using a local drive path. The local path must be specified with an absolute path, including a drive name, and it should look similar to "C:\DATABASE\DATABASE.DB". A path on the FTP server cannot be specified.

You can use a wildcard for the file name of the original files for compression.

Specify *SourceName* within 255 characters. If *SourceName* contains a space character, enclose it in double quotation marks.

/D:*DestName*

Specifies a path name of destination file for compression/expansion using a local path. A path on the FTP server cannot be specified.

You can use a wildcard for the file name of the destination file for expansion.

Specify *DestName* within 255 characters. If *DestName* contains a space character, enclose it in double quotation marks.

If the file name that already exists is specified, it will be overwritten. Note that the same file path name as the one (*SourceName*) specified with /S should not be specified.

/RC:*nn*

Specifies the retry count in case an error occurs in file compression/expansion. Specify an integer value between 0 and 99 for *nn*. Setting any other value results in an error. If omitted, no retry is performed.

If a wildcard is used to specify the compression source files or the expansion destination files, retry is performed only for the failed file.

/RI:*nnnn*

Specifies the retry interval in case an error occurs. Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Setting any other value results in an error. The default value is 0.

This parameter is valid if 1 or a greater value is specified to *nn* in the /RC parameter.

Return Values

Return value	Meaning
0	Completed
Value other than 0	Abended

Command Location

Systemwalker Operation Manager installation directory\MpWalker.JM\bin

Example

In this example, the local file "d:\database\database.prs" on the server is expanded to "c:\database\copy*.*". If expansion fails, retry is performed up to three times at an interval of 60 seconds.

```
FPRSSVR.EXE /E /S:d:\database\database.prs /D:c:\database\copy\*.*
            /RC:3 /RI:60
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

The contents to be output are shown below:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[File compression/Expansion] Name of compression/expansion source files Name of
compression/expansion destination files (*)
[Retry count] Number of retries performed (*)
[Error] Error message (*)
[Process end time] MM-DD-YYYY HH:MM:SS
```

*:

If a wildcard is used, the information of each file is output by file basis.

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

If an error occurs before performing compression/expansion process, a message is output in the following format:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Command error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- The Task Link service must be already working on the server side when you use this command.

- If you compress files using the fprssvr command with the /w parameter on the UNIX-based Systemwalker Operation Manager server and expand them using this command on Windows, they will be extracted as follow:

  - If write permission for the files is not assigned on the compression source, the files are marked with the Read Only attribute on Windows.

  - If write permission for the file is set for any of owner, group, or other on the compression source, the files are marked with the Read/Write attribute on Windows.

- If you compress files using this command, and then expand them on the UNIX-based Systemwalker Operation Manager server, the feature of files after extraction is as follow:

File attribute on Windows before compression	Feature of file extracted on UNIX
Read-only	- The files' owner/group information will be changed to the information of the user issuing the command. - The files will have mode 0444 (readable by everyone). (*1)
Other than Read-only	- The files' owner/group information will be changed to the information of the user issuing the command. - The files will have mode 0666 (readable/writable by everyone). (*2)

*1:

When the files are readable by everyone, it indicates the permission for the file displayed as "-r--r--r--" by using the ls -l command.

*2:

When the files are readable/writable by everyone, it indicates the permission for the file displayed as "-rw-rw-rw-" by using the ls -l command .

- You need the read permissions for the files that you want to compress. However, the superuser can do it without the read permission.

- If this command is canceled, it is carried out at the point when one file has been completely compressed or expanded. Therefore, the command takes a longer time to quit if it is cancelled while a large file is being compressed/expanded.

- If you use a wildcard for the file name specified in this command with the /S parameter, note the following:

The number of files to be processed at a time should be 500 or less, including the files to be processed by other Server Task Link commands.

- A temporary file is created in the following location on the server when executing this command.

  - A directory set in the environment variable TEMP

Note that the compression/expansion fails if there is not enough disk space in the TEMP. The amount of space necessary for the temporary file is determined by adding the files sizes (before compressing or after expanding) together.

- The size per file should be 2GB or less.

9.2 Client Task Link Commands

The Client Task Link commands support job linkage with clients.

Information

....

The Client Task Link commands can create a batch file and register it.

....

9.2.1 PONCLI.EXE Power On Command

Synopsis

```
PONCLI.EXE HostName
          [/IP:IPAddress /MAC:MACAddress /SNM:SubnetMask]
          [/WT:WaitTime] [/RC:nn /RI: nnnn]
```

Description

The PONCLI.EXE command powers on the client that supports a Wakeup on LAN feature, which enables a remote client to be booted automatically.

This command also allows you to power on the client on another subnet if in a TCP/IP environment.

This command is not supported in IPv6 environments. Use this command in IPv4 environments or in environments where dual stack is supported.

Parameters

HostName

Specifies a host name of the client to be powered on.

Specify the host name in *HostName* within 15 alphanumeric characters. If the host name contains a space character, enclose it in double quotation marks.

/IP:*IPAddress*

Specifies an IP address of the client to be powered on. The IP address is specified in a decimal string, which is something similar to "133.161.222.110".

If omitted, the command reads the host information definition file to retrieve the IP address of the target client.

/MAC:*MACAddress*

Specifies a MAC address of the client to be powered on. The MAC address is specified in a hexadecimal string, which is similar to "00-00-0E-9D-2C-38". If omitted, the command reads the host information definition file to retrieve the MAC address of the target client.

/SNM:*SubnetMask*

Specifies a subnet mask of the client to be powered on. The subnet mask is specified in a decimal string, which is something similar to "255.255.255.0".

If omitted, the command reads the host information definition file to retrieve the subnet mask of the target client.

/WT:*WaitTime*

Specifies the wait time to ensure the client is powered on. Specify an integer value between 1 and 999 for *WaitTime* in seconds.

If omitted, waiting is not performed.

/RC:*nn*

Specifies the retry count in case an error occurs in powering on the client.

Specify an integer value between 0 and 99 for *nn*. Setting any other value results in an error. If omitted, no retry is performed.

/RI:*nnnn*

Specifies the retry interval in case an error occurs. This parameter is valid if 1 or a greater value is specified to *nn* in the /RC parameter.

Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Setting any other value results in an error. The default value is 0.

Return Values

Return value	Meaning
0	Completed
Value other than 0	Abended

Command Location

Systemwalker Operation Manager installation directory\MpWalker.JM\bin

Example

In this example, the client "foo" is powered on under the following condition.

IP address:

133.161.222.110

MAC address:

00-00-0E-9D-2C-38

Subnet mask:

255.255.255.0

```
PONCLI.EXE foo /IP:133.161.222.110 /MAC:00-00-0E-9D-2C-38 /SNM:255.255.255.0
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

The contents to be output are shown below:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Host name] HostName
[MAC address] MAC address
[IP address] IP address
[Subnet mask] Subnet mask
[Retry count] Number of retries performed
[Error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

If an error occurs before power on, a message is output in the following format:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Command error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- To execute this command by using the host information definition file, define the client information in the host information definition file in advance. For details, see the S*ystemwalker Operation Manager Installation Guide*.

- The client that can be powered on using this command must be a computer supporting a Wakeup on LAN feature. To turn on the power in a client on another subnet, you need to configure the network hardware (e.g., router) so that the broadcast packet sent from the server can be received by the network of the target computer. For details about the settings method, see the documentation for your networking hardware.

  To check whether the broadcast packet can arrive at the destination or not, you can use the ping command by submitting it on a command prompt on the Systemwalker Operation Manager server side. And if it is not responded, the broadcast packet cannot reach the destination.

  For example, to ping the remote network host of having the IP address "10.90.100.100" and subnet mask "255.255.255.0", use the "ping 10.90.100.255" command.

- In response to connection of a power source, the Wakeup on LAN packet is passed to the network host to which the target client belongs, and immediately after that, this command is returned as completed successfully. To wait to ensure that the computer system is booted after power on, specify the /WT parameter.

9.2.2 POFFCLI.EXE Power Off Command

Synopsis

```
POFFCLI.EXE HostName [/SD:ShutdownMode] [/EF]
            [/RC:nn /RI:nnnn]
```

Description

The POFFCLI.EXE command powers off the client that supports the Windows power-off mechanism. It enables a remote client to be turned off automatically.

Parameters

HostName

Specifies a host name of the client to be powered off.

Specify the host name in *HostName* within 15 alphanumeric characters. If the host name contains a space character, enclose it in double quotation marks.

/SD:*ShutdownMode*

Specifies the mode to turn off the power. Select one of the following:

poff:

Turns off the system power.

reboot:

Reboots the system.

down:

Shuts down the system.

If omitted, the system is turned off as the poff mode is defaulted.

/EF

Specifies forced power off.

Specifying this parameter forces the running applications to be terminated. At this time, the data used by those applications is not saved.

The active desktop on the client should be disabled. Otherwise, the active desktop recovery function may be started automatically the next time you start the OS.

/RC:*nn*

Specifies the retry count in case an error occurs in powering off clients.

Specify an integer value between 0 and 99 for *nn*. Setting any other value results in an error. If omitted, no retry is performed.

/RI:*nnnn*

Specifies the retry interval in case an error occurs.

Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Setting any other value results in an error. The default value is 0.

This parameter is valid if 1 or a greater value is specified to *nn* in the /RC parameter.

Return Values

Return value	Meaning
0	Completed
Value other than 0	Abended

Command Location

Systemwalker Operation Manager installation directory\MpWalker.JM\bin

Example

The following example performs power off for the client "foo".

```
POFFCLI.EXE foo /SD:POFF
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

The contents to be output are shown below.

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Client host name] HostName
[Retry count] Number of retries performed
[Error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

If an error occurs before power off, a message is output in the following format:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Command error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- The Client Task Link service must be already working on the client side when you use this command. At this time, the Client Task Link service must be started by the user having privileges to power off clients.

- The client that can be powered off using this command must be a computer supporting the Windows power-off mechanism.

- If a running application is not terminated in response to a termination request sent from Windows, it may not be possible to power off the client.

- If the /EF parameter is specified, applications that do not respond to a Windows' termination request are forcefully terminated by Windows.

- The specification by the /RC parameter or /RI parameter becomes valid only when the user logged on to the target client does not have the shutdown privilege in Windows operating systems.

9.2.3 EXECACLI.EXE Application Startup Command

Synopsis

```
EXECACLI.EXE HostName [/AM:AppMode] [/WE:WaitEnd]
             [/RC:nn /RI:nnnn] /AE:ExecuteAppli
             [AppliParam]
```

Description

The EXECACLI.EXE command activates any application installed on a client. This command can wait for the application to complete.

Parameters

HostName

Specifies a host name of the client where the application will be activated.

HostName must be no longer than 15 alphanumeric characters. If *HostName* contains a space character, enclose it in double quotation marks.

/AM:*AppMode*

Specifies the mode to start the application. Select one of the following:

NORMAL:

Activates the application in a normal window.

ICON:

Activates the application in iconic state.

If this parameter is omitted, it defaults to the NORMAL mode, activating in a normal window.

/WE:*WaitEnd*

Specifies the time to wait for the activated application to complete. Specify an integer value between 1 and 9600 (in seconds) to *WaitEnd*. Specifying any other value results in an error.

If omitted, waiting for the application to complete is not performed.

/RC:*nn*

Specifies the retry count in case an error occurs in application startup.

Specify an integer value between 0 and 99 for *nn*. Specifying any other value results in an error. If omitted, no retry is performed.

/RI:*nnnn*

Specifies the retry interval in case an error occurs. This parameter is valid if 1 or a greater value is specified to *nn* in the /RC parameter.

Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Specifying any other value results in an error. The default value is 0.

/AE:*ExecuteAppli*

Specifies an application name that you want to activate using the local drive path with no longer than 511 characters. If the name of the target application or any parameter to be passed to the application contains a space character, enclose each in double quotation marks.

AppliParam

Specifies the parameter(s) for the application specified with *ExecuteAppli*. If more than one parameter is specified, put a space between parameters. If a space character is contained in any parameter, enclose it in double quotation marks.

Return Values

Return value	Meaning
0	Completed

Return value	Meaning
Value other than 0	Abended (or an application started with the /WE parameter has terminated with a return value other than 0.)

Command Location

Systemwalker Operation Manager installation directory\MpWalker.JM\bin

Example

In this example, the application "C:\Program files\app\App.exe" on the client "foo" is activated in a normal window, and then waits for this application to complete for 60 seconds.

```
EXECACLI.EXE foo /AM:NORMAL /WE:60 /AE:"C:\Program files\app\App.exe"
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

The contents to be output are shown below:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Client host name] HostName
[Application] Name of the application that was activated
[Retry count] Number of retries performed
[Error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

If an error occurs before starting the application, a message is output in the following format:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Command error] Error message
 [Process end time] MM-DD-YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- The Client Task Link service must be already working on the client side when you use this command.

- This command waits for the application to complete only when the /WE parameter is specified.

- The application started by this command cannot be canceled, even if you cancel this command.

- If this command is canceled, it is carried out at the point when the application has been completely started. Therefore, the command takes a longer time to quit as the application takes more to start.

9.2.4 FCOPYCLI.EXE File Transfer Command

Synopsis

```
FCOPYCLI.EXE HostName /S:SourceName [/SU:SourceUserName]
             /D:DestName [/DU:DestUserName]
            [/LG:LoginGroup] [/A | /B] [/RC:nn /RI:nnnn]
```

Description

The FCOPYCLI.EXE command enables file transfer by clients. This command allows to transfer files on client to server or files on server to client. It performs file transfers using the Windows shared directories (Windows network share method) or file transfers through FTP services (FTP method).

If you wish only to perform file transfer, you can transfer files even to the destination server without having Systemwalker Operation Manager installed.

Parameters

HostName

Specifies a host name of the client which is requested to transfer a file. Specify the host name in *HostName* within 15 alphanumeric characters.

If *HostName* contains a space character, enclose it in double quotation marks.

/S:*SourceName*

Specify a path name of the transfer source file, including the file name. You can use a wildcard for this file name.

Use one of the following methods to specify *SourceName* in the URL format. The parts of path name must be no longer than 255 characters. If the path name contains a space character, enclose it in double quotation marks.

A local drive path:

Specify a path including a drive name.

Specify the path name of desired file directly after "file://" string.

Example:

file://C:\DATABASE\DATABASE.DB

UNC method (Windows network share method):

Specify a path in the format of "\\computer name\share name". For "computer name", specify a Windows-based computer name within 15 alphanumeric characters. Specify a share name within 80 bytes.

Specify the path name of desired file directly after "file://" string.

Example:

file://\\foo\SHARE\DATABASE\DATABASE.DB

A path on the FTP server (FTP method):

Specify a path name to the FTP server.

The path name can be specified with a full path name or relative path name from the current directory in logging in with FTP.

Specify the path name in the format of "server name/path name of desired file" directly after "ftp://" string.

Example) To specify using the full path name:

ftp://foo.bar.com//Usr/DataBase/DataBase.DB

Example) To specify using the relative path name:

ftp://foo.bar.com/DataBase/DataBase.DB

/SU:*SourceUserName*

Specifies a user name to be used to access the file specified in *SourceName*.

And the user name to be specified in *SourceUserName* must be set in the password management book in advance.

SourceUserName must be no longer than 20 alphanumeric characters. If a space character is contained in *SourceUserName*, enclose it in double quotation marks.

If *SourceName* is specified using a local drive path or the UNC method, this parameter's specification is ignored.

However, if *SourceName* is specified using a path on the FTP server, be sure to specify this parameter.

/D:*DestName*

Specify a path name of the transfer destination file, including the file name. You can use a wildcard for this file name.

Specify the path name of the file to *DestName* in the URL format. You can use the same specification methods as those described in the above "/S:SourceName" parameter. See the "/S:SourceName" parameter.

If the file name that already exists is specified, it will be overwritten. Note that the same file path name as the one (*SourceName*) specified with /S should not be specified.

/DU:*DestUserName*

Specifies a user name to be used to access the file specified in *DestName*.

The user name specified in *DestUserName* must be set in the password management book in advance. *DestUserName* must be no longer than 20 alphanumeric characters. If a space character is contained in *DestUserName*, enclose it in double quotation marks.

If *DestName* is specified using a local drive path or the UNC method, this parameter's specification is ignored. In this case, you need to access the target file with the user privileges logged in to the client running the Client Task Link service.

If *DestName* is specified using a path on the FPT server is specified, be sure to specify this parameter.

/LG:*LoginGroup*

Specifies a login definition name used to manage the password management book. Specify the login definition name in *LoginGroup* within 256 characters, and enclose it in double quotation marks if a space character is contained.

If the /SU or /DU parameter is specified, be sure to specify this parameter.

/A

Transfers a file in the ASCII text mode. This parameter is valid only when you transfer files on the FTP server (FTP method).

/B

Transfers a file in the binary mode. This parameter is valid only when you transfer files on the FTP server (FTP method).

If neither /A nor /B is specified, the file is transferred in the binary mode.

/RC:*nn*

Specifies the retry count in case an error occurs in file transfer.

Specify an integer value between 0 and 99 for *nn*. Setting any other value results in an error. If omitted, retry is not performed.

If a wildcard is used to specify the transfer source file and transfer destination file, retry is performed only for the failed file.

/RI:*nnnn*

Specifies the retry interval in case an error occurs. This parameter is valid if 1 or a greater value is specified to *nn* in the /RC parameter.

Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Setting any other value results in an error. The default value is 0.

Return Values

Return value	Meaning
0	Completed
value other than 0	Abended

Command Location

Systemwalker Operation Manager installation directory\MpWalker.JM\bin

Example

In this example, the command is executed to transfer the local file "c:\database\master\database.db" on the client "foo" to the local file "d:\database\database.db". If the transfer failed, retry is performed up to three times at an interval of 60 seconds.

```
FCOPYCLI.EXE foo /S:file://c:\database\master\database.db
              /D:file://d:\database\database.db /RC:3 /RI:60
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

The contents to be output are shown below:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Client host name] HostName
[Server host/User name] Server host name/user name (*)
[Retry count] Number of retries performed (*)
[Error] Error message (*)
[File transfer] Transfer source file name Transfer destination file name (**)
[Retry count] Number of retries performed (**)
[Error] Error message (**)
[Process end time] MM-DD-YYYY HH:MM:SS
```

*:

This information is output if the /SU or /DU parameter is specified.

**:

If a wild card is specified, the information of each file is output by file basis.

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

If an error occurs before transferring a file, a message is output in the following format:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Command error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- The Client Task Link service must be already working on the client side when you use this command.

- Prior to running this command, you need to configure the password management book using the password management command. See "9.3.1 LGMANAGE.EXE Password Management Command" or further details.

- When transferring files by specifying a path on the FTP server, it can be performed only when the path on the FTP server is specified for either the transfer source or the destination. The valid combinations of file path format at source and destination are listed below. Select one of the combinations from the list.

	Specifying source file	Specifying destination file
1	Local path method	Local path method
2	Local path method	UNC method

	Specifying source file	Specifying destination file
3	Local path method	FTP server path method
4	UNC method	Local path method
5	UNC method	UNC method
6	UNC method	FTP server path method
7	FTP server path method	Local path method
8	FTP server path method	UNC method

- The FTP servers available for file transfer by specifying a path on it are limited to those on which the execution results of "DIR" command are displayed in the following format (with permission, number of links, owner, group, size, date, time, and file name are listed in this order, in alphanumerical notation).

```
-rw-rw-r--  1 xxxxx  xxxxxx      605877 Dec 19 18:44 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxxx      560661 Dec 19 18:59 xxxxx.xxx
drwx------  4 xxxxx  xxxxx           96 Jan 20  1998 xxxxx
-rw-rw-r--  1 xxxxx  xxxxx         6042 Jan 25 13:38 xxxxxxxxxxx
-rw-rw-r--  1 xxxxx  xxxxx         5433 Feb 3  17:11 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxx         2155 Feb 3  17:35 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxx         3090 Feb 1  20:13 xxxxx.xxx
```

To verify whether or not file transfer can be performed by using a path on the FTP server, see "9.1.2 FCOPYSVR.EXE File Transfer Command".

- If this command is canceled, it is carried out at the point when one file has been completely transferred. Therefore, the command takes a longer time to quit if it is cancelled while a large file is being transferred.

- When using a wildcard for the file name specified with the /S parameter, the following considerations are needed:

The number of files to be processed at a time should be 500 or less, including the files to be processed simultaneously by other Server Task Link commands.

- The size per file should be 2GB or less.

9.2.5 FCTRLCLI.EXE File Operation Command

Synopsis

```
FCTRLCLI.EXE HostName {/DELETE | /RENAME | /MKDIR | /RMDIR}
             /F:FilePathName [/FN:FileName]
             [/UN:UserName] [/LG:LoginGroup]
             [/RC:nn /RI:nnnn]
```

Description

The FCTRLCLI.EXE command operates files or directories on a client. For file operations, the command deletes or renames files. For directory operations, the command creates or deletes directories. File operations with this command are performed by using the Windows shared directories (Windows network share method) or through FTP services (FTP method).

If you wish only to operate remote files, you can do it even on the remote server without having Systemwalker Operation Manager installed.

Parameters

HostName

Specifies a host name of the client on which files are to be operated. *HostName* must be no longer than 15 characters. If *HostName* contains a space character, enclose it in double quotation marks.

/DELETE

Deletes a file.

/RENAME

Renames a file.

/MKDIR

Creates a directory.

/RMDIR

Deletes a directory.

/F:*FilePathName*

Specifies a path name of the file or directory to be operated. You can use a wildcard for this file name if specified with the /DELETE or /RENAME parameter.

Use one of the following methods to specify the path name of the file to *FilePathName* in the URL format. Specify the path name within 255 characters, and enclose it in double quotation marks if a space character is contained.

A local drive path:

Specify a path including a drive name.

Specify the path name of desired file directly after "file://" string.

Example:

file://C:\DATABASE\DATABASE.DB

UNC method (Windows network share method):

Specify a path in the format of "\\computer name\share name". For "computer name", specify a Windows-based computer name within 15 alphanumeric characters. Specify a share name within 80 bytes.

Specify the path name of desired file directly after "file://" string.

Example:

file://\\foo\SHARE\DATABASE\DATABASE.DB

A path on the FTP server (FTP method):

Specify a path name to the FTP server.

The path name can be specified with a full path name or a relative path name from the current directory in logging in with FTP.

Specify the path name in the format of "server name/path name of desired file" directly after "ftp://" string.

Example) To specify using the full path name:

ftp://foo.bar.com//Usr/DataBase/DataBase.DB

Example) To specify using the relative path name:

ftp://foo.bar.com/DataBase/DataBase.DB

/FN:*FileName*

Specifies the new file name you change to. You can use a wildcard for this parameter. Specify a file name in *FileName* within 255 characters, and enclose it in double quotation marks if a space character is contained.

This parameter is valid if the /RENAME parameter is specified.

/UN:*UserName*

Specifies a user name to be used to access the file or directory specified in *FilePathName*.

The user name you want to specify must be set in the password management book in advance. *UseName* must be no longer than 20 alphanumeric characters. If *UserName* contains a space character, enclose it in double quotation marks.

If *FilePathName* is specified using a local drive path or the UNC method, this parameter's specification is ignored. In this case, you need to access the target file with the user privileges logged in to the client running the Client Task Link service. If *FilePathName* is specified using a path on the FPT server, be sure to specify this parameter.

/LG:*LoginGroup*

Specifies a login definition name used to manage the password management book. The login definition name must be no longer than 256 characters, and enclose it in double quotation marks if a space character is contained.

If the /UN parameter is specified, be sure to specify this parameter.

/RC:*nn*

Specifies the retry count in case an error occurs in file operation.

Specify an integer value between 0 and 99 for *nn*. Setting any other value results in an error. If omitted, no retry is performed.

/RI:*nnnn*

Specifies the retry interval in case an error occurs. This specification is valid if 1 or a greater value is specified in the /RC:*nn* parameter.

Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Setting any other value results in an error. The default value is 0.

Return Values

Return value	Meaning
0	Completed
Value other than 0	Abended

Command Location

Systemwalker Operation Manager installation directory\MpWalker.JM\bin

Example

In this example, the command is executed to delete the local file "c:\database\master\database.db" on the client "foo". If deletion failed, retry is performed up to three times at an interval of 60 seconds.

```
FCTRLCLI.EXE foo /DELETE /F:file://c:\database\master\database.db /RC:3 /RI:60
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

The contents to be output are shown below:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Client host name] HostName
[Server host/User name] Server host name/User name (*)
[Retry count] Number of retries performed (*)
[Error] Error message (*)
[File operation type] File operations/Directory name New file name (**)(***)
[Retry count] Number of retries performed (**)
```

```
[Error] Error message (**)
[Process end time] MM-DD-YYYY HH:MM:SS
```

*:

This information is output if the /UN parameter is specified.

**:

If a wildcard is used, the information of each file is output by file basis.

***:

The new file name you changed to is only output when the /RENAME parameter is specified.

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

In [File operation type], any of "Delete file," "Rename file," "Create directory" or "Delete directory" is output.

If an error occurs before performing file operations, a message is output in the following format:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Command error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- The Client Task Link service must be already working on the client side when you use this command.

- Prior to running this command, you need to configure the password management book using the password management command. See "9.3.1 LGMANAGE.EXE Password Management Command" or further details.

- The FTP servers available for file operations by specifying a path on it are limited to those on which the execution results of "DIR" command are displayed in the following format (with permission, number of links, owner, group, size, date, time, and file name are listed in this order, in alphanumerical notation).

```
-rw-rw-r--  1 xxxxx  xxxxxx     605877 Dec 19 18:44 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxxx     560661 Dec 19 18:59 xxxxx.xxx
drwx------  4 xxxxx  xxxxx          96 Jan 20  1998 xxxxx
-rw-rw-r--  1 xxxxx  xxxxx        6042 Jan 25 13:38 xxxxxxxxxxx
-rw-rw-r--  1 xxxxx  xxxxx        5433 Feb 3  17:11 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxx        2155 Feb 3  17:35 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxx        3090 Feb 1  20:13 xxxxx.xxx
```

To verify whether or not file transfer can be performed by using a path on the FTP server, see "9.1.2 FCOPYSVR.EXE File Transfer Command".

- When using a wildcard for the file name specified using this command with the /F parameter, the following considerations are needed:

The number of files to be processed at a time should be 500 or less, including the files to be processed by other Client Task Link commands.

- The size per file should be 2GB or less.

9.2.6 FPRSCLI.EXE File Compression/Expansion Command

Synopsis

```
FPRSCLI.EXE HostName {/P | /E} /S:SourceName /D:DestName
            [/RC:nn /RI:nnnn]
```

Description

The FPRSCLI.EXE command compresses and expands files on a client.

This command allows you to compress files on the transfer source and expand them on the transfer destination, which helps reduce the file transfer time as well as the load on the network.

Parameters

HostName

Specifies a host name of the client where the files are compressed/expanded. *HostName* must be no longer than 15 alphanumeric characters. If a space character is contained in *HostName*, enclose it in double quotation marks.

/P

Compresses files.

/E

Expands files.

/S:*SourceName*

Specifies a path name of the compression source file/expansion source file using the local drive path. You can use a wildcard for the file name of the compression source file.

SourceName must be no longer than 255 characters. If *SourceName* contains a space character, enclose it in double quotation marks.

/D:*DestName*

Specifies a path name of destination file for compression/expansion using a local path. You can use a wildcard for the file name of the expansion destination file.

Specify the path name in *DestName* within 255 characters. If a space character is contained in *DestName*, enclose it in double quotation marks.

If the file name that already exists is specified, it will be overwritten. Note that the same file path name as the one (*SourceName*) specified with /S should not be specified.

/RC:*nn*

Specifies the retry count in case an error occurs in file compression/expansion.

Specify an integer value between 0 and 99 for *nn*. Setting any other value results in an error. If omitted, no retry is performed.

If a wildcard is used to specify the compression source file or expansion destination file, retry is performed only for the failed files.

/RI:*nnnn*

Specifies the retry interval in case an error occurs. Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Setting any other value results in an error. The default value is 0.

This parameter is valid if 1 or a greater value is specified to *nn* in the /RC parameter.

Return Values

Return value	Meaning
0	Completed
Value other than 0	Abended

Command Location

Systemwalker Operation Manager installation directory\MpWalker.JM\bin

Example

In this example, the command is executed to compress the local file "c:\database\master\database.db" to "d:\database\database.prs" on the client "foo". If the compression fails, retry is performed up to three times at an interval of 60 seconds.

```
FPRSCLI.EXE foo /P /S:c:\database\master\*.* /D:d:\database\database.prs
            /RC:3 /RI:60
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

The contents to be output are shown below:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Client host name] HostName
[File compression/Expansion] Name of compression/expansion source files Name of
compression/expansion destination files (*)
[Retry count] Number of retries performed (*)
[Error] Error message (*)
[Process end time] MM-DD-YYYY HH:MM:SS
```

*:

If a wildcard is used, the information of each file is output by file basis.

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

If an error occurs before performing compression/expansion process, a message is output in the following format:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Command error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- The Client Task Link service must be already working on the client side when you use this command.

- If you compress files using the fprssvr command with the /w parameter on the UNIX-based Systemwalker Operation Manager server, and then expand them using this command on Windows, they will be extracted as follow:

  - If write permission for the files is not assigned on the compression source, the files are marked with the Read Only attribute on Windows.

  - If write permission for the files is set for any of owner, group, or other on the compression source, the files are marked with the Read/Write attribute on Windows.

- If you compress files using this command, and then expand them on the UNIX-based Systemwalker Operation Manager server, the feature of files after extraction is as follow:

File attribute on Windows before compression	Feature of file extracted on UNIX
Read-only	- The files' owner/group information will be changed to the information of the user issuing the command. - The files will have mode 0444 (readable by everyone). (*1)

File attribute on Windows before compression	Feature of file extracted on UNIX
Other than Read-only.	- The files' owner/group information will be changed to the information of the user issuing the command. - The files will have mode 0666 (readable/writable by everyone). (*2)

*1:

　　When the files are readable by everyone, it indicates the permission for the file displayed as "-r--r--r--" by using the ls -l command .

*2:

　　When the files are readable/writable by everyone, it indicates the permission for the file displayed as "-rw-rw-rw-" by using the ls -l command.

- You need the read permissions for the files you want to compress. However, the superuser can do it without the read permission.

- If this command is canceled, it is carried out at the point when one file has been completely compressed or expanded. Therefore, the command takes a longer time to quit if it is cancelled while a large file is being compressed/expanded.

- If you use a wildcard for the file name specified in this command with the /S parameter, note the following:

　　The number of files to be processed at a time on the same client should be 500 or less, including the files to be processed by other Client Task Link commands.

- A temporary file is created in the following location on the client when executing this command.

　　- A directory set in the environment variable TEMP

　　Note that the compression/expansion fails if there is not enough disk space in the TEMP. The amount of space necessary for the temporary file is determined by adding the files sizes (before compressing or after expanding) together.

- The size per file should be 2GB or less.

9.3 Task Link Management Commands

This section explains the Task Link management commands.

9.3.1 LGMANAGE.EXE Password Management Command

Synopsis

```
LGMANAGE.EXE ADD /LG:LoginGroup /UN:UserName /PW:Password
LGMANAGE.EXE DEL /LG:LoginGroup /UN:UserName
LGMANAGE.EXE MOD /LG:LoginGroup /UN:UserName /PW:Password
LGMANAGE.EXE REF /LG:LoginGroup
LGMANAGE.EXE {HELP | ?}
LGMANAGE.EXE
```

Description

The LGMANAGE.EXE command manages the password management book used in Task Link. In the password management book, user names and passwords used in the Task Link commands are managed as a file by login definition name basis.

Parameters

ADD/DEL/MOD/REF

Specifies the type of operation to be performed in the password management book.

ADD:

Adds a user name and password to the password management book.

DEL:

Deletes a user name and password from the password management book.

MOD:

Modifies a password corresponding to the user name in the password management book.

REF:

Refers to a list of user names that are registered in the password management book.

/LG:*LoginGroup*

Specifies a login definition name used in the password management book.

Specify a group name of the OS to which the user issuing the Task Link commands belongs. The login definition name is not case-sensitive and should be specified using up to 200 characters.

/UN:*UserName*

Specifies a user name who manages passwords in the password management book. The user name specified in *UserName* becomes the user who is logged in to the link destination server (such as FTP server) that will be specified using a Task Link command. *UserName* must be no longer than 20 characters. If the user name is specified in the format of containing server name or domain name, the domain name or server name must be no longer than 15 characters, and the user name must be no longer than 20 characters. The user name is case-sensitive.

/PW:*Password*

Specifies a password string. The password string specified in *Password* becomes the password for the user who is logged in to the link destination server (such as FTP server) that will be specified using a Task Link command. The password string must be specified using between 1 and 50 alphanumeric characters. To change the password, specify a new password.

HELP|?

When executing the command with "HELP" or "?", it displays the usage of this command.

No parameters specified

Parameter entry is performed in an interactive mode.

Entry items are displayed, so you need to enter every item in an interactive menu.

Cautions

- For the storage location for the files of the password management book, see "Defining a Password Management Book" in the *Systemwalker Operation Manager Installation Guide*.

- The password management book files can be distributed as the policy information, which is a part of the schedule information in Jobscheduler.

- If the password management book file with the specified login definition name is not found when adding a user name and password, this command creates a new password management book file to add the user name and password that you want.

  If you attempt to add the user name that already exists in the password management book file, it results in error without adding the user name.

- The password management book must be created and managed for each server that uses the Task Link function.

Chapter 10 Task Link Commands [UNIX]

This chapter describes the Task Link commands [UNIX] provided by Systemwalker Operation Manager. Note that the commands must be executed on the server where the Systemwalker Operation Manager server function is installed.

10.1 Server Task Link Commands

The Server Task Link commands support operation with data and job linkage, streamlining server business system configurations.

Information
. . . .
The Server Task Link commands can create a batch file and register it.
. . . .

10.1.1 execasvr Application Startup Command

Synopsis

```
execasvr [/un:UserName] [/rc:nn /ri:nnnn]
         /ae:ExecuteAppli AppliParam
```

Description

The execasvr command activates any application installed on a server. The activated application runs in the background and is not displayed on-screen.

This command will end without waiting for the application to complete.

Parameters

/un:*UserName*

Specifies the right to be used to start an application by specifying the user name.

Specify the user name using up to 20 alphanumeric characters. Space characters and any special characters other than the hyphen (-) and the underscore (_) cannot be specified.

If omitted, the application will be started with the privileges of the user executing this command.

/rc:*nn*

Specifies the retry count in case an error occurs in application startup.

Specify an integer value between 0 and 99 for *nn*. Setting any other value results in an error. If omitted, no retry is performed.

/ri:*nnnn*

Specifies the retry interval in case an error occurs.

Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Setting any other value results in an error. The default value is 0.

This parameter is valid when 1 or a greater value is specified to *nn* in the /rc parameter.

/ae:*ExecuteAppli*

Specifies the application that you want to activate using an absolute path. Specify the application name to *ExecuteAppli*. If the application name contains a space character, enclose it in double quotation marks.

The total number of characters in *ExecuteAppli* including *AppliParam* must be no longer than 511 characters.

AppliParam

Specifies parameters for the application specified in *ExecuteAppli*. If more than one parameter is specified, put a space character between parameters. If a space is contained in any parameter, enclose it in double quotation marks.

Return Values

Return value	Meaning
0	Completed
Value other than 0	Abended

Command Location

Solaris	/opt/FJSVsnjss/bin
HP-UX	/opt/FHPsnjss/bin
AIX	/opt/FAIXsnjss/bin
Linux	/opt/FJSVsnjss/bin

Example

In this example, "/usr/local/bin/test" is activated, being appended with the parameter "test.txt".

```
execasvr /ae:/usr/local/bin/test test.txt
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

The contents to be output are shown below:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Application] Name of the application activated
[Retry count] Number of retries performed
[Error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

If an error occurs before starting the application, a message is output in the following format:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Command error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- This command does not monitor the application process that the command activated. To review the process results, the application itself needs to have the feature to report a completion code.

- This command cannot start the application that requires keyboard entry.

- The application started by this command cannot be canceled, even if you cancel this command.

- If this command is canceled, it is carried out at the point when the application has completely started. Therefore, the command takes a longer time to quit as the application takes more to start.

- To start the application with the user privileges specified with the un:UserName parameter, the user executing this command must have root privileges.

- To start a shell script, specify a shell name at the beginning of the script as follows.

    Example:

```
#! /bin/sh
```

10.1.2 fcopysvr File Transfer Command

Synopsis

```
fcopysvr /s:SourceName [/su:SourceUserName]
         /d:DestName [/du:DestUserName]
         [/lg:LoginGroup] [/a | /b] [/rc:nn /ri:nnnn]
```

Description

The fcopysvr command transfers files between servers. The files that can be transferred using this command are the local files (including directories on the NFS server) or files on the FTP server.

Parameters

/s:SourceName

Specify a path name of the transfer source file, including the file name. You can use a wildcard for the file name.

Since wildcard characters such as an asterisk (*) have special meaning to shell, you need to enclose *SourceName* with either single or double quotation marks as necessary.

Use one of the following methods to specify *SourceName* in the URL format. The part of path name must be no longer than 255 characters. If the path name contains a space character, enclose it in double quotation marks.

A local drive path:

Specify an absolute path from the root directory.

Specify the path name of desired file directly after "file://" string.

Example:

If the file path is "/usr/database/database.db":

file:///usr/database/database.db

A path on the FTP server:

Specify a path name to the FTP server.

The path name can be specified with a full path name or a relative path name from the current directory in logging in with FTP. Specify the path name in the format of "server name/path name of desired file" directly after "ftp://" string.

Example) To specify using the full path name:

ftp://foo.bar.com//usr/database/database.db

Example) To specify using the relative path name:

ftp://foo.bar.com/database/database.db

/su:SourceUserName

Specifies a name of the user to be used to access the file specified in *SourceName*, and the user name to be specified must be set in the password management book in advance. *SourceUserName* must be no longer than 20 alphanumeric characters. If *SourceUserName* contains a space character, enclose it in double quotation marks.

If *SourceName* is specified using a local path, this parameter's specification is ignored. However, if *SourceName* is specified using a path on the FTP server, be sure to specify this parameter.

/d:*DestName*

Specify a path name of the destination file, including the file name. You can use a wildcard for the file name.

Since wildcard characters such as an asterisk (*) have special meaning to shell, you need to enclose *DestName* with either single or double quotation marks as necessary.

You can use the same specification methods as those described in the above "/s:SourceName" parameter. See the "/s:SourceName" parameter.

If the file name that already exists is specified, it will be overwritten. Note that the same file path name as the one (*SourceName*) specified with /s should not be specified.

/du:*DestUserName*

Specifies a user name to be used to access the file specified in *DestName*.

The user name specified in this parameter must be set in the password management book in advance. *DestUserName* must be no longer than 20 alphanumeric characters. If *DestUserName* contains a space character, enclose it in double quotation marks.

If *DestName* is specified using a local path, this parameter's specification is ignored. However, if *DestName* is specified using a path on the FTP server, be sure to specify this parameter.

/lg:*LoginGroup*

Specifies a login definition name used to manage the password management book. *LoginGroup* must be no longer than 256 characters, and enclose it in double quotation marks if a space character is contained.

If the /su or /du parameter is specified, be sure to specify this parameter.

/a

Transfers a file in the ASCII text mode.

This parameter is valid only when you transfer files on the FTP server.

/b

Transfers a file in the binary mode.

This parameter is valid only when you transfer files on the FTP server.

If neither /a nor /b is specified, the file is transferred in the binary mode.

/rc:*nn*

Specifies the retry count in case an error occurs in file transfer.

Specify an integer value between 0 and 99 for *nn*. Setting any other value results in an error. If omitted, no retry is performed.

If a wildcard is used to specify the transfer source file or transfer destination file, retry is performed only for the failed file.

/ri:*nnnn*

Specifies the retry interval in case an error occurs.

Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Setting any other value results in an error. The default value is 0.

This parameter is valid if 1 or a greater value is specified to *nn* in the /rc parameter.

Return Values

Return value	Meaning
0	Completed
Value other than 0	Abended

Command Location

Solaris	/opt/FJSVsnjss/bin
HP-UX	/opt/FHPsnjss/bin
AIX	/opt/FAIXsnjss/bin
Linux	/opt/FJSVsnjss/bin

Example

In the following example, the local file "/usr/database/master/database.db" on the server is transferred to the local file "/usr/database/database.db". If the transfer fails, retry is performed up to three times at an interval of 60 seconds.

```
fcopysvr /s:file:///usr/database/master/database.db
         /d:file:///usr/database/database.db /rc:3 /ri:60
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

```
The contents to be output are shown below:
[Process start time]  MM-DD-YYYY HH:MM:SS
[Server host/User name]  Server host name/user name (*)
[Retry count]  Number of retries performed (*)
[Error]  Error message (*)
[File transfer]  Transfer source file name Transfer destination file name (**)
[Retry count]  Number of retries performed (**)
[Error]  Error message (**)
[Process end time]  MM-DD-YYYY HH:MM:SS
```

*:

This information is output if the /su or /du parameter is specified.

**:

If a wildcard is used, the information of each file is output by file basis.

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

If an error occurs before performing file transfer, a message is output in the following format:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Command error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- Prior to running this command, you need to configure the password management book using the password management command. See "10.3.1 lgmanage Password Management Command" for further details.

- When transferring files by specifying a path on the FTP server, it can be performed only when the path on the FTP server is specified for either the transfer source or the destination. The valid combinations of file path format at source and destination are listed below. Select one of the combinations from the list.

	Specifying source file	Specifying destination file
1	Local path method	Local path method

	Specifying source file	Specifying destination file
2	Local path method	FTP server path method
3	FTP server path method	Local path method

- The FTP servers available for file transfer by specifying a path on it are limited to those on which the execution results of "dir" command are displayed in the following format (with permission, number of links, owner, group, size, date, time, and file name are listed in this order, in alphanumerical notation).

```
-rw-rw-r--  1 xxxxx  xxxxxx     605877 Dec 19 18:44 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxxx     560661 Dec 19 18:59 xxxxx.xxx
drwx------  4 xxxxx  xxxxx          96 Jan 20  1998 xxxxx
-rw-rw-r--  1 xxxxx  xxxxx        6042 Jan 25 13:38 xxxxxxxxxxx
-rw-rw-r--  1 xxxxx  xxxxx        5433 Feb 3  17:11 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxx        2155 Feb 3  17:35 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxx        3090 Feb 1  20:13 xxxxx.xxx
```

To verify whether or not file transfer can be performed by using a path on the FTP server, use the following method.

[Windows]

- Execute the ftp command at a command prompt.

[UNIX]

- Execute the ftp command on a server console or a telnet-connected console.

   Follow the steps below for checking on UNIX.

```
$ ftp XXXXX                          <-ftp command(XXXXX:ftp server name)
Connected to XXXXX.
220 XXXXX FTP server (UNIX(r) System V Release 4.0) ready.
Name (XXXXX:YYYYY): YYYYY     <-User name setting for the ftp server(YYYYY:user name)
331 Password required for YYYYY.
Password:                            <-Password entry
230 User YYYYY logged in.
ftp> dir                             <-dir command entry
200 PORT command successful.

The dir result is output.
```

- If this command is canceled, it is carried out at the point when one file has been completely transferred. Therefore, the command takes a longer time to quit if it is cancelled while a large file is being transferred.

- When using a wildcard for the file name specified with the /s parameter, the following considerations are needed:

   The number of files to be processed at a time should be 500 or less, including the files to be processed simultaneously by other Server Task Link commands.

- The size per file should be 2GB or less.

10.1.3 fctrlsvr File Operation Command

Synopsis

```
fctrlsvr {/delete | /rename | /mkdir | /rmdir} /f:FilePathName
       [/fn:FileName] [/un:UserName] [/lg:LoginGroup]
       [/rc:nn /ri:nnnn]
```

Description

The fctrlsvr command operates files or directories on a server. For file operations, the command deletes or renames files. For directory operations, the command creates or deletes directories. The files that can be operated with this command are the local files (including directories on the NFS server) and files on the FTP server.

If you wish only to operate remote files, you can do it even on the remote server without having Systemwalker Operation Manager installed.

Parameters

/delete

Deletes a file.

/rename

Renames a file.

/mkdir

Creates a directory.

/rmdir

Deletes a directory.

/f:FilePathName

Specify a path name of the file or directory to be operated. You can use a wildcard for the file name only when using with the /delete or /rename parameter.

Since wildcard characters such as an asterisk (*) have special meaning to shell, you need to enclose *FilePathName* with either single or double quotation marks as necessary.

In *FilePathName,* use one of the following methods to specify the file path name in the URL format. The path name must be no longer than 255 characters, and enclose it in double quotation marks if a space character is contained.

A local drive path:

Specify an absolute path from the root directory.

Specify the path name of desired file directly after "file://" string.

Example:

file:///usr/database/database.db

A path on the FTP server:

Specify a path name to the FTP server.

The path name can be specified with a full path name or relative path name from the current directory in logging in with FTP. Specify the path name in the format of "server name/path name of desired file" directly after "ftp://" string.

Example) To specify using the full path name:

ftp://foo.bar.com//usr/database/database.db

Example) To specify using the relative path name:

ftp://foo.bar.com/database/database.db

/fn:FileName

Specifies the new file name you change to. You can use a wildcard for this parameter.

Since wildcard characters such as an asterisk (*) have special meaning to shell, you need to enclose *FileName* with either single or double quotation marks as necessary.

FileName must be no longer than 255 characters, and enclose it in double quotation marks if a space character is contained.

This parameter is valid only when the /rename parameter is specified.

/un:*UserName*

Specifies a user name to be used to access the file or directory specified with *FilePathName*.

The user name to be specified must be set in the password management book in advance. *UserName* must be no longer than 20 alphanumeric characters. If *UserName* contains a space character, enclose it in double quotation marks.

If *FilePathName* is specified using a local path, this parameter's specification is ignored.

However, if *FilePathName* is specified using a path on the FTP server, be sure to specify this parameter.

/lg:*LoginGroup*

Specifies a login definition name used to manage the password management book. The login definition name in *LoginGroup* must be no longer than 256 characters, and enclose it in double quotation marks if a space character is contained.

If the /un parameter is specified, be sure to specify this parameter.

/rc:*nn*

Specifies the retry count in case an error occurs in file operation.

Specify an integer value between 0 and 99 for *nn*. Setting any other value results in an error. If omitted, no retry is performed.

/ri:*nnnn*

Specifies the retry interval in case an error occurs.

Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Setting any other value results in an error. The default value is 0.

This parameter is valid if 1 or a greater value is specified to *nn* in the /rc parameter.

Return Values

Return value	Meaning
0	Completed
Value other than 0	Abended

Command Location

Solaris	/opt/FJSVsnjss/bin
HP-UX	/opt/FHPsnjss/bin
AIX	/opt/FAIXsnjss/bin
Linux	/opt/FJSVsnjss/bin

Example

In this example, the local file "/usr/database/master/database.db" on the server is renamed into "database_backup.db". If renaming fails, retry is performed up to three times at an interval of 60 seconds.

```
fctrlsvr /rename /f:file:///usr/database/master/database.db
        /fn:database_backup.db /rc:3 /ri:60
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

The contents to be output are shown below:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Server host/User name] Server host name/user name (*)
[Retry count] Number of retries performed (*)
[Error] Error message (*)
[File operation type] File operations/Directory name New file name (**)(***)
[Retry count] Number of retries performed (**)
[Error] Error message (**)
[Process end time] MM-DD-YYYY HH:MM:SS
```

*:

This information is output if the /un parameter is specified.

**:

If a wildcard is used, the information of each file is output by file basis.

***:

The new file name you changed to is output only when the /rename parameter is specified.

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

In [File operation type], any of "Delete file", "Rename file", "Create directory", or "Delete directory" is output.

If an error occurs before performing file operations, a message is output in the following format:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Command error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- Prior to running this command, you need to configure the password management book by using the password management command. See "10.3.1 lgmanage Password Management Command" or further details.

- The FTP servers available for file operations by specifying a path on it are limited to those on which the execution results of "dir" command are displayed in the following format (with permission, number of links, owner, group, size, date, time, and file name are listed in this order, in alphanumerical notation).

```
-rw-rw-r--  1 xxxxx  xxxxxx      605877 Dec 19 18:44 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxxx      560661 Dec 19 18:59 xxxxx.xxx
drwx------  4 xxxxx  xxxxx           96 Jan 20  1998 xxxxx
-rw-rw-r--  1 xxxxx  xxxxx         6042 Jan 25 13:38 xxxxxxxxxxx
-rw-rw-r--  1 xxxxx  xxxxx         5433 Feb 3  17:11 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxx         2155 Feb 3  17:35 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxx         3090 Feb 1  20:13 xxxxx.xxx
```

See "10.1.2 fcopysvr File Transfer Command" or how to verify whether or not the path on the FTP server can be specified.

- When using a wildcard for the file name specified with the /f parameter, the following considerations are needed:

The number of files to be processed at a time should be 500 or less, including the files to be processed simultaneously by other Server Task Link commands.

- The size per file should be 2GB or less.

10.1.4 fprssvr File Compression/Expansion Command

Synopsis

```
fprssvr {/p [/w] | /e} /s:SourceName /d:DestName
      [/rc:nn /ri:nnnn]
```

Description

The fprssvr command compresses or expands local files on a server (including directories on the NFS server).

Parameters

/p

Compresses files.

/w

Compresses files in the format for Windows platform, which can be expanded on Windows. If omitted, the files are compressed in the format for UNIX platform, which cannot be expanded on Windows.

/e

Expands files.

/s:*SourceName*

Specifies a path name of the compression source file/expansion source file using a local path. The local path must be specified with an absolute path from the root directory, and it should look similar to "/usr/database/database.db". A path on the FTP server cannot be used to specify this parameter. You can use a wildcard for the file name of the compression source file.

Since wildcard characters such as an asterisk (*) have special meaning to shell, you need to enclose *SourceName* with either single or double quotation marks as necessary.

SourceName must be no longer than 255 characters. If *SourceName* contains a space character, enclose it in double quotation marks.

/d:*DestName*

Specifies a path name of the compression destination file/expansion destination file using a local path. A path on the FTP server cannot be used to specify this parameter. You can use a wildcard for the file name of the expansion destination file.

Since wildcard characters such as an asterisk (*) have special meaning to shell, you need to enclose *DestName* with either single or double quotation marks as necessary.

DestName must be no longer than 255 characters. If *DestName* contains a space character, enclose it in double quotation marks.

If the file name that already exists is specified, it will be overwritten. Note that the same file path name as the one (*SourceName*) specified with /s should not be specified.

/rc:*nn*

Specifies the retry count in case an error occurs in file compression/expansion. Specify an integer value between 0 and 99 for *nn*. Setting any other value results in an error. If omitted, no retry is performed.

If a wildcard is used to specify the compression source files or the expansion destination files, retry is performed only for the failed file.

/ri:*nnnn*

Specifies the retry interval in case an error occurs. Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Setting any other value results in an error. The default value is 0.

Return Values

Return value	Meaning
0	Completed
Value other than 0	Abended

Command Location

Solaris	/opt/FJSVsnjss/bin
HP-UX	/opt/FHPsnjss/bin
AIX	/opt/FAIXsnjss/bin
Linux	/opt/FJSVsnjss/bin

Example

In this example, the local file "/database/database.prs" on the server is expanded to the local file "/database/copy/*.*". If expansion fails, retry is performed up to three times at an interval of 60 seconds.

```
fprssvr /e /s:/database/database.prs /d:/database/copy/*.* /rc:3 /ri:60
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

The contents to be output are shown below:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[File compression/Expansion] Name of compression/expansion source files Name of
compression/expansion destination files (*)
[Retry count] Number of retries performed (*)
[Error] Error message (*)
[Process end time] MM-DD-YYYY HH:MM:SS
```

*:

> If a wildcard is used, the information of each is output by file basis.

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

If an error occurs before performing compression/expansion process, a message is output in the following format:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Command error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- Only a normal file can be compressed by using this command. A link file such as symbolic link (file) cannot be compressed.

- If you compress files using this command with the /w parameter (in the format for Windows platform) and expand them on Windows, they will be extracted as follow:

    - If write permission for the files is not assigned on the compression source, the files are marked with the Read Only attribute on Windows.

    - If write permission for the files is set for any of owner, group, or other on the compression source, the files are marked with the Read/Write attribute on Windows.

- If you expand the compressed files in the format for UNIX platform or Windows platform on UNIX by using this command, the feature of files after extraction is as follow:

Format of compressed file	Feature of file extracted on UNIX (command executed as a superuser)	Feature of file extracted on UNIX (command executed as a general user)
Format for UNIX platform	- The files' owner/group information at compression source will be inherited. - The mode at compression source will be inherited.	- The files' owner/group information will be changed to the information of the user issuing the command. - The mode at compression source will be inherited.

Format for Windows platform	- If the files before compression are Read-only:
	
	  - The files' owner/group information will be changed to the information of the user issuing the command.
	
	  - The files will have mode 0444 (readable by everyone).(*1)
	
	- If the files before compression are other than Read-only:
	
	  - The files' owner/group information will be changed to the information of the user issuing the command.
	
	  - The files will have mode 0666 (readable/writable by everyone). (*2)	

*1:

When the files are readable by everyone, it indicates the permission for the file displayed as "-r--r--r--" by using the ls -l command.

*2:

When the files are readable/writable by everyone, it indicates the permission for the file displayed as "-rw-rw-rw-" by using the ls -l command.

- You need the read permissions for the files you want to compress. However, the superuser can do it without the read permission.

- If this command is canceled, it is carried out at the point when one file has been completely compressed or expanded. Therefore, the command takes a longer time to quit if cancellation is performed while a large file is being compressed/ expanded.

- When using a wildcard for the file name specified with the /s parameter, the following considerations are needed:

  The number of files to be processed at a time should be 500 or less, including the files to be processed simultaneously by other Server Task Link commands.

- A temporary file is created in the following location on the server when executing this command.

  - /tmp(, /var/tmp for Solaris)

  Note that the compression/expansion fails if there is not enough disk space in the /tmp. The amount of space necessary for the temporary file is determined by adding the files sizes (before compressing or after expanding) together.

- The size per file should be 2GB or less.

10.2 Client Task Link Commands

The Client Task Link commands support job linkage with clients.

 Information
....
The Client Task Link commands can create a batch file and register it.
....

10.2.1 poncli Power On Command

Synopsis

```
poncli HostName
        [/ip:IPAddress /mac:MACAddress /snm:SubnetMask]
        [/wt:WaitTime] [/rc:nn /ri:nnnn]
```

Description

The poncli command powers on the client that supports a Wakeup on LAN feature, which enables a remote client to be booted automatically.

This command also allows you to power on the client on another subnet if in a TCP/IP environment.

This command is not supported in IPv6 environments. Use this command in IPv4 environments or in environments where dual stack is supported.

Parameters

HostName

Specifies a host name of the client to be powered on.

Specify the host name in *HostName* within 15 alphanumeric characters. If *HostName* contains a space character, enclose it in double quotation marks.

/ip:*IPAddress*

Specifies an IP address of the client to be powered on. The IP address is specified in a decimal string, which is something similar to "133.161.222.110".

If omitted, the command reads the host information definition file to retrieve the IP address of the target client.

/mac:*MACAddress*

Specifies a MAC address of the client to be powered on. The MAC address is specified in a hexadecimal string, which is similar to "00-00-0e-9d-2c-38". If omitted, the command reads the host information definition file to retrieve the MAC address of the target client.

/snm:*SubnetMask*

Specifies a subnet mask of the client to be powered on. The subnet mask is specified in a decimal string, which is something similar to "255.255.255.0".

If omitted, the command reads the host information definition file to retrieve the subnet mask of the target client.

/wt:*WaitTime*

Specifies the wait time to ensure the client is powered on. Specify an integer value between 1 and 999 for *WaitTime* in seconds.

If omitted, waiting is not performed.

/rc:*nn*

Specifies the retry count in case an error occurs in powering on the client.

Specify an integer value between 0 and 99 for *nn*. Setting any other value results in an error. If omitted, no retry is performed.

/ri:*nnnn*

Specifies the retry interval in case an error occurs. This parameter is valid if 1 or a greater value is specified to *nn* in the /rc parameter.

Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Setting any other value results in an error. The default value is 0.

Return Values

Return value	Meaning
0	Completed
Value other than 0	Abended

Command Location

Solaris	/opt/FJSVsnjss/bin
HP-UX	/opt/FHPsnjss/bin
AIX	/opt/FAIXsnjss/bin
Linux	/opt/FJSVsnjss/bin

Example

In this example, the client "foo" is powered on.

MAC address:

00-00-0e-9d-2c-38

IP address:

133.161.222.110

Subnet mask:

255.255.255.0

```
poncli foo /mac:00-00-0e-9d-2c-38 /ip:133.161.222.110 /snm:255.255.255.0
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output. The contents to be output are shown below:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Host name] HostName
[MAC address] MAC address
[IP address] IP address
[Subnet mask] Subnet mask
[Retry count] Number of retries performed
[Error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- To execute this command by using the host information definition file, define the client information in the host information definition file in advance. For details, see the *Systemwalker Operation Manager Installation Guide*.

- The client that can be powered on using this command must be a computer supporting a Wakeup on LAN feature. To turn on the power in a client on another subnet, you need to configure the network hardware (e.g., router) so that the

broadcast packet sent from the server can be received by the network of the target computer. For details about the settings method, see the documentation for your networking hardware.

To check whether the broadcast packet can arrive at the destination or not, you can use the ping command by submitting it on a command prompt on the Systemwalker Operation Manager server side. And if it is not responded, the broadcast packet cannot reach the destination.

For example, to ping the remote network host of having the IP address "10.90.100.100" and subnet mask "255.255.255.0", use the "ping 10.90.100.255" command.

- In response to connection of a power source, the Wakeup on LAN packet is passed to the network host to which the target client belongs, and immediately after that, this command is returned as completed successfully. To wait to ensure that the computer system is booted after power on, specify the /wt parameter.

10.2.2 poffcli Power Off Command

Synopsis

```
poffcli HostName [/sd:ShutdownMode] [/ef]
      [/rc:nn /ri:nnnn]
```

Description

The poffcli command powers off the client that supports the Windows power-off mechanism. It enables a remote client to be turned off automatically.

Parameters

HostName

Specifies a host name of the client to be powered off.

Specify the host name in *HostName* within 15 alphanumeric characters. If *HostName* contains a space character, enclose it in double quotation marks.

/sd:*ShutdownMode*

Specifies the mode to turn off the power. Select one of the following:

poff:

Turns off the system power.

reboot:

Reboots the system.

down:

Shuts down the system.

If omitted, the system is turned off as the poff mode is defaulted.

/ef

Specifies forced power off.

Specifying this parameter forces the running application to be terminated. At this time, the data used by those applications is not saved.

The active desktop on the client should be disabled. Otherwise, the active desktop recovery function may be started automatically the next time you start the OS.

/rc:*nn*

Specifies the retry count in case an error occurs in powering off clients.

Specify an integer value between 0 and 99 for *nn*. Setting any other value results in an error. If omitted, no retry is performed.

/ri:*nnnn*

Specifies the retry interval in case an error occurs.

Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Setting any other value results in an error. The default value is 0.

This parameter is valid if 1 or a greater value is specified to *nn* in the /rc parameter.

Return Values

Return value	Meaning
0	Completed
Value other than 0	Abended

Command Location

Solaris	/opt/FJSVsnjss/bin
HP-UX	/opt/FHPsnjss/bin
AIX	/opt/FAIXsnjss/bin
Linux	/opt/FJSVsnjss/bin

Example

The following example performs power off for the client "foo".

```
poffcli foo /sd:poff
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

The contents to be output are shown below:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Client host name] HostName
[Retry count] Number of retries performed
[Error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

If an error occurs before power off, a message is output in the following format:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Command error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- The Client Task Link service must be already working on the client side when you use this command. At this time, the Client Task Link service must be started by the user having privileges to power off clients.

- The client that can be powered off using this command must be a computer supporting the Windows power-off mechanism.

- If a running application is not terminated in response to a termination request sent from Windows, it may not be possible to power off the client.

- If the /ef parameter is specified, applications that do not respond to a Windows' termination request are forcefully terminated by Windows.

- The specification by the /rc parameter or /ri parameter becomes valid only when the user logged on to the target client does not have the shutdown privilege in Windows operating systems.

10.2.3 execacli Application Startup Command

Synopsis

```
execacli HostName [/am:AppMode] [/we:WaitEnd]
        [/rc: nn /ri:nnnn]
        /ae:ExecuteAppli [AppliParam]
```

Description

The execacli command activates any application installed on a client. This command can wait for the application to complete.

Parameters

HostName

Specifies a host name of the client where the application will be activated.

Specify a host name in *HostName* within 15 alphanumeric characters. If *HostName* contains a space character, enclose it in double quotation marks.

/am:*AppMode*

Specifies the mode to start the application.

Select one of the following:

normal:

Activates the application in a normal window.

icon:

Activates the application in iconic state.

If this parameter is omitted, it defaults to the normal mode, activating in a normal window.

/we:*WaitEnd*

Specifies the time to wait for the activated application to complete. Specify an integer value between 1 and 9600 (in seconds) in *WaitEnd*. Specifying any other value results in an error.

If omitted, waiting for the application to complete is not performed.

/rc:*nn*

Specifies the retry count in case an error occurs in application startup.

Specify an integer value between 0 and 99 for *nn*. Specifying any other value results in an error. If omitted, no retry is performed.

/ri:*nnnn*

Specifies the retry interval in case an error occurs. This parameter is valid if 1 or a greater value is specified to *nn* in the /rc parameter.

Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Specifying any other value results in an error. The default value is 0.

/ae:*ExecuteAppli*

Specifies an application name that you want to activate using the local drive path with no longer than 511 characters. If the name of the target application or any parameter to be passed to the application contains a space character, enclose each parameter in double quotation marks.

AppliParam

Specifies the parameter(s) for the application specified with *ExecuteAppli*. If more than one parameter is specified, put a space between parameters. If a space character is contained in any parameter, enclose it in double quotation marks.

Return Values

Return value	Meaning
0	Completed
Value other than 0	Abended (or an application started with the /we parameter has terminated with a return value other than 0.)

Command Location

Solaris	/opt/FJSVsnjss/bin
HP-UX	/opt/FHPsnjss/bin
AIX	/opt/FAIXsnjss/bin
Linux	/opt/FJSVsnjss/bin

Example

In this example, the application "c:\program files\app.exe" on the client "foo" is activated in a normal window, and then waits for this application to complete for 60 seconds. "test.txt" is a parameter for the application "app.exe".

```
execacli foo /am:normal /we:60 /ae:"C:\\program files\\app.exe" test.txt
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

The contents to be output are shown below:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Client host name] HostName
[Application] Name of the application that was activated
[Retry count] Number of retries performed
[Error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

If an error occurs before starting the application, a message is output in the following format:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Command error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- The Client Task Link service must be already working on the client side when you use this command.

- This command waits for the application to complete only when the /we parameter is specified.

- The application started by this command cannot be canceled, even if you cancel this command.

- If this command is canceled, it is carried out at the point when the application has been completely started. Therefore, the command takes a longer time to quit as the application takes more to start.

- A backslash (\) in a path name for personal computer can be interpreted as an escape sequence depending on UNIX shells. To use a backslash in such shell string, enter two backlashes (\\).

10.2.4 fcopycli File Transfer Command

Synopsis

```
fcopycli HostName /s:SourceName [/su:SourceUserName]
        /d:DestName [/du:DestUserName]
        [/lg:LoginGroup] [/a | /b] [/rc:nn /ri:nnnn]
```

Description

The fcopycli command enables file transfer by clients. This command allows the transfer of files on client to server or files on server to client. It performs file transfers using the Windows shared directories (Windows network share method) or file transfers through FTP services (FTP method).

If you wish only to perform file transfer, you can transfer files even to the destination server without having Systemwalker Operation Manager installed.

Parameters

HostName

Specifies a host name of the client which is requested to transfer a file. Specify the host name in *HostName* within 15 alphanumeric characters.

If *HostName* contains a space character, enclose it in double quotation marks.

/s:*SourceName*

Specify a path name of the transfer source file, including the file name. You can use a wildcard for the file name.

Since wildcard characters such as an asterisk (*) have special meaning to shell, you need to enclose *SourceName* with either single or double quotation marks as necessary.

Use one of the following methods to specify *SourceName* in the URL format. The parts of path name must be no longer than 255 characters. If the path name contains a space character, enclose it in double quotation marks.

A local drive path:

Specify a path including a drive name.

Specify the path name of desired file directly after "file://" string.

Example:

file://c:\database\database.db

UNC method (Windows network share method):

Specify a path in the format of "\\computer name\share name". For "computer name", specify a Windows-based computer name within 15 alphanumeric characters. Specify a share name within 80 bytes.

Specify the path name of desired file directly after "file://" character string.

Example:

file://\\foo\share\database\database.db

A path on the FTP server (FTP method):

Specify a path name to the FTP server.

The path name can be specified with the full path name or the relative path name from the current directory in logging in with FTP. Specify the path name in the format of "server name/path name of desired file" directly after "ftp://" string.

Example) To specify using the full path name:

ftp://foo.bar.com//usr/database/database.db

Example) To specify using the relative path name:

ftp://foo.bar.com/database/database.db

/su:*SourceUserName*

Specifies a user name to be used to access the file specified in *SourceName*.

And the user name to be specified to *SourceUserName* must be set in the password management book in advance. *SourceUserName* must be specified using up to 20 alphanumeric characters. If *SourceUserName* contains a space character, enclose it in double quotation marks.

If *SourceName* is specified using a local drive path or the UNC method, this parameter's specification is ignored. In this case, you need to access the target file by using the account name of user logged on to the client running the Client Task Link service.

However, if *SourceName* is specified using a path on the FTP server, be sure to specify this parameter.

/d:*DestName*

Specify a path name of the transfer destination file, including the file name. You can use a wildcard for the file name.

Since wildcard characters such as an asterisk (*) have special meaning to shell, you need to enclose *DestName* with either single or double quotation marks as necessary.

For *DestName*, specify the path name of the file in the URL format. You can use the same specification methods as those described in the above "/s:SourceName" parameter. See the "/s:SourceName" parameter.

If the file name that already exists is specified, it will be overwritten. Note that the same file path name as the one (*SourceName*) specified with /s should not be specified.

/du:*DestUserName*

Specifies a user name to be used to access the file specified in *DestName*.

The user name specified in *DestUserName* must be set in the password management book in advance. *DestUserName* must be specified using up to 20 alphanumeric characters. If *DestUserName* contains a space character, enclose it in double quotation marks.

If *DestName* is specified using a local drive path or the UNC method, this parameter's specification is ignored. In this case, you need to access the target file by using the account name of user logged in to the client running the Client Task Link service.

If *DestName* is specified using a path on the FPT server is specified, be sure to specify this parameter.

/lg:*LoginGroup*

Specifies a login definition name used to manage the password management book. Specify the login definition name in *LoginGroup* within 256 characters, and enclose it in double quotation marks if a space character is contained.

If the /su or /du parameter is specified, be sure to specify this parameter.

/a

Transfers a file in the ASCII text mode. This parameter is valid only when you transfer files on the FTP server (FTP method).

/b

Transfers a file in the binary mode. This parameter is valid only when you transfer files on the FTP server (FTP method).

If neither /a nor /b is specified, the file is transferred in the binary mode.

/rc:*nn*

Specifies the retry count in case an error occurs in file transfer.

Specify an integer value between 0 and 99 for *nn*. Setting any other value results in an error. If omitted, retry is not performed.

If a wildcard is used to specify the transfer source file and transfer destination file, retry is performed only for the failed file.

/ri:*nnnn*

Specifies the retry interval in case an error occurs. This parameter is valid if 1 or a greater value is specified to *nn* in the /rc parameter.

Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Setting any other value results in an error. The default value is 0.

Return Values

Return value	Meaning
0	Completed
value other than 0	Abended

Command Location

Solaris	/opt/FJSVsnjss/bin
HP-UX	/opt/FHPsnjss/bin
AIX	/opt/FAIXsnjss/bin
Linux	/opt/FJSVsnjss/bin

Example

In this example, the command is executed to transfer the local file "c:\database\master\database.db" on the client "foo" to the local file "d:\database\database.db". If the transfer failed, retry is performed up to three times at an interval of 60 seconds.

```
fcopycli foo /s:file://c:\\database\\master\\database.db
        /d:file://d:\\database\\database.db /rc:3 /ri:60
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

The contents to be output are shown below:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Client host name] HostName
[Server host/User name] Server host name/user name (*)
[Retry count] Number of retries performed (*)
[Error] Error message (*)
[File transfer] Transfer source file name Transfer destination file name (**)
[Retry count] Number of retries performed (**)
[Error] Error message (**)
[Process end time] MM-DD-YYYY HH:MM:SS
```

*:

This information is output if the /su or /du parameter is specified.

**:

> If a wild card is specified, the information of each file is output by file basis.

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

If an error occurs before transferring a file, a message is output in the following format:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Command error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- The Client Task Link service must be already working on the client side when you use this command.

- Prior to running this command, you need to configure the password management book using the password management command. See "10.3.1 lgmanage Password Management Command" or further details.

- When transferring files by specifying a path on the FTP server, it can be performed only when the path on the FTP server is specified for either the transfer source or the destination. The valid combinations of file path format at source and destination are listed below. Select one of the combinations from the list.

	Specifying source file	Specifying destination file
1	Local path method	Local path method
2	Local path method	UNC method
3	Local path method	FTP server path method
4	UNC method	Local path method
5	UNC method	UNC method
6	UNC method	FTP server path method
7	FTP server path method	Local path method
8	FTP server path method	UNC method

- The FTP servers available for file transfer by specifying a path on it are limited to those on which the execution results of "dir" command are displayed in the following format (with permission, number of links, owner, group, size, date, time, and file name are listed in this order, in alphanumerical notation).

```
-rw-rw-r--  1 xxxxx   xxxxxx     605877 Dec 19 18:44 xxxxx.xxx
-rw-rw-r--  1 xxxxx   xxxxxx     560661 Dec 19 18:59 xxxxx.xxx
drwx------  4 xxxxx   xxxxx          96 Jan 20  1998 xxxxx
-rw-rw-r--  1 xxxxx   xxxxx        6042 Jan 25 13:38 xxxxxxxxxxx
-rw-rw-r--  1 xxxxx   xxxxx        5433 Feb 3  17:11 xxxxx.xxx
-rw-rw-r--  1 xxxxx   xxxxx        2155 Feb 3  17:35 xxxxx.xxx
-rw-rw-r--  1 xxxxx   xxxxx        3090 Feb 1  20:13 xxxxx.xxx
```

> To verify whether or not file transfer can be performed by using a path on the FTP server, see "10.1.2 fcopysvr File Transfer Command".

- If this command is canceled, it is carried out at the point when one file has been completely transferred. Therefore, the command takes a longer time to quit if it is cancelled while a large file is being transferred.

- A backslash (\) in a path name for personal computer can be interpreted as an escape sequence depending on UNIX shells. To use a backslash in such shell string, enter two backlashes (\\).

- When using a wildcard for the file name specified with the /s parameter, the following considerations are needed:

> The number of files to be processed at a time should be 500 or less, including the files to be processed by other Client Task Link commands.

- The size per file should be 2GB or less.

10.2.5 fctrlcli File Operation Command

Synopsis

```
fctrlcli HostName {/delete | /rename | /mkdir | /rmdir}
        /f:FilePathName [/fn:FileName] [/un:UserName]
        [/lg:LoginGroup] [/rc:nn /ri:nnnn]
```

Description

The fctrlcli command operates files or directories on a client. For file operations, the command deletes or renames files. For directory operations, the command creates or deletes directories. File operations with this command are performed using the Windows shared directories (Windows network share method) or through FTP services (FTP method).

If you wish only to operate remote files, you can do it even on the remote server without having Systemwalker Operation Manager installed.

Parameters

HostName

Specifies a host name of the client on which the file is to be operated. The host name in *HostName* must be specified using up to 15 characters. If *HostName* contains a space character, enclose it in double quotation marks.

/delete

Deletes a file.

/rename

Renames a file.

/mkdir

Creates a directory.

/rmdir

Deletes a directory.

/f:*FilePathName*

Specify a path name of the file or directory to be operated. You can use a wildcard for the file name if specified with the /delete or /rename parameter.

Since wildcard characters such as an asterisk (*) have special meaning to shell, you need to enclose *FilePathName* with either single or double quotation marks as necessary.

Use one of the following methods to specify the path name of the file to *FilePathName* in the URL format. The path name must be no longer than 255 characters, and enclose it in double quotation marks if a space character is contained.

A local drive path:

Specify a path including the drive name.

Specify the path name of desired file directly after "file://" character string.

Example:

file://c:\database\database.db

UNC method (Windows network share method):

Specify a path in the format of "\\computer name\share name". For "computer name", specify a Windows-based computer name within 15 alphanumeric characters. Specify a share name within 80 bytes.

Specify the path name of desired file directly after "file://" string.

Example:

file://\\foo\share\database\database.db

By specifying a path on the FTP server (FTP method):

Specify the path name to the FTP server.

The path name can be specified with a full path name or relative path name from the current directory in logging in with FTP.

Specify the path name in the format of "server name/path name of desired file" directly after "ftp://" string.

Example) To specify using the full path name:

ftp://foo.bar.com//usr/database/database.db

Example) To specify using the relative path name:

ftp://foo.bar.com/database/database.db

/fn:*FileName*

Specifies the new file name you change to. You can use a wildcard for this parameter.

Since wildcard characters such as an asterisk (*) have special meaning to shell, you need to enclose *FileName* with either single or double quotation marks as necessary.

Specify a file name in *FileName* within 255 characters, and enclose it in double quotation marks if a space character is contained.

This parameter is valid if the /rename parameter is specified.

/un:*UserName*

Specifies a user name to be used to access the file or directory specified in *FilePathName*.

The user name to be specified must be set in the password management book in advance. *UseName* must be no longer than 20 alphanumeric characters. If *UserName* contains a space character, enclose it in double quotation marks.

If *FilePathName* is specified using a local drive path or the UNC method, this parameter's specification is ignored. In this case, you need to access the target file with the user privileges logged in to the client running the Client Task Link service. If *FilePathName* is specified using a path on the FPT server, be sure to specify this parameter.

/lg:*LoginGroup*

Specifies a login definition name used to manage the password management book. The login definition name in *LoginGroup* must be no longer than 256 characters, and enclose it in double quotation marks if a space character is contained.

If the /un parameter is specified, be sure to specify this parameter.

/rc:*nn*

Specifies the retry count in case an error occurs in file operation.

Specify an integer value between 0 and 99 for *nn*. Setting any other value results in an error. If omitted, no retry is performed.

/ri:*nnnn*

Specifies the retry interval in case an error occurs. This specification is valid if 1 or a greater value is specified in the /rc:*nn* parameter.

Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Setting any other value results in an error. The default value is 0.

Return Values

Return value	Meaning
0	Completed
Value other than 0	Abended

Command Location

Solaris	/opt/FJSVsnjss/bin
HP-UX	/opt/FHPsnjss/bin
AIX	/opt/FAIXsnjss/bin
Linux	/opt/FJSVsnjss/bin

Example

In this example, the command is executed to delete the local file "c:\database\master\database.db" on the client "foo". If deletion failed, retry is performed up to three times at an interval of 60 seconds.

```
fctrlcli foo /delete /f:file://c:\\database\\master\\database.db
/rc:3 /ri:60
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

The contents to be output are shown below:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Client host name] HostName
[Server host/User name] Server host name/user name (***)
[Retry count] Number of retries performed (***)
[Error] Error message (***)
[File operation type] File operations/Directory name New file name (*)(**)
[Retry count] Number of retries performed (*)
[Error] Error message (*)
[Process end time] MM-DD-YYYY HH:MM:SS
```

*:

If a wildcard is used, the information of each file is output by file basis.

**:

The new file name you changed to is only output if the /rename parameter is specified.

***:

This information is output if the /un parameter is specified.

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

In [File operation type], any of "Delete file," "Rename file," "Create directory" or "Delete directory" is output.

If an error occurs before performing file operations, a message is output in the following format:

```
[Process start date and time] MM/DD/YYYY HH:MM:SS
[Command error] Error message
[Process end date and time] MM/DD/YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- The Client Task Link service must be already working on the client side when you use this command.

- Prior to running this command, you need to configure the password management book using the password management command. See "10.3.1 lgmanage Password Management Command" or further details.

- The FTP servers available for file operations by specifying a path on it are limited to those on which the execution results of "dir" command are displayed in the following format (with permission, number of links, owner, group, size, date, time, and file name are listed in this order, in alphanumerical notation).

```
-rw-rw-r--  1 xxxxx  xxxxxx      605877 Dec 19 18:44 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxxx      560661 Dec 19 18:59 xxxxx.xxx
drwx------  4 xxxxx  xxxxx           96 Jan 20  1998 xxxxx
-rw-rw-r--  1 xxxxx  xxxxx         6042 Jan 25 13:38 xxxxxxxxxxx
-rw-rw-r--  1 xxxxx  xxxxx         5433 Feb 3  17:11 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxx         2155 Feb 3  17:35 xxxxx.xxx
-rw-rw-r--  1 xxxxx  xxxxx         3090 Feb 1  20:13 xxxxx.xxx
```

To verify whether or not file transfer can be performed by using a path on the FTP server, see "10.1.2 fcopysvr File Transfer Command".

- A backslash (\) in a path name for personal computer can be interpreted as an escape sequence depending on UNIX shells. To use a backslash in such shell string, enter two backlashes (\\).

- If you use a wildcard for the file name specified using this command with the /f parameter, the following considerations are needed:

The number of files to be processed at a time should be 500 or less, including the files to be processed by other Client Task Link commands.

- The size per file should be 2GB or less.

10.2.6 fprscli File Compression/Expansion Command

Synopsis

```
fprscli HostName {/p | /e} /s:SourceName /d:DestName
      [/rc:nn /ri:nnnn]
```

Description

The fprscli command compresses and expands files on a client.

You can use this command to compress a file at the transfer source and expand it at the transfer destination, which helps reduce the file transfer time as well as the load on the network.

Parameters

HostName

Specifies a host name of the client where the files are compressed/expanded. Specify a host name in *HostName* within 15 alphanumeric characters. If a space character is contained in *HostName*, enclose it in double quotation marks.

/p

Compresses files.

/e

Expands files.

/s:*SourceName*

Specify a path name of the compression source file/expansion source file using the local drive path. You can use a wildcard for the file name of the compression source file.

Since wildcard characters such as an asterisk (*) have special meaning to shell, you need to enclose *SourceName* with either single or double quotation marks as necessary.

SourceName must be no longer than 255 characters. If *SourceName* contains a space character, enclose it in double quotation marks.

/d:*DestName*

Specify a path name of destination file for compression/expansion using a local path. You can use a wildcard for the file name of the expansion destination file.

Since wildcard characters such as an asterisk (*) have special meaning to shell, you need to enclose *DestName* with either single or double quotation marks as necessary.

DestName must be no longer than 255 characters. If *DestName* contains a space character, enclose it in double quotation marks.

If the file name that already exists is specified, it will be overwritten. Note that the same file path name as the one (*SourceName*) specified with /s should not be specified.

/rc:*nn*

Specifies the retry count in case an error occurs in file compression/expansion.

Specify an integer value between 0 and 99 for *nn*. Setting any other value results in an error. If omitted, no retry is performed.

If a wildcard is used to specify the source files for compression or the destination files for expansion, retry is performed only for the failed files.

/ri:*nnnn*

Specifies the retry interval in case an error occurs.

Specify an integer value between 0 and 9999 (in seconds) for *nnnn*. Setting any other value results in an error. The default value is 0.

This parameter is valid if 1 or a greater value is specified to *nn* in the /rc parameter.

Return Values

Return value	Meaning
0	Completed
Value other than 0	Abended

Command Location

Solaris	/opt/FJSVsnjss/bin
HP-UX	/opt/FHPsnjss/bin
AIX	/opt/FAIXsnjss/bin
Linux	/opt/FJSVsnjss/bin

Example

In this example, the command is executed to compress the local file "c:\database\master*.*" to "d:\database\database.prs" on the client "foo". If the compression fails, retry is performed up to three times at an interval of 60 seconds.

```
fprscli foo /p /s:c:\\database\\master\\*.* /d:d:\\database\\database.prs
          /rc:3 /ri:60
```

Execution Results/Output Format

After the command has been through processing, the information at command completion is output to the standard output.

The contents to be output are shown below:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Client host name] HostName
[File compression/Expansion] Name of compression/expansion source files Name of
compression/expansion destination files (*)
[Retry count] Number of retries performed (*)
[Error] Error message (*)
[Process end time] MM-DD-YYYY HH:MM:SS
```

*:

If a wildcard is used, the information of each file is output by file basis.

In [Error], "None" is output if the command completed successfully. A message is output if the command is canceled or failed. For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

If an error occurs before performing compression/expansion process, a message is output in the following format:

```
[Process start time] MM-DD-YYYY HH:MM:SS
[Command error] Error message
[Process end time] MM-DD-YYYY HH:MM:SS
```

For details on the messages, see the *Systemwalker Operation Manager Message Guide*.

Cautions

- The Client Task Link service must be already working on the client side when you use this command.

- A backslash (\) in a path name personal computer can be interpreted as an escape sequence depending on UNIX shells. To use a backslash in such shell string, enter two backslashes (\\).

- If a file is compressed without the /w parameter on a UNIX server running this version or later versions of Systemwalker Operation Manager, it cannot be expanded on a client.

- The compressed file created with this command becomes the format for Windows platform that is the same as the one created with the /w parameter on a UNIX- based server.

- If you compressed files using the fprssvr (server file compression/expansion) command with the /w parameter and expand them using this command on Windows, they will be extracted as follow:

    - If write permission for the files are not assigned on the compression source, the files are marked with the Read Only attribute on Windows.

    - If write permission for the file is set for any of owner, group, or other on the compression source, the files are marked with the Read/Write attribute on Windows.

- If you compress files in the format for Windows platform by using this command, and then expand them on UNIX, the feature of files after extraction is as follow:

File attribute before compression	Feature of file extracted on UNIX
Read-only	- The files' owner/group information will be changed to the information of the user issuing the command. - The files will have mode 0444 (readable by everyone).(*1)
Other than Read-only.	- The files' owner/group information will be changed to the information of the user issuing the command.

File attribute before compression	Feature of file extracted on UNIX
	- The files will have mode 0666 (readable/writable by everyone). (*2)

*1:

When the files are readable by everyone, it indicates the permission for the files displayed as "-r--r--r--" by using the ls -l command.

*2:

When the files are readable/writable by everyone, it indicates the permission for the files displayed as "-rw-rw-rw-" by using the ls -l command.

- You need the read permissions for the files you want to compress. However, the superuser can do it without the read permission.

- If this command is canceled, it is carried out at the point when one file has been completely compressed or expanded. Therefore, the command takes a longer time to quit if it is cancelled while a large file is being compressed/expanded.

- If you use a wildcard for the file name specified in this command with the /s parameter, note the following:

The number of files to be processed at a time on the same client should be 500 or less, including the files to be processed by other Client Task Link commands.

- A temporary file is created in the following location on the client when executing this command.

    - A directory set in the environment variable temp

Note that the compression/expansion fails if there is not enough disk space in the temp. The amount of space necessary for the temporary file is determined by adding the files sizes (before compressing or after expanding) together.

- The size per file should be 2GB or less.

10.3 Task Link Management Commands

This section explains the Task Link management commands.

10.3.1 lgmanage Password Management Command

Synopsis

```
lgmanage add /lg:LoginGroup /un:UserName /pw:Password
lgmanage del /lg:LoginGroup /un:UserName
lgmanage mod /lg:LoginGroup /un:UserName /pw:Password
lgmanage ref /lg:LoginGroup
lgmanage {help | ?}
lgmanage
```

Description

The lgmanage command manages the password management book used in Task Link. In the password management book, user names and passwords used in the Task Link commands are managed as a file by login definition name basis.

Parameters

add/del/mod/ref

Specifies the type of operation to be performed in the password management book.

add:

Adds a user name and password to the password management book.

del:

> Deletes a user name and password from the password management book.

mod:

> Modifies a password corresponding to the user name in the password management book.

ref:

> Refers to a list of user names that are registered in the password management book.

/lg:*LoginGroup*

Specifies a login definition name used in the password management book.

Specify a group name of the OS to which the user issuing the Task Link commands belongs. The login definition name is case-sensitive and must be no longer than 200 characters.

/un:*UserName*

Specifies a user name who manages passwords in the password management book. The user name specified in *UserName* becomes the user who is logged in to the link destination server (such as FTP server) that will be specified using a Task Link command. The user name must be no longer than 20 characters. If the user name is specified in the format of containing server name or domain name, the domain name or server name must be no longer than 15 characters, and the user name must be no longer than 20 characters. The user name is case-sensitive.

/pw:*Password*

Specifies a password string. The password string specified in *Password* becomes the password for the user who is logged in to the link destination server (such as FTP server) that will be specified using a Task Link command. The password string must be specified using between 1 and 50 alphanumeric characters. To change the password, specify a new password.

help|?

When executing the command with "help" or "?", it displays the usage of this command.

No parameters specified

Parameter entry is performed in an interactive mode.

Entry items are displayed, so you need to enter every item in an interactive menu.

Command Location

Solaris	/opt/FJSVsnjss/bin
HP-UX	/opt/FHPsnjss/bin
AIX	/opt/FAIXsnjss/bin
Linux	/opt/FJSVsnjss/bin

Cautions

- For the storage location for the files of the password management book, see "Defining a Password Management Book" in the *Systemwalker Operation Manager Installation Guide*.

- If the password management book file with the specified login definition name is not found when adding a user name and password, this command creates a new password management book file to add the user name and password that you want. If you attempt to add the user name that already exists in the password management book file, it results in error without adding the user name.

- The password management book must be created and managed for each server that uses the Task Link function.

- The password management book files can be distributed as the policy information, which is a part of the schedule information in Jobscheduler.

- For UNIX version, a user name, group name (login definition name) do not accept other than alphanumeric characters, the hyphen (-), and the underscore (_).

- When specifying user name/password using the parameter, the password will be left in the shell history file under the following conditions, and it could be read from other users:

  - The command history feature of the shell command is used.

  - Read permission for the history file is assigned to other users' access privileges.

If you are unsure of your shell specifications, start the lgmanage command without specifying the parameters, and add the user name/password in an interactive manner.

Chapter 11 Utility Commands [Windows]

This chapter describes the Utility commands.

Note

If commands are executed in an environment where the UAC function for Windows Server 2008 or later has been enabled, then a dialog box asking a user's privileges to be promoted may be displayed depending on the privileges of the user that logged in to the operating system, and action may be required.

Refer to "Notes on Executing Commands in Windows [Windows version]" in the *Systemwalker Operation Manager User's Guide* for more information.

11.1 chksrv Service Status Confirmation Command

This command confirms the status of services in batch files to start up/stop services.

Synopsis

```
chksrv service name {start|stop}
```

Operands

service name

Specify the name of service whose status is to be confirmed.

start

Specify to check whether the service has started or not.

stop

Specify to check whether the service has stopped or not.

Return Values

Return value	Meaning
0	When "start" operand is specified, this indicates the service has started. When "stop" operand is specified, this indicates the service has stopped.
10	When "start" operand is specified, this indicates the service has not started. When "stop" operand is specified, this indicates the service has not stopped.
20	The service is in the process of startup.
30	The service is in the process of stopping.
40	The service is in the process of resuming.
50	The service is in the process of pausing.
60	The service has paused.
-1	Another error has occurred.

Standard Output

When executing the chksrv command, the following character strings are logged to the standard output (stdout).

When a service has started running :	SERVICE_RUNNING

When a service has stopped :	SERVICE_STOPPED
When a service is in the process of startup :	SERVICE_START_PENDING
When a service is in the process of stopping :	SERVICE_STOP_PENDING
When a service is in the process of resuming:	SERVICE_CONTINUE_PENDING
When a service is in the process of pausing:	SERVICE_PAUSE_PENDING
When a service has paused:	SERVICE_PAUSED

Examples

The example of use in batch file is shown below:

An example of a batch file that starts services

If the upper limit value for normal completion code of job is 0, the batch file shown below terminates normally when services have started running (completion code is 0), while the batch file terminates abnormally when services have not started running (completion code is 10).

```
net start service name
chksrv service name start
```

An example of a batch file that stops services

If the upper limit value for normal completion code of job is 0, the batch file shown below terminates normally when services have stopped (completion code is 0), while the batch file terminates abnormally when services have not stopped (completion code is 10).

```
net stop service name
chksrv service name stop
```

11.2 f3crhsst Service Startup Command

This command gives instructions to the service control manager to start services.

Synopsis

```
f3crhsst $Serv:service name
```

Operand

service name

Specify the name of service to be started.

Return Values

Return value	Meaning
0	Instruction has been normally given.
3	Service has already started.
7	Failed to operate service.
8	The specified service name does not exist.
9	The specified service is invalid or cannot be started because no valid device is associated with it.
10	There is an error in the parameter.

Return value	Meaning
20	A system error has occurred.

Example

The following example starts a Jobscheduler service:

```
f3crhsst $Serv:Systemwalker MpJobsch
```

11.3 f3crhssp Service Stop Command

This command gives instructions to the service control manager to stop services.

Synopsis

```
f3crhssp $Serv:displayed service name
```

Operand

service name

Specify the displayed name of service to be stopped.

Return Values

Return value	Meaning
0	Instruction has been normally given.
6	Another service in execution has a dependency.
7	Service has already stopped.
8	Failed to operate service.
9	The specified displayed service name does not exist.
10	There is an error in parameter.
20	A system error has occurred.

Example

The following example stops a Jobscheduler service:

```
f3crhssp $Serv:Systemwalker MpJobsch
```

Chapter 12 SSL Environment Setup Commands

12.1 cmdspcert Command

Synopsis

cmdspcert [-ed Environment-directory] {-id CertID | -nn NickName | -fn FileName }
[-sl Select] [-oc OutCODE] [-of OutFile]

Description

The cmdspcert command edits the certificate content for each item and outputs it to standard output or a file. Options that can be specified are shown below.

Options

-ed Environment-directory

Specify the full path of the operation management directory (Environment-directory). If this option is omitted, the information set in the "CMIPATH" environment variable takes effect.

-id CertID

Specify the certificate ID (CertID) displayed by the cmlistcert command as a search condition to display certificates that are already registered.

-nn NickName

Specify the nickname (NickName) that was specified at the time of registration as a search condition to display certificates that are already registered.

-fn FileName

Specify the certificate file name (FileName) as a search condition to display unregistered certificates.
DER and BASE64 format files can be specified.

-sl Select

Specify the display format (Select).
0 : Display each item (default)
1 : Display key

-of OutFile

Specify the full path of the file name (OutFile) where the display results are output.
If this is omitted, the results will be output in standard output. Specify a file that is not yet used.

Command Location

Windows	%CommonProgramFiles%\Fujitsu Shared \F3FSSMEE
Solaris	/opt/FJSVsmee/bin
Linux	/opt/FJSVsmee/bin
Linux x64	/opt/FJSVsmee64/bin

Cautions

- Either -id, -nn, or -fn must be specified.

- The certificate expiry date can be checked in the "VALIDITY" line. The expiry date is displayed in global standard time in the "Start date and time (YYYYMMDDHHMMSS)- End date and time (YYYYMMDDHHMMSS)" format.

- If a BASE64 format certificate is specified in the certificate file, and if the header and footer shown below are attached, that line is skipped. Note that, if the format is other than as shown below, the command ends with an error. Also, if the specified certificate file contains multiple certificates, only the first data is targeted.
  Header: -----Line starting with BEGIN
  Footer: -----Line starting with END

- Items that do not have identification information set are not displayed.

12.2 cmentcert Command

Synopsis

```
cmentcert FileName [-ed Environment-directory ] [-nv] [-ca|-own] [-nn NickName]
```

Description

The cmentcert command registers certificates in the certificate/key management environment. Options that can be specified are shown below.

Options

FileName

Specify the full path of the filename (FileName) where certificate data to be registered is stored.
DER and BASE64 format files can be specified.

-ed Environment-directory

Specify the full path of the operation management directory (Environment-directory). If this option is omitted, the information set in the "CMIPATH" environment variable takes effect.

-nv

Specify this option if certificate verification is not performed during registration. If this option is specified, invalid certificates can be registered, so do not specify this option.

-ca

Specify this option if the certificate to be registered is a certificate issued by the Certificate Authority or an intermediate CA (intermediate Certificate Authority).

-own

Specify this option if the certificate to be registered is a site certificate issued for yourself.

-nn NickName

Specify the nickname (NickName) to identify the certificate. A nickname must be specified to register the certificate that is used for SSL communications. Spaces cannot be specified at the start or end of the character string. Also, nicknames that are already being used cannot be specified.

Command Location

Windows	%CommonProgramFiles%\Fujitsu Shared \F3FSSMEE

Solaris	/opt/FJSVsmee/bin
Linux	/opt/FJSVsmee/bin
Linux x64	/opt/FJSVsmee64/bin

Cautions

- Certificates must be registered in sequence, starting from the root CA certificate.

- If a BASE64 format certificate is specified in the certificate file, and if the header and footer shown below are attached, that line is skipped. Note that, if the format is other than as shown below, the command ends with an error. Also, if the specified certificate file contains multiple certificates, only the first data is targeted.
  Header: -----Line starting with BEGIN
  Footer: -----Line starting with END
  An example of certificate data in BASE64 format is shown below:

  ```
  -----BEGIN CERTIFICATE-----
  ... (Certificate data encoded in Base64) ...
  -----END CERTIFICATE-----
  ```

- Either the -ca option or the -own option must be specified. If these are omitted, the certificate will be registered as another person's certificate, and therefore is invalid for Web servers.

12.3 cmentcrl Command

Synopsis

```
cmentcrl FileName [-ed Environment-directory ] [-nv]
```

Description

The cmentcrl command registers CRLs in the certificate/key management environment. Options that can be specified are shown below.

Options

FileName

Specify the full path of the filename (FileName) where the CRL data to be registered is stored.
DER and BASE64 format CRL files can be specified.

-ed Environment-directory

Specify the full path of the operation management directory (Environment-directory). If this option is omitted, the information set in the "CMIPATH" environment variable takes effect.

-nv

Specify this option if CRL verification is not performed during registration. If this option is specified, invalid CRLs can be registered, so do not specify this option.

Command Location

Windows	%CommonProgramFiles%\Fujitsu Shared \F3FSSMEE
Solaris	/opt/FJSVsmee/bin
Linux	/opt/FJSVsmee/bin
Linux x64	/opt/FJSVsmee64/bin

Cautions

- If there is an old CRL issued by the same Certificate Authority, the old CRL is deleted when the CRL is registered.

- The certificate of the CA that issued the CRL must already be registered.

- If a BASE64 format CRL is specified in the CRL file, and if the header and footer shown below are attached, that line is skipped. Note that, if the format is other than as shown below, the command ends with an error. Also, if the specified CRL file contains multiple CRLs, only the first data is targeted.
  Header: -----Line starting with BEGIN
  Footer: -----Line starting with END
  An example of CRL data in BASE64 format is shown below.

```
-----BEGIN X509 CRL-----
... (CRL data encoded in Base64) ...
-----END X509 CRL-----
```

12.4 cmentpfx Command

Synopsis

```
cmentpfx FileName [-ed Environment-directory ] {-sn slotID|-tl TokenLabel}
[-kl KeyLabel] -nn NickName [-entca] [-sncert]
[-userPIN UserPIN -password Password]
```

Description

This command registers, in the certificate/private key environment, the certificates and private keys stored in the PKCS#12(PFX) data file.
Input of the UserPIN and the password that decrypts PKCS#12(PFX) data is interactive.
Options that can be specified are shown below.

Options

FileName

Specify the full path of the filename that stores PKCS#12(PFX) data.

-ed Environment-directory

Specify the full path of the operation management directory (Environment-directory). If this option is omitted, the information set in the "CMIPATH" environment variable takes effect.

-sn slotID

Specify the slot ID of the token that registers the private key.

-tl TokenLabel

Specify the TokenLabel set for the token that registers the private key.

-kl KeyLabel

Specify an ASCII character string as the label attached to the private key being registered. If omitted, the private key is registered without a label.

-nn NickName

Specify the nickname (NickName). Spaces cannot be specified at the start or end of the character string.

-entca

If a CA certificate is included in the PKCS#12(PFX) data and that certificate is not yet registered, this option registers the CA certificate.

-sncert

Do not specify this option.

-userPIN UserPIN

Specify the User-PIN used to access the token. The space character cannot be specified. Note that a prompt asking for input of the User-PIN (User-PIN) is not displayed.
This option is not displayed in "Usage" when the command is executed.

-password Password

Specify the password that decrypts the PKCS#12(PFX) data. The space character cannot be specified. Note that a prompt asking for input of the password that decrypts the PKCS#12(PFX) data is not displayed.
This option is not displayed in "Usage" when the command is executed.

Command Location

Windows	%CommonProgramFiles%\Fujitsu Shared \F3FSSMEE
Solaris	/opt/FJSVsmee/bin
Linux	/opt/FJSVsmee/bin
Linux x64	/opt/FJSVsmee64/bin

Cautions

- Files created by the following product are PKCS#12(PFX) data file targets:
  Files exported from the Interstage certificate environment by the scsexppfx command

- If the root CA certificate is not registered, a verification error occurs and the certificate is not registered in the PKCS#12(PFX) data.

- CRLs included in the PKCS#12(PFX) data file are not handled.

- Tokens under the slot information directory specified for -sd in the cmsetenv command are the search targets for the token label specified in the -tl option.

- An exclusion error might occur if other applications access the token specified in the -tl option.

- Certificates in the certificate path being registered are registered under the character string of the nickname specified at -nn plus a 4-digit number. The 4-digit numbers are in sequence, starting from 0001. For example, if "nickname" is specified in -nn, from the top level of the path, the certificates are registered with the nicknames nickname0001, nickname0002, etc. Before registration, the cmlistcert command must be used to check that no certificates are already registered with the nickname specified by the user and the nickname plus 00*.

- If a certificate with the same nickname is already registered, or if the certificate is already registered with a different nickname, an error occurs.

12.5 cmlistcert Command

Synopsis

```
cmlistcert [-ed Environment-directory] [-c Country] [-cn CommonName]
[-o Organization] [-ou OrganizationUnit] [-ea EMailAddress]
[-ic Country] [-icn CommonName]
```

```
[-io Organization] [-iou OrganizationUnit] [-iea EMailAddress]
[-ca] [-own] [-nn NickName]
```

Description

The cmlistcert command outputs in the standard output the certificate information corresponding to the specified search key. If -c, -cn, -o, -ou, -ea, -ca, -own, and -nn are all omitted, a list of all the certificates is displayed. Options that can be specified are shown below.

Options

-ed Environment-directory

Specify the full path of the operation management directory (Environment-directory). If this option is omitted, the information set in the "CMIPATH" environment variable takes effect.

-c Country

Specify the country (Country).

-cn CommonName

Specify the alphanumeric name (CommonName).

-o Organization

Specify the alphanumeric organization name (Organization).

-ou OrganizationUnit

Specify the alphanumeric organization unit name (OrganizationUnit).

-ea EMailAddress

Specify the email address (EMailAddress).

-ic Country

Specify the country of the certificate issuer (Country).

-icn CommonName

Specify the alphanumeric name of the certificate issuer (CommonName).

-io Organization

Specify the alphanumeric organization name of the certificate issuer (Organization).

-iou OrganizationUnit

Specify the alphanumeric organization unit name of the certificate issuer (OrganizationUnit).

-iea EMailAddress

Specify the email address of the certificate issuer (EMailAddress).

-ca

Specify this option if the certificate to be registered is a certificate issued by the Certificate Authority.

-own

Specify this option if the certificate to be registered is a certificate issued for yourself.

-nn NickName

Specify the nickname (NickName). Spaces cannot be specified at the start or end of the character string.

Command Location

Windows	%CommonProgramFiles%\Fujitsu Shared \F3FSSMEE
Solaris	/opt/FJSVsmee/bin
Linux	/opt/FJSVsmee/bin
Linux x64	/opt/FJSVsmee64/bin

Cautions

- Only certificates registered in the certificate management directory can be displayed in the list.

- Item names are the same as the option names.

- An item is displayed only if information is set for the corresponding option shown above.

- Partial search is performed for the information corresponding to the specified options.

12.6 cmlistcrl Command

Synopsis

cmlistcrl [-ed Environment-directory] [-c Country] [-cn CommonName]
[-o Organization] [-ou OrganizationUnit] [-ea EMailAddress]

Description

The cmlistcrl command outputs in the standard output the CRL information corresponding to the specified search key.
If -c, -cn, -o, -ou, and -ea are all omitted, a list of all the certificates is displayed.
Options that can be specified are shown below.

Options

-ed Environment-directory

Specify the full path of the operation management directory (Environment-directory). If this option is omitted, the information set in the "CMIPATH" environment variable takes effect.

-c Country

Specify the country (Country).

-cn CommonName

Specify the alphanumeric name (CommonName).

-o Organization

Specify the alphanumeric organization name (Organization).

-ou OrganizationUnit

Specify the alphanumeric organization unit name (OrganizationUnit).

-ea EMailAddress

Specify the email address (EMailAddress).

<u>Command Location</u>

Windows	%CommonProgramFiles%\Fujitsu Shared \F3FSSMEE
Solaris	/opt/FJSVsmee/bin
Linux	/opt/FJSVsmee/bin
Linux x64	/opt/FJSVsmee64/bin

<u>Cautions</u>

- Only CRLs registered in the CRL management directory are targets for display in the list.

- Item names are the same as the option names.

- An item is displayed only if information is set for the corresponding option shown above.

- Partial search is performed for the information corresponding to the specified options.

12.7 cmmakecsr Command

<u>Synopsis</u>

```
cmmakecsr [-ed Environment-directory] -sd Slot-directory -tl TokenLabel
-of OutFile [-f {TEXT|NOHEAD|V2} ] [-c Country] [-cn CommonName]
[-o Organization] [-ou OrganizationUnit] [-ea EMailAddress]
[-t Title] [-tel Phone] [-l Locality] [-s State] [-sa {SHA1|MD5} ]
{ -kl KeyLabel | [-kt RSA] [-kb {512|768|1024|2048} ] } [-p UserPIN]
```

<u>Description</u>

The cmmakecsr command creates a certificate application that has the specified information and outputs it to file. If a key pair is newly created to create the application, either -kt or -kb must be specified. It is recommended to specify 2048 bits in the -kb option.

<u>Options</u>

-ed Environment-directory

Specify the full path of the operation management directory (Environment-directory).
If this option is omitted, the information set in the "CMIPATH" environment variable takes effect.

-sd Slot-directory

Specify the full path of the slot information directory (Slot-directory).

-tl TokenLabel

Register the existing key to be used or register a newly created key. Specify the token label (TokenLabel).

-of OutFile

Specify the full path of the application output file (OutFile).

-f {TEXT|NOHEAD|V2}

Specify the output format.
TEXT : Output is in CSR format. (default)
NOHEAD: Output is without a header.
V2 : Output is in S/MIME format with an application/pkcs10 header attached.

-c Country

Specify the country (Country).

-cn CommonName

Specify the alphanumeric name (CommonName).

-o Organization

Specify the alphanumeric organization name (Organization).

-ou OrganizationUnit

Specify the alphanumeric organization unit name (OrganizationUnit).

-ea EMailAddress

Specify the email address (EMailAddress).

-t Title

Specify the title (Title).

-tel Phone

Specify the telephone number (Phone).

-l Locality

Specify the locality name (Locality).

-s State

Specify the state name (State).

-sa {SHA1|MD5}

Specify the signature algorithm.
SHA1: Use SHA1. (default)
MD5 : Use MD5.

-kl KeyLabel

Specify the label of the key to be used (KeyLabel).

-kt RSA

If a new key is being created, specify the key type.
RSA: Creates an RSA encryption algorithm key pair. (default)

-kb {512|768|1024|2048}

If a new key is being created, specify the key length.
512: Key length is 512 bits (default)
768: Key length is 768 bits
1024: Key length is 1024 bits

2048: Key length is 2048 bits (recommended)

At present, 512 and 768 bit RSA encryption algorithm keys are no longer safe due to improvements in machine processing performance and similar, so do not use them. Also, 1024 bit key is not recommended to use. A specification of 2048 bits is recommended because key lengths are relevant to server security. If 512, 768 or 1024 bit RSA encryption algorithm keys need to be used for operational reasons, be aware of this risk.

-p UserPIN

Specify the User-PIN used to access the token. The space character cannot be specified. Note that a prompt asking for input of the User-PIN is not displayed.
This option is not displayed in "Usage" when the command is executed.

Command Location

Windows	%CommonProgramFiles%\Fujitsu Shared \F3FSSMEE
Solaris	/opt/FJSVsmee/bin
Linux	/opt/FJSVsmee/bin
Linux x64	/opt/FJSVsmee64/bin

Cautions

- For -cn, specify the Web server host name.

- If either the -kt or -kb option is specified, a new key pair is created.

- If the -kl option is specified, the key with the specified label is used.

- Any one of the -c, -cn, -o, -ou, -ea, -t, -tel, -l, and -s options must be specified. Check with the Certificate Authority that you are asking to issue the certificate about items that must be specified.

- If requesting issue of a certificate from VeriSign, Inc., select "Secure Site".

- If requesting issue of a certificate from Cybertrust, Inc., select "SureServer for SSL Certificates".

12.8 cmmkenv Command

Synopsis

```
cmmkenv Environment-directory [-fromdir ValidCert-directory,CRL-directory]
-todir ValidCert-directory,CRL-directory
```

Description

The cmmkenv command creates the directories required for operation of the certificate/key management environment and sets the initial settings.
Options that can be specified are shown below.

Options

Environment-directory

Specify the full path of the operation management directory name (Environment-directory).

-fromdir ValidCert-directory,CRL-directory

Do not specify this option.

-todir ValidCert-directory,CRL-directory

To operate a new ordinary file, specify the directory name.

ValidCert-directory : Specify the full path of the certificate directory.

CRL-directory : Specify the full path of the CRL directory.

Command Location

Windows	%CommonProgramFiles%\Fujitsu Shared \F3FSSMEE
Solaris	/opt/FJSVsmee/bin
Linux	/opt/FJSVsmee/bin
Linux x64	/opt/FJSVsmee64/bin

Cautions

- The operation management directory name specified in this command is also specified when executing other commands. This applies to the following commands:

    - cmdspcert

    - cmentcert

    - cmentcrl

    - cmentpfx

    - cmlistcert

    - cmlistcrl

    - cmmakecsr

    - cmmkpfx

    - cmrmcert

    - cmsetenv

- Ensure that a comma is used to separate ValidCert-directory and CRL-directory when specified in -todir. Do not use a space or any other character.
  Example: -todir /export/home/SSL/sslcert/cert,/export/home/SSL/sslcert/crl

- The same directory cannot be specified as ValidCert-directory and CRL-directory when specified in -todir.

12.9 cmmkpfx Command

Synopsis

```
cmmkpfx FileName [-ed Environment-directory] {-sn slotID|-tl TokenLabel}
-nn NickName [-eeonly] [-userPIN UserPIN]
[-password Password]
```

Description

The cmmkpfx command fetches the EE certificates (including certificates in the path) and private keys corresponding to the specified nickname from the certificate management environment and the key management environment, and outputs a PKCS#12(PFX) data file.

Note that the password that encrypts the UserPIN and PKCS#12(PFX) data is input interactively.

Options that can be specified are shown below.

Options

FileName

Specify the full path of the file name (FileName) that stores the PKCS#12(PFX) data.

Environment-directory

Specify the full path of the operation management directory name (Environment-directory).

-sn SlotID

Specify the slot ID (SlotID) of the token that contains the certificates and keys.

-tl TokenLabel

Specify the TokenLabel set for the token that contains the certificates and keys.
An error occurs if there is more than one of the same label.

-nn NickName

Specify the nickname (NickName).

-eeonly

Specify this option to create PKCS#12(PFX) data that does not include certificates in the path. (Only the certificate specified by the nickname and the corresponding private key are obtained.)
If the PKCS#12(PFX) data created with this option specified is registered in a different environment, a different procedure must be used to register certificates in the path (root CA certificates, etc.).

-userPIN UserPIN

Specify the User-PIN used to access the token.

-password Password

Specify the password that encrypts the PKCS#12(PFX) data.
Spaces cannot be specified.

Command Location

Windows	%CommonProgramFiles%\Fujitsu Shared \F3FSSMEE
Solaris	/opt/FJSVsmee/bin
Linux	/opt/FJSVsmee/bin
Linux x64	/opt/FJSVsmee64/bin

Cautions

- Use from 6 to 128 characters from the character sets below to specify the password.

Category	Characters
Alphabetic	A to Z a to z
Numeric	0 to 9
Symbols	!"#%&'()*+,-./:;<=>?[\]^_{|}~
Space	' '

- Manage passwords in a way that prevents leaks and theft.
  Also, do not set character strings that consist of easily guessed names or words or in which all the same characters are used. Character strings that use a mix of alphanumerics and symbols and are as long as possible are recommended.

- Certificates and private keys are not deleted from the certificate/key management environment when they are obtained.

- A parameter error occurs if the -userPIN and -password options are not specified at the same time.

- If the -userPIN and -password options are specified, the prompt asking for input of the UserPIN and password that encrypts the PKCS#12(PFX) data is not displayed.

- Tokens under the slot information directory specified for -sd in the cmsetenv command are the search targets for the token label specified in the -tl option.

- An exclusion error might occur if other applications access the token specified in the -tl option.

- The certificates targeted to be obtained are the certificates corresponding to the specified nickname and the certificates in the certificate path that is required for verifying that certificate.

- The command ends with an error if the private key corresponding to the certificate with the specified nickname does not exist.

- The command ends with an error if there are multiple certificates with the specified nickname. Use the cmlistcert command to check and, if you want to obtain the certificate, use the cmchgnickname command to change the nickname.

- If the certificate does not exist in the token specified in the -sn | -tl option, the certificates in the certificate management environment are searched.

12.10 cmrmcert Command

Synopsis

```
cmrmcert [-ed Environment-directory ] { -id CertID | -nn NickName }
```

Description

The cmrmcert command deletes certificates from the certificate/key management environment. Options that can be specified are shown below.

Options

-ed Environment-directory

Specify the full path of the operation management directory (Environment-directory). If this option is omitted, the information set in the "CMIPATH" environment variable takes effect.

-id CertID

Specify the certificate ID (CertID).

-nn NickName

Specify the nickname (NickName).

Command Location

Windows	%CommonProgramFiles%\Fujitsu Shared \F3FSSMEE
Solaris	/opt/FJSVsmee/bin
Linux	/opt/FJSVsmee/bin
Linux x64	/opt/FJSVsmee64/bin

Cautions

- Either -id or -nn must be specified.

- The cmlistcert command can be used to check the certificate ID (CertID) and nickname.

12.11 cmsetenv Command

Synopsis

```
cmsetenv Environment-directory -sd Slot-directory -jc LocaleCode
```

Description

The cmsetenv command sets the locale-language code system required for registering a Slot Information Directory, certificate, and CRL into a Certificate/Key Management Environment.

The options and arguments of this command are as follows:

Options

Environment-directory

Specify the full path of the operation management directory name (Environment-directory).

-sd Slot-directory

Specify the full path of the slot information directory (Slot-directory). Specify the directory name that was specified when the makeslot or mkslt command was used to create the slot.

-jc LocaleCode

Specify 0 (that is, UNICODE) as locale code required to register a certificate and CRL.

Command Location

Windows	%CommonProgramFiles%\Fujitsu Shared \F3FSSMEE
Solaris	/opt/FJSVsmee/bin
Linux	/opt/FJSVsmee/bin
Linux x64	/opt/FJSVsmee64/bin

Cautions

- The slot (makeslot) and token (maketoken) creation commands must be executed before the -rc option is specified.

- If a certificate is being registered in the certificate management environment when the -rc option is specified, check that the certificate and nickname are not duplicated. At that time, if the certificate and nickname are both the same, this command will end normally without the certificate being registered. If the same certificate is already registered with a different nickname, or if a different certificate is already registered with the same nickname, this command will end with an error.

12.12 ihsregistupin Command

Synopsis

```
ihsregistupin -f upinfile -d slotdir
```

Description

This command encrypts the user PIN and creates a user PIN management file.
Input of a user PIN is requested when this command is executed. Enter the same character string as the user PIN that was used when the certificate application was created. Note that the entered user PIN character string is not displayed.
The options and parameters of this command are shown below.

Options

-f upinfile

Specify the user PIN management file that stores the encrypted user PIN.

-d slotdir

Specify the absolute path or relative path of the slot information directory in the private key management environment.

Command Location

Windows	%F4AN_INSTALL_PATH%\F4ANswnc\F3FMshs\bin
Solaris	/opt/FJSVshs/bin
Linux	/opt/FJSVshs/bin

Privilege Required

System administrator (user belonging to the Administrators group or superuser) privileges are required.

Cautions

- This command must be executed with administrator privileges.

- For the user PIN, specify the same character string as the user PIN that was used when the certificate application was created.

- The user PIN management file is overwritten and saved.

- In order to prevent user PINs travelling on networks and being leaked externally, it is recommended to execute this command from a local console, or from a terminal connected via a LAN that cannot have external connections.

12.13 makeslot Command

Synopsis

```
makeslot [ -d Slot-directory ]
```

Description

The makeslot command generates slots.
The slot password to be set for a slot is entered interactively.
The set slot password is required when generating and embedding a token.
If slot generation is successful, the slot ID of the created slot is output. The slot ID is the information used to identify a slot and is specified by the maketoken command that is executed next.
Options that can be specified are shown below.

Options

-d Slot-directory

Specify the full path of the directory name that stores the slots that manage keys. Also specify the slot information directory that is specified here when executing the maketoken and cmsetenv commands.

Command Location

Windows	%ProgramFiles%\SecurecryptoLibraryR\PROGRAM\bin
Solaris	/opt/FJSVsclr/bin
Linux	/opt/FJSVsclr/bin
Linux x64	/opt/FJSVsclr64/bin

Cautions

- Use from 6 to 128 characters from the character sets below to specify the slot password. The command ends abnormally if the specified character string does not fulfill these conditions.

Category	Characters
Alphabetic	A to Z a to z
Numeric	0 to 9
Symbols	!"#%&'()*+,-./:;<=>?[\]^_{|}~
Space	' '

- Manage slot passwords in a way that prevents leaks and theft.
  Also, do not set character strings that consist of easily guessed names or words or in which all the same characters are used. Character strings that use a mix of alphanumerics and symbols and are as long as possible are recommended.

Example

[Windows]

```
> makeslot -d d:\sslenv\slot
New Slot-password: (*)
Retype: (*)
makeslot: Succeeded. New Slot-ID is 1.
```

[Solaris] [Linux]

```
# makeslot -d /export/home/slot
New Slot-password: (*)
Retype: (*)
makeslot: Succeeded. New Slot-ID is 1
```

*) The entered slot password is not displayed.

12.14 maketoken Command

Synopsis

```
maketoken [ -d Slot-directory ] -s SlotID [ -t TokenLabel ]
```

Description

The maketoken command generates a token and embeds it in a slot.

The SO-PIN and user PIN to be set in the token are entered interactively.

Options that can be specified are shown below.

Execution of this command generates an FLM03 device model token.

To enable the generated token in an application program that is already started, the application program must be restarted.

Options

-d Slot-directory

Specify the full path of the slot information directory (Slot-directory).

An error occurs if the specified slot information directory does not exist and the command ends abnormally.

-s SlotID

Use the slot ID option (SlotID) to specify the slot where the token being generated will be embedded.

Specify the slot ID that was displayed when the makeslot command was executed.

-t TokenLabel

Specify 32 characters or less as the token label (TokenLabel) to be set for the generated token.

A token label is the character string information to identify a token.

If less than 32 characters are specified, the label is padded with spaces.

The space character cannot be specified at the start of the character string.

If this option is omitted, Token + lower two digits of the slot number (if the slot number is 1, Token01) is assumed as the specification.

Command Location

Windows	%ProgramFiles%\SecurecryptoLibraryR\PROGRAM\bin
Solaris	/opt/FJSVsclr/bin
Linux	/opt/FJSVsclr/bin
Linux x64	/opt/FJSVsclr64/bin

Cautions

- For the token label, specify up to 32 characters from the character sets shown below. If the specified character string does not fulfill these conditions, the command ends abnormally.

   For a SO-PIN and User-PIN, specify between 6 and 128 characters from the character sets shown below.

Category	Characters
Alphabetic	A to Z a to z
Numeric	0 to 9
Symbols	!"#%&'()*+,-./:;<=>?[\]^_{|}~
Space	' '

- Manage passwords in a way that prevents leaks and theft.

   Also, for a SO-PIN or User-PIN, do not set character strings that consist of easily guessed names or words or in which all the same characters are used. Character strings that use a mix of alphanumerics and symbols and are as long as possible are recommended.

- The specified token label and User-PIN might need to be specified when building service environments or starting service operations, so do not forget them.

Example

[Windows]

```
maketoken -d d:\sslenv\slot -s 1 -t token
Slot-password: (*1)
New SO-PIN for token: (*2)
Retype: (*2)
New User-PIN for token: (*3)
Retype: (*3)
```

[Solaris] [Linux]

```
# maketoken -d /export/home/slot -s 1 -t token
Slot-password: (*1)
New SO-PIN for token: (*2)
Retype: (*2)
New User-PIN for token: (*3)
Retype: (*3)
```

*1) Enter the slot password that was entered when the makeslot command was executed. The slot password is not displayed.

*2) Enter to set the SO-PIN (administrator's PIN) in the token. The SO-PIN is not displayed.

*3) Enter to set the user PIN in the token. The user PIN is not displayed.

Chapter 13 Maintenance Commands

This chapter explains the commands that are required for troubleshooting and maintenance of Systemwalker Operation Manager.

📄 **Note**

. . . .

If commands are executed in an environment where the UAC function for Windows Server 2008 or later has been enabled, then a dialog box asking a user's privileges to be promoted may be displayed depending on the privileges of the user that logged in to the operating system, and action may be required.

Refer to "Notes on Executing Commands in Windows [Windows version]" in the *Systemwalker Operation Manager User's Guide* for more information.

. . . .

13.1 mpipcso IPC Resource Status Display Command [UNIX]

Synopsis

```
mpipcso
```

Description

This command displays information about the IPC resources that are currently being used by Systemwalker Operation Manager and the IPC resources that had been used but could not be recovered.

Return value

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Command Location

/opt/systemwalker/bin

Cautions

- When checking IPC resources, use the system command ipcs together to avoid deleting incorrect IPC resources.

- This command should be executed on the Systemwalker Operation Manager Server by the system administrator (superuser).

Example

```
Mpipcso
```

Output Example

An example of command output is shown below:

```
# /opt/systemwalker/bin/mpipcso
IPC status from<running Systemwalker> as of Tue Apr 11 10:27:44 2006
T       ID    KEY        DATE       TIME      PID     CMD
s    131075 0x102029b  2006/04/11 10:23:01  11473 MpFwsec
s    131076 0x102029c  2006/04/11 10:23:01  11473 MpFwsec
```

```
Status of undeletion IPC from<stopping Systemwalker>
T        ID    KEY        DATE        TIME     PID   CMD         DETECTION DATE
s     65536 0x1020297 2006/04/10 22:16:46     611 MpFwsec     2006/04/11 10:23:01
s     65537 0x1020299 2006/04/10 22:16:47     611 MpFwsec     2006/04/11 10:23:01


IPC status from<running Systemwalker Centric Manager> as of Tue Apr 11 10:27:44 2006
T        ID    KEY        DATE        TIME     PID   CMD
m       209 0x2097c   2006/04/11 10:23:24   11798 opagtd
m       111 0x1020b46 2006/04/11 10:24:01   12969 mppmon
s        10 0x1020b48 2006/04/11 10:24:00   12969 mppmon


m       108 0x32027da8 2006/04/11 10:23:46   12697 drmsdemon
m       109 0x64027da8 2006/04/11 10:23:46   12697 drmsdemon
s     65541 0x35027da8 2006/04/11 10:23:46   12697 drmsdemon
s         9 0x62027da8 2006/04/11 10:23:46   12697 drmsdemon
s     65542 0x63027da8 2006/04/11 10:23:46   12697 drmsdemon
s         8 0x66027da8 2006/04/11 10:23:46   12697 drmsdemon
q       912 0x30027da8 2006/04/11 10:23:46   12697 drmsdemon
q       115 0x31027da8 2006/04/11 10:23:46   12697 drmsdemon
q       116 0x36027da8 2006/04/11 10:23:46   12697 drmsdemon
q       117 0x38027da8 2006/04/11 10:23:46   12697 drmsdemon
q       118 0x39027da8 2006/04/11 10:23:46   12697 drmsdemon
q       119 0x61027da8 2006/04/11 10:23:46   12697 drmsdemon
q       120 0x65027da8 2006/04/11 10:23:46   12697 drmsdemon
s     65543 0x67027da8 2006/04/11 10:23:46   12697 drmsdemon


Status of undeletion IPC from<stopping Systemwalker Centric Manager>
T        ID    KEY        DATE        TIME     PID   CMD         DETECTION DATE
m         3 0x1020b47 2006/04/10 22:17:37    1946 mppmon      2006/04/11 09:57:34
s     65538 0x1020b46 2006/04/10 22:17:37    1946 mppmon      2006/04/11 09:57:34


IPC status from<running Systemwalker Operation Manager> as of Tue Apr 11 10:27:45 2006
T        ID    KEY        DATE        TIME     PID   CMD
m        10 0x1020fbf 2006/04/11 10:25:07   14002 tskmnsrv
q        67 0x2020fbf 2006/04/11 10:25:10   14210 tskussrv
q       180 0xa020fbf 2006/04/11 10:25:10   14209 tskbasrv
q       121 0x5020fbf 2006/04/11 10:25:10   14212 tskgrsrv
q       122 0x7020fbf 2006/04/11 10:25:10   14213 tskevsrv
m       112 0x1020833 2006/04/11 10:24:50   13613 mpjmsrv


Status of undeletion IPC from<stopping Systemwalker Operation Manager>
T        ID    KEY        DATE        TIME     PID   CMD         DETECTION DATE

There is no IPC resource of the collection leakage.

#
```

IPC status from<running Systemwalker> as of Tue Apr 11 10:27:44 2006

Meaning: This field shows the IPC resources shared by Systemwalker Centric Manager and Systemwalker Operation Manager, which are running. The date and time at which the mpipcso command was executed is indicated.

Status of undeletion IPC from<stopping Systemwalker>

Meaning: This field shows the IPC resources shared by Systemwalker Centric Manager and Systemwalker Operation Manager, which could not be recovered in the past.

IPC status from<running Systemwalker Centric Manager> as of Tue Apr 11 10:27:44 2006

Meaning: This field shows the IPC resources used by Systemwalker Centric Manager, which is running. The date and time at which the mpipcso command was executed is indicated.

Status of undeletion IPC from <stopping Systemwalker Centric Manager>

Meaning: This field shows the Systemwalker Centric Manager IPC resources that could not be recovered in the past.

```
IPC status from<running Systemwalker Operation Manager> as of Tue Apr 11 10:27:45
2006
```

Meaning: This field shows the IPC resources used by Systemwalker Operation Manager, which is running. The date and time at which the mpipcso command was executed is indicated.

```
Status of undeletion IPC from<stopping Systemwalker Operation Manager>
```

Meaning: This field shows the Systemwalker Operation Manager IPC resources that could not be recovered in the past.

```
T ID KEY DATE TIME PID CMD
```

Meaning: From left to right in this field shows type, ID, key, date of retrieval, time of retrieval, process ID, and process name

```
T ID KEY DATE TIME PID CMD DETECTION DATE
```

Meaning: From left to right in this field shows type, ID, key, date of retrieval, time of retrieval, process ID, process name, and date on which the lack of recovery was recognized.

```
There is no IPC resource of the collection leakage.
```

Meaning: There is no IPC resource that has not been recovered.

13.2 mppviewo Process Running Status Display Command

Synopsis

[Windows]

```
mppviewo.exe [-e]
```

[UNIX]

```
mppviewo [-e]
```

Description

This command displays information about the running status of Systemwalker Operation Manager processes to the standard output.

Option

-e

Specify to display only a process in which an error has occurred.

Return value

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Command Location

Windows	Systemwalker Operation Manager installation directory\mpwalker.jm\bin
Solaris HP-UX AIX Linux	/opt/systemwalker/bin

Privilege Required

System administrator (user belonging to the Administrators group or superuser) privileges are required.

Cautions

- Functions in the following conditions are not displayed in the list:

    - Functions that have not been installed

    - Functions whose startup is disabled

    - Functions to which monitoring is disabled

- In mixed environments that include Systemwalker Centric Manager, the following process information is displayed by the process running status display command (mppviewc) of Systemwalker Centric Manager instead of by this command:

    - Systemwalker MpAosfP

    - Systemwalker MpAosfB

    Refer to the *Systemwalker Centric Manager Reference Guide* for information on the process running status display command (mppviewc) of Systemwalker Centric Manager.

Output Example

This command shows the process running status as follows. In this example, the process "mjsdaemon -sys 1" is in abnormal status:

```
##################### PROCESS INFORMATION BEGIN#####################
** Package  Name **
PROCESS-NAME                                     PID         RUNNING-TIME

** FJSVfwseo **
MpFwsec                                          2345        0:19

** FJSVjmcal **
f3crhcs2                                         2241        0:18

** FJSVJMCMN **
jmnetsv                                          2245        0:15
mpjmsrv                                          2265        0:17

** FJSVJOBSC **
[Subsystem0]
tskmnsrv                                         2745        0:14
[Subsystem1]
tskmnsrv                                         2749        0:12
[Subsystem2]
tskmnsrv                                         2751        0:09

** FJSVMJS **
[Subsystem0]
mjsdaemon -sys 0                                 2821        0:21
mjsdaemon -sys 0                                 2822        0:20
mjsdaemon -sys 0                                 2823        0:19
mjsnetsv 0                                       2825        0:18
[Subsystem1]
mjsdaemon -sys 1                                 2921        0:20
mjsdaemon -sys 1                                 2923        0:19
>>>>> ERROR:Process NOT Found : mjsdaemon -sys 1
mjsnetsv 0                                       2925        0:17
[Subsystem2]
mjsdaemon -sys 2                                 3024        0:15
mjsdaemon -sys 2                                 3025        0:14
mjsdaemon -sys 2                                 3026        0:14
```

```
mjsnetsv 0                                                  3016            0:12
:
##################### PROCESS INFORMATION END   #####################
```

13.3 swcolinf Maintenance Information Collection Command (for Server) [UNIX]

Synopsis

```
[Collection of maintenance information]
  swcolinf [-i Name] -o OutPath [-c Comment]
[Setting of generation management for maintenance information]
  swcolinf -w Count
[List display of maintenance information]
  swcolinf -1 OutPath
```

Description

This command collects information required for troubleshooting of Systemwalker Operation Manager. The number of generations of information to be collected can be defined to enable generation management. In addition, information on collected resources (collection date, collected function, etc.) can be displayed by generation with the list display option.

If no environment has been created due to installation failure etc., this information can be collected using commands in the product media.

Options

-i *Name*

Select a function used to collect information. The functions that can be specified in the Name option are listed below. When specifying more than one function, delimit the functions with a comma. You must specify "all" alone. It cannot be specified with other functions.

Normally, specify "all" to collect information about all Systemwalker Operation Manager functions. Collect information about a specific function only when the function to be collected is clarified or a Fujitsu engineer gives instructions to do so.

If this option is omitted, information about all the functions is collected just like specifying the -i all option.

When using commands in the product media, only "all" can be specified.

all

   Collects information about all functions.

framework

   Collects fundamental information, such as information on security and scripts.

server

   Collects information about the functions installed on servers, such as Jobscheduler, Job Execution Control, Calendar, and Power Control.

tool

   Collects information about installation, backup, and maintenance tools.

-o *OutPath*

Specify the directory in which the collected information is to be stored with a full path using alphanumeric characters with up to 50 bytes for OutPath. The full path must not contain any blanks.

-c *Comment*

Specify a comment, if needed, to be attached to the information that is collected.

Specify a comment with up to 50 bytes using alphanumeric characters and "_". If this option is omitted, no comment is attached to the collected information.

-w *Count*

Specify the number of generations of information to be stored. Specify a number with range from 1 to 10. The initial value is set to 3 generations.

This option is not available when using commands in the product media.

If you do not want generation management, set the value to 1.

-l *OutPath*

Specify to display a list of the collected information. Specify for OutPath the directory that contains the collected information. The dates of collection, collection functions, and comments in the collected information stored under the specified directory are listed by generations.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Command Location

Solaris	/opt/FJSVftlo/swcolinf
HP-UX	/opt/FJSVftlo/swcolinf
AIX	/opt/FJSVftlo/swcolinf
Linux	/opt/FJSVftlo/swcolinf
Product media (Solaris)	<*drive*>/Solaris/tool_unx/swcolinf
Product media (Linux)	<*drive*>/Linux/tool_unx/swcolinf

Privilege Required

System administrator (superuser) privileges are required.

Cautions

- If collection of information about a function fails, a failure message is displayed. Manually copy the applicable file to the storage destination as directed in the window.

- Since this is a silent command, if an error occurs, that error is recorded to the following log.

  - For command option error: Specify the correct option
  - **[Solaris]**

    ```
    /var/tmp/swcolinf.log
    ```

  - **[HP-UX, AIX, Linux]**

    ```
    /tmp/swcolinf.log
    ```

  - For other errors: Copy the failed file to the storage destination manually

    ```
    Storage destination directory/OperationManager/Generation number/swcolinf.log
    ```

Example 1

The following example collects server function and tool information, and stores it in /work:

```
/opt/FJSVftlo/swcolinf/swcolinf -i server.tool -o /work
```

Example 2

The following example collects information about all functions and stores it with a comment attached, in /work:

```
/opt/FJSVftlo/swcolinf/swcolinf -i all -o /work -c 2002.2.2_all_SE001
```

Example 3

The following example displays a list of information about the collected data stored under /work:

```
/opt/FJSVftlo/swcolinf/swcolinf -l /work
```

Execution Result

```
*** Systemwalker Operation Manager Information ***
OUTPATH: /work
FOLDER                          DATE                    FUNCTION    COMMENT
/work/Operation Manager/0001    2002/05/05 10:10:10     all         all
/work/Operation Manager/0002    2002/05/06 10:10:10     tool        tool :error
```

The output items are explained below:

OUTPATH:

Specified directory that contains the collected information

FOLDER:

Generation directory in which information is stored

DATE:

Date of collection

FUNCTION:

Function for which information was collected

COMMENT:

Comment specified by the command

13.4 swcolinf Maintenance Information Collection Command

Synopsis

```
[Collection of maintenance information]
  swcolinf.exe [/i Name] /o OutPath [/c Comment]
[Collection of maintenance information using GUI]
  swcolinf.exe
[Setting of generation management for maintenance information]
  swcolinf.exe /w Count
```

Description

This command collects information required for troubleshooting of Systemwalker Operation Manager. The number of generations to be retained can be specified to enable management of generations of information to be collected.

If no environment has been created due to installation failure etc., this information can be collected using commands in the product media.

In the Server Core environment, execute the swcolinf.exe command without specifying any options, to collect the maintenance information using GUI.

Options

/i *Name*

Select a function used to collect information. The functions that can be specified in the Name option are listed below. When specifying more than one function, delimit the functions with a comma. You must specify "all" alone. It cannot be specified with other functions.

Normally, specify "all" to collect information about all Systemwalker Operation Manager functions. Collect information about a specific function only when the function to be collected is clarified or a Fujitsu engineer gives instructions to do so.

If this option is omitted, information about all the functions is collected just like specifying the /i all option.

When using commands in the product media, only "all" can be specified.

all

    Collects information about all functions.

framework

    Collects fundamental information, such as information on security and scripts.

network

    Collects network-related information.

event

    Collects event monitoring and action management information.

auto [Windows]

    Collects calendar and power control information.

server [Windows]

    Collects information about the functions installed on servers, such as the Jobscheduler, Job Execution Control, Backup Link.

client

    Collects information about the Systemwalker Operation Manager client function, Task Link, etc.

tool

    Collects information about installation, backup, maintenance tools, etc.

/o *OutPath*

Specify the directory in which the collected information is to be stored by using a full path with up to 50 bytes for OutPath. The full path must not contain any blanks.

/c *Comment*

Specify a comment, if needed, to be attached to the information that is collected.

Specify a comment with up to 50 bytes with alphanumeric characters and "_". A comment cannot contain any blanks and number signs (#). If this option is omitted, no comment is attached to the collected information.

/w *Count*

Specify the number of generations of information to be stored. Specify a number with range from 1 to 10. The initial value is set to 3 generations.

This option is not available when using commands in the product media.

If you do not want generation management, set the value to 1.

None

The **Maintenance Information Collection** window will be displayed. In the Server Core environment, execute the command without specifying any options.

Return Values

Return value	Meaning
0	Normal termination
Value other than 0	Abnormal termination

Command Location

Windows	Systemwalker Operation Manager installation directory\mpwalker.jm\mpcmtool\swcolinf
Product media (Windows x86)	*<drive>*\MAIN\tool\swcolinf
Product media (Windows x64)	*<drive>*\Client\tool\swcolinf
Product media (UNIX client)	*<drive>*\Client\tool\swcolinf

Privilege Required/Execution Environment

- System administrator (user belonging to the Administrators group) privileges are required.

- When executing this command on Windows operating systems, do it on a Command Prompt opened with administrator privileges.

Cautions

- Since this is a silent command, if an error occurs, that error is recorded to the following log.

    - For command option error: Specify the correct option

    Windows Directory\swcolinf.log

    - For other errors: Copy the failed file to the storage destination manually

    Storage Destination Directory\OperationManager\Generation number\swcolinf.log

- In the environment where Systemwalker Operation Manager and Systemwalker Centric Manager coexist, collect data on the earlier version product with the Maintenance Information Collection Tool of the later version product.

Example 1

The following example collects server function and tool information, and stores it in c:\tmp.

```
c:\win32app\mpwalker.jm\mpcmtool\swcolinf\swcolinf.exe /i server,tool /o c:\tmp
```

Example 2

The following example collects information about all functions and stores it with a comment attached, in c:\tmp.

```
c:\win32app\mpwalker.jm\mpcmtool\swcolinf\swcolinf.exe /i all /o c:\tmp /c
2002.2.2_all_SE001
```

Example 3

The following example collects maintenance information in the Server Core environment using GUI.

```
c:\win32app\mpwalker.jm\mpcmtool\swcolinf\swcolinf.exe
```

13.5 mpenvset Operation Environment Maintenance Wizard Startup Command [Windows]

Synopsis

```
mpenvset.exe
```

Description

This command starts up the Operation Environment Maintenance Wizard, and backs up/restores the operation environment. The **Operation Environment Maintenance Wizard** window will be displayed when this command is executed. By performing the operations according to the window, it is possible to back up and restore the Systemwalker Operation Manager environment.

Return value

0	Normal termination
Value other than 0	Abnormal termination

Command Location

Windows	<*Systemwalker Operation Manager installation directory*>\mpwalker.jm \mpcmtool\common

Privilege Required/Execution Environment

- System administrator (user belonging to the Administrators group) privileges are required.

- This command can be executed on a server.

Example

<*Systemwalker Operation Manager installation directory*>\mpwalker.jm\mpcmtool
\common\mpenvset.exe

Chapter 14 JCL (Job Control Language)

This chapter explains the Systemwalker Operation Manager JCL (Job Control Language).

JCL (Job Control Language) is a language suitable for Job Execution Control that allows easy handling of complex job procedures. It takes the job step concept and displays the job handling in a job step configuration.

14.1 JCL Overview

This section provides an overview of JCL.

14.1.1 JCL Functions

JCL takes the job step concept and displays the job handling in job step configuration.

It activates programs (executable programs, batch files, shell scripts, etc.) for job steps.

The parameters specified on job submission are carried over into program.

Job steps can be executed selectively by executing the next job step using the job step completion code.

Exclusive batch jobs can be performed.

JCL files cannot be executed as a network job.

14.1.2 JCL Configuration

JCL consists of a group of control statements for total job and control statements related to job steps (to explain the configuration and formats, parameters of control statements are simplified).

JCL configuration is shown below.

```
jobstart  job_name                                              ..... Start of job
host  client_host_name
user  user_name
resource  resource_name
qprty  job_priority_in_queue
dprty  job_execution_priority
path  path_to_executable_program [Windows]
path  path_to_shell_script_or_command [UNIX]
file  file_or_printout_information_used_in_job
    stepstart  step_name,step_execution_condition
    path  path_to_executable_program_in_step [Windows]
    path  path_to_shell_script_or_command_in_step [UNIX]
    file  file_or_printout_information_used_in_step          step
    exec  executable_program_name [Windows]
    exec  shell_script_name [UNIX]
    stepend
    stepstart
        .
        :                                                    step
        :
    stepend
jobend                                                          ..... End of job
```

The JCL format when a network job is submitted is shown below.

Figure 14.1 Job Step Configuration of Network Job

Network job control file
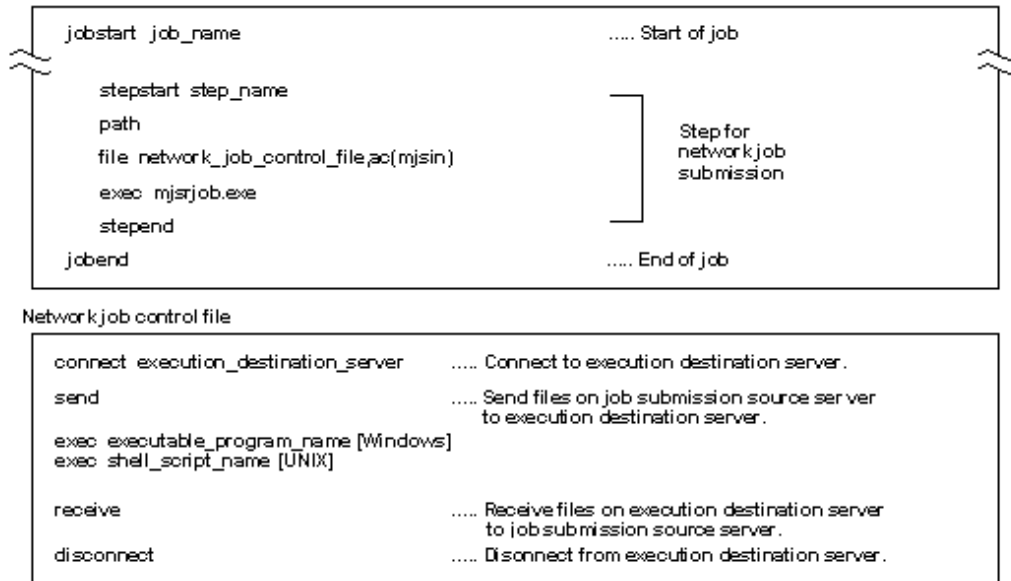
14.1.3 JCL Protocol

JCL protocol is explained below. A list of control statements is also shown below:

```
Control Statement   Operand
```

Control Statement

- Enter from the first column onwards.

- Entry must be made from the first column in some control statements (see table below).

- More than one control statement cannot be entered in one row.

- Rows cannot be continued.

Control Statement	Meaning	Description location
jobstart	Start of job	Description on the first line only. Be sure to enter description from the first column.
jobend	End of job	Description on the last line only. Be sure to enter description from the first column.
stepstart	Start of job step	Be sure to enter description from the first column.
stepend	End of job step	Be sure to enter description from the first column.
host	Client host name	Enter from the first column onwards.
user	Client user name	Enter from the first column onwards.
resource	Resource name and exclusive attributes	Enter from the first column onwards.
qprty	Queuing priority ranking	Enter from the first column onwards.
dprty	Job execution priority	Enter from the first column onwards.
path	Path in which program exists	Enter from the first column onwards.
file	Files used by entire job	Enter from the first column onwards.
exec	Execution of program	Enter from the first column onwards.

Operands

- Leave at least one space between control statements.

- When specifying several operands, separate them with a comma (,).

- Variable symbols may also be included.

  The variable symbols are represented by &n, where n is the nth parameter specified during job submission.

Comments

- If "*" or "#" is input in the first column, that row becomes a comment.

- Comments cannot be inserted in the middle of a control statement row.

14.2 Control Statements for Total Job

This section explains the control statements for the entire job:

14.2.1 jobstart Statement

Synopsis

```
jobstart [job name] [,prt(printer name) [,op] [,oh] [,nop] ]
        [,joblst({0|1})]
```

Description

This statement means the start of a job and must be entered before any other JCL control statements.

It specifies whether job output will be batch or not, and whether job output is to be suspended or not.

Operands

job name

Specify the job name. The job name can be specified with up to 64 bytes.

If the -j option is specified in the qsub command, the job name specified in the -j option is used.

For scheduled jobs where [Job name] is specified in the [Standard information] sheet of the [Add/Change - Job] window, the job name specified in [Job name] is used.

For demand jobs where [Job Name] is specified in the [Standard information] sheet of the [Edit Job Information/Submit] dialog box/window, the job name specified in [Job Name] is used.

The job file name is used if this operand is omitted and none of the following are specified:

- The -j option of the qsub command

- [Job name] in the [Standard information] sheet of the [Add/Change - Job] window

- [Job Name] in the [Standard information] sheet of the [Edit Job Information/Submit] dialog box/window

prt (printer name)

Specify the printer name for outputting JCL's standard output, standard error output, and job list. If the oh, op or nop operand is specified, this operand is necessary.

The printer name specified in this operand will be used as the default value of the prt operand in file control statement. Therefore, if you want to omit the printer name in each file control statement, this operand must be specified.

The printer name to be specified must satisfy the following requirements.

- 32 bytes or less

- "&", "(" and ")" are not included

The printer name can be specified by using its actual name or the alias which has been defined on the [Print Format] sheet in the [Define Operating Information] window of Job Execution Control. [Windows]

op

Specify when outputting the standard output file, standard error output file, and job list file for job batch output.

oh

Specify when changing the job output status to hold after job termination. If this operand is not specified, an output request is automatically issued after job termination and the job is deleted.

nop

If there is a job output (if the prt operand is specified in JCL), after the job output is completed, the job is not deleted but saved as an output held job in the server where Systemwalker Operation Manager is installed.

Held jobs can be output again from the client window or by performing hold release operation with the command on the server. After re-execution, the job will be saved on the server on which Systemwalker Operation Manager is installed.

By specifying this operand, jobs saved on the server on which Systemwalker Operation Manager is installed will not be deleted from this server unless you perform the deletion. Delete unwanted jobs from the client window or by performing deletion operation with the command on the server.

joblst({0|1})

If joblst(1) is specified, the job list will be output. If joblst(0) is specified or the joblst operand is omitted, the job list will not be output.

The job list file will be created with the following names with the same path as the standard output file:

```
... \job name.ljob number
```

The following information is output in the job list:

- Job execution start date and time

- Job step execution start date and time

- Job step execution completion date and time

- Job step completion code

- Job execution completion date and time

- Job completion code

- Network job start/completion date and time

- Network job file transfer volume

The following is an example of job list output:

```
MJS800I MJSRJOB JOB CONTROL START 10/14.11:11: 1
MJS802I CONNECT TO nomonomo
MJS804I send1 SENT 8BYTES
MJS805I JOB START ON nomonomo 10/14.11:11: 3
MJS806I JOB STOP ON nomonomo 10/14.11:11:18 CODE(0)
MJS804I STDOUT RECEIVED 18KB
MJS804I STDERR RECEIVED 0BYTES
MJS804I recv1 RECEIVED 9BYTES
MJS803I DISCONNECT TO nomonomo
MJS801I MJSRJOB JOB CONTROL STOP 10/14.11:11:19
```

In this example, the job name is "updatejob", the output destination is "printer1", and a standard output/standard error output has been specified. When the job ends, the output will not be printed immediately. The output is placed on hold, and even if the hold is later cancelled and the output is printed, the job will not be purged but put on output hold again.

```
jobstart updatejob,prt(printer1),op,oh,nop,joblst(1)
```

14.2.2 host Statement

Synopsis

```
host host name
```

Description

This statement specifies client host name.

Operand

host name

Specify client host name.

The client host name should be specified with up to 64 alphanumeric bytes.

If the -ch option is specified in the qsub command, the client host name specified in the -ch option is used.

Example

```
host konzert
```

konzert: Host name

14.2.3 user Statement

Synopsis

```
user user name
```

Description

This statement specifies client user name.

Operand

user name

Specify client user name.

The client user name can be specified with up to 64 bytes of alphanumeric characters.

If the -cu option is specified in the qsub command, the client user name specified in the -cu option is used.

For scheduled jobs, the project owner is used for the client user name. However, for scheduled jobs where [Effective user name] is specified in the [Detail information] sheet of the [Add/Change - Job] window, the user specified in [Effective user name] is used for the client user name.

Example

```
user s7025
```

s7025: User name

Cautions

Only the system administrator (user belonging to the Administrators group or superuser) can specify this control statement.

14.2.4 resource Statement

Synopsis

```
resource resource name [,{sh|ex}]
```

Description

This statement specifies the resource name used by the relevant job and exclusive attributes.

If there is more than one resource used by the job, describe this control statement using multiple lines.

The maximum number of this control statements to be entered is as follows:

[Windows]

- 16 statements

[UNIX]

- 256 statements

Operands

resource name

Specify resource names used by the job.

The resource name can be specified with up to 64 bytes of alphanumeric characters, underscores (_), and hyphens (-).

If the -R option is specified in the qsub command, the resource information specified in the -R option is used.

For scheduled jobs where [Resource name] is specified in the [Detail information] sheet of the [Add/Change - Job] window, the resource information specified in [Resource name] and [Resource type] is used.

For demand jobs where [Resource Used] is selected in the [Additional Information] sheet of the [Edit Job Information/ Submit] dialog box/window, the resource information specified in [Resource Used] and [Type] are used.

{sh|ex}

Specify exclusivity attributes. "sh" indicates shared attribute and "ex" indicates exclusive attribute.

If omitted, sh is assumed.

Example

```
resource RDB,ex
```

RDB: Resource name

ex: Exclusive attribute

14.2.5 qprty Statement

Synopsis

```
qprty job execution priority
```

Description

This statement specifies job execution priority in a queue.

job execution priority

Specify the execution priority of jobs in a queue using a number with the range of 0 to 63.

The greater the value is, the higher the priority becomes.

When specifying the -p option in the qsub command, the execution priority specified in the -p option is used.

For demand jobs where [Set Execution Priority] is selected in the [Additional Information] sheet of the [Edit Job Information/ Submit] dialog box/window, the execution priority specified in [Set Execution Priority] is used.

Example

```
qprty 32
```

32: Job execution priority

14.2.6 dprty Statement

Synopsis

```
dprty program execution priority specified in exec statement
```

Description

This statement specifies the job execution priority

Operands

program execution priority specified in exec statement

[Windows]

Specify the program's thread priority already defined in the exec statement using a number with the range of 0 to 4.

A larger value has the higher thread priority.

Each of the values (from 0 to 4) has the following meaning:

Specified value	Priority class	Thread priority
4	NORMAL	HIGHEST
3	NORMAL	ABOVE_NORMAL
2	NORMAL	NORMAL
1	NORMAL	BELOW_NORMAL
0	NORMAL	LOWEST

If the -dp option is specified in the qsub command , the execution priority specified in the -dp option is used.

For scheduled jobs where [Priority] is selected in the [Detail information] sheet of the [Add/Change - Job] window, the execution priority specified in [Priority] is used.

For demand jobs where [Set Dispatch Priority] is selected in the [Additional Information] sheet of the [Edit Job Information/ Submit] dialog box/window, the execution priority specified in [Set Dispatch Priority] is used.

[UNIX]

Specify the execution priority between -20 and 19. The lower the value, the higher the priority in the CPU allocation. When specifying the -dp option in the qsub command, the execution priority specified in the -dp option is used

For scheduled jobs where [Priority] is selected in the [Detail information] sheet of the [Add/Change - Job] window, the execution priority specified in [Priority] is used.

For demand jobs where [Set Dispatch Priority] is selected in the [Additional Information] sheet of the [Edit Job Information/Submit] dialog box/window, the execution priority specified in [Set Dispatch Priority] is used.

The value that can be specified is equivalent to the nice value.

"0" represents the standard priority. The negative values show higher priority and the positive values show lower priority. Accordingly, "-20" is the highest priority and "19" is the lowest priority.

Example

```
dprty 4
```

4: Job execution priority

14.2.7 path Statement

Synopsis

```
path path name
```

Description

This statement specifies a path to be added to the PATH environment variable.

By specifying multiple path control statements, multiple paths can be specified.

A path specified in a path control statement is added to the top of the PATH environment variable.

Describe path control statements between the jobstart control statement and first stepstart statement.

Operand

path name

Specify a path to be added to the environment variable.

Define the path name length so that the following does not exceed 4096 bytes.

- "Total path name length specified in the path control statement for the entire job" + "Number of path control statements for the entire job"

Examples

[Windows]

```
path C:\win32app\local
```

C:\win32app\local: Path of executable program

[UNIX]

```
path /home/user1/bin
```

/home/user1/bin: Path of shell script or command

14.2.8 file Statement

Synopsis

[Windows]

```
file [file name],ac (access name) [,prt({printer name | *})
    [,prtform([fontname=font name] [,fontsize=font size]
            [,orient={port | land}] [,form=paper size])]]
```

[UNIX]

```
file [file name],ac(access name) [,prt ({printer name | *})]
```

Description

This statement specifies the file information used by the total job.

By specifying the prt operand, the file can be printed.

Operands

file name

Specify the file name of an input/output file used in the executable program or batch file [Windows]/shell script [UNIX].

The specified file name is set to the same name as the environment variable of the access name. Even if the specified file name contains other environment variable "%character strings%" [Windows]/"$character strings" [UNIX], it is not set to the other environment variable name. The executable program or batch file [Windows] or shell script [UNIX] specified in the exec control statement can obtain the file name from the environment variable.

This operand can be omitted only when the prt operand is specified.

If this operand is omitted, Job Execution Control will create a temporary work file. A unique file name is set for this temporary work file on the system. Note that this temporary work file is deleted when the job is deleted from the job queue after the file contents have been output to the printer.

ac (access name)

Specify any access name. The file name is set in the environment variable with the same name as the specified access name.

Access names must meet the following conditions:

- No more than 64 bytes

- Only alphanumeric characters and "_" (underscores) are used

- The first character is not a numeral

prt ({printer name|*})

After the job has completed, the contents of the file specified in the first argument of this statement are output to the printer. However, only the contents of text file can be output.

If the printer name is specified, it is output to the specified printer. The printer name must meet the following requirements:

- 32 bytes or less

- "&", "(" and ")" are not included

If (*) is specified, the file is output to the printer that is specified in the prt operand of the jobstart statement.

When specifying the printer name, the name defined in the [Print Format] sheet of the [Define Operating Information] window can be used. [Windows]

prtform [Windows]

Specify the attributes related to print format. Make sure to specify one or more sub operands. For the specification value of the sub operand, specify values allowed in the Print Manager for Windows.

When the attributes related to print format are specified, the print format is determined as follows:

- If the printer defined in the prt operand of the file statement is already defined in the [Print Format] sheet of the [Define Operating Information] window for Job Execution Control

  The print form defined in the above sheet and the one specified in this operand will be merged.

- If the printer defined in the prt operand of the file statement is not defined in the [Print Format] sheet of the [Define Operating Information] window for Job Execution Control

  The print form specified in this operand will be used.

When this operand is omitted, the print format is determined as follows:

- If the printer specified in the prt operand of the file statement is already defined in the [Print Format] sheet of the [Define Operation Information] window for Job Execution Control

  The print form defined in the above sheet will be used.

- If the printer specified in the prt operand of the file statement is not defined in the [Print Format] sheet of the [Define Operating Information] window for Job Execution Control

  The defaults (in Windows' definition) of the printer will be used.

  fontname=font name

  Specify the font name.

  If this operand is omitted, the font name specified in initialization parameter "prt" will be used.

  fontsize=font size

  Specify the font size. The unit is measured in points.

  If this operand is omitted, the font size specified in initialization parameter "prt" will be used.

  orient={port|land}

  Specify the print orientation. For portrait printing, specify port, and for landscape printing, specify land.

  If this operand is omitted, the print orientation specified in initialization parameter "prt" will be used.

  form=paper size

  Specify the paper size.

  If this operand is omitted, the paper size specified in initialization parameter "prt" will be used.

Examples

[Windows]

In the following example, the access name "out1" is assigned to the file called "d:\result\data1" (from job process, the file name can be obtained by referring to the environment variable "out1") and the file is set to be output to "prt1" after the job has completed. The output attributes are the font name "system," font size "10 points", and print orientation "landscape."

```
file d:\result\data1,ac(out1),prt(prt1),
prtform(fontname=system,fontsize=10,orient=port)
```

[UNIX]

In the following example, the access name "output1" is assigned to the file called "/result/data1" (from job process, the file name can be obtained by referring to the environment variable "output1") and the file is set to be output to "printer1" after the job has completed.

```
file /result/data1,ac(output1),prt(printer1)
```

The file name can be omitted as follows.

```
file ,ac(output1),prt(printer1)
```

14.2.9 jobend Statement

Synopsis

```
jobend
```

Description

This control statement signifies the end of a job. No other JCL control statements can be described after this control statement row.

This command has no operands.

Example

```
jobend
```

14.3 Control Statements for Job Steps

This section explains the control statements for job steps:

14.3.1 stepstart Statement

Synopsis

```
stepstart job step name{ [,skip (cc{ <| = |> }code
              [, {jobend|job step name}])]
              | [,{even|only|every}]}
```

Description

This statement signifies the start of a job step. The step from this control statement to the next specified stepend control statement is considered to be a single job step.

Using the completion code of the relevant job step, the job step that is to be executed next can be specified.

A maximum number of 186 job steps can be defined in a single JCL file.

Operands

job step name

Specify the job step name.

The job step name can be specified with up to 16 bytes of alphanumeric characters.

skip (cc{<|=|>}code [,{jobend|job step name}])

Specify the next job step to be executed according to the job step completion code. If the condition (cc{<|=|>}code) is met, the control will move to the job step specified in the branching job step ({jobend|job step name}); therefore the job step located next to the job step currently running on JCL will not be executed.

Up to 32 skip statements can be specified simultaneously.

Condition (cc{<|=|>}code)

The "cc" indicates the completion code of the current job step, using a fixed character string.

When specifying the "code," use the completion code value with an integer from 0 to 256.

To terminate a job when cc is smaller than code, specify "cc<code"; when they are equal, specify "cc=code"; when cc is greater than code, specify "cc>code."

Branching job step ({jobend|job step name})

When the specified conditions are satisfied, specify to which job step the control is transferred, using the step name or fixed character string "jobend."

To terminate a job, specify "jobend" or omit this specification.

If a step name located before the current job step is specified, it will end with an error and the job will not be executed.

The control, when moved to the specified job step, will execute the job steps normally from top to bottom. In this case, the code value will be inherited.

even|only|every

Specify one of "even", "only", or "every."

The job step to which this operand is specified should be specified as the last job step of a job.

If this operand is specified when only a single job step exists, the operand will be ignored.

- When "only" is specified:

  The current job step will be executed only when an application error has occurred in any of the preceding job steps.

- When "even" is specified:

  The current job step will be executed regardless of whether the preceding job steps ended abnormally.

  However, if forced termination is performed during preceding job execution or request for stopping a service/daemon is issued, the relevant job step will not be executed.

- When "every" is specified:

  The current job step will always be executed when the job terminates (in all cases of normally, abnormally, and forcibly). But when the service/daemon is not running, the current job step cannot be started. Forced termination cannot be accepted during execution of the current job step. If a service/daemon stop request is received while executing the current job step, the current job step will be halted.

Caution

Abnormal termination of a job with exception code is considered an application error.

Example

The following is an example showing that the current job step "step1" ends with a value greater than 0. The control is moved to the job step named "recovery."

```
stepstart step1,skip(cc>0,recovery)
```

14.3.2 path Statement

Synopsis

```
path path name
```

Description

This statement specifies a path to be added to the PATH environment variable.

By specifying multiple path control statements, multiple paths can be specified.

A path specified in a path control statement is added to the top of the PATH environment variable.

Describe path control statements between the jobstart control statement and first stepstart statement.

When the path statement is specified for the control statement related to the total job, it is added after the path specified in this path statement.

Operand

path name

Specify a path name to be added to the environment variable.

For each job step, define the path name length so that the following does not exceed 4096 bytes.

- "Total path name length specified in the path control statement for the job step" + "Number of path control statements for the job step"

Examples

[Windows]

```
path c:\users\user1\bin
```

C:\users\user1\bin: Path of executable program

[UNIX]

```
path /users/user1/bin
```

/users/user1/bin: Path of shell script or command

14.3.3 file Statement

Synopsis

[Windows]

```
file {[file name],ac(access name)|* ,ac(access name),dlm(delimiter)}
      [,prt({printer name|* })]
      [,prtform([fontname=font name] [,fontsize=font size]
             [,orient={port|land}][,form=paper size])]]
```

[UNIX]

```
file {[file name],ac(access name) | *,ac(access name),dlm(delimiter)}
     [,prt ({printer name | *})]
```

Description

This statement specifies the file used by the program executed by this step.

By specifying the prt operand, the file can be printed.

Operands

[file name],ac (access name)

file name

Specify the file name of an input/output file used in the executable program or batch file [Windows]/shell script [UNIX].

The specified file name is set to the same name as the environment variable of the access name. Even if the specified file name contains other environment variable "%character strings%" [Windows]/"$character strings" [UNIX], it is not set to the other environment variable name. The executable program or batch file [Windows] or shell script [UNIX] specified in the exec control statement can obtain the file name from the environment variable.

This operand can be omitted only when the prt operand is specified.

If this operand is omitted, Job Execution Control will create a temporary work file. A unique file name is set for this temporary work file on the system. Note that this temporary work file will be deleted when the job is deleted from the job queue after the file contents have been output to the printer.

ac (access name)

Specify any access name. Access names must meet the following conditions:

- No more than 64 bytes

- Only alphanumeric characters and "_" (underscores) are used

- The first character is not a numeral

When submitting a network job, specify the file name that stores the command string controlling the network job in the previously mentioned "file name", and then specify "mjsin" as the access name.

See "Figure 14.1 Job Step Configuration of Network Job" for details on job step configuration when a network job is submitted.

*,ac(access name), dlm(delimiter)

*

If "*" is specified, the row from the next row of this control statement to the row immediately before the delimiter appears, are stored in a temporary file created by Job Execution Control.

If "*" is specified in this control statement, it is used when executable program or batch file [Windows], or shell script [UNIX] obtains the input parameter from the file.

There is no need to create a separate file for input parameters since the input parameters can be described in JCL.

dlm (delimiter)

Specify any character string that signifies the last row of data in JCL. Rows up to immediately before the delimiter string specified in this operand appears are saved in a temporary file created by Job Execution Control.

Specify the delimiter character at the start of a row. Rows starting with anything other than delimiter characters, such as spaces, cannot be handled as the final row of data.

The delimiter should be specified with up to eight bytes of alphanumeric characters and it is case sensitive.

prt{printer name|*}

After the job has completed, the contents of the file specified in the first argument of this control statement are output to the printer. However only text file can be output.

If the printer name is specified, it is output to the specified printer. The printer name must meet the following two requirements.

- 32 bytes or less

- "&", "(" and ")" are not included

If (*) is specified, the file is output to the printer specified in the prt operand of the jobstart statement.

Another printer name specified in the [Print Format] sheet of the [Define Operating Information] window for Job Execution Control can also be specified. [Windows]

prtform [Windows]

Specify the attribute related to print format. Make sure to specify one or more sub operands. For the specification value of the sub operand, specify values allowed in the Print Manager for Windows.

When the attributes related to print format is specified, the print format is determined as follows:

- If the printer defined in the prt operand of the file statement is already defined in the prt operand in the operating information file of the Job Execution Control function

The print form defined in the relevant prt operand of the operating information file and the print form specified in this operand will be merged.

- If the printer defined in the prt operand of the file statement is not defined in the prt operand in the operating information file of the Job Execution Control function

The print form specified in this operand will be used.

When this operand is omitted, the print form is determined as follows:

- If the printer defined in the prt operand of the file statement is already defined in the prt operand in the operating information file of the Job Execution Control function

The print form defined in the prt operand in the operating information file will be used.

- If the printer defined in the prt operand of the file statement is not defined in the prt operand in the operating information file of the Job Execution Control function

The defaults (in Windows' definition) of the printer will be used.

fontname=font name

Specify the font name.

If this operand is omitted, the font name specified in initialization parameter "prt" will be used.

fontsize=font size

Specify the font size. The unit is measured in points.

If this operand is omitted, the font size specified in initialization parameter "prt" will be used.

orient={port|land}

Specify the print orientation. For portrait printing, specify port, and for landscape printing, specify land.

If this operand is omitted, the print orientation specified in initialization parameter "prt" will be used.

form=paper size

Specify the paper size.

If this operand is omitted, the paper size specified in initialization parameter "prt" will be used.

Examples

[Windows]

In the following example, the access name "out1" is assigned to the file called "d:\result\data1" (the file name can be obtained by referring to the environment variable "out1" from job process) and the file is set to be output to "prt1" after the job has completed. The output attributes are the font name "system," font size "10 points", and print orientation "portrait."

```
file d:\result\data1,ac(out1),prt(prt1),
prtform(fontname=system,fontsize=10,orient=port)
```

[UNIX]

In the following example, the access name "output1" is assigned to the file called "/result/data1" (the file name can be obtained by referring to the environment variable "output1" from job process) and the file is set to be output to the "printer1" after the job has completed.

```
file /result/data1,ac(output1),prt(printer1)
```

When omitting the file name in the above, describe it as follows:

```
file ,ac(output1),prt(printer1)
```

If "*" is specified in the file name, inline development of any data in JCL is specified. Below is an example of using "ZZZ" as a delimiter string to indicate the end of inline development.

```
file *,ac(indata1),dlm(ZZZ)
....
....
....
ZZZ
```

....: Optional data

14.3.3.1 connect Command

Synopsis

```
connect host(host name)
```

Description

This command establishes connections to another server to which job execution is requested.

Operand

host name

Specify the host name of another server to which job execution is requested. The host name is the host name defined by the network set up with TCP/IP configuration.

If the configuration has not been changed, the host name is the string with which the computer name is converted into lower case characters. [Windows]

Example

```
connect host(host1)
```

host1: Host name of another server to be connected

14.3.3.2 send Command

Synopsis

```
send access name [,file(send destination file name) ]
```

Description

This command sends the files on the server, which has submitted the job to another server specified in the connect command.

Operands

access name

Specify the access name of the file statement specifying the file to send. If an access name that is not described in JCL is specified, an error occurs and the job is terminated.

file(send destination file name)

Specify the destination file name with a full path. If this operand is specified, a file is created with the specified name, and after the job execution is completed, the file won't be deleted.

If this operand is omitted, the file is sent as a temporary file, and the access from the job is handled normally (the temporary file name is set in the access name and environment variable). When the job execution is completed, the temporary file is deleted.

Examples

[Windows]

In the following example, the file specified in the file statement in which access name "data1" is defined is sent to "d:\userinfo\data1" specified in the file operand.

```
send data1,file(d:\userinfo\data1)
```

[UNIX]

In the following example, the file specified in the file statement in which access name "data1" is defined is sent to "/userinfo/data1" specified in the file operand.

```
send data1,file(/userinfo/data1)
```

14.3.3.3 receive Command

Synopsis

```
receive access name [,file(receive source file name) ]
```

Description

This command receives the file specified with the connect command from another server on the local server where the job was submitted.

Operands

access name

Specify the access name of the file statement that specifies the file to be received. If an access name that is not described in JCL is specified, an error occurs and the job is terminated.

file(receive source file name)

Specify the file name of a file that exists on the host connected with the connect command with full path. The specified file will be received.

If this operand is omitted, the file corresponding to the access name will be received.

Examples

[Windows]

The following example receives a file specified in the file statement in which access name "list1" is defined from "d:\userinfo\list1" specified in the file operand.

```
receive list1,file(d:\userinfo\list1)
```

[UNIX]

The following example receives a file specified in the file statement in which access name "list1" is defined from "/userinfo/list1" specified in the file operand.

```
receive list1,file(/userinfo/list1)
```

14.3.3.4 execute Command

Synopsis

```
execute (execution file name execution option) [,queue(queue name)]
```

Description

This command executes jobs on another server specified by the connect command, according to the specifications in the operands.

Operands

execution file name execution option

Specify the execution file name and execution options to be carried out on another server.

Specifying the qsub command as the job file name, a free format network job can be requested. Note that at least one space must be placed between the execution file name and the execution options.

queue(queue name)

Specify on which queue the job requested to another server will be executed.

If this operand is omitted, the queue is determined as follows:

1. The queue that has the name identical to the execution queue of the submission source job (queue name specified at submission or default name).

2. If a queue of the same name does not exist, the queue that has the name defined in "Specify default queue to submit jobs to" (dfltqueue operand of the system parameter) in the [Define Operating Information] window.

3. The job submission fails if "Specify default queue to submit jobs to" (dfltqueue operand in the system parameter) is not selected in the [Define Operating Information] window.

Example

The following example executes application "appl1.exe" on queue "queue1."

```
execute (appl1.exe),queue(queue1)
```

14.3.3.5 disconnect Command

Synopsis

```
disconnect
```

Description

This command disconnects another server that has been connected with the connect command.

This command can be omitted. If omitted, the other server is automatically disconnected at the end of the job step or when the next connect command is executed.

Note that if the disconnect command is executed when there is no connection, the command is ignored and processing continues.

This command has no operands.

14.3.4 exec Statement

Synopsis

```
exec  program name [argument ...]
```

Description

This statement specifies the program or batch file [Windows]/shell script name [UNIX] and arguments to be executed.

Operands

program name

Specify the name of a program to be executed. Batch file name [Windows]/shell script name [UNIX] can also be specified.

arguments...

When an argument is passed to the program to be executed, at least one space should be put between the program name and the argument.

By separating arguments by space, multiple arguments can be set.

When spaces are included in an argument, enclose that argument in quotations ("" or "").

If the arguments are specified as &n, the arguments specified in the qsub command are passed to the program. The n in this case is the nth argument specified in the qsub command.

When executing a network job

When submitting a network job, specify the program name and arguments in the following format:

```
mjsrjob.exe [-r number of retries] (*)
```

*:

Describe only the file name of mjsrjob.exe (full path is not necessary).

-r number of retries

When a network abnormality occurs while requesting a network job execution (executing connect, send or execute commands), a retry is performed. The number of retries can be specified with any number from 1 through 20.

While the execute command is being processed, if a network error occurs in the status that awaits the completion of a job after job execution request has been issued, no retry will be performed.

When submitting a job, if retry processing is specified in the [Additional Information] sheet in the [Edit Job Information/ Submit] dialog box/window of the [Systemwalker Operation Manager] window or the retry option is specified in the qsub command, these specifications take priority.

P Point

The return value of mjsrjob.exe is the equivalent of the return value from an execution file specified by the execute command in the network job control file.

Examples

[Windows]

```
exec appl01.exe work1 &1 &2 &3
```

appl01.exe: Executable program name

[UNIX]

```
exec script1 work1 &1 &2 &3
```

script1: Shell script name

work1: 1st argument

&1: 1st argument specified by qsub command

&2: 2nd argument specified by qsub command

&3: 3rd argument specified by qsub command

14.3.5 stepend Statement

Synopsis

```
stepend
```

Description

This statement specifies the end of job step.

This command has no operands.

```
stepend
```

14.4 Relationships between JCL and Programs

This section explains the relationship between JCL and programs:

14.4.1 path Statement and Program Relationship

The path statement can be used in the control statements for the total job or job steps, and specifies the path to the location of the program.

Regarding searches of a program, it is performed from the path statement for the job steps. If multiple paths are specified in the path statement for the job step or job, the search is performed in the specified order.

An example of the search order is show below.

[Windows]

```
jobstart s7025job
path c:\system\appl    -------- 3
path c:\system\applx   -------- 4

stepstart step1
path c:\user\appl      -------- 1
path c:\user\applx     -------- 2
```

[UNIX]

```
jobstart   s7025job
path /usr/local/bin    -------- 3
path /home/s7025/bin   -------- 4

stepstart step1
path /home/s7025/bin1  -------- 1
path /home/s7025/bin2  -------- 2
```

14.4.2 file Statement and Program Relationship

When manipulating files in programs, the access name defined in JCL can be used. The file name defined in JCL is stored in the environment variable, which has the same name as the access name. That is, the file name is set in the user-defined environment variable.

📖 Note

If the environment variable whose name is the same as the access name in the file statement is set to the startup file of a shell (".profile", ".cshrc", ".login" etc.), the setting of the shell's startup file will be validated (since the setting order is "file statement" first and then "shell's startup file").

14.4.3 exec Statement and Program Relationship

The program whose name is specified in the exec statement is activated. If an argument is specified after the program, the argument is passed to the program.

When "exec appl.exe host1 s7025" is specified in JCL, the arguments of the program are as follows:

1st argument:

host1

2nd argument:

s7025

14.5 JCL/Job Step Completion Codes

This section explains the JCL and job step completion codes:

14.5.1 JCL Completion Code

The largest job step completion code in the job steps entered in JCL is the JCL completion code. However, if the job execution is interrupted, the JCL completion code takes a special value.

Also you can specify the completion code of the job step executed last as the JCL completion code in the [Options] sheet of the [Define Operating Information] window.

See "14.5.2 Job Step Completion Code" for details on the completion code when an executing job is interrupted.

14.5.2 Job Step Completion Code

The return code in the program specified in the exec control statement of the job step becomes the completion code of that job step. For job steps that are not executed according to the execution conditions in the stepstart control statement, there is no completion code.

You can set job step delimiters (step name of job step and completion code) to be output to standard output for the job list by defining settings in the [Options] sheet of the [Define Operating Information] window.

Chapter 15 Systemwalker Script

This chapter explains Systemwalker Scripts (Job Control Language) provided by Systemwalker Operation Manager.

15.1 Overview of Systemwalker Script

This section explains the overview of Systemwalker Scripts.

15.1.1 Functions of Systemwalker Script

Systemwalker Scripts are a new job control language provided by Systemwalker Operation Manager.

Systemwalker Scripts, being based on Tcl (Tool Command Language)/Tk (Tool kit), are platform independent.

Systemwalker Scripts are treated and executed as jobs.

Using the syntax and commands described in this chapter, you can create any jobs with Systemwalker Scripts. For debugging Systemwalker Scripts, commands that run scripts and handle trace information are provided.

15.1.2 Configuration of Systemwalker Script

File Name

A file with the extension "swt" is treated as a Systemwalker Script.

```
~.swt
```

Configuration

A Systemwalker Script consists of a group of command statements. The descriptive contents of scripts are within your discretion as long as you follow the syntax and command forms explained in this chapter. Shown below is the descriptive format of command statements.

```
command   optional argument
```

A script file is a group of command statements as shown below:

```
Command statements
 ...
 ...
 ...
 ...
```

Startup Arguments

When activating a script, you can assign information to be processed within a script as an argument. The startup arguments specified in conjunction with a script filename can be handled in a script by using the following reserved variables.

argc: Stores the number of arguments.

argv: Stores the contents of arguments.

A reference is made to an argument after it is broken down into work variables before being retrieved.

Example:

The following example retrieves the first argument into the work variable "prm1."

```
set prm1 [lindex $argv 0]
```

Description:

set: A command that stores a value in a variable.

prm1: The name of a variable that stores a value.

lindex: A command that retrieves a startup argument.

argv: A variable in which a startup variable is stored (it is identified by the dollar sign, $, prefixed to it)

0: Specifies a value (location of argument to retrieve - 1).

Environment Variables

The environment variables have been already stored in the reserved variable "env."

In a script, each environment variable name following "env" is treated as a parenthesized (enclosed in ()) variable name and referring to and updating environment variables is made on env (a name of environment variable).

Example:

A variable name used to refer to and update PATH environment variable.

```
env(PATH)
```

15.1.3 Syntax of Systemwalker Script

This section describes the syntax of Systemwalker Scripts.

Variables

Variable names can be formed with any number of alphanumeric characters beginning of an alphabet. The variable names are case-sensitive and may include underbars (_).

Syntax

The syntax and interpretation of the Systemwalker Script language are defined as follows.

Separating commands

Commands in script lines are separated by line breaks or semicolons (;).

Example:

The following example stores a value in each of the variables abc and def:

```
set abc ABC
set def DEF
```

or

```
set abc ABC ; set def DEF
```

Variable substitution using dollar sign ($)

Variable substitution uses the dollar sign ($). Variable substitution means that referring to values stored in variables. Variable substitution attempted on a variable whose value is not defined results in a script error, which will cause an abnormal termination.

Example:

The following example stores the value stored in variable abc in other variable def:

```
set abc ABC
set def $abc
```

Command substitution using brackets ([ ])

Some script commands generate return values. Command substitution corresponds to referring to a return value at a command caller when a command is called. Command substitution uses brackets ([]).

Example:

The following example stores the result of arithmetic operation (3 + 2) in variable abc:

```
set abc [ expr 3 + 2 ]
```

Separating words in a command

Words in a command are separated by spaces or tabs. Separation of words in a command indicates that of command name, first argument, second argument and so on.

Variable and command substitutions within a word

When a dollar sign ($) is included in a word, variable substitution is performed, namely the word is replaced by the value of the variable. When brackets ([ ]) are included in a word, command substitution is performed and the word is replaced by the return value of the result of a command.

Command substitution is recursively performed, while variable substitution is done only once.

Example:

The following example recursively performs command substitution and stores the result of computation:

```
set abc [ expr [expr 1 + 1] + 2 ]
```

Note:

Since the expr command is capable of interpreting nested expressions, a math expression should not be written as indicated in the above example, but should be written as follows.

```
set abc [ expr (1 + 1) + 2 ]
```

Specifying a word using double quotations ("")

Words between two double quotations ("") are treated as a single word. Words with missing close double quotations result in a syntax error. The double quotation pair itself is not interpreted as part of a word.

Example:

The following example stores character string 123 in variable abc:

```
set abc "123"
```

The double quotation proceeded by a backslash (\) is treated as an ordinary character rather than being interpreted as a word enclosing symbol. For substitutions of backslash mark, see "Escape sequence (backslash-sign substitution) using backslash sign (\)".

Example:

The following example stores character string 123" in variable abc:

```
set abc "123\""
```

Although the semicolons (;), close brackets (]), and spaces (including line breaks) included in double quoted words are treated as ordinary characters, command and variable substitutions are performed.

Example:

The following example stores a character string containing spaces and variable substitution:

```
set abc "123"
set str "abc $abc def"
```

In this instance, the character string to be stored in variable str is abc 123 def.

Specifying a word using braces ({})

Words between two braces ({}) are treated as a single word. Words with missing closing braces result in a syntax error. The brace pair itself is not interpreted as part of a word.

Example:

The following example stores character string 1{2}3 in variable abc:

```
set abc {1{2}3}
```

The brace preceded by a backslash (\) is not interpreted as a word enclosing symbol. "\{" and "\}" included in a braced word will not be replaced by "{" and "}", respectively.

For details, see "Word enclosing symbols and backslash-sign substitutions".

Example:

The following example stores a character string containing braces:

```
set abc {123\}}
```

Here, character string 123\} is stored in variable abc.

The semicolons (;), close brackets (]), and spaces (including line breaks) included in braced words are treated as ordinary characters, and command and variable substitutions are not performed.

Example:

The following example stores a character string containing spaces:

```
set abc "123"
set str {abc $abc def}
```

Here, the character string to be stored in variable str is abc $abc def.

Escape sequence (backslash-sign substitution) using backslash sign (\)

If a word contains a backslash sign (\), the combination of \ and the character that follows is interpreted as an escape sequence and backslash-sign substitution is performed.

Shown below are types of backslash-sign substitution.

Sequence	Replaced by
\a	Bell
\b	Backspace
\f	Form feed
\n	Line break
\r	Carriage return
\t	Tab
\v	Vertical tab
\<line break>	Blank character (continuation line)
\"	Single double quotation that is not interpreted as a word enclosing symbol
\{	Single open brace that is not interpreted as a word enclosing symbol
\}	Single closing brace that is not interpreted as a word enclosing symbol
\\	Single backslash sign "\"

Note:

Any backslash sign must not be followed by any character other than the above.

Word enclosing symbols and backslash-sign substitutions

While all \s included in double quoted words are replaced by their corresponding characters or symbols, those included in braced words are not replaced except for "\<line break>."

Example:

When \ included in a braced word is not replaced:

```
set abc "C:\\tmp"
set def {C:\tmp}
```

Here, character string C:\tmp is stored in both variables abc and def.

Example:

When \ included in a braced word is replaced:

```
set abc "123\
456"
set def {123\
456}
```

Here, character string 123 456 is stored in both variables abc and def.

The "\{" and "\}" included in braced words are not replaced by their corresponding symbols. "\\" included in braced words are not replaced either; the latter \ loses the special meaning as an escape character in "\<line break>", "\{", and "\}."

Example:

When "\" loses the meaning as an escape character:

```
set abc {123\\}
```

Here, string 123\\ is stored in variable abc.

📬 Note

. . . .

As described above, the following strings cannot be written as braced words.

Example: A string ending with \

```
C:\tmp\
```

Example: A string with a missing brace

```
123}
```

Example: Strings having a \ preceding the line break

```
C:\tmp\
D:\tmp\
```

. . . .

Handling of character strings in conditional expressions

The conditional expressions that are given in if or while statements are interpreted as mathematical expressions. Thus, if you want to have conditional expressions interpreted as string expressions, it is required to enclose the strings with double quotations. This requirement also applies to specify special characters that have true or false values (yes, no, true, and false).

Example 1:

The following example determines whether the value of variable a is string xyz:

```
if {$a == "xyz"} {
     puts ok
}
```

Example 2:

The following example repeats processing as long as the condition is true.

```
set i 0
while {"true"} {
      incr i
      if {$i > 5} {
            break
      }
}
```

Commands for manipulating strings

Double quoted strings are evaluated as mathematical expressions when it is possible to interpret the entire strings as a numeric. To precisely compare strings, it is recommended that the "15.2.19 string (Manipulate Character Strings)" command (a command that manipulates strings) be used to evaluate conditional expressions.

Example:

The following example determines whether the value of variable a is string 01:

```
if {[string compare $a 01] == 0} {
      puts ok
}
```

If the above if statement is replaced as shown below, the condition is true even if the value of variable a is 1 or 001.

```
if {$a == "01"}
```

Comment line

Any line or command that begins with # and ends with a line break is interpreted as a comment.

Example:

```
# Initialize the variable.
set abc 0
```

📝 Note

. . . .

Note the following when entering a comment:

- Do not put an opening brace or closing brace in a comment line.

- Do not put a continuation line (\<line break>) in a comment line or in a line right before the comment line.

- Do not put a comment into conditional expression in the evaluation statements such as "if" or "while" statement.

. . . .

15.1.4 Regular Expression

In a script, you can use regular expressions. The regular expressions used in scripts are explained below.

Symbols of Regular Expression

.	Matches any single character. Example: regexp {ABC.} $text Description: Checks the presence of the characters for the 4-character string containing ABC followed by any single character.
*	Matches zero-or-more iterations of preceding pattern element. Example: regexp {<.*>} $text Description: Checks the presence of the string for any element enclosed in <>. (The enclosed element may be a blank string.)
+	Matches one-or-more iterations of preceding pattern element. Example: regexp {<.+>} $text

	Description: Checks the presence of the string for any character enclosed by <>. (One or more characters must be included between < and >.)
?	Matches zero or one-time incidence of preceding pattern element. Example: regexp {0?1} $text Description: Checks the presence of the string for "1" or "01."
( )	Groups subpatterns. Iteration or selection applies to entire subpatterns. Example: regexp {(.*),(.*)} $text all sub1 sub2 Description: Checks the presence of the string for commas and, if included, extracts all matches to all and extracts the characters before and after the commas to sub1 and sub2. (The characters that precede and follow commas can be blank strings.)
|	Selection Example: regexp {Info|Warning} $text Description: Checks the presence of the string for "Info" or "Warning."
[ ]	Matches characters included in a set of characters. If the first character is "^", it matches a character not included in the set of characters. The range is specified using "-" (hyphen) (e.g. [a-z]). Example: regexp {ABC[0-9][0-9][^]} $text Description: Checks the presence of a 2-digit number following ABC and if a non-blank character follows right after it.
^	Matches the beginning of a string when specified at the beginning of a pattern. Example: regexp{^ABC} $text Description: Checks to see if the beginning of string is "ABC."
$	Matches the end of a character string when specified at the end of a pattern. Example: regexp{ABC$} $text Description: Checks to see if the end of string is "ABC."

Rules of Pattern Search

- The first and longest string that matches a pattern is selected.

- When " |" (selection) is included, the pattern search starts from the left side subpattern.

- In expressions that use "*", "+", or "?", strings having the longest match take precedence.

- For a sequence of two or more partial expressions, the pattern search starts from the left one.

Handling of Special Characters and Regular Expressions

Some of the regular expression symbols correspond to the special characters used in scripts. To use them, write them as shown below.

Special characters used in scripts

[,], $, (,), {, }

Use of special characters

Use the special characters in either of the following manners:

- Place a backslash sign (\) in front of the special character to use it as an escape character.

- Enclose the entire regular expression by braces ({ }).

Example:

The following example specifies regular expression "[0-9]" in the argument of regexp.

```
regexp "\[0-9\]" $text
```

or

```
regexp {[0-9]} $text
```

To treat a regular expression symbol as an ordinary character, place a backslahs (\) in front of it. Note that \ is also a special character in scripts and thus must be written "\\" or the entire regular expression must be enclosed in braces.

Example 1:

The following example checks the text for character "*":

```
regexp "\\*" $text
```

or

```
regexp {\*} $text
```

Example 2:

The following example checks the text for character "[":

```
regexp "\\\[" $text
```

or

```
regexp {\[} $text
```

To treat the backslash sign itself in a regular expression as an ordinary character, it must be written \\. Note that since \ is also a special character in scripts, either backslash signs must be written \\ or the entire regular expression must be enclosed in braces.

Example 3:

The following example checks the text for character "\":

```
regexp "\\\\" $text
```

or

```
regexp {\\} $text
```

Information

Escaping the scripts' special characters using the backslash signs requires multiple backslash signs as illustrated in Examples 2 and 3 and it may cause input errors. Thus, it is recommended that the regular expressions be always enclosed in braces.

Point

To extract strings, describe them using [^].

The following example extracts "BC" enclosed in "<>" to variable Var0 from the string "A<BC>D."

```
regexp {<([^>]*)} "A<BC>D" All Var0
```

The following example shows escaping of "BC" enclosed in "[]" to be extracted from the string "A[BC]D" by putting backslash signs in front of the special characters "[]":

```
regexp {\[([^\]]*)} "A\[BC\]D" All Var0
```

Typical Syntax Error in Regular Expression

The following shows typical syntax errors in regular expressions and corrective measures against them.

- couldn't compile regular expression pattern: parentheses () not balanced

In the description of regular expression, parentheses "()" are not described correctly. Check if either opening parenthesis or closing parenthesis is missing. To use "()" as normal characters, escape them by putting backslash signs (\).

- couldn't compile regular expression pattern: brackets [] not balanced

In the description of regular expression, brackets "[]" are not described correctly. Check if either opening bracket or closing bracket is missing. To use "[]" as normal characters, escape them by putting backslash signs (\).

- couldn't compile regular expression pattern: invalid character range

The range of character set is not specified correctly. Check if there is an error in the description enclosed in "[]." When specifying the range of character set, specify it in ascending order such as [0-9] and [a-z].

- couldn't compile regular expression pattern: quantifier operand invalid

Check if the use of regular expressions is correct.

Caution

If there is an unwanted blank space in regular expression, check cannot be conducted properly since a blank space is identified as a single character in regular expression.

15.2 Commands and Control Statements Used in Systemwalker Scripts

This section explains the commands/control statements used in Systemwalker Scripts.

Symbolic expression of options that can be omitted:

In this section, "**[ ]**" (blue bold brackets) are used to indicate optional options instead of "[ ]" (standard brackets), because the standard brackets are used to indicate a special meaning for Systemwalker Scripts.

15.2.1 after (Stop Processing for a Given Period of Time)

Description

Stops the execution of command for a certain time.

Synopsis

```
after msec
```

Option

msec

Specify the time during which the command sleeps in milliseconds (1/1000 seconds).

Example

The following example executes loop ten times at 1-sec intervals.

```
set i 0
while {$i < 10} {
        incr i
        after 1000
}
```

15.2.2 break (Abort Looping)

Description

Aborts looping.

Synopsis

```
break
```

Example

The following example executes loop five times.

```
set i 0
while {"true"} {
      incr i
      if {$i > 5} {
            break
      }
}
```

15.2.3 catch (Execute Script Lines and Trap Exceptions)

Description

Executes script lines and traps exceptions.

Exceptions generated during execution of a script are errors that cause the current script to be aborted, forcing the process itself to be terminated. Some of the examples of such an error are those generated depending on run-time conditions such as an error on file open (open command), abnormal termination of the process being executed (i.e. completion code is other than 0 with exec command), in addition to a syntax error in the current script being executed, or variable substitutions attempted on variables with no associated values.

The catch command is used to prevent script process from being terminated by running its own recovery process when such exceptions occur.

Synopsis

```
catch {script} [varname]
```

Options

{script}

Script command to run. Two or more commands can be written by separating them by line breaks or semicolons.

varname

The name of a variable that stores an error text giving the cause of an exception, if occurred. If omitted, the error text will not be stored.

Return Values

0:

No exceptions have occurred.

Other than 0:

Exceptions have occurred.

In this case, the following value is stored as detailed information in the global variable errorCdoe:

- When the completion code of the command executed by the script command "exec" is other than 0

```
CHILDSTATUS pid exitcode
```

pid

[Windows]

Numeric characters entered in this field do not have any meaning

[UNIX]

The process ID of the command executed

exitcode

The completion code of the command executed

- When the command executed by the script command "exec" is forcibly terminated by a signal

```
CHILDKILLED pid sigName msg
```

pid

[Windows]

Numeric characters entered in this field do not have any meaning.

[UNIX]

The process ID of the command executed

sigName

The name of the signal sent to the command executed

msg

Detailed message

- When the command executed by the script command "exec" is suspended by a signal

```
CHILDSUSP pid sigName msg
```

pid

[Windows]

Numeric characters entered in this field do not have any meaning.

[UNIX]

The process ID of the command executed

sigName

The name of the signal sent to the process

msg

Detailed message

- When an arithmetic error occurred in the script command "expr"

```
ARITH errName msg
```

errName

Error contents

msg

Detailed message

- When an error occurred in the system call

```
POSIX errName msg
```

errName

　　Error contents

msg

　　Detailed message

　- When there is no information to be returned with the error other than the relevant message

```
NONE
```

Example

The following example opens a file for reading and if fails, enables reading from the standard input.

```
if {[catch {
     set file [open data.txt r]
}]} {
     set file "stdin"
}
gets $file buf
if {$file != "stdin"} {
     close $file
}
```

15.2.4 cd (Change Working Directory)

Description

Changes the working directory.

Synopsis

```
cd dirname
```

Option

dirname

Specify the name of a directory to which the current working directory is changed. The directory names on Windows can be also separated by a slash (/) in addition to the backslash (\). When using a backslash sign, the entire directory name must be enclosed in braces ({}) so as not to be treated as an escape character.

Caution

If a name of a non-existing directory is specified, the script is forced to exit upon detecting the occurrence of an exception.

Example

[Windows]

The following example moves the current directory to c:\tmp.

```
cd {c:\tmp}
```

[UNIX]

The following example moves the current directory to /var/tmp.

```
cd /var/tmp
```

15.2.5 close (Close an Open File)

Description

Closes an open file.

Synopsis

```
close channelID
```

Option

channelID

Specify the channel ID acquired as the return value from the open command.

Reference

open

Caution

Any files opened by the open command must be finally closed by the close command.

Example

The following example closes the channel ID stored in variable fc.

```
close $fc
```

15.2.6 continue (Skip a Loop to the Next Iteration)

Description

Skips to the conditional evaluation of a loop statement, skipping the subsequent iterations of a loop.

Synopsis

```
continue
```

Example

The following example shows the count values on the last three iterations while executing a loop five times.

```
set i 0
while {$i < 5} {
      incr i
      if {$i < 3} {
            continue
      }
      puts $i
}
```

Execution Results/Output Format

```
3
4
5
```

15.2.7 exec (Invoke Subprocess(es))

Description

Invokes subprocess(es) to run program commands.

Synopsis

[Windows]

```
exec cmd /c cmdname [args ...]
```

Note: The exec command must be used with cmd /c.

[UNIX]

```
exec cmdname [args ...]
```

Options

cmdname

Specify thei name of a command file to invoke. The file names on Windows can also be separated by a slash (/) in addition to the backslash sign (\). When using a backslash sign, the entire file name must be enclosed in braces ({}) so as not to be treated as an escape character.

args ...

Specify the argument to be passed to the command to invoke.

If the last *args* is "&", the command is executed in background. Generally, the exec command is executed in foreground and waits for the *cmdname* process to terminate. However, when executed in background, it returns without doing so.

In addition, it is possible to specify a flow control symbol in *args*.

For Windows, when giving a file by specifying its path, the directory must be separated by backslash sign (\) and the entire file name must be enclosed in braces ({}).

Flow control

Writing the following flow control symbols to *args* allows manipulation of standard inputs/outputs of commands invoked or information exchange through activation of multiple commands.

Symbol and form	Meaning	Example
|	Separates individual commands in the pipeline. The standard output of the preceding command will be piped into the standard input of the subsequent command. Three or more commands can be piped in the same way. If "|&" is used in place of "|", the standard output as well as standard error output will be piped into the standard input of the subsequent command.	[Windows] exec cmd /c dir |cmd /c sort [UNIX] exec ls -1 | grep root
< *filename*	The content of file specified by *filename* is read as the standard input of *cmdname*. (Note 1)	[Windows] exec cmd /c sort < data.txt [UNIX] exec sort < data.txt
<< *value*	The value (string) specified by *value* is read as the standard input of *cmdname*. (Note 1)	[Windows] exec cmd /c sort << $data [UNIX] exec sort << $data
> *filename*	The standard output of *cmdname* is overwritten to the file specified by *filename*. (Note 1) Using ">>" in place of ">" causes it to be appended rather than being overwritten.	[Windows] exec cmd /c dir > data [UNIX] exec ls > data

Symbol and form	Meaning	Example
>& *filename*	The standard output of cmdname and standard error output are overwritten to the file specified by filename. (Note 1) Using ">>&" in place of ">&" causes it to be appended rather than being overwritten.	[Windows] exec cmd /c dir data >& out [UNIX] exec rm -f data >& out
2> *filename*	The standard error output of *cmdname* is overwritten to the file specified by *filename*. (Note 2) Using "2>>" in place of "2>" causes it to be appended rather than being overwritten.	[Windows] exec cmd /c dir data 2> out [UNIX] exec rm -f data 2> out

Note 1:

If *cmdname* comprises two or more commands through |, the first command will be used for reading and the last one for writing.

Note 2:

If *cmdname* comprises two or more commands through |, the standard error output of all commands will be stored in the file specified by *filename*.

Return Values

When the command is executed in foreground:

The standard output of the cmdname command is returned. The trailing line break code in the output text will be deleted.

When multiple commands are executed through flow control, the standard output of the last command will be returned.

When the standard output is redirected through flow control, a blank string will be returned.

When the command is executed in background:

The process ID of the command invoked is returned.

When multiple commands are executed through flow control, all process IDs separated by blanks will be returned. Due to this, the standard output that is output by an application cannot be obtained as the return value of the exec command.

Other Input/Output Information

- In the following conditions, a script detects an exception and will be forcibly terminated. Trap the exception detected with the catch command as required.

  - The command is executed in foreground, and

  - The command outputs a message to be sent to the standard error output where the standard error output of the command is not redirected, or

  - The completion code of the command is other than 0.

Example:

When exception trap with the catch command is not performed.

[Windows]

```
#The application is activated. Standard output is stored to variable "outmsg."
set outmsg [exec cmd /c {c:\usr\bin\appl001.exe}]

#Data output to the standard output by the application is output to the standard
output of the job.
puts stdout $outmsg

#When the activation of the application through its execution has been successfully
```

```
terminated, the processing terminates with completion code "0."
exit 0
```

[UNIX]

```
#The application is activated. Standard output is stored to variable "outmsg."
set outmsg [exec /usr/bin/appl001]

#Data output to the standard output by the application is output to the standard
output of the job.
puts stdout $outmsg

#When the activation of the application through its execution has been successfully
terminated, the processing terminates with completion code "0."
exit 0
```

Remark:

The following problems occur when the command executed by exec terminates with completion code other than 0 or abnormally, since exception setting with the catch command does not exist:

- "1" is returned as the completion code of the job and the completion code of the command executed is not applied.

- Error message or other information output to the standard error output by the command executed cannot be obtained.

  To prevent the above problems, it is recommended that you trap exceptions with the catch command when executing a command.

- When trapping exceptions with the catch command, you can refer to the following information.

  - Standard output/standard error output (errorInfo)

  - Completion code (errorCode)

Information

Standard output/standard error output information (errorInfo):

When the standard output or standard error output of the command is not redirected, its output string will be stored in the beginning of global variable errorInfo. In errorInfo, the strings that identify execution errors of the exec statement itself will be stored.

Example:

[Windows]

If the application that implements the following specifications (c:\usr\bin\appl001.exe) terminated abnormally:

Specifications:

Write the data of specified date to the standard output

If the specification of date is invalid, output the following message to the standard error output.

"Error in date specification. Specify in yyyymmdd format."

```
if {[catch {
        exec cmd /c {c:\usr\bin\appl001.exe} 010520
    }]} {
      puts "error information START"
      puts $errorInfo
      puts "error information END"
}
```

When the above code is executed, the standard output shows the following information.

```
Error information START
An error in date specification. Specify in yyyymmdd format. <-  appl001 command output
    while executing                                <- String that indicates execution error
"exec cmd /c {c:\usr\bin\appl001.exe} 010520"            <- of exec statement itself
Error information END
```

[UNIX]

If the ls command is executed by specifying a file (/var/tmp/data) that does not exist:

```
if {[catch {
        exec ls /var/tmp/data
    }]} {
      puts "error information START"
      puts $errorInfo
      puts "error information END"
}
```

When the above code is executed, the standard output shows information as shown below.

```
Error information START
/var/tmp/data: no file or directory exists.                        <- ls command output
    while executing                          <- String that indicates execution error
"exec ls /var/tmp/data"                           <-  of exec statement itself
Error information END
```

Information

Completion code (errorCode)

In case of a non-zero completion code for a command, the completion code will be stored in global variable errorCode in the form shown below. Also other information is stored in it. See the return value of catch (execute script lines and trap exceptions) for the form of errorCode.

```
CHILDSTATUS pid exitcode
```

pid

> **[Windows]**
>
> Numeric characters entered in this field do not have any meaning
>
> **[UNIX]**
>
> The process ID of the command executed

exitcode

> The completion code of the process

- Input/output of executed commands

  The input and output of executed commands having not been redirected are treated as follows:

Input/ Output	When executed in foreground	When executed in background
Standard input	Read from the standard input of the script process itself.	
Standard output	Normally, returned as the return value of the exec command. If exceptions are trapped by the catch command, they are stored in errorInfo.	Written to the standard output of the script process itself.

Input/ Output	When executed in foreground	When executed in background
Standard error output	Treated as an exec command exception. If trapped by the catch command, it is stored in errorInfo.	Written to the standard error output of the script process itself.

Reference

catch

Cautions

- On Windows x64, repeatedly calling the exec command within a single script may cause it to hang up. It is recommended that the number of calls be restricted to 100, more or less.

  If it is required to repeatedly call the exec command, split the script file and decrease the number of times to call exec command per script file. The split script files can be executed within the script as arguments to the swotclsh command by the exec command. Below is an example of executing a split script.

  Example:

  ```
  exec cmd /c swotclsh {c:\script\part1.swt}
  ```

  If you create a script that calls the exec command multiple times, call the command as many times as specified when testing the operation of the script to make sure all calls are properly performed.

- On Windows x64, if a command located in the Windows installation directory\system32 (normally C:\WINDOWS \system32) is specified as the command to be activated by exec, the command located in Windows installation directory \syswow64 will be executed.

- Startup of commands or shells whose descendant processes reside

  In the following conditions, the command directly executed will not return:

  - The command is executed in foreground, and

  - The executed command executes a child process, and

  - The standard output and standard error output are not redirected.

  For this reason, when activating a resident process such as daemon startup shell, activate it in background or redirect it to the standard output and standard error output as shown below.

  - Background startup

  ```
  exec rdbstart &
  ```

  - Redirecting the standard output and standard error output to a file

  ```
  exec rdbstart >& /var/tmp/log.txt
  ```

  - Ignoring the standard output and standard error output

  ```
  exec rdbstart >& /dev/null
  ```

- Windows commands whose parameters require particular interpretation

  For Windows programs, there are two types of commands. One is general commands that interpret double quotation marks for command options as special characters and the other one is special commands that do not interpret such characters as special characters. When passing parameters to such special commands, the parameters may not be passed properly since scripts themselves interpret double quotation marks.

  - Example of parameter interpretation by general commands

Description in command line	Interpretaion in command
"abc"	abc

Description in command line	Interpretaion in command
\"abc\"	"abc"

Special commands interpret the description in the command line as it is.

In order to execute such a command, create a batch file to execute the command and describe statement to execute the batch file by exec.

- Arguments beginning with a flow control symbol

Since a character string beginning with a flow control symbol (< > 2>) is interpreted as a flow control symbol when it is specified as an argument, it cannot be passed as a character string. Thus, specify such a character string using either of the following methods.

- Specifying an argument with a space inserted in its top

Specify the character string as an argument by inserting a space in the top of it so that it can be interpreted as a character string.

You can insert a space in the top by describing the script as shown below when the character string is stored in a variable (msgtxt).

```
if {[regexp {^<|^>|^2>} $msgtxt] == 1} {
    set msgtxt " $msgtxt"
}
```

- Using a shell script or batch file

Create a shell script [UNIX] or batch file [Windows] that executes the target command within the script, and call it by using the exec command. If double quotation marks (") are included in the argument character string, they should be replaced with single quotation marks (') beforehand since character string interpretation does not work correctly within the shell script or batch file under this environment.

Example: Executing the command by specifying a variable (msgtxt) in which arbitrary character string is stored as the argument of the command (usercmd).

[UNIX]

"/var/tmp/usercmd_tmp.sh" is a shell script created temporarily. It is deleted after executed.

```
if {[regexp {^<|^>|^2>} $msgtxt all] == 1} {
    regsub {"} $msgtxt {'} msgtxt
    set fd [open /var/tmp/usercmd_tmp.sh w]
    puts $fd "usercmd \"$msgtxt\""
    close $fd
    exec sh /var/tmp/usercmd_tmp.sh
    exec /usr/bin/rm -f /var/tmp/usercmd_tmp.sh
} else {
    exec usercmd $msgtxt
}
```

[Windows]

"c:\temp\usercmd_tmp.bat" is a batch file created temporarily. It is deleted after executed.

```
if {[regexp {^<|^>|^2>} $msgtxt all] == 1} {
    regsub {"} $msgtxt {'} msgtxt
    set fd [open {c:\temp\usercmd_tmp.bat} w]
    puts $fd "usercmd \"$msgtxt\""
    close $fd
    exec {c:\temp\usercmd_tmp.bat}
    exec cmd /c del {c:\temp\usercmd_tmp.bat}
}else {
    exec usercmd $msgtxt
}
```

Examples

[Windows]

The following example runs the dir command and outputs the command's standard output/standard error output to script's standard output.

```
if {[catch {
        set outmsg [exec cmd /c dir {c:\tmp\data}]

        # Output dir's standard output (return value from exec) to script's standard
           output.
        puts stdout $outmsg
    }]} {
    # Occurrence of an exception.
    # Store $errorInfo as initial value in outmsg.
    set outmsg $errorInfo

    # Output dir's output information (except for trailing two lines of errorInfo) to the
      script's standard output.
    regexp {(.*)\n.* \n.*} $errorInfo all outmsg
    puts stdout $outmsg
}
```

[UNIX]

The following example runs the ls command and outputs the command's standard output/standard error output to script's standard output.

```
if {[catch {
        set outmsg [exec ls -l /var/tmp/data]

        # Output ls's standard output (return value from exec) to script's standard
           output.
        puts stdout $outmsg
    }]} {
    # Occurrence of an exception.
    # Store $errorInfo as initial value in outmsg.
    set outmsg $errorInfo

    # Output ls's output information (except for trailing two lines of errorInfo) to the
      script's standard output.
    regexp {(.*)\n.* \n.*} $errorInfo all outmsg
    puts stdout $outmsg
}
```

Execution Results/Output Format

The result of command execution is displayed on the standard output as shown below.

[Windows]

- When c:\tmp\data file exists:

```
No Drive C volume label.
Volume serial number is 2420-12F3.

Directory c:\tmp
00/12/01  11:37a                   972 data
             One file              972 bytes
                          463,110,144 bytes free area
```

- When c:\tmp\data directory (file) does not exists:

```
No Drive C volume label.
Volume serial number is  2420-12F3.

Directory c:\tmp

Cannot find the file.
```

[UNIX]

- When /var/tmp/data file exists:

```
-rw-rw-r--    1  root   other         0  15:05 Sep. 25  /var/tmp/data
```

- When /var/tmp/data file (directory) does not exist:

```
/var/tmp/data: no file or directory exists.
```

15.2.8 exit (Terminate a Script)

Description

Terminates the process of a script.

Synopsis

```
exit [returncode]
```

Option

returncode

Program completion code. Specify it between 0 and 128. If omitted, 0 is assumed.

Example

If the content of variable rc is a non-zero value, exits abnormally with that value being used as the completion code.

```
if {$rc != 0} {
      exit $rc
}
```

15.2.9 expr (Perform an Arithmetic Operation)

Description

Executes an arithmetic operation.

Synopsis

```
expr arg [arg arg ...]
```

Option

arg

Specify the expression.

Valid operators (listed in decreasing order of precedence)

- + ~ !	Unary minus, unary plus, complement, logical NOT
* / %	Multiply, divide, remainder
+ -	Add, subtract

A part of expression enclosed in parentheses (()) takes precedence over the others included in the same command and is computed first.

Return Value

The result of operation is returned.

Example

The following example displays the result of 3 x (10 - (4 - 2)).

```
set result [expr 3*(10-(4-2))]
puts $result
```

Execution Results/Output Format

```
24
```

15.2.10 flush (Flush Buffered Output)

Description

Flushes buffered output.

Information

Flushing is an operation that removes the data remaining in the write buffer. For example, if the standard output is connected to other commands through a pipe, flushing is needed to immediately have the destination receive the output data.

Synopsis

```
flush channelID
```

Option

channelID

Specify the channel ID acquiired as the return value from the open command. Specifying stdout and stderr flushes the buffered output to the standard output and standard error output.

Example

The following example writes data to the standard output and flushes it.

```
puts stdout $Data
flush stdout
```

15.2.11 gets (Read One Line from a File)

Description

Reads a single line from a file or standard input.

```
gets channelID [varname]
```

Options

channelID

Specify the channel ID acquired as the return value of the open command. Specifying stdin enables reading from the standard input.

varname

Specify the name of a variable to store the content of line read. If this option is omitted, then the content of line read is returned as the return value.

Return Values

When varname is specified:

Returns the number of characters of the string it read. If it reaches the end of file without reading any character, it returns -1.

When varname is not specified:

Returns the string it read.

Caution

One line it reads indicates the location between the current reading position in the file (the beginning of file upon opening the file and, every time it reads, the current position moves correspondingly) and the end-of-line character or the end of file. The end-of-line code will not be stored in the variable given by varname.

Example

The following example reads the data of a line up to the end-of-line character from the standard input and stores it in variable buf.

```
gets stdin buf
```

15.2.12 if-else (Perform Conditional Branching)

Description

Executes scripts on a conditional basis.

Synopsis

```
if {expr1} [then] {body1} [elseif {expr2} [then] {body2}]
[elseif ...] [else {bodyN}]
```

Options

expr

Defines the conditional expression. For a numeric value, 0 is false and anything other than 0 is true. For a string value, true or yes is true and false or no is false. Do not put a comment into conditional expression.

Condition operators available for conditional expressions

!	NOT
< , > , <= , >= , == , !=	Comparison
&& , ||	Logical AND, logical OR

body

> The script command executed when the condition is true. Two or more commands can be written by being separated by line breaks or semicolons.

Caution

Keywords (if, then, elseif, and else) must be separated from "{" by blanks or tabs. If "then" is omitted, "}" must be separated from "{" by blanks or tabs.

When inserting a line break in other than body such as between "}" and "{", add backslash sign (\) to the last of the line to indicate continuation.

- Incorrect example: A line break is inserted in other than body

```
if {$cnt == 1 }
{
      puts "start"
}
```

- Correct example: No line break is inserted between "}" and "{"

```
if {$cnt == 1 } {
      puts "start"
}
```

Or, a line break is inserted and a backslash (\) is added to indicate line continuation.

```
if {$cnt == 1 } \
{
      puts "start"
}
```

Example

The following example determines whether the value of variable a is greater than 0 or not and displays the result.

```
if {$a > 0} {
      puts "a > 0"
} else {
      puts "a <= 0"
}
```

Execution Results/Output Format

Either of the following lines is output depending on the value of a.

```
a > 0
```

or,

```
a <= 0
```

15.2.13 incr (Add a Value to a Variable)

Description

Adds a value to a variable.

Synopsis

```
incr varname [increment]
```

varname

Specify the name of variable to be added. This variable must hold a decimal integer. The result of addition is also stored in this variable.

increment

Specify the value to be added (a decimal integer). If this option is omitted, 1 is added.

Return Value

The result of addition (i.e. $varname + increment$ before incr command is called)

Caution

If both the variable specified by *varname* and the value of *increment* are not decimal integers, the script is forced to exit upon detecting the occurrence of an exception. When using this command, trap exceptions using the catch command where necessary.

Example

The following example shows the count values for the last three looping sequences of the five iterations.

```
set i 0
while {$i < 5} {
      incr i
      if {$i < 3} {
            continue
      }
      puts $i
}
```

Execution Results/Output Format

```
3
4
5
```

15.2.14 open (Open a File)

Description

Opens a file.

Synopsis

```
open filename [access]
```

Options

filename

Specify the name of a file to open. The file names on Windows can be also separated by a slash (/) in addition to the backslash sign (\). When using a backslash sign, the entire file name must be enclosed in braces ({}) so as not to be treated as an escape character. If the filename contains spaces or tabs, it must be enclosed in double quotations ("").

access

Specify the access mode. When omitted, it defaults to read-only (r) access. The values to specify access are as follows:

Types of access modes

Code	Description
r	Opens the file in read-only mode (default value). The file must already exist.
r+	Opens the file in read-and-write mode. The file must already exist.
w	Opens the file in write-only mode. Overwrites the file if exists. If it does not exist, creates a new file.
w+	Opens the file in read-and-write mode. Overwrites the file if exists. If it does not exist, creates a new file.
a	Opens the file in write-only mode. The file must already exist. The write data is added to the end of the file.
a+	Opens the file in read-and-write mode. If the file does not exist, creates a new file. The write data is added to the end of the file.

Return Value

Channel ID:

Since the channel ID must be specified to access an open file (including its closing), the return value needs to be stored in the variable through command substitution.

Reference

close

Caution

If the script fails to open a file (e.g. if a file with no permissions assigned is specified), it is forced to exit upon detecting the occurrence of an exception. When using this command, trap exceptions using the catch command where necessary.

Examples

[Windows]

The following example opens the c:\tmp\data file in write-only mode and stores its channel ID in variable fc.

```
set fc [open {c:\tmp\data} w]
```

[UNIX]

The following example opens the /var/tmp/data file in write-only mode and stores its channel ID in variable fc.

```
set fc [open /var/tmp/data w]
```

15.2.15 puts (Write into a File)

Description

Writes data to file or standard output and standard error output.

Synopsis

```
puts [channelID] string
```

Options

channelID

Specify the channel ID acquired as the return value of the open command. Specifying stdout and stderr enables writing to the standard output and standard error output. If omitted, writing to the standard output is enabled.

string

Specify the text data to write.

Caution

An end-of-line code is always added to the end of the text when written.

Examples

The following example outputs error messages to both the log file and the standard error output.

[Windows]

```
set errlog [open {c:\var\tmp\log.txt} a]
puts $errlog $ErrMsg
puts stderr $ErrMsg
close $errlog
```

[UNIX]

```
set errlog [open /var/tmp/log.txt a]
puts $errlog $ErrMsg
puts stderr $ErrMsg
close $errlog
```

15.2.16 regexp (Perform Regular Expression Matching and Extract the Matched Part)

Description

Matches a regular expression against a string and extracts matched parts.

Synopsis

```
regexp exp string[matchvar] [submatchvar1 submatchvar2 ... ]
```

Options

exp

Specify a regular expression.

string

Specify the string to check.

matchvar

Specify the name of the variable in which parts of string that matched the regular expression are extracted and stored. If omitted, the matching parts will not be stored.

submatchvar

Specify the name of the variable in which parts of string that matched the parenthesized substrings are extracted and stored. If omitted, the matching parts will not be stored.

P Point

. . . .

A regular expression is a shorthand for describing the structure of a string using symbols. This command checks whether the string specified in the *string* option has a part that matches the structure indicated in regular expression *exp*.

If it does, the command extracts it from *string* and stores it in the *matchvar* variable. In regular expressions, you can specify the strings to extract as many as you want at any desired position. These strings will be stored in *submatchvar* variables in order.

For how to write a regular expression, see "15.1.4 Regular Expression".

•••

Return Values

0:      Not matched.

1:      Matched.

Examples

- Extracting before and after a specified character

    The following example checks whether the string contains commas "," and, if contained, extracts all matches to all and extracts those characters preceding and following a comma to variables sub1 and sub2, respectively. (The characters that precede and follow a comma can be blank characters.)

    ```
    regexp {(.*),(.*)} $text all sub1 sub2
    ```

- Specifying a part to be extracted

    The following example extracts the part enclosed in "<>" from the following message and stores it to variable Var0.

    ```
    A "LinkUp" has occurred on the network. <An error has occurred with the DB server.>
    ```

    ```
    regexp {<([^>]*)} $EventText All Var0
    ```

    Message part "An error has occurred with the DB server." is stored to variable Var0.

- Escaping special characters using backslash sign

    The following example extracts the part after JobName= to the next blank, the bracketed part after RC= and the parenthesized part after device=, and then stores them to Var0, Var1 and Var2 respectively.

    ```
    JOBNET Error. JobName=SCRJB_1 RC=[0022] xxx-device=(DX038)
    ```

    ```
    regexp {JobName=([^ ]*) RC=\[([^\]]*)\] .*device=\(([^\)]*)\)} $EventText All Var0 Var1
    Var2
    ```

    "SCRJB_1", "0022" and "DX038" are stored to Var0, Var1 and Var2 respectively.

15.2.17 regsub (Perform Substitutions based on Regular Expression Pattern Matching)

Description

Performs substitutions based on regular expression pattern matching.

Synopsis

```
regsub [-all] exp string subspec varname
```

Options

-all

Specify to substitute all portions that matched a regular expression. If omitted, only the first matching portion is substituted.

exp

Specify the regular expression that identifies the portion to substitute.

string

Specify the string to be replaced.

subspec

Specify the string to replace. The string to replace must be enclosed in braces ({}). To include "\" or "&" in the string to replace, they must be written "\\" or "\&", respectively.

varname

Specify the name of the variable to store the string that was replaced.

 P Point
. . . .

A regular expression is a shorthand for describing the structure of a string using symbols. This command searches the string specified by *string* for the portion that matches the structure represented in the regular expression exp. If there is a match, it replaces it with the string specified by *subspec* and stores in the variable whose name is given by *varname*.

For how to write a regular expression, see "15.1.4 Regular Expression".

. . . .

Return Values

0:    Not matched.

1:    Matched.

Example

The following example replaces all spaces between strings with commas.

```
set text "a b c d"
regsub -all " " $text {,} result
puts $result
```

Execution Results/Output Format

```
a,b,c,d
```

15.2.18 set (Store a Value in a Variable)

Description

Stores a value in a variable.

Synopsis

```
set varname [value]
```

Options

varname

Specify the name of the variable that stores the value.

value

The value to store. Specify a numeric value or string. If omitted, the value of the variable specified by *varname* is not changed.

Return Values

The value stored in the variable *varname* after processing has been completed (if *value* has been specified, the value of *value*; if it has not been specified, the value having been stored in the variable *varname*)

Caution

If *value* is omitted, the value stored in the variable *varname* as the return value will be returned. Thus, a new variable specified by *varname* will result in an error.

Example

The following example stores a numeric value in variable rc and a text in variable msg.

```
set rc 1
set msg "File open error"
```

15.2.19 string (Manipulate Character Strings)

Description

Performs string operations, such as comparisons, extractions, conversions, and fragmentation.

Synopsis

```
string option arg [arg ... ]
```

Options

option

Specify the option keyword that defines a string operation. How to define available options and their descriptions are shown below.

arg ...

Specify the argument required for string manipulation.

Each option is explained in the following table:

string compare *string1* *string2*	Performs a string-by-string comparison. Returns -1, 0, or 1 depending on whether *string1* lexicographically is less than *string2*.
string first *string1* *string2*	Searches *string2* for occurrences of *string1*. If found, it returns the index of the first character (see Note) of *string1* in the first match within *string2*. If not found, it returns -1.
string index *string* *charIndex*	Returns the string of the place indicated by *charIndex* (index (see Note)).
string last *string1* *string2*	Searches *string2* for occurrences of *string1*. If found, it returns the index of the first character (see Note)) of *string1* in the last match within *string2*. If not found, it returns -1.
string length *string*	Returns the length of string in *string*.
string match *pattern* *string*	Checks if *pattern* matches *string*. Returns 1 if it does, 0 if it does not. Special characters used to specify filenames in UNIX C Shell can be also used in *pattern*. Shown below are the characters treated as the special characters in *pattern*. *: Interpreted as a sequence of characters, including a null string. E.g.) Pattern ab* matches any strings that begin with ab. ?: Interpreted as any single character. E.g.) Pattern ab? matches a three-character sting that begins with ab.

	[]: Interpreted as any single character enclosed in brackets. [a-z] is interpreted as any character between a and z. E.g.) Pattern ab[0-9] matches any three-character strings that begins with ab followed by a number. \: If written just before any of the special characters (*?[] \) in pattern, it voids their special interpretation to treat them as a single character. E.g.) Pattern ab* matches the string ab*. If any of these special characters is not included in the pattern, the two strings will match if string perfectly matches *pattern*.
string range *string i j*	Returns the string of the place (index (see Note)) indicated from character *i* to character *j*.
string tolower *string*	Returns the *string* which has been converted to lower case.
string totitle *string*	Returns the *string* the first character of which has been converted to upper case and the rest of which has been converted to lower case.
string toupper *string*	Returns the *string* which has been converted to upper case.
string trim *string* [*chars*]	If a character contained in the character set specified by *chars* exists at the beginning or end of *string*, this option returns a character string from which that character has been removed. If *chars* is not specified, white spaces are removed.
string trimleft *string* [*chars*]	If a list of characters specified by *chars* exists at the beginning of *string*, this option returns a character string from which that list of characters has been removed. If *chars* is not specified, white spaces are removed.
string trimright *string* [*chars*]	If a character contained in the character set specified by *chars* exists at the end of *string*, this option returns a character string from which that character has been removed. If *chars* is not specified, blank spaces are removed.
string wordend *string index*	Returns the position of the character just after the last one in the word containing the indexed (see Note) character in *string*.
string wordstart *string index*	Returns the position of the first character in the word containing the indexed (see Note) character in *string*.

Note:

The index represents the position in a string (nth character), when 0 corresponds to the first character.

🅿 Point

When using the string command to determine character strings, use the logical expression as shown below since complicated conditions cannot be specified with the command alone.

Example:

When string "AAA" or "BBB" is stored to variable EventText, it is displayed to standard output.

```
if {[string first "AAA" $EventText] > 0 || [string first "BBB" $EventText] > 0} {
  puts $EventText
}
```

Return Values

See the above table for each option.

Example

The following example extracts the first three characters from the string stored in the variable buf and stores them to the variable top.

```
set top [string range $buf 0 2]
```

15.2.20 while (Execute a Loop)

Description

Repeats processing as long as a condition is met.

Synopsis

```
while {test} {body}
```

Options

test

Specify the conditional expression that repeats a loop (body). For a numeric value, 0 is false and anything other than 0 is true. For a string value, true and yes are true and false and no are false. Do not put a comment into conditional expression.

Condition operators valid for conditional expressions

!	NOT
< , > , <= , >= , == , !=	Comparison
&& , ||	Logical AND, Logical OR

body

The script command executed as long as the condition is true. Two or more commands can be written by being separated by line breaks or semicolons.

Cautions

- "while" must be separated from "{}" by blanks or tabs. "}" in the conditional expression must be separated from "{" in the body must be separated by blanks or tabs.

  When inserting a line break in other than body such as between "}" and "{", add a backslash sign to the end of the line to indicate continuation.

- A while statement involving an error in the conditional evaluation logic may result in an infinite loop. To avoid this, it is recommend that a protective mechanism as described below be provided so as to prevent the loop being executed over a certain time.

  - Provide a while statement with a variable for the counter that is not used in any other elements in the statement and have that value checked at an early stage in that while statement for each iteration of a loop.

Example

The following example executes a loop five times.

```
set i 0
while {$i < 5} {
     incr i
}
```

15.3 Commands Used in Debugging

Unlike other programming languages such as C, Systemwalker Script runs without requiring compilation procedures. Then again it has such a drawback that simple syntax errors, logical errors, or lack of consideration on execution environments cannot be easily detected before being run.

It is, therefore, required to check and debug scripts as soon as you have created them.

This section explains the commands used to debug Systemwalker Scripts.

15.3.1 Extended Trace Commands

The following commands enable you to debug Systemwalker Scripts by writing those commands in the scripts and by collecting traces.

- sw_TcOpenTrace (Open Trace File Command)

- sw_TcWriteTrace (Output Trace Information Command)

- sw_TcCloseTrace (Close Trace File Command)

15.3.1.1 sw_TcOpenTrace (Open Trace File Command)

Description

This command opens a trace file and performs initial processing for collecting traces. To invoke the collecting of script execution traces, this command must be called first.

Synopsis

```
sw_TcOpenTrace [-size fsize] [-level tlevel]
[-mode group|other] filename
```

Options

-size *fsize*

Specify the maximum size of one trace file in *fsize*. The value can be any number from 1 to 100,000 (KB). It defaults to 50 (KB).

In the trace collection, two files are created to cyclically collect traces and, thus, the estimated maximum file size will be twice as much as the size specified here.

-level *tlevel*

Specify the trace level in *tlevel* in the range from 0 to 2. It defaults to 1. The following table lists what is collected and recommended examples for each trace level.

0	Disables the subsequent sw_TcWriteTrace commands and no trace is collected.
1	Only level-1 traces are collected in the subsequent sw_TcWriteTrace. Example: Using to output the information collected when Systemwalker Scripts are actually running. - Output of information that is kept as operation logs such as results of startup or processing - Output of information that is kept as error logs such as details for errors detected
2	Both level 1 and 2 traces are collected in the subsequent sw_TcWriteTrace. Example: Using to output the information collected when performing operation verification test and debugging - Output of information that is kept as script root check, such as breakpoints and the points before and after external programs are invoked - Output of information that is kept as information within scripts, such as the contents of read files and values of variables used in handling loops

-mode group|other

Specify the write mode for the trace file to create. If omitted, the write permission attribute is set for the owner. This option is ignored on Windows.

group

Specify the write permission attribute for the owner and group.

other

Specify the write permission attribute for the owner, group, and others.

filename

Specify the name of a trace file. If the filename is specified without its path, the trace is collected into the default trace storage directory. If it is specified with its path, the trace is collected in accordance with the path.

For Windows, the file names on Windows can be also separated by a slash (/) in addition to the backslash sign (\). When using a backslash sign, the entire file name must be enclosed in braces ({ }) so as not to be treated as an escape character.

By default, the traces are stored in the following directory:

[Windows]

```
Systemwalker Operation Manager installation directory \MpWalker.JM\mpfwtco\trc
```

[UNIX]

```
/var/opt/FJSVfwtco/trc
```

Note

The directory where a trace file is created should be the directory for which the Systemwalker Script execution user has update rights.

Information

Two files are created to cyclically collect traces. Thus "#1.txt" or "#2.txt" is appended to the end of the filename you specified.

For example, if you specify {c:\tmp\trc}, trc#1.txt and trc#2.txt will be created under c:\tmp. [Windows]

If you specify "/var/tmp/trc", trc#1.txt and trc#2.txt will be created under /var/tmp. [UNIX]

Return Values

Positive value:	Normal termination. A handle used in a trace command
-9:	Insufficient memory
-15:	Code conversion error
-17:	File open error

Reference

sw_TcCloseTrace

Examples

[Windows]

The following example opens traces with c:\var\tmp\usrtrc given as the trace file and stores the handle in the variable hwnd.

```
set hwnd [sw_TcOpenTrace {c:\var\tmp\usrtrc}]
```

[UNIX]

The following example opens traces with /var/tmp/usrtrc given as the trace file and stores the handle in the variable hwnd.

```
set hwnd [sw_TcOpenTrace /var/tmp/usrtrc]
```

15.3.1.2 sw_TcWriteTrace (Output Trace Information Command)

Description

This command outputs trace information to a trace file.

Synopsis

```
sw_TcWriteTrace [-level tlevel] thandle text
```

Options

-level *tlevel*

Specify trace level 1 or 2 in *tlevel*. It defaults to 1.

thandle

Specify the handle obtained by sw_TcOpenTrace. If a null string ("") is specified, traces will not be collected.

text

Specify a string to be output to the trace file. It should be specified within 900 bytes. Part of a string exceeding 900 bytes will be truncated to 900 bytes.

Return Values

TCOK:	Normal termination
TCERR_WRITE:	File write error
TCERR_CODECONV:	Code conversion error

Reference

sw_TcOpenTrace

Example

The following example writes the string "File open error"to the trace file given by hwnd if the return value of sw_TcOpenTrace has been stored in the variable hwnd.

```
sw_TcWriteTrace $hwnd "File open error"
```

Execution Results/Output Format

```
Fri Apr  4 10:37:00 2014     18879 File open error
            (1)               (2)       (3)
```

Description of Output Items

The following explains items that are output in the above example:

(1) Time:

The time that the trace information was output in the form of "Day of the week Month Day Hour:Min:Sec Year".

(2) Process ID:

Process ID of Systemwalker Script

(3) String:

String specified by sw_TcWriteTrace

15.3.1.3 sw_TcCloseTrace (Close Trace File Command)

Description

This command closes a trace file and terminates the collection of traces.

Synopsis

```
sw_TcCloseTrace thandle
```

Option

thandle

Specify the handle obtained by sw_TcOpenTrace.

Return Values

TCOK: Normal termination

TCERR: Abnormal termination

Reference

sw_TcOpenTrace

Example

The following example closes the trace file given by hwnd if the return value of sw_TcOpenTrace has been stored in the variable hwnd.

```
sw_TcCloseTrace $hwnd
```

15.3.2 swotclsh (Run Script Command)

This command allows you to run any Systemwalker Script you created by itself. It is used to pre-check the operation of scripts.

Description

Using this command, you can execute Systemwalker Scripts. The scripts run by this command include control statements, commands and Systemwalker extended commands standard with Tcl/Tk offered by Systemwalker. It is used to debug the script you created.

Synopsis

```
swotclsh scriptfile [parameter1 parameter2 ... ]
```

Option

scriptfile

Specify the name of a script file to run with its full path.

Note

. . . .

While the extended commands offered by Systemwalker Operation Manager are executable, those offered by Systemwalker Centric Manager are not.

. . . .

parameter1 parameter2 ...

Specify the arguments passed to *scriptfile*.

Return Values

The return value can be manipulated in the description of script files to run. Note that the return values that scripts may return must be within a range from 0 to 128.

0 - 128:	Result code of execution script
129 - 255:	Error in script execution command
	The return value is the same as the error message (messages starting with "swotclsh:") ID, which is simultaneously output to the standard error output.

Command Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris	/usr/bin

Caution

If an error caused by a script file to run has occurred, a message indicating script file name or line number without its label is output to the standard error output. Below are examples of the error message output.

- Syntax error

```
wrong # args: no expression after "if" argument
    while executing
"if"
    (file "err.swt" line 5)
```

- Command execution error

```
invalid command name "noentry"
    while executing
"noentry"
    (file "err.swt" line 2)
```

- Command execution result error [Windows]

```
couldn't open "c:\aaa\bbb": no such file or directory
    while executing
"open {c:\aaa\bbb}"
    (file "err.swt" line 17)
```

- Command execution result error [UNIX]

```
couldn't open "/aaa/bbb": no such file or directory
    while executing
"open /aaa/bbb"
    (file "err.swt" line 17)
```

Examples

The following example performs a test run of the script file c:\home\scripts\a.swt.

```
swotclsh c:\home\scripts\a.swt
```

[UNIX]

The following example performs a test run of the script file /home/scripts/a.swt.

```
swotclsh /home/scripts/a.swt
```

Execution Results/Output Format

The output format of execution results depends on the processing of a.swt.

The output result appears on a terminal if a.swt is writing information to the standard output/standard error output within a script (puts stdout xxxxx, puts stderr xxxxx).

15.4 Examples of Scripts

This section describes Systemwalker Scripts using two examples.

Example 1

The following example determines whether a certain file exists and, if it exists, starts an application. If it does not exist, the script does not start the application.

- Input

    Specify the name of the file to determine its existence as an argument.

- Output

Exit code	Meaning
0	Normal termination
4	Argument is not specified.
8	The specified file does not exist.
16	Application exits with a non-zero code, or application outputs a message to the standard error output.

[Windows]

```
#
# This script is sample.
#
# check argument
# check the number of arguments.
if { $argc == 0 } {
        # If the number of arguments is zero, an error message is output.
        puts stderr "file name was not specified."
        # In this case, the script exits with exit code 4.
        exit 4
}

# set file name
# Set the string given by the argument to the variable "filename."
set filename  $argv

# check file
# Set to trap exceptions.
```

```
if {[catch {
        # Open the file.
        set file1 [open $filename r]
   }]} {
        # Output an error message if an exception is trapped.
        puts stderr "file not found."
        # In this case, the script exits with exit code 8.
        exit 8
}
# Close the opened file.
close $file1

# Set to trap exceptions.
if {[catch {
        # Invoke the application.   Store the standard output in the variable "outmsg."
        set outmsg [exec cmd /c {c:\usr\bin\appl001.exe}]
        # Output the data output to the standard output by the application to the job's
           standard.
        output.
        puts stdout $outmsg
   }]} {
        # Output process ID and exit code if an exception is trapped.
        puts stderr $errorCode
        # Output an error message if an exception is trapped.
        puts stderr $errorInfo
        # In this case, the script exits with exit code 16.
        exit 16
}
# The script exits with exit code 0 when the application has been successfully executed.
exit 0
```

[UNIX]

```
#
# This script is sample.
#
# check argument
# check the number of arguments.
if { $argc == 0 } {
        # If the number of arguments is zero, an error message is output.
        puts stderr "file name was not specified."
        # In this case, the script exits with exit code 4.
        exit 4
}

# set file name
# Set the string given by the argument to the variable "filename."
set filename  $argv
# check file
# Set to trap exceptions.
if {[catch {
      # Open the file.
      set file1 [open $filename r]
   }]} {
      # Output an error message if an exception is trapped.
      puts stderr "file not found."
      # In this case, the script exits with exit code 8.
      exit 8
}
# Close the opened file.
close $file1

# Set to trap exceptions.
```

```
if {[catch {
          # Invoke the application.   Store the standard output in the variable "outmsg."
          set outmsg [exec /usr/bin/appl001]
          # Output the data output to the standard output by the application to the job's
            standard.
          output.
          puts stdout $outmsg
     }]} {
          # Output process ID and exit code if an exception is trapped.
          puts stderr $errorCode
          # Output an error message if an exception is trapped.
          puts stderr $errorInfo
          # In this case, the script exits with exit code 16.
          exit 16
     }
}
# The script exits with exit code 0 when the application has been successfully executed.
exit 0
```

Note

Standard output of commands running in a script

The standard output is not inherited to the commands run in a script. Thus, if you change the point at which to invoke the application in example 1 as shown below, the standard output of the application will not be output as the job's standard output.

Note that, if a Systemwalker Script is executed as a scheduled job, you cannot check processing result messages output by the application run in the script on the [Previous log] sheet.

[Windows]

```
if {[catch {
      # Invoke the application.
      exec cmd /c {c:\usr\bin\appl001.exe}
}]} {
puts stderr $errorCode
puts stderr $errorInfo
exit 16
}
exit 0
```

[UNIX]

```
if {[catch {
      # Invoke the application.
      exec /usr/bin/appl001
}]} {
      puts stderr $errorCode
      puts stderr $errorInfo
      exit 16
}
exit 0
```

Example 2

The following example reads a file in which a list of names of files to be processed by an application and invokes the application for all filenames written in the file. The script passes a filename as an argument when invoking the application.

- Input

  Specify the name of the file with which the list of file names to be processed as arguments has been confirmed.

- Output

Exit code	Meaning
0	Normal termination
4	Argument is not specified
8	The specified file does not exist.
16	Application exits with a non-zero code, or application outputs a message to the standard error output.

[Windows]

```
#
# This script is sample.
#
# check argument
# check the number of arguments.
if { $argc == 0 } {
        # If the number of arguments is zero, an error message is output.
        puts stderr "file name was not specified."
        # In this case, the script exits with exit code 4.
        exit 4
}


# set file name
# Set the string given by the argument to the variable "filename."
set filename  $argv
# check file
# Set to trap exceptions.
if {[catch {
      # Open the file.
      set file1 [open $filename r]
   }]} {
      # Output an error message if an exception is trapped.
      puts stderr "file not found."
      # In this case, the script exits with exit code 8.
      exit 8
}
# Load a single line and execute loop until the end of the file is reached.
while { [gets $file1 param] != -1 } {
      # Set to trap exceptions.
      if {[catch {
          # Invoke the application. Pass the loaded strings at this time.
          # Store the standard output in the variable "outmsg."
          set outmsg [exec cmd /c {c:\usr\bin\appl002.exe} $param]
          # Output the data output to the standard output by the application to the
            job's standard output.
          puts stdout $outmsg
      }]} {
          # Output process ID and exit code if an exception is trapped.
          puts stderr $errorCode
          # Output an error message if an exception is trapped.
          puts stderr $errorInfo
          close $file1
          # In this case, the script exits with exit code 16.
          exit 16
      }
}
#Close the file.
close $file1
```

```
# The script exits with exit code 0 when the application has been successfully executed.
exit 0
```

[UNIX]

```
#
# This script is sample.
#
# check argument
# check the number of arguments.
if { $argc == 0 } {
        # If the number of arguments is zero, an error message is output.
        puts stderr "file name was not specified."
        # In this case, the script exits with exit code 4.
        exit 4
}
# set file name
# Set the string given by the argument to the variable "filename."
set filename  $argv
# check file
# Set to trap exceptions.
if {[catch {
      # Open the file.
      set file1 [open $filename r]
   }]} {
      # Output an error message if an exception is trapped.
      puts stderr "file not found."
      # In this case, the script exits with exit code 8.
      exit 8
}
# Load a single line and execute loop until the end of the file is reached.
while { [gets $file1 param] != -1 } {
      # Set to trap exceptions.
      if {[catch {
            # Invoke the application.  Pass the loaded strings at this time.
            # Store the standard output in the variable "outmsg."
            set outmsg [exec /usr/bin/appl002 $param]
            # Output the data output to the standard output by the application to the
              job's standard output.
            puts stdout $outmsg
        }]} {
            # Output process ID and exit code if an exception is trapped.
            puts stderr $errorCode
            # Output an error message if an exception is trapped.
            puts stderr $errorInfo
            close $file1
            # In this case, the script exits with exit code 16.
            exit 16
      }
}
#Close the file.
close $file1
# The script exits with exit code 0 when the application has been successfully executed.
exit 0
```

15.5 Examples of Script Debugging

This section describes the debugging of Systemwalker Scripts using the following two examples.

- Example of embedding a command used for debugging

- Example of how to change the trace level

Example of Embedding a Command Used for Debugging

Below is an example of "15.3 Commands Used in Debugging" that have been embedded in a script.

Example: A script that reads a file given by an argument

Script file	Level (Note)	Output contents
| ```
# Open trace file.
  # Delete "level 2" when the script actually
runs.
  set Trchwnd [sw_TcOpenTrace -level 2 /var/log/
G001]
  if {$Trchwnd < 0} {
       puts stderr "Trace open err"
       exit 1
  }
# Output a start log record.
  sw_TcWriteTrace $Trchwnd "JOB001 START"
# Acquire file name.
  set filename [lindex $argv 0]
  sw_TcWriteTrace -level 2 $Trchwnd "Input file
name: $filename"
  # Open operation control file.
  if {[catch {
       set cid [open $filename r]
       }]} {
       # When an open error is captured,
       # output an error log record.
       sw_TcWriteTrace $Trchwnd "File open err
(filename :$filename)"
       sw_TcCloseTrace $Trchwnd
       exit 1
  }
  sw_TcWriteTrace -level 2 $Trchwnd "File opened "
  # Load file.
  set rc [gets $cid buf]
  if {$rc == -1} {
       # Since the file is empty, set the default
in the read buffer.
       sw_TcWriteTrace -level 2 $Trchwnd "File
empty "
       set buf 100
  }
  sw_TcWriteTrace -level 2 $Trchwnd "Gets data:
$buf"
  close $cid
  ...
# Output an end log record.
  sw_TcWriteTrace $Trchwnd "JOB001 END"
  sw_TcCloseTrace $Trchwnd
``` | 1<br><br>2<br><br><br>1<br><br><br>2<br><br><br>2<br><br>2<br><br><br>1 | Operation log<br><br>Information in script (Route check)<br><br>Error log<br><br><br>Route check<br><br>Route check<br><br><br>Information in script (Route check)<br><br>Operation log |

Note:

The numbers in the "Level" column indicate trace levels.

Since the trace level defaults to 1, its specification in commands is omitted.

Example of How to Change the Trace Level

To specify the trace level, use options of extended trace commands that are entered in Systemwalker Scripts. It is difficult to directly change the content of a script if the trace level needs to be changed when the script is running on Systemwalker Operation Manager. Thus, the following means is provided to allow you to change the trace level without changing scripts.

- Specification using a script's argument

  A trace level is passed via an argument used when a script is invoked.

  Example: When the first argument represents a trace level [Windows]

```
# Obtain trace level from first argument
set trclevel [lindex $argv 0]
# Check the setting value
if {$trclevel < 0 || $trclevel > 2} {
        # Setting value error
        puts stderr "trace level err :$trclevel"
        exit 1
}
# Open trace.
set Trchwnd [sw_TcOpenTrace -level $trclevel {c:\var\log\G001.trc}]
...
sw_TcCloseTrace $Trchwnd
exit 0
```

  Example: When the first argument represents a trace level [UNIX]

```
# Obtain trace level from first argument
set trclevel [lindex $argv 0]
# Check the setting value
if {$trclevel < 0 || $trclevel > 2} {
        # Setting value error
        puts stderr "trace level err :$trclevel"
        exit 1
}
# Open trace.
set Trchwnd [sw_TcOpenTrace -level $trclevel /var/log/G001.trc]
...
sw_TcCloseTrace $Trchwnd
exit 0
```

- Specification using an external file

  A trace level is read from a file in which the trace level is entered.

  Example: When a trace level has been entered in the trace level definition file c:\home\etc\trclevel [Windows]

```
# Open, read, and close the trace level description file.
if {[catch {
        set cid [open {c:\home\etc\trclevel} r]
        gets $cid trclevel
        close $cid
        }]} {
# Trace level description file error
puts stderr "Can not read {c:\home\etc\trclevel}"
exit 1
}
# Check setting value.
if {$trclevel < 0 || $trclevel > 2} {
        # Setting value error
        puts stderr "trace level err :$trclevel"
        exit 1
}
# Open trace.
set Trchwnd [sw_TcOpenTrace -level $trclevel {c:\var\log\G001.trc}]
...
sw_TcCloseTrace $Trchwnd
exit 0
```

Example: When a trace level has been entered in the trace level definition file "/home/etc/trclevel" [UNIX]

```
# Open, read, and close the trace level description file.
if {[catch {
        set cid [open /home/etc/trclevel r]
        gets $cid trclevel
        close $cid
        }]} {
# Trace level description file error
puts stderr "Can not read /home/etc/trclevel"
exit 1
}
# Check setting value.
if {$trclevel < 0 || $trclevel > 2} {
        # Setting value error
        puts stderr "trace level err :$trclevel"
        exit 1
}
# Open trace.
set Trchwnd [sw_TcOpenTrace -level $trclevel /var/log/G001.trc]
...
sw_TcCloseTrace $Trchwnd
exit 0
```

Chapter 16 API Information

This chapter explains the information on APIs (Application Programming Interfaces) provided by Systemwalker Operation Manager.

⚑ Note

Notes for using APIs

- When using the APIs on the Windows x86 version

  Create 32-bit applications when using these APIs because they are 32-bit APIs.

- When using the APIs on the Windows x64 version

  Create 64-bit applications when using these APIs because they are 64-bit APIs.

- When using the APIs on the Linux x86 version

  Create 32-bit applications when using these APIs because they are 32-bit APIs.

- When using the APIs on the Linux x64 version

  Create 64-bit applications when using these APIs because they are 64-bit APIs.

- When using the APIs on the Solaris 32-bit version

  Create 32-bit applications when using these APIs because they are 32-bit APIs.

- When using the APIs on the Solaris 64-bit version

  Create 64-bit applications when using these APIs because they are 64-bit APIs.

⚑ Note

Notes on using APIs in a Windows x64 environment or Linux x64 environment

Use the x86 version of Systemwalker Operation Manager if 32-bit APIs are already being used as part of the migration from V13.3.1 or earlier.

⚑ Note

Notes on using APIs in a Solaris 64-bit environment

Use the 32-bit version of Systemwalker Operation Manager if 32-bit APIs are already being used as part of the migration from V13.3.1 or earlier.

⚑ Note

Notes on using 64-bit APIs

The sample programs provided in this chapter are 32-bit programs. Replace with 64-bit API files if the 64-bit APIs are to be used.

16.1 Calendar API

In this section, calendar APIs are explained.

Note that the following APIs, provided by previous versions of Systemwalker Operation Manager, are used in the same way as they are in this version. The operating environment, precautionary items and required files are the same as for this version.

- Calendar Registration API (Mp_AddCalendar)

- Calendar Update API (Mp_SetCalendar)

- Calendar Information Acquisition API (Mp_GetCalendar)

Previous versions of Systemwalker Operation Manager indicate the following versions:

[Windows]

SystemWalker/OperationMGR V4.0L10/V4.0L20

[UNIX]

SystemWalker/OperationMGR 4.0

The section below explains the operating environments, precautionary items, and required files which are common to calendar APIs.

Operating Environment

These APIs can only be operated on machines where the Systemwalker Operation Manager server is installed. They will not operate on machines where only the Systemwalker Operation Manager client is installed.

The APIs can be used only for Windows, Solaris and Linux x64 versions. They are not available for HP-UX, AIX and Linux x86 versions.

Cautions

- Do not perform optimization in a compiling environment.

- In a compiling environment, use Visual C++ (version 2005) as the compiler and "multithread (DLL) " as the runtime library. [Windows]

- Use the same version/levels of the LIB file and INCLUDE file.

Required Files

The following files are required to use the calendar APIs shown above:

[Windows x86]

LIB:

f3crhcap.lib

INCLUDE:

f3crhcap.h

[Windows x64]

LIB:

f3crhcap_x64.lib

INCLUDE:

f3crhcap_x64.h

[Solaris 32-bit]

LIB:

libcalapi.so

INCLUDE:

f3crhcap.h

[Solaris 64-bit]

LIB:

libcalapi_64.so

INCLUDE:

f3crhcap_64.

[Linux x64]

LIB:

libcalapi_x64.so

INCLUDE:

f3crhcap_x64.h

Each calendar API is stored in the LIB file, and the definitions and structures used by each API are declared in the INCLUDE file.

These files are stored in the following directories:

[Windows]

LIB:

Systemwalker Operation Manager installation directory: MpWalker.JM\lib

INCLUDE:

Systemwalker Operation Manager installation directory: MpWalker.JM\include

[Common to Solaris 32-bit, Solaris 64-bit and Linux x64]

LIB:

/opt/FJSVjmcal/lib

INCLUDE:

/opt/FJSVjmcal/include

16.1.1 Calendar Name List Acquisition API

This section explains the calendar name list acquisition API (Mp_ListCalendar).

Synopsis

```
int Mp_ListCalendar (char *APL_name,
             CAL_LIST *calendar_list) ;
```

Description

The Mp_ListCalendar function acquires a list of the calendar names registered on the server.

Parameters

APL_name

Specify the storage area address for the requested source application name (64 bytes or less).

Specify NULL at the end of the requested source application name. Do not use names that begin with "Mp_" or "_CAL_."

calendar_list

Specify the address for the calendar name acquisition information structure (CAL_LIST).

Return Values

Return Value	Meaning	Action
0	Normal termination.	-
-7	Calendar list area insufficient.	Review the size of the calendar list area using the procedure shown in "Remarks".
-10	Calendar service/daemon is not running.	Start the calendar service.
-11	There is not enough memory.	Use one of the following methods to increase the amount of available memory: - Close other applications. - Add more physical memory.

Remarks

You must first obtain the calendar name storage field structure as an array. Set the address and number of arrays (number of possible storage locations for calendar names) in the calendar name acquisition information structure, then call up this function.

If the number of calendars exceeds the number of locations where calendar names can be stored, since the actual number of calendars is set in cal_num in the calendar name acquisition information structure, the extra number of calendar name storage area structures are re-acquired. The calendar name storage area structure address (pcal_name) and the number of possible storage locations for calendar names (cal_area_num) are updated, and this function is called up again.

Format of Calendar Name Acquisition Information Structure (CAL_LIST)

```
typedef struct CAL_LIST_TAG {
    short       cal_area_num ;   // number of possible storage locations for calendar names
    short       cal_num ;        // actual number of calendars
    CAL_NAME_AREA *pcal_name ;   // calendar name storage area structure address
} CAL_LIST ;
```

cal_area_num

Specify the number of calendar name storage locations (specified by the number of calendars).

cal_num

The actual number of calendars (also set when insufficient) stored on the server is set.

pcal_name

Specify the address of calendar name storage area structure (CAL_NAME_AREA).

Format of Calendar Name Storage Area Structure (CAL_NAME_AREA)

```
typedef struct CAL_NAME_AREA_TAG {
     char   cal_name[25] ;              // calendar name storage area
} CAL_NAME_AREA ;
```

cal_name

The calendar name is set.

Program Example

Below is an example program of the calendar name list acquisition API:

```
#include <stdio.h>
#include <stdlib.h>
#include <malloc.h>
#include <string.h>

#include <f3crhcap.h>

#define CAL_NUM 20

extern void put_calendar_list( CAL_LIST * ) ;

void main() {

        int ret ;
        CAL_LIST calendar_list ;

      /* initialization */
      memset( &calendar_list, 0x00, sizeof( CAL_LIST) ) ;

      /* calendar name area acquisition */
     calendar_list.pcal_name = (CAL_NAME_AREA *) malloc (sizeof (CAL_NAME_AREA)
*CAL_NUM);
      if ( calendar_list.pcal_name == NULL ) {
                printf( "Lack of memory !!\n") ;
                exit ( 1 ) ;

      }
      /* calendar name area setting */
      calendar_list.cal_area_num = CAL_NUM ;

      /* calendar list acquisition */
     ret = Mp_ListCalendar( "Test Pro", &calendar_list) ;

     printf( "ret= %d  calendar num = %d\n", ret, calendar_list.cal_num ) ;

      /* calendar list post-acquisition processing */
     if ( ret == 0 )
           put_calendar_list( &calendar_list ) ;

      /* calendar name area return */
      free( calendar_list.pcal_name ) ;

    exit ( 0 ) ;
}
```

16.1.2 Calendar Registration API (Mp_AddCalendar2)

This section explains the calendar registration API (Mp_AddCalendar2).

Synopsis

```
int Mp_AddCalendar2 (char *APL_name,
          char *cal_name,
          CALENDAR_WDAT2 *cal_dat2) ;
```

Description

The Mp_AddCalendar2 function registers new calendars.

Holidays and substitute holidays for which the date changes from year to year can also be registered. Past dates can not be set.

Parameters

APL_name

Specify the storage area address for the requested source application name (64 bytes or less).

Specify NULL at the end of the requested source application name. Do not use names that begin with "Mp_" or "_CAL_."

cal_name

Specify the storage area address for the calendar name to be registered. Specify the calendar name with up to 24 bytes and set NULL at the end.

cal_dat2

Specify the address of the calendar update information structure (CALENDAR_WDAT2).

Return Values

Return Value	Meaning	Action
0	Normal termination.	-
-2	Same calendar name already exists.	Change the specified calendar name.
-5	SYSTEM_CALENDAR specified.	Specify something other than "SYSTEM_CALENDAR".
-6	Not system administrator (a user in the Administrators group on Windows or a superuser on UNIX).	Execute as a system administrator (a member of the Administrators group or the superuser).
-8	Error in calendar information.	Review the specified calendar information (the calendar update information structure "CALENDAR_WDAT2").
-10	Calendar service/daemon is not running.	Start the calendar service or daemon.
-11	There is not enough memory.	Use one of the following methods to increase the amount of available memory: - Close other applications. - Add more physical memory.
-15	Unspecifiable character specified for calendar name.	Do not use invalid characters in the calendar name. Calendar names cannot contain spaces, user-defined characters, or any of the following characters: ? \ * " < >	/ : , ;
-16	Error in file operation (write error).	Check that there is enough space available on the hard disk and that the hard disk is functioning normally.
-17	Illegal calendar name length.	Calendar names can be no more than 24 bytes long.

Remarks

Past dates are all set as working days.

If quoting the calendar in Jobscheduler, the relevant day is calculated as a working day.

Format of Calendar Update Information Structure (CALENDAR_WDAT2)

[32-bit version]

```
typedef struct CALENDAR_WDAT2_TAG {
  short year ;                     // This year in 4 digits
  short sub_holiday ;              // Specify whether to enable or disable substitute
                                   // holiday
  unsigned char holiday_w[12] ;    // Each years' day of the week information (for 12
                                   // months)
  unsigned long holiday_d[12] ;    // Each years' date information (for 12
                                   // months)
  unsigned char holiday_m[12][6] ; // Each years' week information (for 12 months x 6
                                   // weeks)
  short cal_dat[3][12][31] ;       // 3 years' (from year) calendar information

}CALENDAR_WDAT2 ;
```

[64-bit version]

```
typedef struct CALENDAR_WDAT2_TAG {
  short year ;                     // This year in 4 digits
  short sub_holiday ;              // Specify whether to enable or disable substitute
                                   // holiday
  unsigned char holiday_w[12] ;    // Each years' day of the week information (for 12
                                   // months)
  unsigned int holiday_d[12] ;     // Each years' date information (for 12
                                   // months)
  unsigned char holiday_m[12][6] ; // Each years' week information (for 12 months x 6
                                   // weeks)
  short cal_dat[3][12][31] ;       // 3 years' (from year) calendar information

}CALENDAR_WDAT2 ;
```

year

The standard year (first of the 3 years) in the Western calendar is set, with 4 digits, with 3 years' calendar information.

sub_holiday

Specify whether or not the substitute holiday is enabled. Specify the following values:

Value	Substitute holiday valid/invalid
0	Invalid
1	Valid

holiday_w

This is the days of the week information for each year (12 month array). Specify the day(s) of the week of every year's holidays. Set the first bit for Sunday, the second for Monday, and the third through seventh bits for Tuesday through Saturday respectively. Of the months specified in the array, the days of the week whose bits are set to ON will be holidays.

holiday_d

This is the date information for each year (12 month array). Specify the date(s) of every year's holidays. From the top, specify the first bit for the first day, the second bit for the second day, and the third through thirty first bits for the third through thirty first days respectively. Every year, of the months specified in the array, the dates whose bits are set to ON will be holidays.

holiday_m

This is the weekly information for each year (an array of 12 months x 6 weeks). Specify the holidays for which the date changes every year. The first array specifies the month (1 to 12), the second array specifies the week number (1 to 5 is first

to fifth week, 6 is the last week). Set the first bit for Sunday, the second for Monday, and the third through seventh bits for Tuesday through Saturday respectively. Every year, the days of the week whose bits are set to ON of the week specified in the second array of the month specified in the first array will be holidays.

cal_dat

This is a three-year calendar from the year specified in year. It reflects annual holiday information. Specify the following values:

Value	Calendar information
0	Impossible dates (February 30, etc.)
1	Holiday
2	Working day

Program Example

Below is an example program of the calendar registration API:

```
#include <stdio.h>
#include <stdlib.h>
#include <malloc.h>
#include <string.h>

#include <f3crhcap.h>

extern  void set_cal_dat2( CALENDAR_WDAT2 *) ;

void main() {

        int ret ;
        int count ;
        CALENDAR_WDAT2 cal_dat2 ;

        /*.......... initialization ..........*/
        memset( &cal_dat2, 0x00, sizeof( CALENDAR_WDAT2) ) ;

        /*.......... calendar information setting ..........*/
        /* this year's settings */
        cal_dat2.year = 2000 ;

        /* enable substitute holiday */
        cal_dat2.sub_holiday = 1 ;

        /* make Saturday every week a holiday */
        for ( count = 0 ; count < 12 ; count++ ) {
                cal_dat2.holiday_w[count] = (unsigned char)0x82 ;
        }

        /* set calendar information for 3 years */
        set_cal_dat2( &cal_dat2 ) ;

        /* calendar registration */
        ret = Mp_AddCalendar2( "Test Pro", "calendar_1", &cal_dat2 ) ;

        printf( "ret= %d\n", ret ) ;

        exit ( 0 ) ;

}
```

16.1.3 Calendar Update API (Mp_SetCalendar2)

This section explains the calendar update API (Mp_SetCalendar2).

Synopsis

```
int Mp_SetCalendar2 (char *APL_name,
                char *cal_name,
                CALENDAR_WDAT2 *cal_dat2) ;
```

Description

The Mp_SetCalendar2 function updates the calendar information (if the specified calendar does not exist, it is registered first). Holidays and substitute holidays for which the date changes from year to year can also be updated.

The SYSTEM_CALENDAR cannot be updated.

Past dates cannot be set.

Parameters

APL_name

Specify the storage area address for the requested source application name (64 bytes or less).

Specify NULL at the end of the requested source application name. Do not use names that begin with "Mp_" or "_CAL_."

cal_name

Specify the storage area address for the calendar name to be updated. Specify the calendar name with up to 24 bytes and set NULL at the end.

cal_dat2

Specify the address of the calendar update information structure (CALENDAR_WDAT2).

See "16.1.2 Calendar Registration API (Mp_AddCalendar2)" for details on the calendar update information structure.

Return Values

Return Value	Meaning	Action
0	Normal termination.	-
-5	SYSTEM_CALENDAR specified.	Specify something other than "SYSTEM_CALENDAR".
-6	Not system administrator (a user in the Administrators group on Windows or a superuser on UNIX).	Execute as a system administrator (a member of the Administrators group or the superuser).
-8	Error in calendar information.	Review the specified calendar information (the calendar update information structure "CALENDAR_WDAT2").
-10	Calendar service/daemon is not running.	Start the calendar service or daemon.
-11	There is not enough memory.	Use one of the following methods to increase the amount of available memory: - Close other applications. - Add more physical memory.
-15	Unspecifiable character specified in calendar name.	Do not use invalid characters in the calendar name.

Return Value	Meaning	Action
		Calendar names cannot contain spaces, user-defined characters or any of the following characters: ? \ * " < > | / : , ;
-16	Error in file operation (write error).	Check that there is enough space available on the hard disk and that the hard disk is functioning normally.
-17	Illegal calendar name length.	Calendar names can be no more than 24 bytes long.

Program Example

Below is an example program of the calendar update API:

```c
#include <stdio.h>
#include <stdlib.h>
#include <malloc.h>
#include <string.h>

#include <f3crhcap.h>

extern  void set_cal_dat2( CALENDAR_WDAT2 *) ;

void main() {

        int ret ;
        int count ;
        CALENDAR_WDAT2 cal_dat2 ;

        /*..........initialization ..........*/
        memset( &cal_dat2, 0x00, sizeof( CALENDAR_WDAT2) ) ;

        /*..........calendar information setting..........*/
        /*this year's settings */
        cal_dat2.year = 2000 ;

        /*  enable substitute holiday */
        cal_dat2.sub_holiday = 1 ;

        /*make Sunday every week a holiday */
        for ( count = 0 ; count < 12 ; count++ ){
                cal_dat2.holiday_w[count] = (unsigned char)0x80 ;
        }

        /*set calendar information for 3 years */
        set_cal_dat2( &cal_dat2 ) ;

        /*calendar update */
        ret = Mp_SetCalendar2( "Test Pro", "calendar_1", &cal_dat2 ) ;

        printf( "ret= %d\n", ret ) ;

        exit ( 0 ) ;

}
```

16.1.4 Calendar Update API (Mp_SetCalendar3)

This section explains the calendar update API (Mp_SetCalendar3).

Synopsis

```
int Mp_SetCalendar3 (char *APL_name,
                     char *cal_name,
                     CALENDAR_WDAT3 *cal_dat3) ;
```

Description

The Mp_SetCalendar3 function updates the calendar information (if the specified calendar does not exist, it is handled as a calendar to be registered) and holidays and substitute holidays for which the date changes from year to year. Update start date applied to job nets can also be updated.

The SYSTEM_CALENDAR cannot be updated.

If update start date is omitted, update will be applied to all dates.

Past dates cannot be set.

Parameters

APL_name

Specify the storage area address for the requested source application name (64 bytes or less).

Specify NULL at the end of the requested source application name. Do not use names that begin with "Mp_" or "_CAL_."

cal_name

Specify the storage area address for the calendar name to be updated. Specify the calendar name with up to 24 bytes and set NULL at the end.

cal_dat3

Specify the address of the calendar update information structure (CALENDAR_WDAT3).

Return Values

Return Value	Meaning	Action
0	Normal termination.	-
-5	SYSTEM_CALENDAR specified.	Specify something other than "SYSTEM_CALENDAR".
-6	Not system administrator (a user in the Administrators group on Windows or a superuser on UNIX).	Execute as a system administrator (a member of the Administrators group or the superuser).
-8	Error in calendar information.	Review the specified calendar information (the calendar update information structure "CALENDAR_WDAT3").
-10	Calendar service/daemon is not running.	Start the calendar service or daemon.
-11	There is not enough memory.	Use one of the following methods to increase the amount of available memory: - Close other applications. - Add more physical memory.
-15	Unspecifiable character specified in calendar name.	Do not use invalid characters in the calendar name.

Return Value	Meaning	Action
		Calendar names cannot contain spaces, user-defined characters or any of the following characters: ? \ * " < > | / : , ;
-16	Error in file operation (write error).	Check that there is enough space available on the hard disk and that the hard disk is functioning normally.
-17	Illegal calendar name length.	Calendar names can be no more than 24 bytes long.

Format of Calendar Update Information Structure (CALENDAR_WDAT3)

[32-bit version]

```
typedef struct CALENDAR_WDAT3_TAG {
 short year ;                      // This year in 4 digits
 short sub_holiday ;               // Specify whether to enable or disable substitute
                                   // holiday
 unsigned char holiday_w[12] ;     // Each year's day of the week information (for 12
                                   // months)
 unsigned long holiday_d[12] ;     // Each year's date information (for 12 months)
 unsigned char holiday_m[12][6] ;  // Each year's week information (for 12 months x 6
                                   // weeks)
 short cal_dat[3][12][31] ;        // 3 year's (from year) calendar information
 short start_y ;                   // The year of the update start date in 4 digits
 short start_m ;                   // The month of the update start date (1 to 12)
 short start_d ;                   // The day of the update start date (1 to 31)
}CALENDAR_WDAT3 ;
```

[64-bit version]

```
typedef struct CALENDAR_WDAT3_TAG {
 short year ;                      // This year in 4 digits
 short sub_holiday ;               // Specify whether to enable or disable substitute
                                   // holiday
 unsigned char holiday_w[12] ;     // Each year's day of the week information (for 12
                                   // months)
 unsigned int holiday_d[12] ;     // Each year's date information (for 12 months)
 unsigned char holiday_m[12][6] ;  // Each year's week information (for 12 months x 6
                                   // weeks)
 short cal_dat[3][12][31] ;        // 3 year's (from year) calendar information
 short start_y ;                   // The year of the update start date in 4 digits
 short start_m ;                   // The month of the update start date (1 to 12)
 short start_d ;                   // The day of the update start date (1 to 31)
}CALENDAR_WDAT3 ;
```

year

The standard year (first of the 3 years) in the Western calendar is set, with 4 digits, with 3 years calendar information.

sub_holiday

Specify whether or not the substitute holiday is enabled. Specify the following values:

Value	Substitute holiday valid/invalid
0	Invalid

Value	Substitute holiday valid/invalid
1	Valid

holiday_w

This is the days of the week information for each year (12 month array). Specify the day(s) of the week of every year's holidays. Set the first bit for Sunday, the second for Monday, and the third through seventh bits for Tuesday through Saturday respectively. Of the months specified in the array, the days of the week whose bits are set to ON will be holidays.

holiday_d

This is the date information for each year (12 month array). Specify the date(s) of every year's holidays. From the top, specify the first bit for the first day, the second bit for the second day, and the third through thirty first bits for the third through thirty first days respectively. Every year, of the months specified in the array, the dates whose bits are set to ON will be holidays.

holiday_m

This is the weekly information for each year (an array of 12 months x 6 weeks). Specify the holidays for which the date changes every year. The first array specifies the month (1 to 12), the second array specifies the week number (1 to 5 is first to fifth week, n6 is the last week). Set the first bit for Sunday, the second for Monday, and the third through seventh bits for Tuesday through Saturday respectively. Every year, the days of the week whose bits are set to ON of the week specified in the second array of the month specified in the first array will be holidays.

cal_dat

This is a three-year calendar from the year specified in year. It reflects annual holiday information. Specify the following values:

Value	Calendar information
0	Impossible dates (February 30, etc.)
1	Holiday
2	Working day

start_y

Specify the update start date (from the current year to two years ahead) in 4 digits in western calendar.

Specify 0 when ignoring the update start date.

start_m

Specify the month of the update start date (1 to 12).

Specify 0 when ignoring the update start date.

start_d

Specify the day of the update start date (1 to 31).

Specify 0 when ignoring the update start date.

Program Example

Below is an example program of the calendar update API:

```
#include <stdio.h>
#include <string.h>
#include <f3crhcap.h>
```

```
int main( int argc,char** argv)
{
/*..........variable declaration........ */
          int ret ;
          int count ;
          CALENDAR_WDAT3 cal_dat3 ;

          /*..........initialization ..........*/
          memset( &cal_dat3, 0x00, sizeof( CALENDAR_WDAT3) ) ;

          /*..........set calendar information ..........*/
          /*set this year */
          cal_dat3.year = 2006 ;

          /*  enable substitute holiday */
          cal_dat3.sub_holiday = 1 ;

          /*setting every Saturday and Sunday as holidays */
          for ( count = 0 ; count < 12 ; count++ ){
                    cal_dat3.holiday_w[count] = (unsigned char)0x82 ;
          }
          /* set the update start date (update is applied from 2006/08/09) */
          cal_dat3.start_y = 2006 ;
          cal_dat3.start_m = 8 ;
          cal_dat3.start_d = 9 ;

          /*update calendar */
          ret = Mp_SetCalendar3( "Test Pro", "calendar_1", &cal_dat3 ) ;

          printf( "ret= %d\n", ret ) ;

          return 0 ;
}
```

16.1.5 Calendar Information Acquisition API

This section explains the calendar information acquisition API (Mp_GetCalendar2).

Synopsis

```
int Mp_GetCalendar2 (char *APL_name,
          char *cal_name,
          short mode,
          CALENDAR_RDAT2 *cal_dat2) ;
```

Description

The Mp_GetCalendar2 function acquires calendar information. Holidays and substitute holidays which change from year to year can also be acquired.

When acquiring the information on SYSTEM_CALENDAR on Windows, you can specify whether operation pattern type is included or only information on working days or holidays is acquired. [Windows]

Parameters

APL_name

Specify the storage area address for the requested source application name (64 bytes or less).

Specify NULL at the end. Do not use names that begin with "Mp_" or "_CAL_."

cal_name

Specify the calendar name storage area address for the calendar information to be acquired. Specify the calendar name within 24 bytes and set NULL at the end.

mode

[Windows]

Specify whether or not you acquire the pattern type as follows (this is valid when you acquire the SYSTEM_CALENDAR information).

0:

The pattern type is not acquired.

1:

The pattern type is acquired.

[UNIX]

Specify 0.

cal_dat2

Specify the address of the calendar reference information structure (CALENDAR_RDAT2).

Return Values

Return Value	Meaning	Action
0	Normal termination.	-
-1	Calendar does not exist.	Specify the name of an existing calendar.
-10	Calendar service/daemon is not running.	Start the calendar service or daemon.
-11	There is not enough memory.	Use one of the following methods to increase the amount of available memory: - Close other applications. - Add more physical memory.
-16	Error in file operation (read error).	Check that there is enough space available on the hard disk and that the hard disk is functioning normally.
-17	Illegal calendar name length.	Calendar names can be no more than 24 bytes long.

Format of Calendar Reference Information Structure (CALENDAR_RDAT2)

[32-bit version]

```
typedef struct CALENDAR_RDAT2_TAG {
 short year ;                     // This year in 4 digits
 short sub_holiday ;              //  Specify whether to enable substitute holiday
 unsigned char holiday_w[12] ;    // Each year's day of the week information (for 12
                                  // months)
 unsigned long holiday_d[12] ;    // Each year's date information (for 12 months)
 unsigned char holiday_m[12][6] ; // Each year's week information (for 12 months x 6
                                  // weeks)
 short cal_dat[4][12][31] ;       // 4 year's (from year-1) calendar information
} CALENDAR_RDAT2 ;
```

[64-bit version]

```
typedef struct CALENDAR_RDAT2_TAG {
 short year ;                      // This year in 4 digits
 short sub_holiday ;               //  Specify whether to enable substitute holiday
 unsigned char holiday_w[12] ;     // Each year's day of the week information (for 12
                                   // months)
 unsigned int holiday_d[12] ;     // Each year's date information (for 12 months)
 unsigned char holiday_m[12][6] ;  // Each year's week information (for 12 months x 6
                                   // weeks)
 short cal_dat[4][12][31] ;        // 4 year's (from year-1) calendar information
} CALENDAR_RDAT2 ;
```

year

The standard year (the second of the 4 years) for the set 4 years calendar information is set in Western calendar with 4 digits. Normally, this year is set as the standard.

sub_holiday

Sets whether or not substitute holidays are enabled. The following values are set:

Value	Substitute holiday valid/invalid
0	Invalid
1	Valid

holiday_w

This is the days of the week information for each year (12 month array). Each year's holidays' day(s) of the week are set. Set the first bit for Sunday, the second for Monday, and the third through seventh bits for Tuesday through Saturday respectively. Of the months specified in the array, the days of the week whose bits are set to ON will be holidays.

holiday_d

This is the date information for each year (12 month array). Each year's holiday's dates are set. From the top, specify the first bit for the first day, the second bit for the second day, and the third through thirty first bits for the third through thirty first days respectively. Every year, of the months specified in the array, the dates whose bits are set to ON will be holidays.

holiday_m

This is the weekly information for each year (an array of 12 months x 6 weeks). Specify the holidays for which the date changes every year. The first array specifies the month (1 to 12), the second array specifies the week number (1 to 5 is first to fifth week, n6 is the last week). Set the first bit for Sunday, the second for Monday, and the third through seventh bits for Tuesday through Saturday respectively. Every year, the days of the week whose bits are set to ON of the week specified in the second array of the month specified in the first array will be holidays.

cal_dat

A four-year calendar from the year before year is set. The information to be set is shown below:

[Windows]

Value	Calendar information
0	Impossible dates (February 30, etc.)
1	Holiday
2	Working day (when the specified mode is 0)
3	Standard pattern (when the specified mode is 1)

Value	Calendar information
4	Non-standard pattern 1 (when the specified mode is 1)
5	Non-standard pattern 2 (when the specified mode is 1)
6	Non-standard pattern 3 (when the specified mode is 1)
7	Non-standard pattern 4 (when the specified mode is 1)
8	Non-standard pattern 5 (when the specified mode is 1)
9	Non-standard pattern 6 (when the specified mode is 1)
10	Non-standard pattern 7 (when the specified mode is 1)
11	Non-standard pattern 8 (when the specified mode is 1)
12	Separate 1 (when the specified mode is 1)
11+n	Separate n (when the specified mode is 1)
2xxx	Holiday/Separate (when the specified mode is 1). xxx corresponds to the Separate ID.

[UNIX]

Value	Calendar information
0	Impossible dates (February 30, etc.)
1	Holiday
2	Working day

Program Example

Below is an example program of the calendar information acquisition API

```
#include <stdio.h>
#include <stdlib.h>
#include <malloc.h>
#include <string.h>

#include <f3crhcap.h>

extern  void put_cal_dat2( CALENDAR_RDAT2*) ;

void main() {

        int ret ;
        CALENDAR_RDAT2 cal_dat2 ;

        /*.......... initialization............*/
        memset( &cal_dat2, 0x00, sizeof( CALENDAR_RDAT2) ) ;

        /* calendar information acquisition */
        ret = Mp_GetCalendar2( "Test Pro", "calendar_1", 0, &cal_dat2 ) ;

        printf( "ret= %d\n", ret ) ;

        if ( ret == 0 )
                put_cal_dat2( &cal_dat2 ) ;

        exit ( 0 ) ;
}
```

16.1.6 Day Change Time Acquisition API

This section gives an explanation of the day change time acquisition API (Mp_GetDayStartTime).

Synopsis

```
int Mp_GetDayStartTime (char *APL_name,
                char *start_time) ;
```

Description

The Mp_GetDayStartTime function acquires day change time.

Parameters

APL_name

Specify the storage area address for the requested source application name (64 bytes or less).

Specify NULL at the end. Do not use names that begin with "Mp_" or "_CAL_."

start_time

Specify the day change time storage area address (HHMM format, 5 byte area). NULL is specified at the end.

Return Values

Return Value	Meaning	Action
0	Normal termination.	-
-10	Calendar service/daemon is not running.	Start the calendar service or daemon.

Program Example

Below is an example program of the day change time acquisition API:

```
#include <stdio.h>
#include <stdlib.h>
#include <malloc.h>
#include <string.h>

#include <f3crhcap.h>

void main() {

        int ret ;
        char start_time[5] ;

        /*.......... initialization..........*/
        memset( start_time, 0x00, sizeof( start_time) ) ;

        /* day change time acquisition */
        ret = Mp_GetDayStartTime( "Test Pro", start_time ) ;

        printf( "ret= %d  start_time=%s\n", ret, start_time ) ;

        exit ( 0 ) ;
}
```

16.2 Jobscheduler API

This section explains the APIs provided by Jobscheduler.

Operating environments and precautionary notes that are common to Jobscheduler APIs are shown below.

Operating Environment

These APIs can only operate on machines where a Systemwalker Operation Manager server is installed. They will not operate on machines where only the Systemwalker Operation Manager client is installed.

The job net manipulation API, job net manipulation (startup parameter specification) API, group manipulation API and job net start time change API are available for Windows, Solaris and Linux versions.

EE

The job net manipulation API/EE, job net manipulation (startup parameter specification) API/EE, group manipulation API/EE and job net start time change API/EE are available for Windows, Solaris and Linux versions.

Cautions

- Use VisualC++ (version 2005) for compiler and multithread (DLL) for runtime library as the compile environment. [Windows]

- Use the same version/levels of the LIB file and INCLUDE file.

- Do not call Jobscheduler APIs from multiple threads in a single process simultaneously.

- Do not execute the APIs from an environment where standard output and standard error output are closed.

- Jobscheduler APIs are not synchronized with the Operation Management Server. Due to this, the completion of the APIs does not guarantee the completion of API processing executed on the server.

- Also, there may be a slight time lag during which the API is executed and the processing is accepted on the server.

- When the Extended User Management function is enabled, the access right authentication depends on the OS user that executes the API [UNIX]:

    - If the OS user is a system administrator:

    The API is executed with the Operation Manager administrator privileges (has update rights for all projects).

    - If the OS user is a project owner:

    The API is executed with the assumption that the owner has update rights over their projects.

    - If the OS user is neither a system administrator nor project owner:

    The API is executed if the OS user that executed the API is included in the OS users corresponding to all Operation Manager users that have access to the project, and the API is executed if the user has the privileges. If the OS user that executed the API corresponds to more than one Operation Manager user, the OS user will be given the most powerful privileges (update rights > change and operation rights > reference rights) among the privileges that the corresponding Operation Manager users have. If the corresponding Operation Manager user has both change rights and operation rights, then the OS user will also have both privileges.

- When the Systemwalker authentication repository is enabled, the access right authentication depends on the OS user that executes the API:

    - If the OS user is a system administrator

    The API is executed with the Operation Manager administrator privileges (has update rights for all projects).

    - If the OS user is a project owner

    The API is executed with update rights over the project they are the owner of.

    - If the OS user is neither a system administrator nor a project owner

    The API is executed, if the OS user has access rights over the project (update rights, change rights, operation rights, and reference rights).

16.2.1 Job Net Manipulation API

This section explains the job net manipulation API (Mp_JobschControl).

Synopsis

[Windows]

```
#include "f3csbJobschControl.h"
int Mp_JobschControl (char *project,
            char *netname,
            int action,
            int rerunopt) ;
```

[UNIX]

```
#include "jobschapi.h"
int Mp_JobschControl (char *project,
            char *netname,
            int action,
            int rerunopt) ;
```

Description

This API is an interface to use the jobschcontrol command from the C interface.

Parameters

project

Specify the name of the project where the job net to be manipulated exists. Note that you can only specify the names of the projects that the API execution user can use.

This can be omitted if there is only one project having the job net that the API execution user can use. In this case, NULL must be specified.

netname

Specify the name of the job net to be manipulated. This cannot be omitted.

action

Specify the job net manipulation method from the following:

1:

Start

2:

Cancel

3:

Restart

4:

Pause

5:

Continue

6:

Disable

7:

Enable

8:

Confirm

9:

Reinstate

10:

Revoke

rerunopt

Specify the option when 3 (Restart) is selected for action. If 3 (Restart) is not selected for action, 0 should be specified.

0:

Performs normal re-activation. Activates from an abended job. If the restart point is set at a job located before the abended job, restart is executed from that job.

1:

Restarts the job net from the top job.

2:

Re-activates a job net set in "Invalid at next restart" mode in relation to an abended job.

(Skip the abended job and activates the next job).

Jobs forcibly terminated are handled in the same way as abended jobs are handled.

Return Values

Return Value	Meaning	Action
0	Normal termination.	-
4	Mode in which job net cannot function.	Perform the operation again after the status of the job net has changed.
8	Jobscheduler service/daemon is not running.	Start the Jobsheduler service or daemon and perform the operation again.
12	Error in parameter setting or invalid user privilege.	Take one of the following measures: - Set the correct parameter (project name, job net name, operation method or restart option). - Grant the executing user access privileges to operate the project or higher privileges.
16	Fatal error occurred.	A system error may have occurred. Check the event log or syslog for any system abnormalities. If an error occurs, collect Jobscheduler information using the Maintenance Information Collection Tool, and then contact a Fujitsu SE. If none are found, the problem may only have been temporary, so perform the operation again.

Required Files

The following files are required for use of the job net manipulation API shown above:

[Windows x86]

LIB:

f3csbJobschControl.lib

INCLUDE:

f3csbJobschControl.h

[Windows x64]

LIB:

f3csbJobschControl_x64.lib

INCLUDE:

f3csbJobschControl_x64.h

[Common to Solaris 32-bit and Linux x86]

LIB:

libjobsch.so

INCLUDE:

jobschapi.h

[Solaris 64-bit]

LIB:

libjobsch_64.so

INCLUDE:

jobschapi_64.h

[Linux x64]

LIB:

libjobsch_x64.so

INCLUDE:

jobschapi_x64.h

The job net manipulation API is stored in the LIB file, and the definitions and structures used by this API are declared in the INCLUDE file. These files are stored under the following directories.

[Windows]

LIB:

Systemwalker Operation Manager installation directory: MpWalker.JM\lib

INCLUDE:

Systemwalker Operation Manager installation directory: MpWalker.JM\include

[Common to Solaris and Linux]

LIB:

/opt/FJSVJOBSC/usr/lib

INCLUDE:

/opt/FJSVJOBSC/usr/include

Program Example

Below is an example program of the job net manipulation API:

[Windows]

```
#include <windows.h>
#include <stdio.h>
#include <stdlib.h>
#include "f3csbJobschControl.h"

void main()
{
        int ret;

        ret = Mp_JobschControl("user1",          // project name
                               "net001",          // jobnet name
                               1,                 // action : start
                               0);                //  rerun opt : none

        if (ret == 0) {
            printf("jobnet(net001) is controlled normaly.\n");
        }
        else{
            printf("ERROR: return code = %d\n", ret);
            exit(1);
        }
        exit(0);
}
```

[UNIX]

```
#include <stdio.h>
#include <stdlib.h>#include "jobschapi.h"

main()
{
        int ret;

        ret = Mp_JobschControl("user1",          /* project name*/
                               "net001",          /* jobnet name*/
                               1,                 /* action : start */
                               0);                /* rerun opt : none*/

        if (ret == 0) {
            printf("jobnet(net001) was normally controlled.\n");
         }
        else{
            printf("ERROR: return code = %d\n", ret);
            exit(1);
        }
        exit(0);
}
```

EE

16.2.2 Job Net Manipulation API/EE

This section explains the job net manipulation API/EE (Mp_JobschControlEx)

Synopsis

[Windows]

```
#include "f3csbJobschControlEx.h"
int Mp_JobschControlEx (char *project,
            char *netname,
```

```
                    int action,
                    int rerunopt,
                    int system_num) ;
```

[UNIX]

```
#include "jobschapiex.h"
int Mp_JobschControlEx (char *project,
                    char *netname,
                    int action,
                    int rerunopt,
                    int system_num) ;
```

Description

This API is an interface to use the jobschcontrol command from the C interface in Systemwalker Operation Manager EE.

Parameters

project

Specify the name of the project where the job net to be manipulated exists. Note that you can only specify the names of the projects that the API execution user can use. This can be omitted if there is only one project having the job net that the API execution user can use. In this case, NULL must be specified.

netname

Specify the name of the job net to be manipulated. This cannot be omitted.

action

Specify the job net manipulation method from the following:

1:

Start

2:

Cancel

3:

Restart

4:

Pause

5:

Continue

6:

Disable

7:

Enable

8:

Confirm

9:

Reinstate

10:

Revoke

rerunopt

Specify the option when 3 (Restart) is selected for action. If 3 (Restart) is not selected for action, 0 should be specified.

0:

Performs normal re-activation. Activates from an abended job. If the restart point is set at a job located before the abended job, restart is executed from that job.

1:

Restarts the job net from the top job.

2:

Re-activates a job net set in "Invalid at next restart" mode in relation to an abended job.

(Skips the abended job and activates the next job).

Jobs forcibly terminated are handled in the same way as abended jobs are handled.

system_num

Specify operating target subsystem number using a range from 0 to 9.

Return Values

Return Value	Meaning	Action
0	Normal termination.	-
4	Mode in which job net cannot function.	Perform the operation again after the status of the job net has changed.
8	Jobsheduler service/daemon is not running.	Start the Jobsheduler service or daemon and perform the operation again.
12	Error in parameter setting or invalid user privilege.	Take one of the following measures: - Set the correct parameter (project name, job net name, operation method, restart option or subsystem number). - Grant the executing user access privileges to operate the project or higher privileges.
16	Fatal error occurred.	A system error may have occurred. Check the event log or syslog for any system abnormalities. If an error occurs, collect Jobscheduler information using the Maintenance Information Collection Tool, and then contact a Fujitsu SE. If none are found, the problem may only have been temporary, so perform the operation again.

Required Files

The following files are required for use of the job net manipulation API shown above:

[Windows x86]

LIB:

f3csbJobschControlEx.lib

INCLUDE:

f3csbJobschControlEx.h

[Windows x64]

LIB:

f3csbJobschControlEx_x64.lib

INCLUDE:

f3csbJobschControlEx_x64.h

[Common to Solaris 32-bit and Linux x86]

LIB:

libjobschex.so

INCLUDE:

jobschapiex.h

[Solaris 64-bit]

LIB:

libjobschex_64.so

INCLUDE:

jobschapiex_64.h

[Linux x64]

LIB:

libjobschex_x64.so

INCLUDE:

jobschapiex_x64.h

The job net manipulation API is stored in the LIB file, and the definitions and structures used by the job net manipulation API are declared in the INCLUDE file. These files are stored under the following directories.

[Windows]

LIB:

Systemwalker Operation Manager installation directory: MpWalker.JM\lib

INCLUDE:

Systemwalker Operation Manager installation directory: MpWalker.JM\include

[Common to Solaris and Linux]

LIB:

/opt/FJSVJOBSC/usr/lib

INCLUDE:

/opt/FJSVJOBSC/usr/include

Program Example

Below is an example program of the job net manipulation API/EE:

[Windows]

```
#include <windows.h>
#include <stdio.h>
```

```
#include <stdlib.h>#include  "f3csbJobschControlEx.h"

void main()
{
    int ret;

    ret = Mp_JobschControlEx("user1",          // project name
                             "net001",          // jobnet name
                             1,                 // action : start
                             0,                 // rerun opt : none
                             1);                // sub system number

    if (ret == 0) {
        printf("jobnet(net001) is controlled normally.\n");
    }
    else{
        printf("ERROR: return code = %d\n", ret);
        exit(1);
    }
    exit(0);
}
```

[UNIX]

```
#include <stdio.h>
#include <stdlib.h>
#include "jobschapiex.h"

main ()
{

    int ret;

    ret = Mp_JobschControlEx ("user1",        /* project name*/
                              "net1",         /* jobnet name*/
                              1,              /* action : start */
                              0,              /* rerun opt : none*/
                              1);             /* sub system number*/

    if (ret == 0) {
        printf ("jobnet(net1) was normally controlled.\n");
    }
    else{
        printf ("ERROR: return code = %d\n", ret);
        exit(1);
    }
    exit(0);
}
```

16.2.3 Job Net Manipulation (Startup Parameter Specification) API

This section explains the job net manipulation (startup parameter specification) API (Mp_JobschControlParam).

<u>**Synopsis**</u>

[Windows]

```
#include "f3csbJobschControl.h"
int Mp_JobschControlParam (char *project,
            char *netname,
```

```
              char *parameter,
              int action,
              int rerunopt,) ;
```

[UNIX]

```
#include "jobschapi.h"
int Mp_JobschControlParam (char *project,
              char *netname,
              char *parameter,
              int action,
              int rerunopt,) ;
```

Description

This API is an interface to use the jobschcontrol command from the C interface. Using the API allows you to pass parameters to each job in the job net to start.

Only the job nets with the Job Execution Control attribute can be used.

Parameters

project

Specify the name of a project in which the job net to be manipulated exists. Note that you can only specify the project name of the projects that the API execution user can use.

This can be omitted if there is only one project registered in the job net that the API execution user can use. In this case, NULL must be specified.

netname

Specify the name of the job net to be manipulated. This cannot be omitted.

parameter

Specify a parameter to be passed to each job of the job net to be started by a string within 200 bytes for startup or restart.

Note that for a job net manipulation other than startup and restart, specify NULL (this value is ignored when a string is specified).

When a string is specified:

The parameter specified is appended to the end of job argument.

When NULL is specified:

At restart, the parameter specified when the job was started last is appended to the end of job argument. At startup, no parameter is specified.

action

Specify the job net manipulation method from the following:

1:

Start

2:

Cancel

3:

Restart

4:

Pause

5:

> Continue

6:

> Disable

7:

> Enable

8:

> Confirm

9:

> Reinstate

10:

> Revoke

rerunopt

Specify the option when 3 (Restart) is selected for action. If 3 (Restart) is not selected for action, 0 should be specified.

0:

> Performs normal re-activation. Activates from an abended job. If the restart point is set at a job located before the abended job, restart is executed from that job.

1:

> Restarts the job net from the top job.

2:

> Re-activates a job net set in "Invalid at next restart" mode in relation to an abended job.
>
> (Skips the abended job and activates the next job).
>
> Jobs forcibly terminated are handled in the same way as abended jobs are handled.

Return Values

Return Value	Meaning	Action
0	Normal termination.	-
4	Mode in which job net cannot function.	Perform the operation again after the status of the job net has changed.
8	Jobscheduler service/daemon is not running.	Start the Jobsheduler service or daemon and perform the operation again.
12	One of the following: - Error in parameter setting or invalid user privilege. - Error in user access privileges setting - Either the specified project or job net does not exist, or the job net has been copied using "copy and startup" operation, or the job net does not have	Take one of the following measures: - Set the correct parameter (project name, job net name, parameters passed to job, operation method or restart option). - Grant the executing user access privileges to operate the project or higher privileges. - Specify a job net that can be operated on.

Return Value	Meaning	Action
	Job Execution Control attributes.	
16	Fatal error occurred.	A system error may have occurred. Check the event log or syslog for any system abnormalities. If an error occurs, collect Jobscheduler information using the Maintenance Information Collection Tool, and then contact a Fujitsu SE. If none are found, the problem may only have been temporary, so perform the operation again.
20	Job nets that have been started with variable parameters cannot be restarted using this API.	Use the job net manipulation API to restart the job net using the variable parameter specified when the job net was started.

Required Files

The following files are required for use of the job net manipulation API shown above:

[Windows x86]

LIB:

f3csbJobschControl.lib

INCLUDE:

f3csbJobschControl.h

[Windows x64]

LIB:

f3csbJobschControl_x64.lib

INCLUDE:

f3csbJobschControl_x64.h

[Common to Solaris 32-bit and Linux x86]

LIB:

libjobsch.so

INCLUDE:

jobschapi.h

[Solaris 64-bit]

LIB:

libjobsch_64.so

INCLUDE:

jobschapi_64.h

[Linux x64]

LIB:

libjobsch_x64.so

INCLUDE:

jobschapi_x64.h

The job net manipulation API is stored in the LIB file, and the definitions and structures used by the job net manipulation API are declared in the INCLUDE file. These files are stored under the following directories.

LIB:

Systemwalker Operation Manager installation directory: MpWalker.JM\lib

INCLUDE:

Systemwalker Operation Manager installation directory: MpWalker.JM\include

[Common to Solaris and Linux]

LIB:

/opt/FJSVJOBSC/usr/lib

INCLUDE:

/opt/FJSVJOBSC/usr/include

Caution

- Job nets that have been started using variable parameters cannot be restarted using this API.

Program Example

Below is an example program of the job net manipulation API:

[Windows]

```
#include <windows.h>
#include <stdio.h>
#include <stdlib.h>
#include "f3csbJobschControl.h"

void main()
{
    int ret;

    ret = Mp_JobschControlParam("user1",              // project name
                                "net001",             // jobnet name
                                "param1",             // parameter
                                1,                    // action : start
                                0);                   // rerun opt : none

    if (ret == 0) {
        printf("jobnet(net001) was normally controlled.\n");
    }
    else{
        printf("ERROR: return code = %d\n", ret);
        exit(1);
    }
    exit(0);
}
```

[UNIX]

```
#include <stdio.h>
#include <stdlib.h>
#include "jobschapi.h"

main()
{
    int ret;
```

```
    ret = Mp_JobschControlParam("user1",                /* project name*/
                                "net001",               /* jobnet name  */
                                "param1",               /* parameter    */
                                1,                      /* action : start */
                                0);                     /* rerun opt : none*/

    if (ret == 0) {
        printf("jobnet(net001) was normally controlled.\n");
    }
    else{
        printf("ERROR: return code = %d\n", ret);
        exit(1);
    }
    exit(0);
}
```

EE

16.2.4 Job Net Manipulation (Startup Parameter Specification) API/EE

This section explains the job net manipulation (startup parameter specification) API/EE (Mp_JobschControlParamEx).

Synopsis

[Windows]

```
#include "f3csbJobschControlEx.h"
int Mp_JobschControlParamEx (char *project,
                char *netname,
                char *parameter,
                int action,
                int rerunopt,
                int system_num) ;
```

[UNIX]

```
#include "jobschapiex.h"
int Mp_JobschControlParamEx (char *project,
                char *netname,
                char *parameter,
                int action,
                int rerunopt,
                int system_num) ;
```

Description

This API is an interface to use the jobschcontrol command from the C interface in Systemwalker Operation Manager EE. Using the API allows you to pass parameters to each job in the job net to start.

Only the job nets with the Job Execution Control attribute can be used.

Parameters

project

Specify the name of a project in which the job net is manipulated. Note that you can only specify the project name of the projects that the API execution user can use.

This can be omitted if there is only one project registered in the job net that the API execution user can use. In this case, NULL must be specified.

netname

Specify the name of the job net to be manipulated. This cannot be omitted.

parameter

Specify a parameter to be passed to each job of the job net to be started by a string within 200 bytes for startup or restart.

Note that for a job net manipulation other than startup and restart, specify NULL (this value is ignored when a string is specified).

When a string is specified:

The parameter specified is appended to the end of job argument.

When NULL is specified:

At restart, the parameter specified when the job was started last is appended to the end of job argument.

At startup, no parameter is specified.

action

Specify the job net manipulation method from the following:

1:

Start

2:

Cancel

3:

Restart

4:

Pause

5:

Continue

6:

Disable

7:

Enable

8:

Confirm

9:

Reinstate

10:

Revoke

rerunopt

Specify the option when 3 (Restart) is selected for action. If 3 (Restart) is not selected for action, 0 should be specified.

0:

Performs normal re-activation. Activates from an abended job. If the restart point is set at a job located before the abended job, restart is executed from that job.

1:

Restarts the job net from the top job.

2:

Re-activates a job net set in "Invalid at next restart" mode in relation to an abended job.

(Skips the abended job and activates the next job).

Jobs forcibly terminated are handled in the same way as abended jobs are handled.

system_num

Specify operating target subsystem number using a range of 0 to 9.

Return Values

Return Value	Meaning	Action
0	Normal termination.	-
4	Mode in which job net cannot function.	Perform the operation again after the status of the job net has changed.
8	Jobscheduler service/daemon is not running.	Start the Jobsheduler service or daemon and perform the operation again.
12	One of the following: - Error in parameter setting or invalid use privilege. - Error in user access privileges setting - Either the specified project or job net does not exist, or the job net has been copied using "copy and startup" operation, or the job net does not have Job Execution Control attributes.	Take one of the following measures: - Set the correct parameter (project name, job net name, parameters passed to job, restart option or subsystem number). - Grant the executing user access privileges to operate the project or higher privileges. - Specify a job net that can be operated on.
16	Fatal error occurred.	A system error may have occurred. Check the event log or syslog for any system abnormalities. If an error occurs, collect Jobscheduler information using the Maintenance Information Collection Tool, and then contact a Fujitsu SE. If none are found, the problem may only have been temporary, so perform the operation again.
20	Job nets that have been started with variable parameters cannot be restarted using this API.	Use the job net manipulation API to restart the job net using the variable parameter specified when the job net was started.

Required Files

The following files are required for use of the job net manipulation API shown above:

[Windows x86]

LIB:

f3csbJobschControlEx.lib

INCLUDE:

f3csbJobschControlEx.h

[Windows x64]

LIB:

f3csbJobschControlEx_x64.lib

INCLUDE:

f3csbJobschControlEx_x64.h

[Common to Solaris 32-bit and Linux x86]

LIB:

libjobschex.so

INCLUDE:

jobschapiex.h

[Solaris 64-bit]

LIB:

libjobschex_64.so

INCLUDE:

jobschapiex_64.h

[Linux x64]

LIB:

libjobschex_x64.so

INCLUDE:

jobschapiex_x64.h

The job net manipulation API is stored in the LIB file, and the definitions and structures used by the job net manipulation API are declared in the INCLUDE file. These files are stored under the following directories.

[Windows]

LIB:

Systemwalker Operation Manager installation directory: MpWalker.JM\lib

INCLUDE:

Systemwalker Operation Manager installation directory: MpWalker.JM\include

[Common to Solaris and Linux]

LIB:

/opt/FJSVJOBSC/usr/lib

INCLUDE:

/opt/FJSVJOBSC/usr/include

Caution

- Job nets that have been started using variable parameters cannot be restarted using this API.

Program Example

Below is an example program of the job net manipulation API/EE:

[Windows]

```
#include <windows.h>
```

```
#include <stdio.h>
#include <stdlib.h>
#include "f3csbJobschControlEx.h"

void main()
{
    int ret;

    ret = Mp_JobschControlParamEx("user1",              // project name
                                  "net001",             // jobnet name
                                  "param1",             // parameter
                                  1,                    // action : start
                                  0,                    // rerun opt : none
                                  1);                   // sub system number
    if (ret == 0) {
        printf("jobnet(net001) was normally controlled.\n");
    }
    else{
        printf("ERROR: return code = %d\n", ret);
        exit(1);
    }
    exit(0);
}
```

[UNIX]

```
#include <stdio.h>
#include <stdlib.h>
#include "jobschapiex.h"

main ()
{
    int ret;

    ret = Mp_JobschControlParamEx ("user1",         /* project name      */
                                   "net1",           /* jobnet name       */
                                   "param1",         /* parameter         */
                                   1,                /* action : start    */
                                   0;                /* rerun opt : none  */
                                   1);               /* sub system number */

    if (ret == 0) {
        printf ("jobnet(net1) was normally controlled.\n");
    }
    else{
        printf ("ERROR: return code = %d\n", ret);
        exit(1);
    }
    exit(0);
}
```

16.2.5 Group Manipulation API

This section explains the group manipulation API (Mp_JobschCtlGrp).

Synopsis

[Windows]

```
#include "f3csbJobschControl.h"
int Mp_JobschCtlGrp (char *project,
            char *grpname,
            int action) ;
```

[UNIX]

```
#include "jobschapi.h"
int Mp_JobschCtlGrp (char *project,
            char *grpname,
            int action) ;
```

Description

This API is an interface to use the jobschctlgrp command from the C interface.

Parameters

project

Specify the name of the project where the group to be manipulated exists. Note that you can only specify the project names that the API execution user can use.

This can be omitted if there is only one project having the group that the API execution user can use. In this case, NULL must be specified.

grpname

Specify the name of a group to be manipulated. This cannot be omitted.

action

Specify a job group manipulation method from the following:

1:

Start

2:

Cancel

3:

Restart

4:

Pause

5:

Continue

6:

Disable

7:

Enable

8:

Confirm

Return Values

Return Value	Meaning	Action
0	Normal termination.	-
4	No job group can be manipulated.	Perform the operation again after the status of the group has changed.
8	Jobscheduler service/daemon is not running.	Start the Jobsheduler service or daemon and perform the operation again.
12	Error in parameter setting or invalid user privilege.	Take one of the following measures: - Set the correct parameter (project name, group name or operation method). - Grant the executing user access privileges to operate the project or higher privileges.
16	Fatal error occurred.	A system error may have occurred. Check the event log or syslog for any system abnormalities. If an error occurs, collect Jobscheduler information using the Maintenance Information Collection Tool, and then contact a Fujitsu SE. If none are found, the problem may only have been temporary, so perform the operation again.

Required Files

The following files are required for use with the group manipulation API shown above:

[Windows x86]

LIB:

f3csbJobschCtlGrp.lib

INCLUDE:

f3csbJobschControl.h

[Windows x64]

LIB:

f3csbJobschCtlGrp_x64.lib

INCLUDE:

f3csbJobschControl_x64.h

[Common to Solaris 32-bit and Linux x86]

LIB:

libjobsch.so

INCLUDE:

jobschapi.h

[Solaris 64-bit]

LIB:

libjobsch_64.so

INCLUDE:

jobschapi_64.h

[Linux x64]

LIB:

    libjobsch_x64.so

INCLUDE:

    jobschapi_x64.h

The group manipulation API is stored in the LIB file, and the definitions and structures used by this API are declared in the INCLUDE file.

These files are stored under the following directory:

[Windows]

LIB:

    Systemwalker Operation Manager installation directory
    MpWalker.JM\lib

INCLUDE:

    Systemwalker Operation Manager installation directory
    MpWalker.JM\include

[Common to Solaris and Linux]

LIB:

    /opt/FJSVJOBSC/usr/lib

INCLUDE:

    /opt/FJSVJOBSC/usr/include

Program Example

Below is an example program of the group manipulation API:

[Windows]

```
#include <windows.h>
#include <stdio.h>
#include <stdlib.h>
#include "f3csbJobschControl.h"

void main()
{
    int ret;

    ret = Mp_JobschCtlGrp("user1",          // project name
                          "grp001",         // group name
                          1);               // action : start
    if (ret == 0) {
        printf("group(grp001) is controlled normally.\n");
    }
    else{
        printf("ERROR: return code = %d\n", ret);
        exit(1);
    }
    exit(0);
}
```

[UNIX]

```
#include <stdio.h>
```

```
#include <stdlib.h>
#include "jobschapi.h"

main()
{
    int ret;

    ret = Mp_JobschCtlGrp("user1",        /* project name*/
                          "grp001",       /* group name */
                          1);             /* action : start */

    if (ret == 0) {
        printf("group(grp001) was normally controlled.\n");
    }
    else{
        printf("ERROR: return code = %d\n", ret);
        exit(1);
    }
    exit(0);
}
```

EE

16.2.6 Group Manipulation API/EE

This section explains the group manipulation API/EE (Mp_JobschCtlGrpEx).

Synopsis

[Windows]

```
#include "f3csbJobschControlEx.h"
int Mp_JobschCtlGrpEx (char *project,
           char *grpname,
           int action) ;
           int system_num) ;
```

[UNIX]

```
#include "jobschapiex.h"
int Mp_JobschCtlGrpEx (char *project,
           char *grpname,
           int action),
           int system_num);
```

Description

This API is an interface to use the jobschctlgrp command from the C interface in Systemwalker Operation Manager EE.

Parameters

project

Specify the name of the project where the group to be manipulated exists. Note that you can only specify the project names that the API execution user can use.

This can be omitted if there is only one project having the group that the API execution user can use. In this case, NULL must be specified.

grpname

Specify the name of a group to be manipulated. This cannot be omitted.

action

Specify a job group manipulation method from the following:

1:

Start

2:

Cancel

3:

Restart

4:

Pause

5:

Continue

6:

Disable

7:

Enable

8:

Confirm

system_num

Specify operating target subsystem number using a range of 0 to 9.

Return Values

Return Value	Meaning	Action
0	Normal termination.	-
4	No job group can be manipulated.	Perform the operation again after the status of the group has changed.
8	Jobscheduler service/daemon is not running.	Start the Jobsheduler service or daemon and perform the operation again.
12	Error in parameter setting or invalid user privilege.	Take one of the following measures: - Set the correct parameter (project name, group name, operation method or subsystem number). - Grant the executing user access privileges to operate the project or higher privileges.
16	Fatal error occurred.	A system error may have occurred. Check the event log or syslog for any system abnormalities. If an error occurs, collect Jobscheduler information using the Maintenance Information Collection Tool, and then contact a Fujitsu SE. If none are found, the problem may only have been temporary, so perform the operation again.

Required Files

The following files are required for use with the group manipulation API shown above:

[Windows x86]

LIB:

　f3csbJobschControlEx.lib

INCLUDE:

　f3csbJobschControlEx.h

[Windows x64]

LIB:

　f3csbJobschControlEx_x64.lib

INCLUDE:

　f3csbJobschControlEx_x64.h

[Common to Solaris 32-bit and Linux x86]

LIB:

　libjobschex.so

INCLUDE:

　jobschapiex.h

[Solaris 64-bit]

LIB:

　libjobschex_64.so

INCLUDE:

　jobschapiex_64.h

[Linux x64]

LIB:

　libjobschex_x64.so

INCLUDE:

　jobschapiex_x64.h

The group manipulation API is stored in the LIB file, and the definitions and structures used by the group manipulation API are declared in the INCLUDE file. These files are stored under the following directories.

[Windows]

LIB:

　Systemwalker Operation Manager installation directory: MpWalker.JM\lib

INCLUDE:

　Systemwalker Operation Manager installation directory: MpWalker.JM\include

[Common to Solaris and Linux]

LIB:

　/opt/FJSVJOBSC/usr/lib

INCLUDE:

　/opt/FJSVJOBSC/usr/include

Program Example

Below is an example program of the group manipulation API/EE:

[Windows]

```
#include <windows.h>
#include <stdio.h>
##include <stdlib.h>
include "f3csbJobschControlEx.h"

void main()
{
    int ret;

    ret = Mp_JobschCtlGrpEx("user1",              // project name
                            "grp001",             // group name
                            1,                    // action : start
                            1);                   // sub system number
    if (ret == 0) {
        printf("group(grp001) is controlled normaly.\n");
    }
    else{
        printf("ERROR: return code = %d\n", ret);
        exit(1);
    }
    exit(0);
}
```

[UNIX]

```
#include <stdio.h>
#include <stdlib.h>
#include "jobschapiex.h"

main ()
{
    int ret;

    ret = Mp_JobschCtlGrpEx ("user1",        /* project name      */
                             "grp1",         /* group name        */
                             1,              /* action : start    */
                             1);             /*sub system number */
    if (ret == 0) {
        printf ("group(grp1) was normally controlled.\n");
    }
    else{
        printf ("ERROR: return code = %d\n", ret);
        exit(1);
    }
    exit(0);
}
```

16.2.7 Job Net Execution API [Windows]

This section explains the job net execution API (JSNetStart).

Synopsis

```
#include "f3csbNetStart.h"
int JSNetStart (char *project,
          char *netname,
          char *text,
```

```
            int kind,
            char *parm,
            int action);
```

Description

With the features of the job net execution API, you can:

- Start up a job net from a user application

- Output a desired message (event) at job net startup time

- Pass a desired parameter to the job

- Wait to start up a job net by connecting to the job net startup queue when the same job net has already been running

Parameters

project

Specify the project name of the project that has job nets to be manipulated. As for the specifiable project name, the API execution user must be able to use the project of that name.

This can be omitted if there is only one project having the job net that the API execution user can use. In this case, NULL must be specified.

netname

Specify the name of a job net to be started up. This cannot be omitted.

text

Specify the string to be output to the event log with up to 256 bytes. If NULL is specified, the message won't be output to the event log.

kind

Specify the type of the messages to be output to the event log. The following three types are available to choose from. If the text parameter is NULL, this parameter will be ignored.

LOG_INFO:

Message at information level (message identifier:10000)

LOG_WARN:

Message at warning level (message identifier:10001)

LOG_ERR:

Message at error level (message identifier:10002)

parm

Specify the parameter string to be passed to the job with up to 512 bytes.

The parameters specified in all the jobs in the job net to be started up will be passed.

action

Specify the actions to take when the job net specified in the netname parameter is already running. The instructions and their meaning are shown below:

ACT_QUE

Connects to the job net startup queue.

ACT_NO

Cancels the startup request.

Return Values

Return Value	Meaning	Action
0	The startup request was accepted normally, and the job net started up.	-
1	Connected to the job net startup queue (when ACT_QUE is specified), because the job net was already running.	-
2	The startup request was cancelled (when ACT_NO is specified), because the job net was already running.	-
-1	The project name specified in the project parameter is either illegal or no such project name exists.	Set the correct project name.
-2	The job net name specified in the netname parameter is either illegal or no such job net name exists.	Set the correct job net name.
-3	The message type specified in the kind parameter is illegal.	Set the correct message type.
-4	The operation was illegal for executing of the job net specified in the action parameter.	Set the correct action.
-5	The character string specified in the text parameter is too long.	The character string specified as the text parameter can be no more than 256 bytes long.
-6	The character string specified in the parm parameter is too long.	The character string specified as the param parameter can be no more than 512 bytes long.
-7	The job net could not be started because it has been in Paused or Disabled. Cancel the Paused or Disabled for the job net and re-execute. Otherwise, the job net could not be started because it belongs to a group.	Take one of the following measures and then perform the operation again: - Check the status of the job net. - Remove the job net from the group.
-8	The job net could not be started because the startup processing was congested. Wait for a while and re-execute it.	Wait a few moments and attempt the operation again.
-9	Either the Jobscheduler service is not running or the job net execution API function is disabled. Enable the job net execution API function and start up the Jobscheduler service, then re-execute.	Take one of the following measures and then perform the operation again: - Start the Jobscheduler service. - Reset the startup parameter of the Jobscheduler service and then restart the Jobscheduler service.
-10	The Jobscheduler service is carrying out shutdown procedures. Start the Jobscheduler service after the procedures have stopped, then re-execute.	After the Jobscheduler service has stopped, restart the Jobscheduler service and then attempt the operation again.

Return Value	Meaning	Action
-20	Failed to accept the job net startup request. Refer to the event log for details and re-execute after removing the cause of the error.	Use the error details output to the event log to remove the cause of the problem and then attempt the operation again.

Required Files

The following files are required to use the job net execution API shown above:

[Windows x86]

LIB:

    f3csbNetStart.lib

INCLUDE:

    f3csbNetStart.h

[Windows x64]

LIB:

    f3csbNetStart_x64.lib

INCLUDE:

    f3csbNetStart_x64.h

These files are stored under "MpWalker.JM\lib" and "MpWalker.JM\include" where Systemwalker Operation Manager was installed.

Program Example

Below is an example program of the job net execution API:

```
#include <windows.h>
#include <stdio.h>
#include <stdlib.h>
#include "f3csbNetStart.h"

void main()
{
    int ret;

    ret = JSNetStart("user1",            // project name
                     "net001",           // jobnet name
                     "net001 start.",    // messages for event log
                     LOG_INFO,           // kind of event log : information
                     "-d c:\\data -x",   // parameters for jobs
                     ACT_QUE);           // if jobnet is running, that is queued.

    if (ret == 0) {
        printf("jobnet(net001) is started.\n");
    }
    else if (ret == 1) {
        printf("jobnet(net001) is queued, because now running.\n");
    }
    else if (ret < 0) {
        printf("ERROR: return code = %d\n", ret);
        exit(1);
    }
    exit(0);
}
```

16.2.8 Job Net Execution API/EE [Windows]

This section explains the job net execution API/EE (JSNetStartEx).

Synopsis

```
#include "f3csbNetStartEx.h"
int JSNetStartEx (char *project,
          char *netname,
          char *text,
          int kind,
          char *parm,
          int action,
          int system_num);
```

Description

With the features of job net execution API/EE, you can:

- Start up a job net from a user application

- Output a desired message (event) at job net startup time

- Pass a desired parameter to the job

- Wait to start up a job net by connecting to the job net startup queue when the same job net has already been running

Parameters

project

Specify the project name of the project that has job nets to be manipulated. As for the specifiable project name, the API execution user must be able to use the project of that name.

This can be omitted if there is only one project having the job net that the API execution user can use. In this case, NULL must be specified.

netname

Specify the name of a job net to be started up. This cannot be omitted.

text

Specify the string to be output to the event log with up to 256 bytes. If NULL is specified, the message won't be output to the event log.

kind

Specify the type of the messages to be output to the event log. The following three types are available to choose from. If the text parameter is NULL, this parameter will be ignored.

LOG_INFO:

Message at information level (message identifier:10000)

LOG_WARN:

Message at warning level (message identifier:10001)

LOG_ERR:

Message at error level (message identifier:10002)

parm

Specify the parameter string to be passed to the job with up to 512 bytes.

The parameters specified in all the jobs in the job net to be started up will be passed.

action

Specify the actions to take when the job net specified in the netname parameter is already running. The instructions and their meaning are shown below:

ACT_QUE

Connects to the job net startup queue.

ACT_NO

Cancels the startup request.

system_num

Specify operating target subsystem number using a range of 0 to 9.

Return Values

Return Value	Meaning	Action
0	The startup request was accepted normally, and the job net started up.	-
1	Connected to the job net startup queue (when ACT_QUE is specified), because the job net was already running.	-
2	The startup request was cancelled (when ACT_NO is specified), because the job net was already running.	-
-1	The project name specified in the project parameter is either illegal or no such project name exists.	Set the correct project name.
-2	The job net name specified in the netname parameter is either illegal or no such job net name exists.	Set the correct job net name.
-3	The message type specified in the kind parameter is illegal.	Set the correct message type.
-4	The operation was illegal for executing of the job net specified in the action parameter.	Set the correct action.
-5	The character string specified in the text parameter is too long.	The character string specified as the text parameter can be no more than 256 bytes long.
-6	The character string specified in the parm parameter is too long.	The character string specified as the param parameter can be no more than 512 bytes long.
-7	The job net could not be started because it has been in Paused or Disabled. Cancel the Paused or Disabled for the job net and re-execute. Otherwise, the job net could not be started because it is subordinate to a group.	Take one of the following measures and then perform the operation again: - Check the status of the job net. - Remove the job net from the group.
-8	The job net could not be started because the startup processing was congested. Wait for a while and re-execute it.	Wait a few moments and attempt the operation again.
-9	Either the Jobscheduler service is not running or the job net execution API function	Take one of the following measures and then perform the operation again:

Return Value	Meaning	Action
	is disabled. Enable the job net execution API function and start up the Jobscheduler service, then re-execute.	- Start the Jobscheduler service. - Reset the startup parameter of the Jobscheduler service and then restart the Jobscheduler service.
-10	The Jobscheduler service is carrying out shutdown procedures. Start the Jobscheduler service after the procedures have stopped, then re-execute.	After the Jobscheduler service has stopped, restart the Jobscheduler service and then attempt the operation again.
-20	Failed to accept the job net startup request. Refer to the event log for details and re-execute after removing the cause of the error.	Use the error details output to the event log to remove the cause of the problem and then attempt the operation again.

Required Files

The following files are required to use the job net execution API shown above:

[Windows x86]

LIB:

f3csbNetStartEx.lib

INCLUDE:

f3csbNetStartEx.h

[Windows x64]

LIB:

f3csbNetStartEx_x64.lib

INCLUDE:

f3csbNetStartEx_x64.h

These files are stored under "MpWalker.JM\lib" and "MpWalker.JM\include" where Systemwalker Operation Manager was installed.

Program Example

Below is an example program of the job net execution API/EE:

```
#include <windows.h>
#include <stdio.h>
#include <stdlib.h>
#include "f3csbNetStartEx.h"

void main()
{
    int ret;

    ret = JSNetStartEx("user1",            // project name
                       "net001",           // jobnet name
                       "net001 start.",    // messages for event log
                       LOG_INFO,           // kind of event log : information
                       "-d c:\\data -x",   // parameters for jobs
                       ACT_QUE,            // if jobnet is running, that is queued.
                       1);                 // sub system number
    if (ret == 0) {
        printf("jobnet(net001) is started.\n");
    }
```

```
    else if (ret == 1) {
        printf("jobnet(net001) is queued, because now running.\n");
    }
    else if (ret < 0) {
        printf("ERROR: return code = %d\n", ret);
        exit(1);
    }
    exit(0);
}
```

16.2.9 Job Net Start Time Change API

This section explains the job net start time change API (Mp_JobschCtlStartTime).

Synopsis

[Windows]

```
#include "f3csbJobschControl.h"
int Mp_JobschCtlStartTime (
      int mode,
      struct jobsch_start_time *get_time,
      struct jobsch_start_time *before_time,
      struct jobsch_start_time *set_time,
      char *project_name,
      char *jobnet_name) ;
```

[UNIX]

```
#include "jobschapi.h"
int Mp_JobschCtlStartTime (
      int mode,
      struct jobsch_start_time *get_time,
      struct jobsch_start_time *before_time,
      struct jobsch_start_time *set_time,
      char *project_name,
      char *jobnet_name) ;
```

Description

This API enables functions for obtaining or changing the job net start time to be used from a C language interface.

The start time acquisition process stores the start time of the specified job net in the area specified by the get_time parameter.

The start time change process changes the start time of the specified job net to the value specified by the set_time parameter and then the modified start time and startup day information are used to reschedule the scheduled startup date/time.

Parameters

mode

Specify the operation mode of this API. This option cannot be omitted.

0:

Obtains the start time. The before_time and set_time parameters will be ignored. To use this mode, the API must be executed by a user with the reference right (or a higher right) for the project to which the job net belongs.

1:

Changes the start time. The get_time parameter will be ignored. To use this mode, the API must be executed by a user with the update or change right for the project to which the job net belongs.

get_time

Specify the area of the jobsch_start_time structure that will obtain the start time. NULL is specified if this parameter is omitted. This parameter cannot be omitted if 0 is specified for the mode parameter. If this API terminates normally when 0 is specified for the mode parameter, the acquisition result will be stored in the structure. Refer to the explanation of the time parameter of the jobsch_start_time structure for more information about the value that is stored.

before_time

Specify the pre-change startup time. Refer to the explanation for the time parameter in the jobsch_start_time structure for the value to be specified. NULL is specified if this parameter is omitted. This parameter cannot be omitted if 1 is specified for the mode parameter. The specified start time type and start time are compared with the start time type and start time specified for the job net, and the start time is changed only if these values match. In the event of a mismatch, the API will return an error.

set_time

Specify the start time to be set. Refer to the time parameter in the jobsch_start_time structure for the value to be specified. This parameter cannot be omitted if 1 is specified for the mode parameter.

project_name

Specify the name of the project to which to the job net to be manipulated belongs. Only the names of projects that can be used by the user running the API can be specified. This parameter can be omitted if only one of the projects that can be used by the user running the API has job nets registered with it. NULL is specified if this parameter is omitted.

jobnet_name

Specify the name of the job net to be manipulated. This parameter cannot be omitted.

Return Values

Return Value	Meaning	Action
0	Normal termination.	-
4	The pre-change start time of the specified job net does not match.	Specify the same start time type and start time as the job net to be changed in before_time.
8	The execution conditions of the job net for which the start time is to be obtained do not specify a start time. Or, the execution conditions of the specified job net differ from the start time type to be changed.	Specify a job net with a specified start time as the job net to be obtained. When changing the start time, specify the same type in set_time as the start time of the relevant job net.
12	The specified job net belongs to a group.	Specify a job net that does not belong to a group.
16	The status of the specified job net is either "executing", "warning" or "end delayed".	Perform the operation again after the status of the job net has changed.
24	The job net is using a different character encoding from that used by Systemwalker.	Ensure that the execution environment of the program invoking the API uses the same character encoding as that used by Systemwalker.
28	The Jobscheduler service/daemon is not running.	Start the Jobscheduler service/daemon and perform the operation again.
32	There is an error in the parameter specifications or the user privileges, or the specified project or job net does not exist.	Take whichever of the following actions is necessary:

Return Value	Meaning	Action
		- Set the correct parameter (operation mode, start time, project name or job net name). - Grant the executing user the correct project access privileges.
36	Fatal error occurred.	A system error may have occurred. Check the event log or syslog for any system abnormalities. If an error occurs, collect Jobscheduler information using the Maintenance Information Collection Tool, and then contact a Fujitsu SE. If none are found, the problem may only have been temporary, so perform the operation again.
40	The specified job net is being manipulated or changed.	Perform the operation again after the status of the job net has changed.

Format of the jobsch_start_time structure

The format of the jobsch_start_time structure is as follows:

```
struct jobsch_start_time {
    int type;
    char rsv1[4];
    char time[128];
    char rsv2[24];
} ;
```

type:

The following table lists the values that indicate the different start time types.

Value	Type
1	A job net with only one start time specified
2	A job net with multiple start times specified
3	A job net with an execution interval

rsv1:

Reserved. (Not used.)

time:

A character string that indicates the start time. The format of this parameter differs according to the start time type.

1: When the job net has only one start time specified

A 4-byte (5 bytes with the NULL character) string of the format "HHMM" (representing hours and minutes).

Example) "0900"

2: When the job net has more than one start time specified

A character string up to 29 bytes long (30 bytes including the NULL character) that is made up of 4-byte strings of the format "HHMM" (representing hours and minutes) separated by commas. (Six 4-byte strings + five commas + one NULL character = 30 bytes)

Example) "0900,1100,1300,1500,1700"

3: When the job net has an execution interval

A character string up to 13 bytes long (14 bytes including the NULL character) of the format "HHMM,HHMM,n" (representing start time, end time and execution interval).Each parameter is separated by a comma. The start and end times consist of 4-byte strings of the format "HHMM" (representing hours and minutes) and the execution interval in minutes is represented by a character string between one and three bytes long. The minimum execution interval is 1 and the maximum is 240 (minutes).

Example) "0900,1700,60"

rsv2:

Reserved. (Not used.)

Required Files

The following files are needed when using the Job Net Start Time Change API.

[Windows x86]

LIB:

Systemwalker Operation Manager installation directory\MpWalker.JM\lib\f3csbJobschControl.lib

INCLUDE:

Systemwalker Operation Manager installation directory\MpWalker.JM\include\f3csbJobschControl.h

[Windows x64]

LIB:

Systemwalker Operation Manager installation directory\MpWalker.JM\lib\f3csbJobschControl_x64.lib

INCLUDE:

Systemwalker Operation Manager installation directory\MpWalker.JM\include\f3csbJobschControl_x64.h

[Common to Solaris 32-bit and Linux x86]

LIB:

/opt/FJSVJOBSC/usr/lib/libjobsch.so

INCLUDE:

/opt/FJSVJOBSC/usr/include/jobschapi.h

[Solaris 64-bit]

LIB:

/opt/FJSVJOBSC/usr/lib/libjobsch_64.so

INCLUDE:

/opt/FJSVJOBSC/usr/include/jobschapi_64.h

[Linux x64]

LIB:

/opt/FJSVJOBSC/usr/lib/libjobsch_x64.so

INCLUDE:

/opt/FJSVJOBSC/usr/include/jobschapi_x64.h

Privileges needed for execution

The executing user must be a system administrator or a user with permission to access the target project.

Cautions

- This API is only valid for job nets with the execution conditions indicated by the start time in the jobsch_start_time structure.

- The start time type cannot be changed.

- The start time of a job net belonging to a group cannot be changed.

- Programs that use this API must have the correct time zone setting.

- This API will return an error if it is called by a program specifies a different locale from that in which Systemwalker Operation Manager is operating.

Program Example

A program that uses the job net start time change API is provided below.

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <locale.h>
#include <string.h>
#include "jobschapi.h"

main ()
{
    int ret;
    char t_str[8];
    struct jobsch_start_time b_time, a_time;

    tzset ();
    setlocale (LC_ALL, "");

    memset(&b_time, 0x00, sizeof(b_time));

    ret = Mp_JobschCtlStartTime (
                    0,          /* mode : GET       */
                    &b_time,    /* get start time   */
                    NULL,       /* NULL             */
                    NULL,       /* NULL             */
                    "test",     /* project name     */
                    "net1");    /* jobnet name      */
    if (ret == 0) {
        printf ("Start type is %d.\n", b_time.type);
        printf ("Start time is %s.\n", b_time.time);
    }
    else{
        printf ("ERROR: get return code = %d\n", ret);
        exit(1);
    }

    memset(&a_time, 0x00, sizeof(a_time));
    a_time.type = b_time.type;
    strcpy(a_time.time, b_time.time);
    strcat(a_time.time, ",1900");

    ret = Mp_JobschCtlStartTime (
                    1,          /* mode : CHANGE    */
                    NULL,       /* NULL             */
                    &b_time,    /* before time      */
                    &a_time,    /* set time         */
                    "test",     /* project name     */
                    "net1");    /* jobnet name      */
```

```
    if (ret == 0) {
        printf ("Change type is %d.\n", a_time.type);
        printf ("Change time is %s.\n", a_time.time);
    }
    else{
        printf ("ERROR: set return code = %d\n", ret);
        exit(1);
    }

    exit(0);
}
```

16.2.10 Job Net Start Time Change API/EE

This section explains the job net start time change API/EE (Mp_JobschCtlStartTimeEx).

Synopsis

[Windows]

```
#include "f3csbJobschControlEx.h"
int Mp_JobschCtlStartTimeEx (
        int mode,
        struct jobsch_start_time *get_time,
        struct jobsch_start_time *before_time,
        struct jobsch_start_time *set_time,
        char *project_name,
        char *jobnet_name,
        int system_num) ;
```

[UNIX]

```
#include "jobschapiex.h"
int Mp_JobschCtlStartTimeEx (
        int mode,
        struct jobsch_start_time *get_time,
        struct jobsch_start_time *before_time,
        struct jobsch_start_time *set_time,
        char *project_name,
        char *jobnet_name,
        int system_num) ;
```

Description

This API enables functions for obtaining or changing the job net start time to be used from a C language interface in Systemwalker Operation Manager EE.

The start time acquisition process stores the start time of the specified job net in the area specified by the get_time parameter.

The start time change process changes the start time of the specified job net to the value specified by the set_time parameter and then the modified start time and startup day information are used to reschedule the scheduled startup date/time.

Parameters

mode

Specify the operation mode of this API. This option cannot be omitted.

0:

Obtains the start time. The before_time and set_time parameters will be ignored. To use this mode, the API must be executed by a user with the reference right (or a higher right) for the project to which the job net belongs.

1:

> Changes the start time. The get_time parameter will be ignored. To use this mode, the API must be executed by a user with the update or change right for the project to which the job net belongs.

get_time

Specify the area of the jobsch_start_time structure that will obtain the start time. NULL is specified if this parameter is omitted. This parameter cannot be omitted if 0 is specified for the mode parameter. If this API terminates normally when 0 is specified for the mode parameter, the acquisition result will be stored in the structure. Refer to the explanation of the time parameter of the jobsch_start_time structure for more information about the value that is stored.

before_time

Specify the pre-change startup time. Refer to the explanation for the "time" parameter in the "jobsch_start_time" structure for the value to be specified. NULL is specified if this parameter is omitted. This parameter cannot be omitted if 1 is specified for the mode parameter. The specified start time type and start time are compared with the start time type and start time specified for the job net, and the start time is changed only if these values match. In the event of a mismatch, the API will return an error.

set_time

Specify the start time to be set. Refer to the time parameter in the jobsch_start_time structure for the value to be specified. NULL is specified if this parameter is omitted. This parameter cannot be omitted if 1 is specified for the mode parameter.

project_name

Specify the name of the project to which to the job net to be manipulated belongs. Only the names of projects that can be used by the user running the API can be specified. This parameter can be omitted if only one of the projects that can be used by the user running the API has job nets registered with it. NULL is specified if this parameter is omitted.

jobnet_name

Specify the name of the job net to be manipulated. This parameter cannot be omitted.

system_num

Specify the subsystem to be operated as a number between 0 and 9. This parameter cannot be omitted.

Return Values

Return Value	Meaning	Action
0	Normal termination.	-
4	The pre-change start time of the specified job net does not match.	Specify the same start time type and start time as the job net to be changed in before_time.
8	The execution conditions of the job net to be obtained do not specify a start time. Or, the execution conditions of the specified job net differ from the start time type to be changed.	Specify a job net with a specified start time as the job net to be obtained. When changing the start time, specify the same type in set_time as the start time of the relevant job net.
12	The specified job net belongs to a group.	Specify a job net that does not belong to a group.
16	The status of the specified job net is either "executing", "warning" or "end delayed". The specified job net is running.	Perform the operation again after the status of the job net has changed.
24	The job net is using a different character encoding from that used by Systemwalker.	Ensure that the execution environment of the program invoking the API uses the

Return Value	Meaning	Action
		same character encoding as that used by Systemwalker.
28	The Jobscheduler service/daemon is not running.	Start the Jobscheduler service/daemon and perform the operation again.
32	There is an error in the parameter specifications or the user privileges, or the specified project or job net does not exist.	Take whichever of the following actions is necessary: - Set the correct parameter (operation mode, start time, project name or job net name). - Grant the executing user the correct project access privileges.
36	Fatal error occurred.	A system error may have occurred. Check the event log or syslog for any system abnormalities. If an error occurs, collect Jobscheduler information using the Maintenance Information Collection Tool, and then contact a Fujitsu SE. If none are found, the problem may only have been temporary, so perform the operation again.
40	The specified job net is being manipulated or changed.	Perform the operation again after the status of the job net has changed.

Format of the jobsch_start_time structure

The format of the jobsch_start_time structure is as follows:

```
struct jobsch_start_time {
    int type;
    char rsv1[4];
    char time[128];
    char rsv2[24];
} ;
```

type:

The following table lists the values that indicate the different start time types.

Value	Type
1	A job net with only one start time specified
2	A job net with multiple start times specified
3	A job net with an execution interval

rsv1:

Reserved. (Not used.)

time:

A character string that indicates the start time. The format of this parameter differs according to the start time type.

1: When the job net has only one start time specified

A 4-byte (5 bytes with the NULL character) string of the format "HHMM" (representing hours and minutes).

Example) "0900"

2: When the job net has more than one start time specified

A character string up to 29 bytes long (30 bytes including the NULL character) that is made up of 4-byte strings of the format "HHMM" (representing hours and minutes) separated by commas. (Six 4-byte strings + five commas + one NULL character = 30 bytes)

Example) "0900,1100,1300,1500,1700"

3: When the job net has an execution interval

A character string up to 13 bytes long (14 bytes including the NULL character) of the format "HHMM,HHMM,n" (representing start time, end time and execution interval).Each parameter is separated by a comma. The start and end times consist of 4-byte strings of the format "HHMM" (representing hours and minutes) and the execution interval in minutes is represented by a character string between one and three bytes long. The minimum execution interval is 1 and the maximum is 240 (minutes).

Example) "0900,1700,60"

rsv2:

Reserved. (Not used.)

Required Files

The following files are needed when using the Job Net Start Time Change API.

[Windows x86]

LIB:

Systemwalker Operation Manager installation directory\MpWalker.JM\lib\f3csbJobschControlEx.lib

INCLUDE:

Systemwalker Operation Manager installation directory\MpWalker.JM\include\f3csbJobschControlEx.h

[Windows x64]

LIB:

Systemwalker Operation Manager installation directory\MpWalker.JM\lib\f3csbJobschControlEx_x64.lib

INCLUDE:

Systemwalker Operation Manager installation directory\MpWalker.JM\include\f3csbJobschControlEx_x64.h

[Common to Solaris 32-bit and Linux x86]

LIB:

/opt/FJSVJOBSC/usr/lib/libjobschex.so

INCLUDE:

/opt/FJSVJOBSC/usr/include/jobschapiex.h

[Solaris 64-bit]

LIB:

/opt/FJSVJOBSC/usr/lib/libjobschex_64.so

INCLUDE:

/opt/FJSVJOBSC/usr/include/jobschapiex_64.h

[Linux x64]

LIB:

/opt/FJSVJOBSC/usr/lib/libjobschex_x64.so

INCLUDE:

/opt/FJSVJOBSC/usr/include/jobschapiex_x64.h

Privileges needed for execution

The executing user must be a system administrator or a user with permission to access the target project.

Cautions

- The executing user must be a system administrator or a user with permission to access the target project.

- This API is only valid for job nets with the execution conditions indicated by the start time in the jobsch_start_time structure.

- The start time type cannot be changed.

- The start time of a job net belonging to a group cannot be changed.

- Programs that use this API must have the correct time zone setting.

- This API will return an error if it is called by a program specified with a different locale from that in which Systemwalker Operation Manager is operating.

Program Example

A program that uses the job net start time change API is provided below.

```c
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <locale.h>
#include <string.h>
#include "jobschapiex.h"

main ()
{
    int ret;
    char t_str[8];
    struct jobsch_start_time b_time, a_time;

    tzset ();
    setlocale (LC_ALL, "");

    memset(&b_time, 0x00, sizeof(b_time));

    ret = Mp_JobschCtlStartTimeEx (
                    0,          /* mode : GET       */
                    &b_time,    /* get start time   */
                    NULL,       /* NULL             */
                    NULL,       /* NULL             */
                    "test",     /* project name     */
                    "net1",     /* jobnet name      */
                    1);         /* sub system number*/
    if (ret == 0) {
        printf ("Start type is %d.\n", b_time.type);
        printf ("Start time is %s.\n", b_time.time);
    }
    else{
        printf ("ERROR: get return code = %d\n", ret);
        exit(1);
    }

    memset(&a_time, 0x00, sizeof(a_time));
    a_time.type = b_time.type;
    strcpy(a_time.time, b_time.time);
    strcat(a_time.time, ",1900");
```

```
    ret = Mp_JobschCtlStartTimeEx (
                      1,           /* mode : CHANGE    */
                      NULL,        /* NULL             */
                      &b_time,     /* before time      */
                      &a_time,     /* set time         */
                      "test",      /* project name     */
                      "net1",      /* jobnet name      */
                      1);          /* sub system number*/
    if (ret == 0) {
        printf ("Change type is %d.\n", a_time.type);
        printf ("Change time is %s.\n", a_time.time);
    }
    else{
        printf ("ERROR: set return code = %d\n", ret);
        exit(1);
    }

    exit(0);
}
```

16.3 Job Execution Control API

This section explains information on the APIs provided by Job Execution Control:

The operating environments, precautionary items, and required files which are common to Job Execution Control APIs are explained below.

Operating Environment

This API can only operate on the machine where a Systemwalker Operation Manager server has been installed. It cannot operate on the machine where only the Systemwalker Operation Manager client has been installed.

Cautions

- In a compiling environment, use Visual C++ (version 2005) as the compiler and "multithread" for the runtime library. **[Windows]**

- As for applications that have link with LIB, establish their links again by using the file and INCLUDE file provided in this version/level.

- Do not call the Job Execution Control APIs from multiple threads within a single process simultaneously.

Required Files

The following files are required for use with the Job Execution Control APIs shown above:

[Windows x86]

LIB:

f3cuapi.lib

INCLUDE:

f3cuapi.h

[Windows x64]

LIB:

f3cuapi_x64.lib

INCLUDE:

f3cuapi_x64.h

[Common to Solaris 32-bit, Solaris 64-bit, Linux x86 and Linux x64]

LIB:

  libmjes.so

INCLUDE:

  mjesapi.h

The Job Execution Control APIs are contained in the LIB file, and declarations for the constants and structures used by these APIs are entered in the INCLUDE file.

These files are stored in the following directories:

[Windows]

LIB:

  Systemwalker Operation Manager installation directory: MpWalker.JM\lib

INCLUDE:

  Systemwalker Operation Manager installation directory: MpWalker.JM\include

[Common to Solaris and Linux]

LIB:

  /opt/FJSVMJS/usr/lib

INCLUDE:

  /opt/FJSVMJS/usr/include

16.3.1 Installation/Operation Information Acquisition API [Windows]

This section explains the installation/operation information acquisition API (Mp_StatusMJES).

Synopsis

```
#include "f3cuapi.h"
long Mp_StatusMJES()
```

Description

This API obtains the version/level and operation status of Job Execution Control.

Return Values

Return Value	Meaning	Action
0	Job Execution Control is not running.	-
1 or greater	Job Execution Control is running. The precise return value varies depending on the product version/level. - For V13.8.0: 140*65536+40=(9175080)10=(8C0028)16 - For V13.3.0/V13.3.1: 130*65536+40=(8519720)10=(820028)16 In the case of Systemwalker Operation Manager V13.2.0 or earlier, the first 2 bytes indicate the version, and the last 2 bytes indicate the level. In the case of Systemwalker Operation Manager V13.2.0, the version is 130 (the integer preceding the first period x 10) and the level is 30 (the integer preceding the second period x 10 + 10 + the integer following the second period). - For V13.2.0:	-

Return Value	Meaning	Action
	130*65536+30=(8519710)10=(82001E)16	
-1	A system error occurred.	Collect "Job Execution Control" information with the maintenance information collection tool and contact a Fujitsu SE.

Note that in the event of a system error, GetLastError()can be used to obtain detailed error codes.

Program Example

Below is an example program of the installation/operation information acquisition API:

```
#include <windows.h>
#include <stdio.h>
#include "f3cuapi.h"

int main (int argc, char **argv)
{
        long rtn;

        rtn = Mp_StatusMJES ();
        switch (rtn) {
        case 0:
                printf ("Systemwalker MpMjes service is not running.\n");
                return (1);
        case MJSESYSER:
                printf ("GetLastError() is %d\n", GetLastError());
                printf ("System error occurred.\n");
                return (1);
        }
        printf ("Version : %d\n", rtn / 65536);
        printf ("Level   : %d\n", rtn % 65536);
        return (0);
}
```

EE

16.3.2 Installation/Operation Information Acquisition API/EE [Windows]

This section explains the installation/operation information acquisition API/EE (Mp_StatusMJES_Ex).

Synopsis

```
#include "f3cuapi.h"
long Mp_StatusMJES_Ex(int system_num)
```

Description

This API obtains the version/level and operation status of Job Execution Control in Systemwalker Operation Manager EE.

Parameter

system_num

Specify operating target subsystem number using a range of 0 to 9.

Return Values

Return Value	Meaning	Action
0	Job Execution Control is not running.	-
1 or greater	Job Execution Control is running. The precise return value varies depending on the product version/level. - For V13.8.0: 140*65536+40=(9175080)10=(8C0028)16 - For V13.3.0/V13.3.1: 130*65536+40=(8519720)10=(820028)16 In the case of Systemwalker Operation Manager V13.2.0 or earlier, the first 2 bytes indicate the version, and the last 2 bytes indicate the level. In the case of Systemwalker Operation Manager V13.2.0, the version is 130 (the integer preceding the first period x 10) and the level is 30 (the integer preceding the second period x 10 + 10 + the integer following the second period). - For V13.2.0: 130*65536+30=(8519710)10=(82001E)16	-
-1	A system error occurred.	Collect "Job Execution Control" information with the maintenance information collection tool and contact a Fujitsu SE.

Note that in the event of a system error, GetLastError()can be used to obtain detailed error codes.

Program Example

Below is an example program of the installation/operation information acquisition API/EE:

```
#include <windows.h>
#include <stdio.h>
#include "f3cuapi.h"

int main (int argc, char **argv)
{
        long rtn;
        int system_num;

        system_num = 0;
        rtn = Mp_StatusMJES_Ex (system_num);
        switch (rtn) {
        case 0:
                printf ("Systemwalker MpMjes service is not running.\n");
                return (1);
        case MJSESYSER:
                printf ("GetLastError() is %d\n", GetLastError());
                printf ("System error occurred.\n");
                return (1);
        }
        printf ("Version : %d\n", rtn / 65536);
        printf ("Level    : %d\n", rtn % 65536);
        return (0);
}
```

16.3.3 Job Submission API

This section explains the job submission API (Mp_SubmitJob).

Synopsis

[Windows]

```
#include "f3cuapi.h"
long Mp_SubmitJob (char *rhost,
            char *submitinf,
            char *env[ ],
            char *jobhost,
            char *jobname,
            long *jobno)
```

[UNIX]

```
#include "mjesapi.h "
int Mp_SubmitJob (char *rhost,
            char *submitinf,
            char *env[ ],
            char *jobhost,
            char *jobname,
            int *jobno)
```

Description

This API submits a job to the server where Systemwalker Operation Manager has been installed. If the submission of a job completes successfully, the job submission host name, job name and job number identifying the job will be returned.

Parameters

rhost

Specify NULL here.

submitinf

Specify the arguments of the qsub command as the job submission information. The arguments can be described in the same way as specifying options and operands in the qsub command.

See qsub Job Submit Command for further details on the Synopsis of arguments in the 7.20 qsub Job Submit Command.

env[ ]

Specify the environment variable to be inherited by the job. When there is no need for inheritance, specify NULL.

This environment variable will only be inherited by the job if you specify "-x" in the submitinf option.

jobhost

If the submission of a job completes successfully, the job submission host name identifying the job will be returned. This name can be used as information identifying the job.

jobname

If the submission of the job completes successfully, the job name determined by Job Execution Control will be returned. This name can be used as information identifying the job. If you specify the job name during the job submission, then the job is submitted with that name. However, when the job name is omitted and is declared in the job file, you can find the job name of the submitted job by checking the job name that was returned to the area specified in this parameter.

jobno

If the submission of a job completes successfully, the job number assigned by Job Execution Control will be returned. This number can be used as information identifying the job. By using this job number, you can manipulate the job.

Return Values

Return Value	Meaning	Action
0	The job was submitted normally.	-
-1	A system error occurred.	Collect "Job Execution Control" information with the maintenance information collection tool and contact a Fujitsu SE.
-1000 or less	Submission processing was aborted because an error was detected in Job Execution Control. Error messages can be obtained using the error message acquisition API.	Use the error message acquisition API to obtain the error message and take whatever action is required.

Caution

When the argument is pointing to an invalid address, this function ends abnormally, and no operations will be defined.

Program Example

Below is an example program of the job submission API

[Windows]

```
#include <windows.h>
#include <stdio.h>
#include <stdlib.h>
#include "f3cuapi.h"

int main (int argc, char **argv)
{
        char submitinf[1024+1];
        char jobhost [64+1];
        char jobname [64+1];
        long jobno = 0;
        long rtn = 0;
        char errmsg [256+1];

        /*
         * initialization working area
         */
        memset(submitinf, 0x00, sizeof(submitinf));
        memset(jobhost, 0x00, sizeof(jobhost));
        memset(jobname, 0x00, sizeof(jobname));
        memset(errmsg, 0x00, sizeof(errmsg));

        if (argc != 4) {
                printf ("Usage: %s <queue-name> <job-name> <job-file>\n", argv[0]);
                return (1);
        }

        /*
         * set submitinf
         */
        sprintf(submitinf,"-q %s -j %s %s",argv[1],argv[2],argv[3]);
```

```c
        /*
         * call Mp_SubmitJob() API
         */
        rtn = Mp_SubmitJob (NULL, submitinf, NULL, jobhost, jobname, &jobno);

        /*
         * check return code of Mp_SubmitJob() API
         */
        if (rtn == 0) {
                printf ("Job %s(%d.%s) was submitted.\n", jobname, jobno, jobhost);
                return (0);
        } else if (rtn == -1) {
                printf ("GetLastError() is %d.\n", GetLastError());
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
                printf ("errmsg = %s\n", errmsg);
        }
        return (1);
}
```

[UNIX]

```c
#include <stdio.h>
#include <stdlib.h>
#include <errno.h>
#include "mjesapi.h"

int main (int argc, char **argv)
{
        char submitinf[1024+1];
        char jobhost [64+1];
        char jobname [64+1];
        int jobno = 0;
        int rtn = 0;
        char errmsg [256+1];

        /*
         * initialization working area
         */
        memset(submitinf, 0x00, sizeof(submitinf));
        memset(jobhost, 0x00, sizeof(jobhost));
        memset(jobname, 0x00, sizeof(jobname));
        memset(errmsg, 0x00, sizeof(errmsg));

        if (argc != 4) {
                printf ("Usage: %s <queue-name> <job-name> <job-file>\n", argv[0]);
                return (1);
        }

        /*
         * set submitinf
         */
        sprintf(submitinf,"-q %s -j %s %s",argv[1],argv[2],argv[3]);

        /*
         * call Mp_SubmitJob() API
         */
        rtn = Mp_SubmitJob (NULL, submitinf, NULL, jobhost, jobname, &jobno);

        /*
```

```
         * check return code of Mp_SubmitJob() API
         */
        if (rtn == 0) {
                printf ("Job %s(%d.%s) was submitted.\n", jobname, jobno, jobhost);
                return (0);
        } else if (rtn == -1) {
                printf ("errno is %d.\n", errno);
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
                printf ("errmsg = %s\n", errmsg);
        }
        return (1);
}
```

16.3.4 Job Hold API [Windows]

This section explains the job hold API (Mp_HoldJob).

Synopsis

```
#include "f3cuapi.h"
long Mp_HoldJob (char *rhost,
         char *jobname,
         long jobno,
         char *jobhost,
         char *userforadmin)
```

Description

This API changes the job status to on hold and prevents it from being executed.

Parameters

rhost

Specify NULL here.

jobname

Specify the job name to hold. NULL can be specified. When NULL is specified, specify the job to hold with the job number specified in the jobno parameter.

jobno

Specify the job number of a job to be held. When specifying NULL in the jobname and when there are multiple jobs that have the same name, you must specify this parameter. If you specify jobname and there is no job with the same name, you can omit the job number. Specify 0 when omitting the job number.

jobhost

Specify NULL here.

userforadmin

Specify this parameter when a system administrator (users belonging to the Administrators group) holds users' jobs. Specify the owner name of the held job. If the owner name is not specified, NULL should be specified.

Return Values

Return Value	Meaning	Action
0	The job was successfully put on hold.	-
-1	A system error occurred.	Collect "Job Execution Control" information with the maintenance information collection tool and contact a Fujitsu SE.
-1000 or less	Job hold processing was aborted because an error was detected in Job Execution Control. Error messages can be obtained using the error message acquisition API.	Use the error message acquisition API to obtain the error message and take whatever action is required.

Caution

When the argument is pointing to an invalid address, this function ends abnormally, and no operations will be defined.

Program Example

Below is an example program of the job hold API:

```c
#include <windows.h>
#include <stdio.h>
#include "f3cuapi.h"
int main (int argc, char **argv)
{
        long jobno = 0;
        long rtn = 0;
        char errmsg[256+1];

        /*
         * initialization working area
         */
        memset(errmsg, 0x00, sizeof(errmsg));

        if (argc != 2) {
                printf ("Usage: %s <jobno>\n", argv[0]);
                return (1);
        }

        /*
         * set jobno
         */
        jobno = atoi(argv[1]);

        /*
         *  call Mp_HoldJob() API
         */
        rtn = Mp_HoldJob (NULL, NULL, jobno, NULL, NULL);

        /*
         * check return code of Mp_HoldJob() API
         */
        if (rtn == 0) {
                printf ("Job %d has been held.\n", jobno);
                return (0);
        } else if (rtn == -1) {
                printf ("GetLastError() is %d.\n", GetLastError());
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
```

```
                    printf ("errmsg = %s\n", errmsg);
        }
        return (1);
}
```

16.3.5 Job Hold API/EE [Windows]

This section explains the job hold API/EE (Mp_HoldJob_Ex).

Synopsis

```
#include "f3cuapi.h"
long Mp_HoldJob_Ex (char *rhost,
           char *jobname,
           long jobno,
           char *jobhost,
           char *userforadmin)
           int system_num)
```

Description

This API changes the job status to on hold and prevents it from being executed in Systemwalker Operation Manager EE.

Parameters

rhost

Specify NULL here.

jobname

Specify the job name to hold. NULL can be specified. When NULL is specified, specify the job to hold with the job number specified in the jobno parameter.

jobno

Specify the job number of a job to be held. When specifying NULL in the jobname and when there are multiple jobs that have the same name, you must specify this parameter. If you specify jobname and there is no job with the same name, you can omit the job number. Specify 0 when omitting the job number.

jobhost

Specify NULL here.

userforadmin

Specify this parameter when a system administrator (users belonging to the Administrators group) holds users' jobs. Specify the owner of the held job. If the owner name is not specified, NULL should be specified.

system_num

Specify operating target subsystem number using a range of 0 to 9.

Return Values

The job hold API (Mp_HoldJob_Ex) returns the values as below:

Return Value	Meaning	Action
0	The job was successfully put on hold.	-
-1	A system error occurred.	Collect "Job Execution Control" information with the maintenance

Return Value	Meaning	Action
		information collection tool and contact a Fujitsu SE.
-1000 or less	Job hold processing was aborted because an error was detected in Job Execution Control. Error messages can be obtained using the error message acquisition API.	Use the error message acquisition API to obtain the error message and take whatever action is required.

Caution

When the argument is pointing to an invalid address, this function ends abnormally, and no operations will be defined.

Program Example

Below is an example program of the job hold API/EE:

```
#include <windows.h>
#include <stdio.h>
#include "f3cuapi.h"
int main (int argc, char **argv)
{
        long jobno = 0;
        long rtn = 0;
        char errmsg[256+1];
        int system_num;

         /*
          * initialization working area
          */
        memset(errmsg, 0x00, sizeof(errmsg));

        if (argc != 2) {
                printf ("Usage: %s <jobno>\n", argv[0]);
                return (1);
        }

        /*
         * set jobno
         */
        jobno = atoi(argv[1]);

        /*
         *  call Mp_HoldJob_Ex() API
         */
        system_num = 0;
        rtn = Mp_HoldJob_Ex (NULL, NULL, jobno, NULL, NULL, system_num);

        /*
         * check return code of Mp_HoldJob Ex () API
         */
        if (rtn == 0) {
                printf ("Job %d has been held.\n", jobno);
                return (0);
        } else if (rtn == -1) {
                printf ("GetLastError() is %d.\n", GetLastError());
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
                printf ("errmsg = %s\n", errmsg);
        }
```

```
            return (1);
}
```

16.3.6 Job Hold Release API [Windows]

This section explains the job hold release API(Mp_RlsJob).

Synopsis

```
#include "f3cuapi.h"
long Mp_RlsJob (char *rhost,
            char *jobname,
            long jobno,
            char *jobhost,
            char *userforadmin)
```

Description

This API releases the job from the hold status so that it can be executed.

Parameters

rhost

Specify NULL here.

jobname

Specify the job name to be released. NULL can be specified. When NULL is specified, specify the job to be released with the job number specified in the jobno parameter.

jobno

Specify the job number to be released. When specifying NULL in the jobname and when there are multiple jobs that have the same name, you must specify this parameter. If you specify jobname and there is no job with the same name, you can omit the job number. Specify 0 when omitting the job number.

jobhost

Specify NULL here.

userforadmin

Specify this parameter when a system administrator (users belonging to the Administrators group) releases the users' jobs. Specify the owner name of the released job. If the owner name is not specified, NULL should be specified.

Return Values

Return Value	Meaning	Action
0	The job's hold status was successfully released.	-
-1	A system error occurred.	Collect "Job Execution Control" information with the maintenance information collection tool and contact a Fujitsu SE.
-1000 or less	Hold release processing was aborted because an error was detected in Job Execution Control. Error messages can be	Use the error message acquisition API to obtain the error message and take whatever action is required.

Return Value	Meaning	Action
	obtained using the error message acquisition API.	

Caution

When the argument is pointing to an invalid address, this function ends abnormally, and no operations will be defined.

Program Example

Below is an example program of the job hold release API:

```
#include <windows.h>
#include <stdio.h>
#include "f3cuapi.h"
int main (int argc, char **argv)
{
        long jobno = 0;
        long rtn = 0;
        char errmsg[256+1];

        /*
         * initialization working area
         */
        memset(errmsg, 0x00, sizeof(errmsg));

        if (argc != 2) {
                printf ("Usage: %s <jobno>\n", argv[0]);
                return (1);
        }

        /*
         * set jobno
         */
        jobno = atoi(argv[1]);

        /*
         *  call Mp_RlsJob() API
         */
        rtn = Mp_RlsJob (NULL, NULL, jobno, NULL, NULL);

        /*
         * check return code of Mp_RlsJob() API
         */
        if (rtn == 0) {
                printf ("Job %d has been released.\n", jobno);
                return (0);
        } else if (rtn == -1) {
                printf ("GetLastError() is %d.\n", GetLastError());
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
                printf ("errmsg = %s\n", errmsg);
        }
        return (1);
}
```

EE

16.3.7 Job Hold Release API/EE [Windows]

This section explains the job hold release API/EE (Mp_RlsJob_Ex).

Synopsis

```
#include "f3cuapi.h"
long Mp_RlsJob_Ex (char *rhost,
          char *jobname,
          long jobno,
          char *jobhost,
          char *userforadmin)
          int system_num)
```

Description

This API releases the job from the hold status so that it can be executed in Systemwalker Operation Manager EE.

Parameters

rhost

Specify NULL here.

jobname

Specify the name of job to be released. NULL can be specified. When NULL is specified, specify the job to be released with the job number specified in the jobno parameter.

jobno

Specify the job number to be released. When specifying NULL in the jobname and when there are multiple jobs that have the same name, you must specify this parameter. If you specify jobname and there is no job with the same name, you can omit the job number. Specify 0 when omitting the job number.

jobhost

Specify NULL here.

userforadmin

Specify this parameter when a system administrator (users belonging to the Administrators group) releases the users' jobs. Specify the owner of the released job. If the owner name is not specified, NULL should be specified.

system_num

Specify operating target subsystem number using a range from 0 to 9.

Return Values

Return Value	Meaning	Action
0	The job's hold status was successfully released.	-
-1	A system error occurred.	Collect "Job Execution Control" information with the maintenance information collection tool and contact a Fujitsu SE.
-1000 or less	Hold release processing was aborted because an error was detected in Job Execution Control. Error messages can be obtained using the error message acquisition API.	Use the error message acquisition API to obtain the error message and take whatever action is required.

Caution

When the argument is pointing to an invalid address, this function ends abnormally, and no operations will be defined.

Program Example

Below is an example program of the job hold release API/EE:

```c
#include <windows.h>
#include <stdio.h>
#include "f3cuapi.h"

int main (int argc, char **argv)
{
        long jobno = 0;
        long rtn = 0;
        char errmsg[256+1];
        int system_num;

        /*
         * initialization working area
         */
        memset(errmsg, 0x00, sizeof(errmsg));

        if (argc != 2) {
                printf ("Usage: %s <jobno>\n", argv[0]);
                return (1);
        }

        /*
         * set jobno
         */
        jobno = atoi(argv[1]);

        /*
         *  call Mp_RlsJob_Ex() API
         */
        system_num = 0;
        rtn = Mp_RlsJob_Ex (NULL, NULL, jobno, NULL, NULL, system_num);

        /*
         * check return code of Mp_RlsJob_Ex() API
         */
        if (rtn == 0) {
                printf ("Job %d has been released.\n", jobno);
                return (0);
        } else if (rtn == -1) {
                 printf ("GetLastError() is %d.\n", GetLastError());
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
                printf ("errmsg = %s\n", errmsg);
        }
        return (1);
}
```

16.3.8 Job Delete API

This section explains the job delete API (Mp_DeleteJob).

Synopsis

[Windows]

```
#include "f3cuapi.h"
long Mp_DeleteJob (char *rhost,
          char *jobname,
          long jobno,
          char *jobhost,
          char *userforadmin)
```

[UNIX]

```
#include "mjesapi.h "
int Mp_DeleteJob (char *rhost,
          char *jobname,
          int jobno,
          char *jobhost,
          char *userforadmin)
```

Description

This API deletes the job from the server where Systemwalker Operation Manager is installed. If the job is currently being executed, the execution is cancelled.

Parameters

rhost

Specify NULL here.

jobname

Specify the job name to be deleted. NULL can be specified. When NULL is specified, specify the job to be deleted with the job number specified in the jobno parameter.

jobno

Specify the job number of a job to be deleted. When specifying NULL in the jobname and when there are multiple jobs that have the same name, you must specify this parameter. If you specify jobname and there is no job with the same name, you can omit the job number. Specify 0 when omitting the job number.

jobhost

Specify NULL here.

userforadmin

Specify this parameter when a system administrator (users belonging to the Administrators group/ the superuser) deletes the users' jobs. Specify the owner of the deleted job. If the owner name is not specified, NULL should be specified.

Return Values

Return Value	Meaning	Action
0	The job was deleted normally.	-
-1	A system error occurred.	Collect "Job Execution Control" information with the maintenance information collection tool and contact a Fujitsu SE.

Return Value	Meaning	Action
-1000 or less	Deletion processing was aborted because an error was detected in Job Execution Control. Error messages can be obtained using the error message acquisition API.	Use the error message acquisition API to obtain the error message and take whatever action is required.

Caution

When the argument is pointing to an invalid address, this function ends abnormally, and no operations will be defined.

Program Example

Below is an example program of the job delete API:

[Windows]

```c
#include <windows.h>
#include <stdio.h>
#include "f3cuapi.h"

int main (int argc, char **argv)
{
        long jobno = 0;
        long rtn = 0;
        char errmsg[256+1];

        /*
         * initialization working area
         */
        memset(errmsg, 0x00, sizeof(errmsg));

        if (argc != 2) {
                printf ("Usage: %s <jobno>\n", argv[0]);
                return (1);
        }

        /*
         * set jobno
         */
        jobno = atoi(argv[1]);

        /*
         *  call Mp_DeleteJob() API
         */
        rtn = Mp_DeleteJob (NULL, NULL, jobno, NULL, NULL);

        /*
         * check return code of Mp_DeleteJob() API
         */
        if (rtn == 0) {
                printf ("Job %d has been deleted.\n", jobno);
                return (0);
        } else if (rtn == -1) {
                printf ("GetLastError() is %d.\n", GetLastError());
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
                printf ("errmsg = %s\n", errmsg);
        }
```

```
        return (1);
}
```

[UNIX]

```c
#include <stdio.h>
#include <errno.h>
#include "mjesapi.h"

int main (int argc, char **argv)
{
        int jobno = 0;
        int rtn = 0;
        char errmsg[256+1];

        /*
         * initialization working area
         */
        memset(errmsg, 0x00, sizeof(errmsg));

        if (argc != 2) {
                printf ("Usage: %s <jobno>\n", argv[0]);
                return (1);
        }

        /*
         * set jobno
         */
        jobno = atoi(argv[1]);

        /*
         *  call Mp_DeleteJob() API
         */
        rtn = Mp_DeleteJob (NULL, NULL, jobno, NULL, NULL);

        /*
         * check return code of Mp_DeleteJob() API
         */
        if (rtn == 0) {
                printf ("Job %d has been deleted.\n", jobno);
                return (0);
        } else if (rtn == -1) {
                printf ("errno is %d.\n", errno);
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
                printf ("errmsg = %s\n", errmsg);
        }
        return (1);
}
```

EE

16.3.9 Job Delete API/EE

This section explains the job delete API/EE (Mp_DeleteJob_Ex).

Synopsis

[Windows]

```
#include "f3cuapi.h"
long Mp_DeleteJob_Ex (char *rhost,
                char *jobname,
                long jobno,
                char *jobhost,
                char *userforadmin,
                int system_num)
```

[UNIX]

```
#include "mjesapi.h "
int Mp_DeleteJob_Ex (char *rhost,
                char *jobname,
                int jobno,
                char *jobhost,
                char *userforadmin,
                int system_num)
```

Description

This API deletes the job from the server where Systemwalker Operation Manager is installed. If the job is currently being executed, the execution is cancelled.

Parameters

rhost

Specify NULL here.

jobname

Specify the job name to be deleted. NULL can be specified. When NULL is specified, specify the job to be deleted with the job number specified in the jobno parameter.

jobno

Specify the job number of a job to be deleted. When specifying NULL in the jobname and when there are multiple jobs that have the same name, you must specify this parameter. If you specify jobname and there is no job with the same name, you can omit the job number. Specify 0 when omitting the job number.

jobhost

Specify NULL here.

userforadmin

Specify this parameter when a system administrator (users belonging to the Administrators group/ the superuser) deletes the users' jobs. Specify the owner of the deleted job. If the owner name is not specified, NULL should be specified.

system_num

Specify operating target subsystem number using a range of 0 to 9.

Return Values

Return Value	Meaning	Action
0	The job was deleted normally.	-
-1	A system error occurred.	Collect "Job Execution Control" information with the maintenance information collection tool and contact a Fujitsu SE.

Return Value	Meaning	Action
-1000 or less	Deletion processing was aborted because an error was detected in Job Execution Control. Error messages can be obtained using the error message acquisition API.	Use the error message acquisition API to obtain the error message and take whatever action is required.

Caution

When the argument is pointing to an invalid address, this function ends abnormally, and no operations will be defined.

Program Example

Below is an example program of the job delete API/EE:

[Windows]

```c
#include <windows.h>
#include <stdio.h>
#include "f3cuapi.h"

int main (int argc, char **argv)
{
        long jobno = 0;
        long rtn = 0;
        char errmsg[256+1];
        int system_num;

        /*
         * initialization working area
         */
        memset(errmsg, 0x00, sizeof(errmsg));

        if (argc != 2) {
                printf ("Usage: %s <jobno>\n", argv[0]);
                return (1);
        }

        /*
         * set jobno
         */
        jobno = atoi(argv[1]);

        /*
         *  call Mp_DeleteJob_Ex() API
         */
        system_num = 0;
        rtn = Mp_DeleteJob_Ex (NULL, NULL, jobno, NULL, NULL, system_num);

        /*
         * check return code of Mp_DeleteJob_Ex() API
         */
        if (rtn == 0) {
                printf ("Job %d has been deleted.\n", jobno);
                return (0);
        } else if (rtn == -1) {
                printf ("GetLastError() is %d.\n", GetLastError());
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
                printf ("errmsg = %s\n", errmsg);
        }
```

```
                return (1);
}
```

[UNIX]

```
#include <stdio.h>
#include <errno.h>
#include "mjesapi.h"

int main (int argc, char **argv)
{
        int jobno = 0;
        int rtn = 0;
        char errmsg[256+1];
        int system_num;

        /*
         * initialization working area
         */
        memset(errmsg, 0x00, sizeof(errmsg));

        if (argc != 2) {
                printf ("Usage: %s <jobno>\n", argv[0]);
                return (1);
        }

        /*
         * set jobno
         */
        jobno = atoi(argv[1]);

        /*
         *  call Mp_DeleteJob_Ex() API
         */
        system_num = 0;
        rtn = Mp_DeleteJob_Ex (NULL, NULL, jobno, NULL, NULL, system_num);

        /*
         * check return code of Mp_DeleteJob_Ex() API
         */
        if (rtn == 0) {
                printf ("Job %d has been deleted.\n", jobno);
                return (0);
        } else if (rtn == -1) {
                printf ("errno is %d.\n", errno);
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
                printf ("errmsg = %s\n", errmsg);
        }
        return (1);
}
```

16.3.10 Job Move API [Windows]

This section explains the job move API (Mp_MoveJob).

Synopsis

```
#include "f3cuapi.h"
long Mp_MoveJob (char *rhost,
          char *jobname,
          long jobno,
          char *jobhost,
          char *queue,
          char *userforadmin)
```

Description

This API moves a job to another queue.

Parameters

rhost

Specify NULL here.

jobname

Specify the job name to be moved. NULL can be specified. When NULL is specified, specify the job to be moved with the job number specified in the jobno parameter.

jobno

Specify the job number to be moved. When specifying NULL in the jobname and when there are multiple jobs that have the same name, you must specify this parameter. If you specify jobname and there is no job with the same name, you can omit the job number. Specify 0 when omitting the job number.

jobhost

Specify NULL here.

queue

Specify the destination queue name.

userforadmin

Specify this parameter when a system administrator (users belonging to the Administrators group) moves the users' jobs. Specify the owner name of the moved job. If the owner name is not specified, NULL should be specified.

Return Values

Return Value	Meaning	Action
0	The job was moved normally.	-
-1	A system error occurred.	Collect "Job Execution Control" information with the maintenance information collection tool and contact a Fujitsu SE.
-1000 or less	Movement processing was aborted because an error was detected in Job Execution Control. Error messages can be obtained using the error message acquisition API.	Use the error message acquisition API to obtain the error message and take whatever action is required.

Caution

When the argument is pointing to an invalid address, this function ends abnormally, and no operations will be defined.

You cannot move a job from the distributed execution queue to another queue and vice versa.

Program Example

Below is an example program of the job move API:

```c
#include <windows.h>
#include <stdio.h>
#include "f3cuapi.h"

int main (int argc, char **argv)
{
        long jobno = 0;
        long rtn = 0;
        char errmsg[256+1];

        /*
         * initialization working area
         */
        memset(errmsg, 0x00, sizeof(errmsg));

        if (argc != 3) {
                printf ("Usage: %s <jobno> <queuename>\n", argv[0]);
                return (1);
        }

        /*
         * set jobno
         */
        jobno = atoi(argv[1]);

        /*
         *  call Mp_MoveJob() API
         */
        rtn = Mp_MoveJob (NULL, NULL, jobno, NULL, argv[2],NULL);

        /*
         * check return code of Mp_MoveJob() API
         */
        if (rtn == 0) {
                printf ("Job %d has been moved.\n", jobno);
                return (0);
        } else if (rtn == -1) {
                printf ("GetLastError() is %d.\n", GetLastError());
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
                printf ("errmsg = %s\n", errmsg);
        }
        return (1);
}
```

16.3.11 Job Move API/EE [Windows]

This section explains the job move API/EE (Mp_MoveJob_Ex).

Synopsis

```c
#include "f3cuapi.h"
long Mp_MoveJob_Ex (char *rhost,
        char *jobname,
```

```
            long jobno,
            char *jobhost,
            char *queue,
            char *userforadmin
            int system_num)
```

Description

This API moves a job to another queue in Systemwalker Operation Manager EE.

Parameters

rhost

Specify NULL here.

jobname

Specify the job name to be moved. NULL can be specified. When NULL is specified, specify the job to be moved with the job number specified in the jobno parameter.

jobno

Specify the job number to be moved. When specifying NULL in the jobname and when there are multiple jobs that have the same name, you must specify this parameter. If you specify jobname and there is no job with the same name, you can omit the job number. Specify 0 when omitting the job number.

jobhost

Specify NULL here.

queue

Specify the destination queue name.

userforadmin

Specify this parameter when a system administrator (users belonging to the Administrators group) moves the users' jobs. Specify the owner name of the moved job. If the owner name is not specified, NULL should be specified.

system_num

Specify operating target subsystem number using a range of 0 to 9.

Return Values

Return Value	Meaning	Action
0	The job was moved normally.	-
-1	A system error occurred.	Collect "Job Execution Control" information with the maintenance information collection tool and contact a Fujitsu SE.
-1000 or less	Movement processing was aborted because an error was detected in Job Execution Control. Error messages can be obtained using the error message acquisition API.	Use the error message acquisition API to obtain the error message and take whatever action is required.

Cautions

When the argument is pointing to an invalid address, this function ends abnormally, and no operations will be defined.

You cannot move a job from the distributed execution queue to another queue and vice versa.

Program Example

Below is an example program of the job move API/EE:

```c
#include <windows.h>
#include <stdio.h>
#include "f3cuapi.h"

int main (int argc, char **argv)
{
        long jobno = 0;
        long rtn = 0;
        char errmsg[256+1];
        int system_num;

        /*
         * initialization working area
         */
        memset(errmsg, 0x00, sizeof(errmsg));

        if (argc != 3) {
                printf ("Usage: %s <jobno> <queuename>\n", argv[0]);
                return (1);
        }

        /*
         * set jobno
         */
        jobno = atoi(argv[1]);

        /*
         *  call Mp_MoveJob_Ex() API
         */
        system_num = 0;
        rtn = Mp_MoveJob_Ex (NULL, NULL, jobno, NULL, argv[2], NULL, system_num);

        /*
         * check return code of Mp_MoveJob_Ex() API
         */
        if (rtn == 0) {
                printf ("Job %d has been moved.\n", jobno);
                return (0);
        } else if (rtn == -1) {
                printf ("GetLastError() is %d.\n", GetLastError());
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
                printf ("errmsg = %s\n", errmsg);
        }
        return (1);
}
```

16.3.12 Job Status Acquisition API

This section explains the job status acquisition API (Mp_StatusJob).

Synopsis

[Windows]

```
#include "f3cuapi.h"
long Mp_StatusJob (char *rhost,
            char *jobname,
            long jobno,
            char *jobhost,
            ulong *compcode,
            char *userforadmin)
```

[UNIX]

```
#include "mjesapi.h"
int Mp_StatusJob (char *rhost,
            char *jobname,
            int jobno,
            char *jobhost,
            uint *compcode,
            char *userforadmin)
```

Description

This API obtains the status of a job. When the job is in the completion status, you can obtain the job's completion code.

Parameters

rhost

Specify NULL here.

jobname

Specify the job name of a job to be obtained. NULL can be specified. When NULL is specified, specify the job to be obtained with the job number specified in the jobno parameter.

jobno

Specify the job number of the job to be obtained. When specifying NULL in the jobname and when there are multiple jobs that have the same name, you must specify this parameter. If you specify jobname and there is no job with the same name, you can omit the job number. Specify 0 when omitting the job number.

jobhost

Specify NULL here.

compcode

Specify the address of the return area for the completion code of a job. When the job has a completed condition, the completion code will be stored in this area.

userforadmin

Specify this parameter when a system administrator (users belonging to the Administrators group/ the superuser) obtains the status of users' jobs. Specify the owner name of the obtained job. If the owner name is not specified, NULL should be specified.

Return Values

Return Value	Meaning	Action
0	Specified job does not exist.	-
1	Execution waiting status.	-
2	Execution on hold status.	-

Return Value	Meaning	Action
3	Executing.	-
4	Output on hold status.	-
5	Job is completed.	-
-1	A system error occurred.	Collect "Job Execution Control" information with the maintenance information collection tool and contact a Fujitsu SE.
-1000 or less	Status acquisition processing was aborted because an error was detected in Job Execution Control. Error messages can be obtained using the error message acquisition API.	Use the error message acquisition API to obtain the error message and take whatever action is required.

Caution

When the argument is pointing to an invalid address, this function ends abnormally, and no operations will be defined.

Program Example

Below is an example program of the job status acquisition API:

[Windows]

```
#include <windows.h>
#include <stdio.h>
#include "f3cuapi.h"

int main (int argc, char **argv)
{
        long jobno = 0;
        long rtn = 0;
        unsigned long compcode = 0;
        char errmsg[256+1];

        /*
         * initialization working area
         */
        memset(errmsg, 0x00, sizeof(errmsg));

        if (argc != 2) {
                printf ("Usage: %s <jobno>\n", argv[0]);
                return (1);
        }

        /*
         * set jobno
         */
        jobno = atoi(argv[1]);

        /*
         *  call Mp_StatusJob() API
         */
        rtn = Mp_StatusJob (NULL, NULL, jobno, NULL, &compcode, NULL);

        /*
         * check return code of Mp_StatusJob() API
         */
```

```
        if (rtn == 0) {
                printf ("Job %d is not found.\n", jobno);
                return (0);
        } else if (rtn == 1) {
                printf ("Job %d is waiting for execution.\n", jobno);
                return (0);
        } else if (rtn == 2) {
                printf ("Job %d is holding for execution.\n", jobno);
                return (0);
        } else if (rtn == 3) {
                printf ("Job %d is executing.\n", jobno);
                return (0);
        } else if (rtn == 4) {
                printf ("Job %d is holding for output.\n", jobno);
                return (0);
        } else if (rtn == 5) {
                printf ("Job %d was completed. completion code = %d\n", jobno, compcode);
                return (0);
        } else if (rtn == -1) {
                printf ("GetLastError() is %d.\n", GetLastError());
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
                printf ("errmsg = %s\n", errmsg);
        }
        return (1);
}
```

[UNIX]

```
#include <stdio.h>
#include <errno.h>
#include "mjesapi.h"

int main (int argc, char **argv)
{
        int jobno = 0;
        int rtn = 0;
        unsigned int compcode = 0;
        char errmsg[256+1];

        /*
         * initialization working area
         */
        memset(errmsg, 0x00, sizeof(errmsg));

        if (argc != 2) {
                printf ("Usage: %s <jobno>\n", argv[0]);
                return (1);
        }

        /*
         * set jobno
         */
        jobno = atoi(argv[1]);

        /*
         *  call Mp_StatusJob() API
         */
        rtn = Mp_StatusJob (NULL, NULL, jobno, NULL, &compcode, NULL);
```

```
        /*
         * check return code of Mp_StatusJob() API
         */
        if (rtn == 0) {
                printf ("Job %d is not found.\n", jobno);
                return (0);
        } else if (rtn == 1) {
                printf ("Job %d is waiting for execution.\n", jobno);
                return (0);
        } else if (rtn == 2) {
                printf ("Job %d is holding for execution.\n", jobno);
                return (0);
        } else if (rtn == 3) {
                printf ("Job %d is executing.\n", jobno);
                return (0);
        } else if (rtn == 4) {
                printf ("Job %d is holding for output.\n", jobno);
                return (0);
        } else if (rtn == 5) {
                printf ("Job %d was completed. completion code = %d\n", jobno, compcode);
                return (0);
        } else if (rtn == -1) {
                printf ("errno is %d.\n", errno);
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
                printf ("errmsg = %s\n", errmsg);
        }
        return (1);
}
```

16.3.13 Job Status Acquisition API/EE

This section explains the job status acquisition API/EE (Mp_StatusJob_Ex).

Synopsis

[Windows]

```
#include "f3cuapi.h"
long Mp_StatusJob_Ex (char *rhost,
        char *jobname,
        long jobno,
        char *jobhost,
        ulong *compcode,
        char *userforadmin,
        int system_num)
```

[UNIX]

```
#include "mjesapi.h"
int Mp_StatusJob_Ex (char *rhost,
        char *jobname,
        int jobno,
        char *jobhost,
        uint *compcode,
        char *userforadmin,
        int system_num)
```

Description

This API obtains the status of a job in Systemwalker Operation Manager EE. When the job is in the completion status, you can obtain the job's completion code.

Parameters

rhost

Specify NULL here.

jobname

Specify the job name of a job to be obtained. NULL can be specified. When NULL is specified, specify the job to be obtained with the job number specified in the jobno parameter.

jobno

Specify the job number of the job to be obtained. When specifying NULL in the jobname and when there are multiple jobs that have the same name, you must specify this parameter. If you specify jobname and there is no job with the same name, you can omit the job number. Specify 0 when omitting the job number.

jobhost

Specify NULL here.

compcode

Specify the address of the return area for the completion code of a job. When the job has a completed condition, the completion code will be stored in this area.

userforadmin

Specify this parameter when a system administrator (users belonging to the Administrators group/ the superuser) obtains the status of users' jobs. Specify the owner name of the obtained job. If the owner name is not specified, NULL should be specified.

system_num

Specify operating target subsystem number using a range of 0 to 9.

Return Values

Return Value	Meaning	Action
0	Specified job does not exist.	-
1	Execution waiting status.	-
2	Execution on hold status.	-
3	Executing.	-
4	Output on hold status.	-
5	Job is completed.	-
-1	A system error occurred.	Collect "Job Execution Control" information with the maintenance information collection tool and contact a Fujitsu SE.
-1000 or less	Status acquisition processing was aborted because an error was detected in Job Execution Control. Error messages can be obtained using the error message acquisition API.	Use the error message acquisition API to obtain the error message and take whatever action is required.

Caution

When the argument is pointing to an invalid address, this function ends abnormally, and no operations will be defined.

Program Example

Below is an example program of the job status acquisition API/EE:

[Windows]

```c
#include <windows.h>
#include <stdio.h>
#include "f3cuapi.h"

int main (int argc, char **argv)
{
        long jobno = 0;
        long rtn = 0;
        unsigned long compcode = 0;
        char errmsg[256+1];
        int system_num;

        /*
         * initialization working area
         */
        memset(errmsg, 0x00, sizeof(errmsg));

        if (argc != 2) {
                printf ("Usage: %s <jobno>\n", argv[0]);
                return (1);
        }

        /*
         * set jobno
         */
        jobno = atoi(argv[1]);

        /*
         *  call Mp_StatusJob_Ex() API
         */
        system_num = 0;
        rtn = Mp_StatusJob_Ex (NULL, NULL, jobno, NULL, &compcode, NULL, system_num);

        /*
         * check return code of Mp_StatusJob_Ex() API
         */
        if (rtn == 0) {
                printf ("Job %d is not found.\n", jobno);
                return (0);
        } else if (rtn == 1) {
                printf ("Job %d is waiting for execution.\n", jobno);
                return (0);
        } else if (rtn == 2) {
                printf ("Job %d is holding for execution.\n", jobno);
                return (0);
        } else if (rtn == 3) {
                printf ("Job %d is executing.\n", jobno);
                return (0);
        } else if (rtn == 4) {
                printf ("Job %d is holding for output.\n", jobno);
                return (0);
        } else if (rtn == 5) {
```

```
                printf ("Job %d was completed. completion code = %d\n", jobno, compcode);
                return (0);
        } else if (rtn == -1) {
                printf ("GetLastError() is %d.\n", GetLastError());
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
                printf ("errmsg = %s\n", errmsg);
        }
        return (1);
}
```

[UNIX]

```
#include <stdio.h>
#include <errno.h>
#include "mjesapi.h"

int main (int argc, char **argv)
{
        int jobno = 0;
        int rtn = 0;
        unsigned int compcode = 0;
        char errmsg[256+1];
        int system_num;

        /*
         * initialization working area
         */
        memset(errmsg, 0x00, sizeof(errmsg));

        if (argc != 2) {
                printf ("Usage: %s <jobno>\n", argv[0]);
                return (1);
        }

        /*
         * set jobno
         */
        jobno = atoi(argv[1]);

        /*
         *  call Mp_StatusJob_Ex() API
         */
        system_num = 0;
        rtn = Mp_StatusJob_Ex (NULL, NULL, jobno, NULL, &compcode, NULL, system_num);

        /*
         * check return code of Mp_StatusJob_Ex() API
         */
        if (rtn == 0) {
                printf ("Job %d is not found.\n", jobno);
                return (0);
        } else if (rtn == 1) {
                printf ("Job %d is waiting for execution.\n", jobno);
                return (0);
        } else if (rtn == 2) {
                printf ("Job %d is holding for execution.\n", jobno);
                return (0);
        } else if (rtn == 3) {
                printf ("Job %d is executing.\n", jobno);
```

```
                    return (0);
        } else if (rtn == 4) {
                    printf ("Job %d is holding for output.\n", jobno);
                    return (0);
        } else if (rtn == 5) {
                    printf ("Job %d was completed. completion code = %d\n", jobno, compcode);
                    return (0);
        } else if (rtn == -1) {
                    printf ("errno is %d.\n", errno);
        } else if (rtn <= -1000) {
                    printf ("Error code is %d.\n", rtn);
                    Mp_GetMJESerror (errmsg);
                    printf ("errmsg = %s\n", errmsg);
        }
        return (1);
}
```

16.3.14 Job List Information Acquisition API [Windows]

This section explains the job list information acquisition API (Mp_ListJob).

Synopsis

```
#include "f3cuapi.h"
long Mp_ListJob (char *rhost,
            long mode,
            struct Mp_JOBINF_1 *jobinf[ ]
            char *userforadmin)
```

Description

This API obtains a list of job information. It obtains the information on all the jobs that are owned by the user who issued this function. If a system administrator (users belonging to the Administrators group) issued this function, then the information on all the jobs that exist on the server where Systemwalker Operation Manager is installed is gathered. However, if userforadmin is specified, only the information on the jobs that are owned by the corresponding user will be obtained.

Parameters

rhost

Specify NULL here.

mode

Specify one of the following process modes. After GETJOB, you must always specify FREEJOB.

GETJOB:

Obtains the job list information.

FREEJOB:

Frees the area that was used for the job list information obtained by GETJOB.

jobinf[ ]

GETJOB:

Specify the area address to which the address of the job information structure (Mp_JOBINF_1) array will be notified.

FREEJOB:

Specify the area address to which the address of the job information structure array was notified in GETJOB. After issuing GETJOB and until the issuing of FREEJOB, you should not change the contents of the area.

userforadmin

Specify this parameter when a system administrator (users belonging to the Administrators group) obtains only the users' jobs status. Specify the owner name of the obtained job. If the owner name is not specified, NULL should be specified.

Return Values

Return Value	Meaning	Action
0	The relevant job does not exist.	-
1 or greater	The job information list was obtained successfully. The number of jobs was returned.	-
-1	A system error occurred.	Collect "Job Execution Control" information with the maintenance information collection tool and contact a Fujitsu SE.
-1000 or less	Processing was aborted because an error was detected in Job Execution Control. Error messages can be obtained using the error message acquisition API.	Use the error message acquisition API to obtain the error message and take whatever action is required.

Caution

When the argument is pointing to an invalid address, this function ends abnormally, and no operations will be defined.

Format of the Job Information Structure (Mp_JOBINF_1)

The following is the format of the job information structure:

This structure is defined in the INCLUDE file f3cuapi.h".

```
typedef struct _Mp_JOBINF_1 {
  char      jobname[];        /* job name                                   */
  char      jobhost[];        /* submission source host name                */
  long      jobno;            /* job number                                 */
  char      user[];           /* job owner and user                         */
  char      queue[];          /* queue name                                 */
  long      jobtype;          /* job type                                   */
  long      status;           /* job status                                 */
  char      comment[];        /* comment entered during the submission process */

} Mp_JOBINF_1
```

Following provides the explanations about each item.

jobname

The job name is set.

jobhost

The submission source host name is set.

jobno

The job number is set.

user

The job owner and user are set.

queue

The queue name is set.

jobtype

The job type is set. The following types can be set for a job:

Value	Type
1	Demand job
2	Schedule job

status

The job status is set. The following statuses can be set for a job:

Value	Status
1	Waiting for execution
2	Execution on hold
3	Executing
4	Output on hold

comment

The comment specified during the submission process is set.

Program Example

Below is an example program of the job list information acquisition API:

```
#include <windows.h>
#include <stdio.h>
#include "f3cuapi.h"

int main (int argc, char **argv)
{
        Mp_JOBINF_1 *jobinf = 0;
        Mp_JOBINF_1 *jinf = 0;

        long mode;
        long idx;
        long rtn = 0;
        char errmsg[256+1];

        /*
         * initialization working area
         */
        memset(errmsg, 0x00, sizeof(errmsg));

        mode = GETJOB;
        /*
         *  call Mp_ListJob() API
         */
        rtn = Mp_ListJob (NULL, mode, &jobinf, NULL);

        /*
         * check return code of Mp_ListJob() API
         */
        if (rtn == -1) {
                printf ("GetLastError() is %d.\n", GetLastError());
                return (1);
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
```

```
                  printf ("errmsg = %s\n", errmsg);
                  return (1);
         } else if (rtn == 0) {
                  printf ("Job is nothing.\n", rtn);
                  return (0);
         }

         jinf = jobinf;

         printf("job name            jobno user name            status\n");
         for (idx = 0; idx < rtn; idx++) {

                  printf("%20.20s %5.5d %20.20s ",
                          jinf->jobname, jinf->jobno, jinf->user);
                  switch(jinf->status) {
                  case 1  :
                          printf("waiting   ");
                          break;
                  case 2  :
                          printf("holding   ");
                          break;
                  case 3  :
                          printf("running   ");
                          break;
                  case 4  :
                          printf("outhold   ");
                          break;
                  }
                  printf("\n");
                  jinf++;

         }

         mode = FREEJOB;

         /*
          *  call Mp_ListJob() API
          */
         Mp_ListJob (NULL, mode, &jobinf, NULL);

         return(0);
}
```

16.3.15 Job List Information Acquisition API/EE [Windows]

This section explains the job list information acquisition API/EE (Mp_ListJob_Ex).

Synopsis

```
#include "f3cuapi.h"
long Mp_ListJob_Ex (char *rhost,
          long mode,
          struct Mp_JOBINF_1 *jobinf[ ],
          char *userforadmin,
          int system_num)
```

Description

This API obtains a list of job information in Systemwalker Operation Manager EE. It obtains the information on all the jobs that are owned by the user who issued this function. If a system administrator (users belonging to the Administrators group) issued this function, then all the job information on every server that exist on the server where Systemwalker Operation

Manager is installed is gathered. However, when userforadmin is specified, only the information on the jobs that are owned by the corresponding user will be obtained.

Parameters

rhost

Specify NULL here.

mode

Specify one of the following process modes. After GETJOB, you must always specify FREEJOB.

GETJOB:

Obtains the job list information.

FREEJOB:

Frees the area that was used for the job list information obtained by GETJOB.

jobinf[ ]

GETJOB:

Specify the area address to which the address of the job information structure (Mp_JOBINF_1) array will be notified.

FREEJOB:

Specify the area address to which the address of the job information structure array was notified in GETJOB. After issuing GETJOB and until the issuing of FREEJOB, you should not change the contents of the area.

userforadmin

Specify this parameter when a system administrator (users belonging to the Administrators group) obtains only the users' jobs status. Specify the owner name of the obtained job. If the owner name is not specified, NULL should be specified.

system_num

Specify operating target subsystem number using a range of 0 to 9.

Return Values

Return Value	Meaning	Action
0	The relevant job does not exist.	-
1 or greater	The job information list was obtained successfully. The number of jobs was returned.	-
-1	A system error occurred.	Collect "Job Execution Control" information with the maintenance information collection tool and contact a Fujitsu SE.
-1000 or less	Processing was aborted because an error was detected in Job Execution Control. Error messages can be obtained using the error message acquisition API.	Use the error message acquisition API to obtain the error message and take whatever action is required.

Caution

When the argument is pointing to an invalid address, this function ends abnormally, and no operations will be defined.

Format of the Job Information Structure (Mp_JOBINF_1)

The following is the format of the job information structure:

This structure is defined in the INCLUDE file "f3cuapi.h".

```
typedef struct _Mp_JOBINF_1{
  char      jobname[];          /* job name                                  */
  char      jobhost[];          /* submission source host name               */
  long      jobno;              /* job number                              */
  char      user[];             /* job owner and user                       */
  char      queue[];            /* queue name                               */
  long      jobtype;            /* job type                                 */
  long      status;             /* job status                               */
  char      comment[];          /* comment entered during the submission process */
}Mp_JOBINF_1
```

Following provides the explanations about each item.

jobname

The job name is set.

jobhost

The submission source host name is set.

jobno

The job number is set.

user

The job owner and user are set.

queue

The queue name is set.

jobtype

The job type is set. The following types can be set for a job:

Value	Type
1	Demand job
2	Schedule job

status

The job status is set. The following statuses can be set for a job:

Value	Status
1	Waiting for execution
2	Execution on hold
3	Executing
4	Output on hold

comment

The comment specified during the submission process is set.

Program Example

Below is an example program of the job list information acquisition API/EE:

```
#include <windows.h>
#include <stdio.h>
```

```c
#include "f3cuapi.h"

int main (int argc, char **argv)
{
        Mp_JOBINF_1 *jobinf = 0;
        Mp_JOBINF_1 *jinf = 0;

        long mode;
        long idx;
        long rtn = 0;
        char errmsg[256+1];
        int system_num;

        /*
         * initialization working area
         */
        memset(errmsg, 0x00, sizeof(errmsg));

        mode = GETJOB;

        /*
         *  call Mp_ListJob_Ex() API
         */
        system_num = 0;
        rtn = Mp_ListJob_Ex (NULL, mode, &jobinf, NULL, system_num);

        /*
         * check return code of Mp_ListJob_Ex() API
         */
        if (rtn == -1) {
                printf ("GetLastError() is %d.\n", GetLastError());
                return (1);
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
                printf ("errmsg = %s\n", errmsg);
                return (1);
        } else if (rtn == 0) {
                printf ("Job is nothing.\n", rtn);
                return (0);
        }

        jinf = jobinf;

        printf("job name             jobno user name            status\n");
        for (idx = 0; idx < rtn; idx++) {

                printf("%20.20s %5.5d %20.20s ",
                        jinf->jobname, jinf->jobno, jinf->user);
                switch(jinf->status) {
                case 1  :
                        printf("waiting   ");
                        break;
                case 2  :
                        printf("holding   ");
                        break;
                case 3  :
                        printf("running   ");
                        break;
                case 4  :
                        printf("outhold   ");
                        break;
                }
```

```
                printf("\n");
                jinf++;

        }

        mode = FREEJOB;

        /*
         *  call Mp_ListJob_Ex() API
         */
        Mp_ListJob_Ex (NULL, mode, &jobinf, NULL, system_num);

        return(0);
}
```

16.3.16 Queue List Information Acquisition API [Windows]

This section explains the queue list information acquisition API (Mp_ListQueue).

Synopsis

```
#include "f3cuapi.h"
long Mp_ListQueue (char *rhost,
                   long mode,
                   struct Mp_QUEUEINF_1 *queueinf[ ])
```

Description

This API obtains the list of queue information.

Parameters

rhost

Specify NULL here.

mode

Specify one of the following process modes. After GETQST, you must always specify FREEQST.

GETQST:

Obtains the queue list information.

FREEQST:

Frees the area that was used for the queue list information obtained by GETQST.

queueinf[ ]

GETQST:

Specify the address to which the address of the queue information structure (Mp_QUEUEINF_1) will be notified.

FREEQST:

Specify the area address to which the address of the job information structure's array was notified in GETQST. Between the issuing GETQUE and FREEQST, you should not change the contents of the area.

Return Values

Return Value	Meaning	Action
1 or greater	This indicates that the queue information list was obtained normally and the number of queues is returned.	-
-1	A system error occurred.	Collect "Job Execution Control" information with the maintenance information collection tool and contact a Fujitsu SE.
-1000 or less	Processing was aborted because an error was detected in Job Execution Control. Error messages can be obtained using the error message acquisition API.	Use the error message acquisition API to obtain the error message and take whatever action is required.

Caution

When the argument is pointing to an invalid address, this function ends abnormally, and no operations will be defined.

Format of the Queue Information Structure (Mp_QUEUEINF_1)

The following is the format of the queue information structure:

This structure is defined in the INCLUDE file f3cuapi.h".

```
typedef struct _Mp_QUEUEINF_1 {
  char      queuename[];             /*queue name                              */
  long      queuetype;               /*queue type                              */
  char      status;                  /*queue status                            */
  long      maxjob;                  /*maximum possible job submission quantity */
  long      maxexec;                 /*maximum possible job execution quantity  */
  long      elapsed;                 /*maximum time lapse limit (seconds)      */
  long      dfltprty;                /*default priority order                  */
  long      dfltdprty;               /*default priority                        */
  long      job_wait;                /*jobs waiting to be executed             */
  long      job_hold;                /*jobs on hold                            */
  long      job_run;                 /*jobs in execution                       */
  char      hostgrp[];               /*distributed host group name             */

} Mp_QUEUEINF_1
```

Following provides the explanations about each item.

queuename

The queue name is set.

queuetype

The queue type is set. The following types can be set for a queue:

Value	Type
1	Local queue
2	Distributed execution queue

status

The queue status is set. The following statuses can be set for a queue:

Value	Status
1	Operating

Value	Status
2	Paused

maxjob

The maximum number of possible jobs to be submitted is set.

maxexec

The maximum number of possible jobs to be executed is set.

elapsed

The maximum time lapse limit is set in seconds.

dfltprty

The default priority order is set.

dfltdprty

The default priority is set.

job_wait

The number of jobs waiting to be executed at the moment when the API was called up is set.

job_hold

The number of jobs held at the moment when the API was called is set.

job_run

The number of jobs being executed at the moment when the API was called is set.

hostgrp

The distributed host group name is set. NULL is set when it is a local queue.

Program Example

Below is an example program of the queue list information acquisition API:

```c
#include <windows.h>
#include <stdio.h>
#include "f3cuapi.h"

int main (int argc, char **argv)
{

        Mp_QUEUEINF_1 *queinf = 0;
        Mp_QUEUEINF_1 *qinf = 0;

        long mode;
        long idx;
        long rtn = 0;
        char errmsg[256+1];

        /*
         * initialization working area
         */
        memset(errmsg, 0x00, sizeof(errmsg));

        mode = GETQST;
        /*
         *  call Mp_ListQueue() API
         */
        rtn = Mp_ListQueue (NULL, mode, &queinf);
```

```
        /*
         * check return code of Mp_ListQueue() API
         */
        if (rtn == -1){
                printf ("GetLastError() is %d.\n", GetLastError());
                return (1);
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
                printf ("errmsg = %s\n", errmsg);
                return (1);
        }

        qinf = queinf;

        printf("queue name      wait  hold  run   status\n");
        for (idx = 0; idx < rtn; idx++) {

                printf("%15.15s %5.5d %5.5d %5.5d ",
                        qinf->queuename, qinf->job_wait, qinf->job_hold, qinf->job_run);
                switch(qinf->status){
                case 1  :
                        printf("active  ");
                        break;
                case 0  :
                default :
                        printf("inactive");
                        break;
                }
                printf("\n");
                qinf++;

        }

        mode = FREEQST;

        /*
         *  call Mp_ListQueue() API
         */
        Mp_ListQueue (NULL, mode, &queinf);

        return(0);

}
```

16.3.17 Queue List Information Acquisition API/EE [Windows]

This section explains the queue list information acquisition API/EE (Mp_ListQueue_Ex).

Synopsis

```
#include "f3cuapi.h"
long Mp_ListQueue_Ex (char *rhost,
          long mode,
          struct Mp_QUEUEINF_1 *queueinf[ ],
          int system_num)
```

Description

This API obtains the list of queue information in Systemwalker Operation Manager EE.

Parameters

rhost

Specify NULL here.

mode

Specify one of the following process modes. After GETQST, you must always specify FREEQST.

GETQST:

Obtains the queue list information.

FREEQST:

Frees the area that was used for the queue list information obtained by GETQST.

queueinf[ ]

GETQST:

Specify the address to which the address of the queue information structure (Mp_QUEUEINF_1).

FREEQST:

Specify the area address to which the address of the job information structure's array was notified in GETQST. Between the issuing GETQST and FREEQST, you should not change the contents of the area.

system_num

Specifies operating target subsystem number using a range of 0 to 9.

Return Values

Return Value	Meaning	Action
1 or greater	This indicates that the queue information list was obtained normally and the number of queues is returned.	-
-1	A system error occurred.	Collect "Job Execution Control" information with the maintenance information collection tool and contact a Fujitsu SE.
-1000 or less	Processing was aborted because an error was detected in Job Execution Control. Error messages can be obtained using the error message acquisition API.	Use the error message acquisition API to obtain the error message and take whatever action is required.

Caution

When the argument is pointing to an invalid address, this function ends abnormally, and no operations will be defined.

Format of the Queue Information Structure (Mp_QUEUEINF_1)

The following is the format of the queue information structure.

This structure is defined in the INCLUDE file "f3cuapi.h".

```
typedef struct _Mp_QUEUEINF_1 {
  char     queuename[];                /*queue name                              */
  long     queuetype;                  /*queue type                              */
  char     status;                     /*queue status                            */
  long     maxjob;                     /*maximum possible job submission quantity */
```

```
    long      maxexec;                      /*maximum possible job execution quantity   */
    long      elapsed;                      /*maximum time lapse limit (seconds)        */
    long      dfltprty;                     /*default priority order                    */
    long      dfltdprty;                    /*default priority                          */
    long      job_wait;                     /*jobs waiting to be executed               */
    long      job_hold;                     /*jobs on hold                              */
    long      job_run;                      /*jobs in execution                         */
    char      hostgrp[];                    /*distributed host group name               */
} Mp_QUEUEINF_1
```

Following provides the explanations about each item.

queuename

> The queue name is set.

queuetype

> The queue type is set. The following types can be set for a queue:

Value	Type
1	Local queue
2	Distributed execution queue

status

> The queue status is set. The following statuses can be set for a queue:

Value	Status
1	Operating
2	Paused

maxjob

> The maximum number of possible jobs to be submitted is set.

maxexec

> The maximum number of possible jobs to be executed is set.

elapsed

> The maximum time lapse limit is set in seconds.

dfltprty

> The default priority order is set.

dfltdprty

> The default priority is set.

job_wait

> The number of jobs waiting to be executed at the moment when the API was called up is set.

job_hold

> The number of jobs held at the moment when the API was called is set.

job_run

> The number of jobs being executed at the moment when the API was called is set.

hostgrp

> The distributed host group name is set. NULL is set when it is a local queue.

Program Example

Below is an example program of the queue list information acquisition API/EE:

```c
#include <windows.h>
#include <stdio.h>
#include "f3cuapi.h"

int main (int argc, char **argv)
{

        Mp_QUEUEINF_1 *queinf = 0;
        Mp_QUEUEINF_1 *qinf = 0;
        int system_num;

        long mode;
        long idx;
        long rtn = 0;
        char errmsg[256+1];

        /*
         * initialization working area
         */
        memset(errmsg, 0x00, sizeof(errmsg));

        mode = GETQST;
        /*
         *  call Mp_ListQueue_Ex() API
         */
        system_num = 0;
        rtn = Mp_ListQueue_Ex (NULL, mode, &queinf, system_num);

        /*
         * check return code of Mp_ListQueue_Ex() API
         */
        if (rtn == -1){
                printf ("GetLastError() is %d.\n", GetLastError());
                return (1);
        } else if (rtn <= -1000) {
                printf ("Error code is %d.\n", rtn);
                Mp_GetMJESerror (errmsg);
                printf ("errmsg = %s\n", errmsg);
                return (1);
        }

        qinf = queinf;

        printf("queue name      wait  hold  run   status\n");
        for (idx = 0; idx < rtn; idx++) {

                printf("%15.15s %5.5d %5.5d %5.5d ",
                        qinf->queuename, qinf->job_wait, qinf->job_hold,
                        qinf->job_run);
                switch(qinf->status){
                case 1  :
                        printf("active  ");
                        break;
                case 0  :
                default :
                        printf("inactive");
                        break;
                }
                printf("\n");
```

```
            qinf++;

    }

    mode = FREEQST;

    /*
     *  call Mp_ListQueue_Ex() API
     */
    Mp_ListQueue_Ex (NULL, mode, &queinf, system_num);

    return(0);

}
```

16.3.18 Error Message Acquisition API

This section explains the error message acquisition API (Mp_GetMJESerror).

Synopsis

[Windows]

```
#include "f3cuapi.h"
long Mp_GetMJESerror (char *errormsg)
```

[UNIX]

```
#include "mjesapi.h"
int Mp_GetMJESerror (char *errormsg)
```

Description

When using the APIs provided by Job Execution Control, this API obtains the error messages of errors (up to 1000 errors) that occurred recently.

You can obtain the messages of errors that occurred within the same process. You cannot obtain them from other processes.

Parameter

errormsg

Specify the address of the area to which an error message is returned.

You must reserve 256 bytes for this area.

Return Values

Return Value	Meaning	Action
0	Normal termination.	-

Caution

When the argument is pointing to an invalid address, this function ends abnormally, and no operations will be defined.

16.4 Action Control API [Windows]

This section explains the APIs that control actions.

The APIs that control actions can be categorized into the action execution API, action stop API, and action status notification API.

The action control APIs' operating environment, cautions and required files are explained below.

Operating Environment

These APIs can only operate on machines where a Systemwalker Operation Manager server is installed. They will not operate on machines where only the Systemwalker Operation Manager client is installed.

Cautions

- In a compiling environment, use Visual C++ (version 2005) as the compiler and "multithread (DLL) " for the runtime library.

- Use the same version/levels of the LIB file and INCLUDE file.

- In the environment where Systemwalker Centric Manager 13.2.0 or later and Systemwalker Operation Manager 13.2.0 or later coexist or, only Systemwalker Centric Manager 13.2.0 or later is uninstalled from that environment, privileges required for executing this API vary. See the *Systemwalker Centric Manager API Reference Guide* for details.

Required Files

The following files are required in order to use the action control APIs:

[Windows x86]

LIB:

f3crhxac.lib

INCLUDE:

f3crhxac.h

[Windows x64]

LIB:

f3crhxac_x64.lib

INCLUDE:

f3crhxac_x64.h

Action control APIs are stored in a LIB file. The constants and structures used in each AP should be declared in an INCLUDE file.

These files are stored under "MpWalker\lib" and "MpWalker\include" where Systemwalker Operation Manager was installed.

16.4.1 Action Execution API

This section explains the action execution APIs.

A list of the Action Execution APIs is shown below:

- Sound Notification API (Mp_PlaySound2)

- Stop Sound API (Mp_StopSound)

- Pop-up Message Transmission (Specified Users) API (MP_PopupUser2)

- Pop-up Message Transmission (Domains) API (MP_PopupDomain2)

- Pop-up Message Transmission (All Users) API (MP_PopupSession2)

- MS-Mail Transmission API (Mp_SendMSMail2)

- E-Mail Transmission API (Mp_SendEMail)

- Short Mail Message Transmission API (Mp_CallPager2)

16.4.1.1 Sound Notification API

This section explains the sound notification API (Mp_PlaySound2).

Synopsis

```
long Mp_PlaySound2 (unsigned char *AppName,
           int kind,
           unsigned char *text_or_filename,
           int repeat,
           MpSoundParam *SoundParam,
           char *ExeHost)
```

Description

The following types of sound are notified:

- Message read out

- WAV file playback

- Beep.

Parameters

AppName

Specify the address of the application that made the call. The application name must be 64 bytes or less in length.

kind

Specify the type of sound. Specify one of the followings in this parameter.

SOUND_VOICE:

Reads out the message text.

SOUND_WAVE:

Plays a WAV file.

SOUND_BEEP:

Emits a beep.

text_or_filename

Specify one of the following depending on the value of parameter kind.

When specifying SOUND_VOICE:

Specify the address of the message text that is to be read out.

When specifying SOUND_WAVE

Specify the full path address of the file to be played to a value with up to 260 bytes. The file you specify must exist on the host specified by ExeHost.

When specifying SOUND_BEEP

This is invalid.

repeat

Specify the number of times the sound should be emitted. If 0 (zero) has been specified, repeats until stop processing has been completed.

SoundParam

Specify the address of the sound option information structure (MpSoundParam). If SOUND_VOICE has been specified, specify this when changing the system settings. This is invalid if SOUND_VOICE has not been specified. Specify NULL if this does not have to be set. If 0 (zero) has been specified for all items, the system settings will be used.

ExeHost

Specify the address of the name of host (client that chose execution of action) that executes actions. If NULL is specified, the action will be executed on the host you specified in the **Action Environment Setup** window.

Return Values

If the request was successfully accepted, an integer 0 or above will be returned. Any value less than 0 will be an error.

The following table provides details about errors that may be returned.

Return Value	Meaning	Action
MPACT_PARAMERR	There is a parameter error.	Check for incorrect parameter settings.
MPACT_NOMEMORY	There is not enough memory.	Either increase the size of the page file or add more physical memory.
MPACT_NOTSOUND	The environment for playing sound has not been installed.	Check that a product equipped with a voice synthesizing engine compatible with Microsoft Speech API(SAPI)5.x has been installed.
MPACT_NOTMOVE	The Action Control Server has not been started.	Check that Systemwalker MpAosfB is running.
MPACT_COMFAIL	An error has occurred in communication with the Action Control Server.	Check that Systemwalker MpAosfB and Systemwalker MpAosfX are running.
MPACT_SYSERR	An error has occurred on the Action Control Server.	Refer to the message output to the event log or syslog, eliminate the cause of the error and then restart Systemwalker. If the problem persists after the cause of the error has been eliminated and Systemwalker has been restarted, use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_NOTDLL	DLL for the action request cannot be loaded.	Check that the file f3crhxac.dl (see Note), which is needed for action management, exists in the following folder: Systemwalker installation directory\mpwalker\bin If the file is not located in the above folder, either reinstall Systemwalker or use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_EXCEPT	The address of an invalid area has been specified.	An internal logic error has occurred. Collect data with the maintenance information collection tool and contact a Fujitsu SE.

Note: f3crhxac.dll" is replaced by "f3crhxac_64.dll" in 64-bit platforms.

Format of the Sound Option Information Structure (MpSoundParam)

```
typedef struct MpSoundParam_tag {
    DWORD dwVoiceType ;     // voice type
```

```
    DWORD dwIntonation ;    // voice intonation level
    DWORD dwPitch ;         // voice pitch
    DWORD dwSpeed ;         // voice speed
    DWORD dwSpeedRate ;     // fluctuation of voice speed
    DWORD dwTone ;          // emphasis in voice tone
    DWORD dwVolume ;        // voice volume
} MpSoundParam ;
```

dwVoiceType

Specify the voice type.

VOICE_MALE:

This plays a male voice.

VOICE_FEMALE:

This plays a female voice.

dwIntonation

Specify an intonation level between 1 and 4.

dwPitch

Specify the pitch of the voice between 1 and 5.

dwSpeed

Specify the speed of the voice between 1 and 10.

dwSpeedRate

Specify the fluctuation of the speed of the voice between 1 and 10.

dwTone

Specify whether there is emphasis in high voice tone.

No:

1

Yes:

2

DWORD dwVolume

Specify the volume of the voice between 1 and 10.

Program Example

Below is an example program of the sound notification API:

```
#include "f3crhxac.h"

unsigned char *AppName = "Sound notification API" ;  // requested source application name
int kind              = SOUND_VOICE ;                // read out text
unsigned char *text   = "an error has occurred. please contact your operator." ;
int repeat            = 0 ;                          // repeat until stop process is executed.
MpSoundParam  *SndPrm = NULL ;                       // default value is used.
long          lrc ;

/* request for sound notification */
```

```
lrc = Mp_PlaySound2(AppName,kind,text,repeat,NULL,NULL) ;
if ( lrc < 0 ) {
    /* error processing */
}

/* normal completion */
```

16.4.1.2 Stop Sound API

This section explains the stop sound API (Mp_StopSound).

Synopsis

```
long Mp_StopSound (unsigned char *AppName,
            long handle)
```

Description

This API stops the sound notification.

Parameters

AppName

Specify the address of the caller application specified in the sound stop API (Mp_PlaySound).

handle

Specify the value returned by the Mp_PlaySound function. If 0 (zero) has been specified, The sound that is currently being played will be stopped.

Return Values

If the request was successfully accepted, 0 will be returned. Any value less than 0 is an error. The following table provides details about errors that may be returned.

Return Value	Meaning	Action
MPACT_PARAMER R	There is a parameter error.	Check for incorrect parameter settings.
MPACT_NOMEMO RY	There is not enough memory.	Either increase the size of the page file or add more physical memory.
MPACT_NOTACTI ON	Specified action does not exist (already finished).	-
MPACT_NOTSTOP	Cannot stop the specified action.	-
MPACT_NOTMOV E	The Action Control Server has not been started.	Check that Systemwalker MpAosfB is running.
MPACT_COMFAIL	An error has occurred in communication with the Action Control Server.	Check that Systemwalker MpAosfB and Systemwalker MpAosfX are running.
MPACT_SYSERR	An error has occurred on the Action Control Server.	Refer to the message output to the event log or syslog, eliminate the cause of the error and then restart Systemwalker. If the problem persists after the cause of the error has been eliminated and Systemwalker has been restarted, use the maintenance information collection tool to

Return Value	Meaning	Action
		collect event monitoring data and contact a Fujitsu SE.
MPACT_NOTDLL	DLL for the action request cannot be loaded.	Check that the file f3crhxac.dl (see Note), which is needed for action management, exists in the following folder: Systemwalker installation directory\mpwalker\bin If the file is not located in the above folder, either reinstall Systemwalker or use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_EXCEPT	The address of an invalid area has been specified.	An internal logic error has occurred. Collect data with the maintenance information collection tool and contact a Fujitsu SE.

Note: f3crhxac.dll" is replaced by "f3crhxac_64.dll" in 64-bit platforms.

Program Example

Below is an example program of the stop sound API:

```
#include "f3crhxac.h"

unsigned char *AppName = "Sound notification API" ;  // application name specified in the
Mp_PlaySound()
long handle           = 1 ;                          // return value of Mp_PlaySound()
long lrc ;

lrc = Mp_StopSound(AppName,handle) ;
if ( lrc != 0 ) {
    /* error processing */
}

/* normal completion */
```

16.4.1.3 Pop-up Message Transmission (Specified Users) API

This section explains the pop-up message transmission (specified users) API (Mp_PopupUser2).

Synopsis

```
long Mp_PopupUser2 (unsigned char *AppName,
          MpAddress *AddressList,
          int nAddressList,
          unsigned char *MsgText,
          char *ExeHost)
```

Description

This API displays messages on specified computers or on computers to which specified users have logged on.

Parameters

AppName

Specify the address of the application that made the call. The application name must be 64 bytes or less in length.

AddressList

Specify the address of the destination information structure (MpAddress). Specify the name of the user or computer that is sending the pop-up message. If there are multiple names, specify them in an array. If NULL is specified, the pop-up message will be sent to all users logged on to the domain.

nAddressList

Specify the number of arrays that were specified in AddressList.

MsgText

Specify the address of the message that will appear in the pop-up window. The message has to be within 128 bytes in length.

ExeHost

Specify the address of the name of host (client that chose execution of action) that executes actions. If NULL is specified, the action will be executed on the host you specified in the **Action Environment Setup** window.

Return Values

If the request was successfully accepted, 0 or a higher integer will be returned. Any value less than 0 is an error. The following table provides details about errors that may be returned.

Return Value	Meaning	Action
MPACT_PARAMERR	There is a parameter error.	Check for incorrect parameter settings.
MPACT_NOMEMORY	There is not enough memory.	Either increase the size of the page file or add more physical memory.
MPACT_NOTADDRESS	There is no valid user name or computer name.	There is an error in the address specified in the address information structure (MpAddress). Correct the address specification.
MPACT_NOTMOVE	The Action Control Server has not been started.	Check that Systemwalker MpAosfB is running.
MPACT_COMFAIL	An error has occurred in communication with the Action Control Server.	Check that Systemwalker MpAosfB and Systemwalker MpAosfX are running.
MPACT_SYSERR	An error has occurred on the Action Control Server.	Refer to the message output to the event log or syslog, eliminate the cause of the error and then restart Systemwalker. If the problem persists after the cause of the error has been eliminated and Systemwalker has been restarted, use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_NOTDLL	DLL for the action request cannot be loaded.	Check that the file f3crhxac.dl (see Note), which is needed for action management, exists in the following folder: Systemwalker installation directory\mpwalker \bin If the file is not located in the above folder, either reinstall Systemwalker or use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_EXCEPT	The address of an invalid area has been specified.	An internal logic error has occurred. Collect data with the maintenance information collection tool and contact a Fujitsu SE.

Note: f3crhxac.dll" is replaced by "f3crhxac_64.dll" in 64-bit platforms.

Caution

- Pop-up messages can only be used on Microsoft networks. For the action execution host, you cannot specify a computer on which Windows(R) 98, or Windows(R) Me has been installed.

Format of the Destination Information Structure (MpAddress)

```
typedef struct MpAddress_tag {
    int type ;
    unsigned char *address ;
    char *addrid ;
} MpAddress ;
```

type

Specify the type of address that was specified in address.

TYPE_ADDRESS:

This is the user name or computer name.

TYPE_NAME:

This is the destination name.

address

Specify the user name/computer name or the address of the destination.

addrid

This parameter is enabled when Systemwalker Centric Manager is installed and "TYPE_NAME" is specified as the type parameter. Refer to the *Systemwalker Centric Manager Reference Guide* for details.

Program Example

Below is an example program of the pop-up message transmission (specified users) API:

```
#include "f3crhxac.h"

unsigned char *AppName = "pop-up API" ;        // requested source application name
MpAddress addr[2] ;                            // destination address storage area
int naddr ;
unsigned char *text = "this will stop the system. " ;   // message text to be sent
long lrc ;

addr[0].type = TYPE_ADDRESS ;        // send a message to users "operator" and "user01"
addr[0].address = "operator" ;
addr[1].type = TYPE_ADDRESS ;
addr[1].address = "user01" ;
naddr = 2 ;

lrc = Mp_PopupUser2(AppName,addr,naddr,text,NULL) ;
if ( lrc < 0 ) {
    /* error processing */
}

/* normal completion */
```

16.4.1.4 Pop-up Message Transmission (Domains) API

This section explains the pop-up message transmission (domain) API (Mp_PopupDomain2).

Synopsis

```
long Mp_PopupDomain2 (unsigned char *AppName,
            char *domain,
            unsigned char *MsgText,
            char *Exehost)
```

Description

This API displays pop-up messages on all computers that are connected to a specified domain.

Parameters

AppName

Specify the address of the application that made the call. The application name must be 64 bytes or less length.

domain

Specify the address of the domain that will display the message. The domain name has to be within 15 bytes in length.

MsgText

Specify the address of the message that will appear in the pop-up window. The message has to be within 128 bytes in length.

ExeHost

Specify the address of the name of host (client that chose execution of action) that executes actions. If NULL is specified, the action will be executed on the host you specified in the **Action Environment Setup** window.

Return Values

If the request was successfully accepted, 0 or a higher integer will be returned. Any value less than 0 is an error. The following table provides details about errors that may be returned.

Return Value	Meaning	Action
MPACT_PARAMERR	There is a parameter error.	Check for incorrect parameter settings.
MPACT_NOMEMORY	There is not enough memory.	Either increase the size of the page file or add more physical memory.
MPACT_NOTMOVE	The Action Control Server has not been started.	Check that Systemwalker MpAosfB is running.
MPACT_COMFAIL	An error has occurred in communication with the Action Control Server.	Check that Systemwalker MpAosfB and Systemwalker MpAosfX are running.
MPACT_SYSERR	An error has occurred on the Action Control Server.	Refer to the message output to the event log or syslog, eliminate the cause of the error and then restart Systemwalker. If the problem persists after the cause of the error has been eliminated and Systemwalker has been restarted, use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.

Return Value	Meaning	Action
MPACT_NOTDLL	DLL for the action request cannot be loaded.	Check that the file f3crhxac.dl (see Note), which is needed for action management, exists in the following folder: Systemwalker installation directory\mpwalker \bin If the file is not located in the above folder, either reinstall Systemwalker or use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_EXCEPT	The address of an invalid area has been specified.	An internal logic error has occurred. Collect data with the maintenance information collection tool and contact a Fujitsu SE.

Note: f3crhxac.dll" is replaced by "f3crhxac_64.dll" in 64-bit platforms.

Caution

- Pop-up messages can only be used on Microsoft networks. For the action execution host, you cannot specify a computer on which Windows(R) 98, or Windows(R) Me has been installed.

Program Example

Below is an example program of the pop-up message transmission (domains) API:

```
#include "f3crhxac.h"

unsigned char *AppName = "pop-up API" ;  // requested source application name
char *domain = "Domain" ;                 // destination domain name
unsigned char *text = "this will stop the system. " ;   // message text to be sent
long lrc ;

lrc = Mp_PopupDomain2(AppName,domain,text,NULL) ;
if ( lrc < 0 ) {
    /* error processing */
}

/* normal completion */
```

16.4.1.5 Pop-up Message Transmission (All Users) API

This section explains the pop-up message transmission (all users) API (Mp_PopupSession2).

Synopsis

```
long Mp_PopupSession2 (unsigned char *AppName,
          unsigned char *MsgText,
          char *ExeHost)
```

Description

This API sends pop-up messages to all sessions connected to the server.

Parameters

AppName

Specify the address of the application that made the call. The application name must be 64 bytes or less in length.

MsgText

Specify the address of the message that will appear in the pop-up window. The message must be 128 bytes or less in length.

ExeHost

Specify the address of the name of host (client that chose execution of action) that executes actions. If NULL is specified, the action will be executed on the host you specified in the **Action Environment Setup** window.

Return Values

If the request was successfully accepted, 0 or a higher integer will be returned. Any value less than 0 is an error. The following table provides details about errors that may be returned.

Return Value	Meaning	Action
MPACT_PARAMERR	There is a parameter error.	Check for incorrect parameter settings.
MPACT_NOMEMORY	There is not enough memory.	Either increase the size of the page file or add more physical memory.
MPACT_NOTMOVE	The Action Control Server has not been started.	Check that Systemwalker MpAosfB is running.
MPACT_COMFAIL	An error has occurred in communication with the Action Control Server.	Check that Systemwalker MpAosfB and Systemwalker MpAosfX are running.
MPACT_SYSERR	An error has occurred on the Action Control Server.	Refer to the message output to the event log or syslog, eliminate the cause of the error and then restart Systemwalker. If the problem persists after the cause of the error has been eliminated and Systemwalker has been restarted, use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_NOTDLL	DLL for the action request cannot be loaded.	Check that the file f3crhxac.dl (see Note), which is needed for action management, exists in the following folder: Systemwalker installation directory\mpwalker\bin. If the file is not located in the above folder, either reinstall Systemwalker or use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_EXCEPT	The address of an invalid area has been specified.	An internal logic error has occurred. Collect data with the maintenance information collection tool and contact a Fujitsu SE.

Note: f3crhxac.dll" is replaced by "f3crhxac_64.dll" in 64-bit platforms.

Caution

- Pop-up messages can only be used on Microsoft networks. For the action execution host, you cannot specify a computer on which Windows(R) 98, or Windows(R) Me has been installed.

Program Example

Below is an example program of the pop-up message transmission (all users) API:

```
#include "f3crhxac.h"

unsigned char *AppName = "pop-up API" ;            // requested source application name
unsigned char *text = "this will stop the system. " ;    // message text to be sent
long lrc ;

lrc = Mp_PopupSession2(AppName,text,NULL) ;
if ( lrc < 0 ) {
    /* error processing */
}

/* normal completion */
```

16.4.1.6 MS-Mail Transmission API

This section explains the MS-Mail transmission API (Mp_SendMSMail2).

Synopsis

```
long Mp_SendMSMail2 (unsigned char *AppName,
          MpAddress *AddressList,
          int nAddressList,
          MpDataList *MailData,
          int nMailData,
          unsigned char *MailTitle,
          MpMMParam *MmailParam,
          char *ExeHost)
```

Description

This API requests transmission of MS-Mail to the Action Control Server.

Parameters

AppName

Specify the address of the application that made the call. The application name must be 64 bytes or less in length.

AddressList

Specify the address of the destination information structure (MpAddress). Specify the name of the user that is sending the mail. If there are multiple names, specify them in an array.

nAddressList

Specify the number of arrays specified in AddressList.

MailData

Specify the address of the mail transmission data information structure (MpDataList). If you are sending multiple data sets, specify them in an array.

nMailData

Specify the number of arrays specified in MailData.

MailTitle

Specify the title of the mail.

MMailParam

Specify the address of the mail transmission control information structure (MpMMParam). Specify the information needed to send the mail. If NULL is specified, use each item's default value.

ExeHost

Specify the address of the name of host (client that chose execution of action) that executes actions. If NULL is specified, the action will be executed on the host you specified in the **Action Environment Setup** window.

Return Values

If the request was successfully accepted, 0 or a higher integer will be returned. Any value less than 0 is an error. The following table provides details about errors that may be returned.

Return Value	Meaning	Action
MPACT_PARAMERR	There is a parameter error.	Check for incorrect parameter settings.
MPACT_NOMEMORY	There is not enough memory.	Either increase the size of the page file or add more physical memory.
MPACT_NOTADDRESS	There is no valid mail address number.	There is an error in the address specified in the address information structure (MpAddress). Correct the address specification.
MPACT_NOTMOVE	The Action Control Server has not been started.	Check that Systemwalker MpAosfB is running.
MPACT_COMFAIL	An error has occurred in communication with the Action Control Server.	Check that Systemwalker MpAosfB and Systemwalker MpAosfX are running.
MPACT_SYSERR	An error has occurred on the Action Control Server.	Refer to the message output to the event log or syslog, eliminate the cause of the error and then restart Systemwalker. If the problem persists after the cause of the error has been eliminated and Systemwalker has been restarted, use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_NOTDLL	DLL for the action request cannot be loaded.	Check that the file f3crhxac.dl (see Note), which is needed for action management, exists in the following folder: Systemwalker installation directory\mpwalker\bin If the file is not located in the above folder, either reinstall Systemwalker or use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_EXCEPT	The address of an invalid area has been specified.	An internal logic error has occurred. Collect data with the maintenance information collection tool and contact a Fujitsu SE.

Note: f3crhxac.dll" is replaced by "f3crhxac_64.dll" in 64-bit platforms.

Remarks

- You can only specify one piece of memory data as mail data. Memory data should also be specified at the front of an array. If there is no memory data, all mails should be sent as an attached file.

- Memory data must be ended with "\0." Insert a line break with the CR(0x0d)+LF(0x0a) code.

Caution

Since machines with Microsoft Outlook Express installed (e.g. Windows 2000) are unable to receive MS mails, use e-mails for reception.

Format of the Destination Information Structure (MpAddress)

```
typedef struct MpAddress_tag {
    int type ;
    unsigned char *address ;
    char *addrid ;
} MpAddress ;
```

type

Specify the type of address that was specified in address.

TYPE_ADDRESS:

This is the mail address of the MS-Mail destination.

TYPE_NAME:

This is the destination name.

address

Specify the mail address or destination address.

addrid

This parameter is enabled when Systemwalker Centric Manager is installed and "TYPE_NAME" is specified as the type parameter. Refer to the *Systemwalker Centric Manager Reference Guide* for details.

Format of the Mail Transmission Data Information Structure (MpDataList)

```
typedef struct MpDataList_tag {
    int type ;
    unsigned char *data ;
} MpDataList ;
```

type

Specify the type of data that will be specified in data.

TYPE_LMEMORY:

This is memory data.

TYPE_FILENAME:

This is a file name.

TYPE_TEMPFILE:

This is a file name. It will be deleted by the action control function after the mail has been sent successfully.

data

If TYPE_LMEMORY has been specified, specify the address of the memory data you specified using a maximum of 1,028 bytes.

If TYPE_FILENAME or TYPE_TEMPFILE has been specified, set the address of the file specified within MAX_PATH.

```
typedef struct MpMMParam_tag {
    char *Profile ;
    char *Password ;
} MpMMParam ;
```

Profile

Specify the address of the user that is logging in. If NULL is specified, the default value is used.

Password

Specify the address of the password of the user who logs in. If NULL is specified, the default value is used.

Program Example

Below is an example program of the MS-Mail Transmission API

```
#include "f3crhxac.h"

unsigned char *AppName = "MS-Mail transmission API" ; // requested source application name
MpAddress addr[2] ;                                   // destination address storage area
int naddr ;
MpDataList data[2] ;
int ndata ;
unsigned char *Title  = "system abnormality notification" ;  // mail title
MpMMParam  MailPrm = NULL ;                                   // default value is used
long lrc ;

addr[0].type = TYPE_ADDRESS ;        // send a message to users "operator" and "user01"
addr[0].address = "operator" ;
addr[1].type = TYPE_ADDRESS ;
addr[1].address = "user01" ;
naddr = 2 ;

data[0].type = TYPE_LMEMORY ;        // setting for transmitting data
data[0].data = "abnormality has occurred. \r\n refer to the attached file for more detail.
\r\n" ;
data[1].type = TYPE_FILENAME ;
data[1].data = "D:\\TEMP\\ERR001.LOG" ;
ndata = 2 ;

lrc = Mp_SendMSMail2(AppName,addr,naddr,data,ndata,Title,MailPrm,NULL) ;
if ( lrc < 0 ) {
    /* error processing */
}

/* normal completion */
```

16.4.1.7 E-Mail Transmission API

This section explains the e-mail transmission API (Mp_SendEMail).

Synopsis

```
long Mp_SendEMail (unsigned char *AppName,
        MpAddress *AddressList,
        int nAddressList,
        MpDataList *MailData,
        int nMailData,
```

```
              char *FromAddress,
              unsigned char *MailTitle,
              char *SMTPServer)
```

Description

This API requests transmission of e-mail to the Action Control Server.

Parameters

AppName

Specify the address of the application that made the call. The application name must be 64 bytes or less in length.

AddressList

Specify the address of the destination information structure (MpAddress). If there are multiple names, specify them in an array.

See "16.4.1.6 MS-Mail Transmission API" for more information on the destination information structure.

nAddressList

Specify the number of addresses (arrays) specified in AddressList.

MailData

Specify the address of the mail transmission data information structure (MpDataList). If you are sending multiple data sets, specify them in an array.

See "16.4.1.6 MS-Mail Transmission API" for more information on the mail transmission data information structure.

nMailData

Specify the number of arrays specified in MailData.

FromAddress

Specify the mail address of the sender. If NULL is specified, the default value is used.

MailTitle

Specify the title of the mail.

SMTPServer

Specify the name of the SMTP server. If NULL is specified, the default value is used.

Return Values

If the request was successfully accepted, 0 or a higher integer will be returned. Any value less than 0 is an error. The following table provides details about errors that may be returned.

Return Value	Meaning	Action
MPACT_PARAMERR	There is a parameter error.	Check for incorrect parameter settings.
MPACT_NOMEMORY	There is not enough memory.	Either increase the size of the page file or add more physical memory.
MPACT_NOTADDRESS	There is no valid mail address number.	There is an error in the address specified in the address information structure (MpAddress). Correct the address specification.

Return Value	Meaning	Action
MPACT_NOTMOVE	The Action Control Server has not been started.	Check that Systemwalker MpAosfB is running.
MPACT_COMFAIL	An error has occurred in communication with the Action Control Server.	Check that Systemwalker MpAosfB and Systemwalker MpAosfX are running.
MPACT_SYSERR	An error has occurred on the Action Control Server.	Refer to the message output to the event log or syslog, eliminate the cause of the error and then restart Systemwalker. If the problem persists after the cause of the error has been eliminated and Systemwalker has been restarted, use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_NOTDLL	DLL for the action request cannot be loaded.	Check that the file f3crhxac.dl (see Note), which is needed for action management, exists in the following folder: Systemwalker installation directory\mpwalker\bin If the file is not located in the above folder, either reinstall Systemwalker or use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_EXCEPT	The address of an invalid area has been specified.	An internal logic error has occurred. Collect data with the maintenance information collection tool and contact a Fujitsu SE.

Note: f3crhxac.dll" is replaced by "f3crhxac_64.dll" in 64-bit platforms.

Remarks

- The data specified at the beginning of MailData will be sent as the body (text) of the mail. If there is no body (text) in the mail, specify NULL for the first "data" member of the array.

- You can only specify one TYPE_LMEMORY in MailData. TYPE_LMEMORY should also be specified at the front of an array.

- Memory data must be ended with "\0." Insert a line with the CR(0x0d)+LF(0x0a) code.

Program Example

Below is an example program of the e-mail transmission API:

```
#include "f3crhxac.h"

unsigned char *AppName = "E-Mail transmission API" ;  // requested source application name
MpAddress addr[2] ;                                   // destination address storage area
int naddr ;
MpDataList data[2] ;
int ndata ;
char *FromAddr        = "MpWalker@fujitsu.co.jp" ;     // source address
unsigned char *Title  = "System Report" ;             // mail title
char *SMTPSrv         = NULL ;                         // default value is used
long lrc ;
```

```
addr[0].type = TYPE_ADDRESS ;    // send a message to user's "operator" and "user()1"
addr[0].address = "operator@fujitsu.co.jp" ;
addr[1].type = TYPE_ADDRESS ;
addr[1].address = "user01@fujitsu.co.jp" ;
naddr = 2 ;

data[0].type = TYPE_LMEMORY ;                   // setting for transmitting data
data[0].data = "abnormality has occurred.  \r\n refer to the attached file for more detail.
\r\n" ;
data[1].type = TYPE_FILENAME ;
data[1].data = "D:\\TEMP\\ERR001.LOG" ;
ndata = 2 ;

lrc = Mp_SendEMail(AppName,addr,naddr,data,ndata,FromAddr,Title,SMTPSrv) ;
if ( lrc < 0 ) {
  /* error processing */
}

/* normal completion */
```

16.4.1.8 Short Mail Message Transmission API

This section explains the short mail message transmission API (Mp_CallPager2).

Synopsis

```
long Mp_CallPager2 (unsigned char *AppName,
        MpPagerList *AddressList,
        int nAddressList,
        char *MsgText,
        int msgtype,
        char *ExeHost)
```

Description

This API sends a short mail message.

Parameters

AppName

Specify the address of the application that made the call. The application name must be 64 bytes or less in length.

AddressList

Specify the address of the short mail transmission information structure (MpPagerList). Specify the telephone number of the short mail. If there are multiple telephone numbers, specify them in an array.

nAddressList

Specify the number of addresses (arrays) specified in AddressList.

MsgText

Specify the address of the short mail message to be sent. Specify the transmission code (including code of fixed type messages) of the message.

Or, specify the characters supported by the short mail company.

MsgType

Specify the format of the message that was specified in MsgText.

PAGER_CODE:

This is the number format.

PAGER_TEXT:

This is the message text.

exeHost

Specify the address of the name of host (client that chose execution of action) that executes actions. If NULL is specified, the action will be executed on the host you specified in the **Action Environment Setup** window.

Return Values

If the request was successfully accepted, 0 or a higher integer will be returned. Any value less than 0 is an error. The following table provides details about errors that may be returned.

Return Value	Meaning	Action
MPACT_PARAMERR	There is a parameter error.	Check for incorrect parameter settings.
MPACT_NOMEMORY	There is not enough memory.	Either increase the size of the page file or add more physical memory.
MPACT_NOTADDRESS	There is no valid mail address number.	There is an error in the address specified in the address information structure (MpAddress). Correct the address specification.
MPACT_NOTMODEM	A modem has not been installed.	Check that the modem is correctly installed and connected. Refer to the operating system help files for information on checking the status of a modem.
MPACT_NOTMOVE	The Action Control Server has not been started.	Check that Systemwalker MpAosfB is running.
MPACT_COMFAIL	An error has occurred in communication with the Action Control Server.	Check that Systemwalker MpAosfB and Systemwalker MpAosfX are running.
MPACT_SYSERR	An error has occurred on the Action Control Server.	Refer to the message output to the event log or syslog, eliminate the cause of the error and then restart Systemwalker. If the problem persists after the cause of the error has been eliminated and Systemwalker has been restarted, use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_NOTDLL	DLL for the action request cannot be loaded.	Check that the file f3crhxac.dl (see Note), which is needed for action management, exists in the following folder: Systemwalker installation directory\mpwalker \bin If the file is not located in the above folder, either reinstall Systemwalker or use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.

Return Value	Meaning	Action
MPACT_EXCEPT	The address of an invalid area has been specified.	An internal logic error has occurred. Collect data with the maintenance information collection tool and contact a Fujitsu SE.

Note: f3crhxac.dll" is replaced by "f3crhxac_64.dll" in 64-bit platforms.

Remarks

The maximum length of a message varies according to the type of short mail and the type of modem. Refer to the NTT DoCoMo mobile phone and modem instruction manuals for more information. If the message is longer than the maximum permissible length, the message may not be sent.

Format of the Short Mail Transmission Information Structure (MpPagerList)

```
typedef struct MpPagerList_tag {
    int type ;
    unsigned char *address ;
    char *addrid ;

    char *vender ;
} MpPagerList ;
```

type

Specify the type of address that was specified in address.

TYPE_ADDRESS:

This is the telephone number of the short mail destination.

TYPE_NAME:

This is the destination name.

address

Specify the telephone number or address of the destination.

In the case of a telephone number, the telephone number of the short mail destination should be specified in the following format:

telephone number of short mail center + "T" + telephone number of notification destination

Telephone number of short mail center

Set the telephone number of the NTT DoCoMo Short Mail Center.

Reference: As of September 2007, the telephone number of the NTT DoCoMo Short Mail Center was 0903101655. Contact the contract company for more information.

T

A delimiting character. Set "T" or "t".

Telephone number of notification destination

Set the telephone number of the notification destination.

Example: To send a short mail message to a notification destination having the telephone number 09012345678:

0903101655T09012345678

addrid

> This parameter is enabled when Systemwalker Centric Manager is installed and "TYPE_NAME" is specified as the type parameter. Refer to the *Systemwalker Centric Manager Reference Guide* for details.

vender

> Set the short mail type. The short mail type supported by Systemwalker is provided below. To specify a different type, it will be necessary to specify the short mail company name registered using the **Action Environment Setup** window and create a message conversion exit (an exit used to convert unsupported short mail messages).
>
> See "17.9 Unsupported Short Mail Message Conversion Exit [Windows]" for more about the exit for converting unsupported short mail messages.

> **MPPAGER_DOCOMO_SHORTMAIL:**
>
> > NTT DoCoMo Short Mail

Program Example

> Below is an example program of the short mail message transmission API:

```c
#include "f3crhxac.h"

unsigned char *AppName = "short mail API" ;    // requested source application name
MpAddress addr[2] ;                            // destination address storage area
int naddr ;
char *text           = "an abnormality has occurred." ;   // message to be sent
int type             = PAGER_TEXT ;
long lrc ;



addr[0].type = TYPE_ADDRESS ;                  // Sends Short Mail to 2 destinations
addr[0].address = "0903101655T1112223333" ;
addr[0].vender = MPPAGER_DOCOMO_SHORTMAIL ;
addr[1].type = TYPE_ADDRESS ;
addr[1].address = "0903101655T1114445555" ;
addr[1].vender = MPPAGER_DOCOMO_SHORTMAIL ;
naddr = 2 ;

lrc = Mp_CallPager2(AppName,addr,naddr,text,type,NULL) ;
if ( lrc < 0 ) {
    /* error processing */
}

/* normal completion */
```

16.4.2 Action Execution API (Compatible with Old Versions)

> This section explains the action execution APIs provided by the previous versions of Systemwalker Operation Manager.
>
> A list of the Action Execution APIs is shown below:
>
> - Sound Notification API (Mp_PlaySound)
>
> - Pop-up Message Transmission (Specified Users) API (MP_PopupUser)
>
> - Pop-up Message Transmission (Domains) API (MP_PopupDomain)
>
> - Pop-up Message Transmission (All Users) API (Mp_PopupSession)
>
> The previous version of Systemwalker Operation Manager is as follows:
>
> - SystemWalker/OperationMGR V4.0

16.4.2.1 Sound Notification API

This section explains the sound notification API (Mp_PlaySound).

Synopsis

```
long Mp_PlaySound (unsigned char *AppName,
                   int kind,
                   unsigned char *text_or_filename,
                   int repeat,
                   MpSoundParam *SoundParam)
```

Description

This emits the following sounds:

- Message read out

- WAV file playback

- Beep

Parameters

AppName

Specify the address of the application that made the call. The application name must be 64 bytes or less in length.

kind

Specify the type of sound. Specify one of the following in this parameter.

SOUND_VOICE:

Reads out the message text.

SOUND_WAVE:

Plays a WAV file.

SOUND_BEEP:

Emits a beep.

text_or_filename

Specify one of the followings depending on the value of parameter kind.

When specifying SOUND_VOICE:

Specify the address of the message text that is to be read out.

When specifying SOUND_WAVE:

Specify the full path address of the file to be played to a value within the range specified in MAX_PATH.

When specifying SOUND_BEEP:

This parameter is invalid.

repeat

Specify the number of times the sound should be emitted. If 0 (zero) has been specified, the sound is repeated until stop processing has been completed.

SoundParam

Specify the address of the sound option information structure (MpSoundParam). If SOUND_VOICE has been specified, specify this when changing the system settings. This is invalid if SOUND_VOICE has not been specified. Specify NULL if this does not have to be set. If 0 (zero) is specified for all items, the system settings will be used.

Return Values

If the request is successfully accepted, an integer above 0 will be returned. Any value less than 0 is an error. The following table provides details about errors that may be returned.

Return Value	Meaning	Action
MPACT_PARAMERR	There is a parameter error.	Check for incorrect parameter settings.
MPACT_NOMEMORY	There is not enough memory.	Either increase the size of the page file or add more physical memory.
MPACT_NOTSOUND	The environment for playing sound has not been installed.	Check that a product equipped with a voice synthesizing engine compatible with Microsoft Speech API(SAPI)5.x has been installed.
MPACT_NOTMOVE	The Action Control Server has not been started.	Check that Systemwalker MpAosfB is running.
MPACT_COMFAIL	An error has occurred in communication with the Action Control Server.	Check that Systemwalker MpAosfB and Systemwalker MpAosfX are running.
MPACT_SYSERR	An error has occurred on the Action Control Server.	Refer to the message output to the event log or syslog, eliminate the cause of the error and then restart Systemwalker. If the problem persists after the cause of the error has been eliminated and Systemwalker has been restarted, use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_NOTDLL	DLL for the action request cannot be loaded.	Check that the file f3crhxac.dl (see Note), which is needed for action management, exists in the following folder: Systemwalker installation directory\mpwalker\bin If the file is not located in the above folder, either reinstall Systemwalker or use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_EXCEPT	The address of an invalid area has been specified.	An internal logic error has occurred. Collect data with the maintenance information collection tool and contact a Fujitsu SE.

Format of the Sound Option Information Structure (MpSoundParam)

```
typedef struct MpSoundParam_tag {
    DWORD dwVoiceType ;
    DWORD dwIntonation ;
    DWORD dwPitch ;
    DWORD dwSpeed ;
    DWORD dwSpeedRate ;
    DWORD dwTone ;
```

```
    DWORD dwVolume ;
} MpSoundParam ;
```

dwVoiceType

Specify the voice type.

VOICE_MALE:

This plays a male voice.

VOICE_FEMALE:

This plays a female voice.

dwIntonation

Specify an intonation level between 1 and 4.

dwPitch

Specify the pitch of the voice between 1 and 5.

dwSpeed

Specify the speed of the voice between 1 and 10.

dwSpeedRate

Specify the fluctuation of the speed of the voice between 1 and 10.

dwTone

Specify whether there is emphasis in high voice tone.

No:

1

Yes:

2

DWORD dwVolume

Specify the volume of the voice between 1 and 10.

Program Example

Below is an example program of the sound notification API:

```
#include "f3crhxac.h"

unsigned char *AppName = "Sound notification API" ;  // requested source application name
int kind              = SOUND_VOICE ;                     // read out text
unsigned char *text   = "an error has occurred. please contact your operator." ;
int repeat            = 0 ;                       // repeat until stop process is executed.
MpSoundParam  *SndPrm = NULL ;                          // default value is used.
long          lrc ;

/* request for sound notification */
lrc = Mp_PlaySound(AppName,kind,text,repeat,NULL) ;
if ( lrc < 0 ) {
    /* error processing */
}
```

```
/* normal completion */
```

16.4.2.2 Pop-up Message Transmission (Specified Users) API

This section explains the pop-up message transmission (specified users) API (Mp_PopupUser).

Synopsis

```
long Mp_PopupUser (unsigned char *AppName,
            MpAddress *AddressList,
            int nAddressList,
            unsigned char *MsgText)
```

Description

This API displays messages on specified computers or on computers to which specified users have logged on.

Parameters

AppName

Specify the address of the application that made the call. The application name must be 64 bytes or less in length.

AddressList

Specify the address of the destination information structure (MpAddress). Specify the name of the user or computer that is sending the pop-up message. If there are multiple names, specify them in an array. If NULL is specified, the pop-up message will be sent to all users logged on to the domain.

nAddressList

Specify the number of arrays specified in AddressList.

MsgText

Specify the address of the message that will appear in the pop-up window. The message has to be 128 bytes or less in length.

Return Values

If the request was successfully accepted, 0 or a higher integer will be returned. Any value less than 0 is an error. The following table provides details about errors that may be returned.

Return Value	Meaning	Action
MPACT_PARAMERR	There is a parameter error.	Check for incorrect parameter settings.
MPACT_NOMEMORY	There is not enough memory.	Either increase the size of the page file or add more physical memory.
MPACT_NOTADDRESS	There is no valid user name or computer name.	There is an error in the address specified in the address information structure (MpAddress). Correct the address specification.
MPACT_NOTMOVE	The Action Control Server has not been started.	Check that Systemwalker MpAosfB is running.
MPACT_COMFAIL	An error has occurred in communication with the Action Control Server. An error has occurred in communication with the Action Control Server.	Check that Systemwalker MpAosfB and Systemwalker MpAosfX are running.

Return Value	Meaning	Action
MPACT_SYSERR	An error has occurred on the Action Control Server.	Refer to the message output to the event log or syslog, eliminate the cause of the error and then restart Systemwalker. If the problem persists after the cause of the error has been eliminated and Systemwalker has been restarted, use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_NOTDLL	DLL for the action request cannot be loaded.	Check that the file f3crhxac.dl (see Note), which is needed for action management, exists in the following folder: Systemwalker installation directory\mpwalker \bin If the file is not located in the above folder, either reinstall Systemwalker or use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_EXCEPT	The address of an invalid area has been specified.	An internal logic error has occurred. Collect data with the maintenance information collection tool and contact a Fujitsu SE.

Caution

Pop-up messages can only be used on Microsoft networks.

Format of the Destination Information Structure (MpAddress)

```
typedef struct MpAddress_tag {
    int type ;
    unsigned char *address ;
    char *addrid ;
} MpAddress ;
```

type

Specify the type of address that was specified in address.

TYPE_ADDRESS:

This is the user name or computer name.

TYPE_NAME:

This is the destination name.

address

Specify the user name/computer name or the address of the destination.

addrid

This parameter is enabled when Systemwalker Centric Manager is installed and "TYPE_NAME" is specified as the type parameter. Refer to the *Systemwalker Centric Manager Reference Guide* for details.

Program Example

Below is an example program of the pop-up message transmission (specified users) API:

```
#include "f3crhxac.h"

unsigned char *AppName = "pop-up API" ;    // requested source application name
MpAddress addr[2] ;                        // destination address storage area
int naddr ;
unsigned char *text   = "this will stop the system. " ;  // message text to be sent

long lrc ;

addr[0].type = TYPE_ADDRESS ;   // send a message  to users "operator" and "user01"
addr[0].address = "operator" ;
addr[1].type = TYPE_ADDRESS ;
addr[1].address = "user01" ;
naddr = 2 ;

lrc = Mp_PopupUser(AppName,addr,naddr,text) ;
if ( lrc < 0 ) {
    /* error processing */
}

/* normal completion */
```

16.4.2.3 Pop-up Message Transmission (Domains) API

This section explains the pop-up message transmission (domain) API (Mp_PopupDomain).

Synopsis

```
long Mp_PopupDomain (unsigned char *AppName,
            char *domain,
            unsigned char *MsgText)
```

Description

This API displays pop-up messages on all computers that are connected to a specified domain.

Parameters

AppName

Specify the address of the application that made the call. The application name must be 64 bytes or less in length.

domain

Specify the address of the domain that will display the message. The domain name must be 15 bytes or less in length.

MsgText

Specify the address of the message that will appear in the pop-up window. The message has to be 128 bytes or less in length.

Return Values

If the request was successfully accepted, 0 or a higher integer will be returned. Any value less than 0 is an error. The following table provides details about errors that may be returned.

Return Value	Meaning	Action
MPACT_PARAMERR	There is a parameter error.	Check for incorrect parameter settings.
MPACT_NOMEMORY	There is not enough memory.	Either increase the size of the page file or add more physical memory.

Return Value	Meaning	Action
MPACT_NOTMOVE	The Action Control Server has not been started.	Check that Systemwalker MpAosfB is running.
MPACT_COMFAIL	An error has occurred in communication with the Action Control Server.	Check that Systemwalker MpAosfB and Systemwalker MpAosfX are running.
MPACT_SYSERR	An error has occurred on the Action Control Server.	Refer to the message output to the event log or syslog, eliminate the cause of the error and then restart Systemwalker. If the problem persists after the cause of the error has been eliminated and Systemwalker has been restarted, use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_NOTDLL	DLL for the action request cannot be loaded.	Check that the file f3crhxac.dl (see Note), which is needed for action management, exists in the following folder: Systemwalker installation directory\mpwalker\bin If the file is not located in the above folder, either reinstall Systemwalker or use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_EXCEPT	The address of an invalid area has been specified.	An internal logic error has occurred. Collect data with the maintenance information collection tool and contact a Fujitsu SE.

Caution

Pop-up messages can only be used on Microsoft networks.

Program Example

Below is an example program of the pop-up message transmission (domains) API:

```
#include "f3crhxac.h"

unsigned char *AppName = "pop-up API" ;                 // requested source application name
char *domain          = "Domain" ;                      // destination domain name
unsigned char *text   = "this will stop the system. " ; // message text to be sent
long lrc ;

lrc = Mp_PopupDomain(AppName,domain, text) ;
if ( lrc < 0 ) {
    /* error processing */
}

/* normal completion */
```

16.4.2.4 Pop-up Message Transmission (All Users) API

This section explains the pop-up message transmission (all users) API (Mp_PopupSession).

Synopsis

```
long Mp_PopupSession (unsigned char *AppName,
                      unsigned char *MsgText)
```

Description

This API sends pop-up messages to all sessions connected to the server.

Parameters

AppName

Specify the address of the application that made the call. The application name must be 64 bytes or less in length.

MsgText

Specify the address of the message that will appear in the pop-up window. The message has to be 128 bytes or less in length.

Return Values

If the request was successfully accepted, 0 or a higher integer will be returned. Any value less than 0 is an error. The following table provides details about errors that may be returned.

Return Value	Meaning	Action
MPACT_PARAMERR	There is a parameter error.	Check for incorrect parameter settings.
MPACT_NOMEMORY	There is not enough memory.	Either increase the size of the page file or add more physical memory.
MPACT_NOTMOVE	The Action Control Server has not been started.	Check that Systemwalker MpAosfB is running.
MPACT_COMFAIL	An error has occurred in communication with the Action Control Server.	Check that Systemwalker MpAosfB and Systemwalker MpAosfX are running.
MPACT_SYSERR	An error has occurred on the Action Control Server.	Refer to the message output to the event log or syslog, eliminate the cause of the error and then restart Systemwalker. If the problem persists after the cause of the error has been eliminated and Systemwalker has been restarted, use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_NOTDLL	DLL for the action request cannot be loaded.	Check that the file f3crhxac.dl (see Note), which is needed for action management, exists in the following folder: Systemwalker installation directory\mpwalker \bin If the file is not located in the above folder, either reinstall Systemwalker or use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_EXCEPT	The address of an invalid area has been specified.	An internal logic error has occurred. Collect data with the maintenance information collection tool and contact a Fujitsu SE.

Pop-up messages can only be used on Microsoft networks.

Program Example

Below is an example program of the pop-up message transmission (all users) API:

```
#include "f3crhxac.h"

unsigned char *AppName = "pop-up API" ;            // requested source application name
unsigned char *text    = "this will stop the system. " ; //  message text to be sent
long lrc ;

lrc = Mp_PopupSession(AppName,text) ;
if ( lrc < 0 ) {
    /* error processing */
}

/* normal completion */
```

16.4.3 Action Stop API

This section explains the action stop API.

The list of action stop APIs is shown below:

- Action Stop API (Mp_StopAction)

16.4.3.1 Action Stop API

This section explains the action stop API (Mp_StopAction).

Synopsis

```
long Mp_StopAction (unsigned char *AppName,
            long handle)
```

Description

This API stops actions that have been requested to the Action Control Server.

Parameters

AppName

Specify the address of the application (specified in the action execution API) that requested the action that you want to stop.

handle

Specify the action control number (return value of the action execution API).

Return Values

Return Value	Meaning	Action
0	This stops the action.	-
MPACT_NOTACTION	Specified action does not exist (already terminated).	-

Return Value	Meaning	Action
MPACT_NOTSTOP	Cannot stop the action because it is being executed.	The action cannot be stopped because it is currently running.
MPACT_NOTACCESS	You do not have permission to stop the action.	Execute the action as a user with Administrator privileges.
MPACT_PARAMERR	There is a parameter error.	Check for incorrect parameter settings.
MPACT_NOMEMORY	There is not enough memory.	Either increase the size of the page file or add more physical memory.
MPACT_NOTMOVE	The Action Control Server has not been started.	Check that Systemwalker MpAosfB is running.
MPACT_COMFAIL	Communication with the Action Control Server has failed.	Check that Systemwalker MpAosfB and Systemwalker MpAosfX are running.
MPACT_SYSERR	An error has occurred on the Action Control Server.	Refer to the message output to the event log or syslog, eliminate the cause of the error and then restart Systemwalker. If the problem persists after the cause of the error has been eliminated and Systemwalker has been restarted, use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_NOTDLL	DLL for the action request cannot be loaded.	Check that the file f3crhxac.dl (see Note), which is needed for action management, exists in the following folder: Systemwalker installation directory\mpwalker\bin If the file is not located in the above folder, either reinstall Systemwalker or use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_EXCEPT	The address of an invalid area has been specified.	An internal logic error has occurred. Collect data with the maintenance information collection tool and contact a Fujitsu SE.

Program Example

Below is an example program of the action stop API:

```
#include "f3crhxac.h"

unsigned char *AppName = "execution API" ;  // application name specified in the action
                                            // execution API
long handle            = 1 ;                // return value of the action execution API
long lrc ;

lrc = Mp_StopAction(AppName,handle) ;
if ( lrc != 0 ) {
    /* error processing */
}

/* normal completion */
```

16.4.4 Action Status Notification API

This section explains the action status notification API.

A list of action status notification APIs is shown below:

- Action Status Notification API (Mp_GetActionInfo2)

- Action Information Area Release API (Mp_FreeActionInfo2)

16.4.4.1 Action Status Notification API

This section explains the action status notification API (Mp_GetActionInfo2).

Synopsis

```
long Mp_GetActionInfo2 (long handle,
            MpActionInfo2 **ActionInfo,
            int *nActionInfo)
```

Description

This API posts the current status of the specified action.

Parameters

handle

Specify the action control number (return value of the action execution API).

ActionInfo

Specify the address of the area for the action status information structure (MpActionInfo2). The action status information structure will be obtained in the API. The action status information structure will be set in the area indicated by this parameter. The action status information structure is released by the release API (Mp_FreeActionInfo2).

nActionInfo

Specify the address of the area that stores the number of actions. The number of arrays of the action status information structure will be set in the area indicated by this parameter.

Return Values

Return Value	Meaning	Action
0	This stops the action.	-
MPACT_NOTACTION	Specified action does not exist (already terminated).	-
MPACT_PARAMERR	There is a parameter error.	Check for incorrect parameter settings.
MPACT_ALLOCFAIL	Failed to obtain area.	Either increase the size of the page file or add more physical memory.
MPACT_NOTMOVE	The Action Control Server has not been started.	Check that Systemwalker MpAosfB is running.
MPACT_COMFAIL	Communication with the Action Control Server has failed.	Check that Systemwalker MpAosfB and Systemwalker MpAosfX are running.
MPACT_SYSERR	An error has occurred on the Action Control Server.	Refer to the message output to the event log or syslog, eliminate the cause of the error and then restart Systemwalker. If the problem persists after the cause of the error has been eliminated and

Return Value	Meaning	Action
		Systemwalker has been restarted, use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_NOTDLL	DLL for the action request cannot be loaded.	Check that the file f3crhxac.dl (see Note), which is needed for action management, exists in the following folder: Systemwalker installation directory\mpwalker\bin If the file is not located in the above folder, either reinstall Systemwalker or use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_EXCEPT	The address of an invalid area has been specified.	An internal logic error has occurred. Collect data with the maintenance information collection tool and contact a Fujitsu SE.

Format of Action Status Information Structure (MpActionInfo2)

```
typedef struct MpActionInfo2_tag {
    long no ;
    long action ;
    char *exehost;
    MpActionAddr *addr ;
    int  naddr ;
    int  status ;
    int  code ;
} MpActionInfo2 ;
```

no

The action control number is set here.

action

The action type is set here.

ACTION_EMAIL:

This sends an e-mail.

ACTION_MSMAIL:

This sends an MS-mail.

ACTION_POPUP:

This sends a pop-up message.

ACTION_PAGER:

This sends a short mail message.

ACTION_SOUND:

This plays a sound.

exehost

The name of the host that executes an action is set.

addr

Specify the address of the action execution destination information structure (MpActionAddr). This stores the address of the action's execution destination.

naddr

The number of addresses specified in addr is set here.

status

The action status is set here.

STATUS_NORMAL:

Action has been completed.

STATUS_EXEC:

Action is being executed.

STATUS_EXEWAIT:

Action is waiting.

STATUS_ERROR:

Action has abended.

STATUS_STOP:

Action has paused.

code

Set this if status is abended. See "Detail Code" described later for more about detail codes.

Format of Action Execution Destination Information Structure (MpActionAddr)

```
typedef struct MpActionAddr_tag {
    unsigned char *name ;
    char *address ;
} MpActionAddr ;
```

name

If a destination has been specified in the action execution API, its address will be set here. If the address has been specified, NULL will be set.

address

The address of the transmission destination is set.

Using email:

The mail address is set.

Using a popup message:

The user name or computer name is set.

Using short mail:

The telephone number is set.

Detail Code

This describes the status of the action obtained in the action status notification API (Mp_GetActionInfo).

Detail Code	Meaning	Action
MPACT_D_NOMEMORY	A shortage of memory occurred while executing the action.	All
MPACT_D_FILEACCESS	An error occurred in file operations while executing the action.	
MPACT_D_NOFILE	The specified file does not exist.	
MPACT_D_DEFFILE	The action execution environment has not been defined.	
MPACT_D_FAIL	Action execution failed for some reason.	
MPACT_D_NOTMOVE	The action execution server has not been started.	
MPACT_D_NOHOST	The specified action execution host does not exist.	
MPACT_D_SENDFAIL	Failed to send a message.	Popup
MPACT_D_NOTSERVER	The SMTP server could not be found.	E-Mail
MPACT_D_NOTMYHOST	The local host name cannot be obtained.	
MPACT_D_SMTPCOMMAND	An error occurred during processing on the SMTP server.	
MPACT_D_SMTPCOMM	An error occurred during communication with the SMTP server.	
MPACT_D_NOTCONNECT	A connection could not be made with the SMTP server.	
MPACT_D_FROMADDR	The sender's address has not been specified.	
MPACT_D_SIGNIN	Sign-in failed.	MS-Mail
MPACT_D_NOSEND	An error occurred during e-mail sending processing.	
MPACT_D_COMMOPEN	The COM port (modem) could not be opened.	Short mail
MPACT_D_COMMFAIL	An error occurred in the COM port (modem) I/O.	
MPACT_D_INITCOMM	Configuration of the COM port (modem) failed.	
MPACT_D_LINEBUSY	Signal could not be sent because the line was busy.	
MPACT_D_LINEFAIL	Signal could not be sent due to a problem with the telephone line.	
MPACT_D_CONVERTFAIL	The message conversion processing failed.	
MPACT_D_NOTSOUND	The environment for sound replay has not been configured.	Sound notification
MPACT_D_NOTDEVICE	Another application is using the sound replay driver.	
MPACT_D_CREATESOUND	Sound replay processing failed.	

Program Example

Below is an example program of the action status notification API:

```
#include "f3crhxac.h"

long handle = 1 ;                  // control number of the action that obtains the status
MpActionInfo *pInfo = NULL ;  // address setting range of information area
int nInfo = 0 ;

lrc = Mp_GetActionInfo2(handle,&pInfo,&nInfo);
if ( lrc != 0 ) {
    /* error processing */
}
```

```
/* reference to the action information (pInfo) */

/* release the action information area (pInfo) */
Mp_FreeActionInfo2(pInfo,nInfo) ;
```

16.4.4.2 Action Information Area Release API

This section explains the action information area release API (Mp_FreeActionInfo2).

Synopsis

```
void Mp_FreeActionInfo2 (MpActionInfo2 *ActionInfo,
                         int nAction)
```

Description

This API releases the area that was obtained in the action status notification API (Mp_GetActionInfo2).

Parameters

ActionInfo

Specify the address of the action status information structure (MpActionInfo2) that was obtained in the action status notification API.

See "16.4.4.1 Action Status Notification API" for more about the action status information structure.

nAction

This shows the number of actions set in ActionInfo.

Program Example

Below is an example program of the action information area release API:

```
#include "f3crhxac.h"

long handle = 1 ;               // control number of the action that obtains the status
MpActionInfo2 *pInfo = NULL ;   // address setting range of information area
int nInfo = 0 ;

lrc = Mp_GetActionInfo2(handle,&pInfo,&nInfo);
if ( lrc != 0 ) {
    /* error processing */
}

/* reference to the action information (pInfo) */

/* release the action information area (pInfo) */
Mp_FreeActionInfo2(pInfo,nInfo) ;
```

16.4.5 Action Status Notification API (Compatible with Old Versions)

This section explains the action status notification APIs provided by the previous versions of Systemwalker Operation Manager.

A list of action status notification APIs is shown below:

- Action Status Notification API (Mp_GetActionInfo)

- Action Information Area Release API (Mp_FreeActionInfo)

The previous version of Systemwalker Operation Manager is as follows:

- SystemWalker/OperationMGR V4.0

16.4.5.1 Action Status Notification API

This section explains the action status notification API (Mp_GetActionInfo).

Synopsis

```
ong Mp_GetActionInfo (long handle,
                      MpActionInfo **ActionInfo,
                      int *nActionInfo)
```

Description

This API posts the current status of the specified action.

Parameters

handle

Specify the action control number (return value of the action execution API).

ActionInfo

Specify the address of the area for the action status information structure (MpActionInfo). The action status information structure will be obtained in the API. The action status information structure will be set in the area indicated by this parameter. The action status information structure is released by the release API (Mp_FreeActionInfo).

nActionInfo

Specify the address of the area that stores the number of actions. The number of arrays of the action status information structure will be set in the area indicated by this parameter.

Return Values

Return Value	Meaning	Action
0	This stops the action.	-
MPACT_NOTACTION	Specified action does not exist (already terminated).	-
MPACT_PARAMERR	There is a parameter error.	Check for incorrect parameter settings.
MPACT_ALLOCFAIL	Failed to obtain area.	Either increase the size of the page file or add more physical memory.
MPACT_NOTMOVE	The Action Control Server has not been started.	Check that Systemwalker MpAosfB is running.
MPACT_COMFAIL	Communication with the Action Control Server has failed.	Check that Systemwalker MpAosfB and Systemwalker MpAosfX are running.
MPACT_SYSERR	An error has occurred on the Action Control Server.	Refer to the message output to the event log or syslog, eliminate the cause of the error and then restart Systemwalker. If the problem persists after the cause of the error has been eliminated and Systemwalker has been restarted, use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.

Return Value	Meaning	Action
MPACT_NOTDLL	DLL for the action request cannot be loaded.	Check that the file f3crhxac.dl (see Note), which is needed for action management, exists in the following folder: Systemwalker installation directory\mpwalker\bin If the file is not located in the above folder, either reinstall Systemwalker or use the maintenance information collection tool to collect event monitoring data and contact a Fujitsu SE.
MPACT_EXCEPT	The address of an invalid area has been specified.	An internal logic error has occurred. Collect data with the maintenance information collection tool and contact a Fujitsu SE.

Format of Action Status Information Structure (MpActionInfo)

```
typedef struct MpActionInfo_tag {
    long no ;
    long action ;
    MpActionAddr *addr ;
    int  naddr ;
    int  status ;
    int  code ;
} MpActionInfo ;
```

no

The action control number is set.

action

The action type is set.

ACTION_EMAIL:

This sends e-mail.

ACTION_MSMAIL:

This sends an MS-mail.

ACTION_POPUP:

This sends a pop-up message.

ACTION_PAGER:

This sends a short mail message.

ACTION_SOUND:

This plays a sound.

addr

Specify the address of the action execution destination information structure (MpActionAddr). This stores the address of the action's execution destination.

naddr

The number of addresses specified in addr is set here.

status

The action status is set.

STATUS_NORMAL:

Action has been completed.

STATUS_EXEC:

Action is being executed.

STATUS_EXEWAIT:

Action is waiting.

STATUS_ERROR:

Action has abended.

STATUS_STOP:

Action has paused.

code

The detail code is set if the status indicates an abnormal termination. See Detail code described later for more about detail codes.

Format of Action Execution Destination Information Structure (MpActionAddr)

```
typedef struct MpActionAddr_tag {
    unsigned char *name ;
    char *address ;
} MpActionAddr ;
```

name

If a destination has been specified in the action execution API, its address will be set here. If the address has been specified, NULL will be set.

address

The address of the transmission destination is set.

Using email:

The mail address is set.

Using a popup message:

The user name or computer name is set.

Using short mail:

The telephone number is set.

Detail Code

This describes the status of the action obtained in the action status notification API (Mp_GetActionInfo).

Detail Code	Meaning	Action
MPACT_D_NOMEMORY	A shortage of memory occurred while executing the action.	All
MPACT_D_FILEACCESS	An error occurred in file operations while executing the action.	
MPACT_D_NOFILE	The specified file does not exist.	

Detail Code	Meaning	Action
MPACT_D_DEFFILE	The action execution environment has not been defined.	
MPACT_D_FAIL	Action execution failed for some reason.	
MPACT_D_NOTMOVE	The action execution server has not been started.	
MPACT_D_NOHOST	The specified action execution host does not exist.	
MPACT_D_SENDFAIL	Failed to send a message.	Popup
MPACT_D_NOTSERVER	The SMTP server could not be found.	E-Mail
MPACT_D_NOTMYHOST	The local host name cannot be obtained.	
MPACT_D_SMTPCOMMAND	An error occurred during processing on the SMTP server.	
MPACT_D_SMTPCOMM	An error occurred during communication with the SMTP server.	
MPACT_D_NOTCONNECT	A connection could not be made with the SMTP server.	
MPACT_D_FROMADDR	The sender's address has not been specified.	
MPACT_D_SIGNIN	Sign-in failed.	MS-Mail
MPACT_D_NOSEND	An error occurred during e-mail sending processing.	
MPACT_D_COMMOPEN	The COM port (modem) could not be opened.	Short mail
MPACT_D_COMMFAIL	An error occurred in the COM port (modem) I/O.	
MPACT_D_INITCOMM	Configuration of the COM port (modem) failed.	
MPACT_D_LINEBUSY	Signal could not be sent because the line was busy.	
MPACT_D_LINEFAIL	Signal could not be sent due to a problem with the telephone line.	
MPACT_D_CONVERTFAIL	The message conversion processing failed.	
MPACT_D_NOTSOUND	The environment for sound replay has not been configured.	Sound notification
MPACT_D_NOTDEVICE	Another application is using the sound replay driver.	
MPACT_D_CREATESOUND	Sound replay processing failed.	

Program Example

Below is an example program of the action status notification API:

```
#include "f3crhxac.h"

long handle = 1 ;                  // control number of the action that obtains the status
MpActionInfo *pInfo = NULL ;       // address setting range of information area
int nInfo = 0 ;

lrc = Mp_GetActionInfo(handle,&pInfo,&nInfo);
if ( lrc != 0 ) {
    /* error processing */
}

/* reference to the action information (pInfo) */

/* release the action information area (pInfo) */
Mp_FreeActionInfo(pInfo,nInfo) ;
```

16.4.5.2 Action Information Area Release API

This section explains the action information area release API (Mp_FreeActionInfo).

Synopsis

```
void Mp_FreeActionInfo (MpActionInfo *ActionInfo,
                            int nAction)
```

Description

This API releases the area that was obtained in the action status notification API (Mp_GetActionInfo).

Parameters

ActionInfo

Specify the address of the action status information structure (MpActionInfo) that was obtained in the action status notification API.

See "16.4.5.1 Action Status Notification API" for more about the action status information structure.

nAction

This parameter shows the number of actions set in ActionInfo.

Program Example

Below is an example program of the action information area release API:

```
#include "f3crhxac.h"

long handle = 1 ;                  // control number of the action that obtains the status
MpActionInfo *pInfo = NULL ;      //  address setting range of information area
int nInfo = 0 ;

lrc = Mp_GetActionInfo(handle,&pInfo,&nInfo);
if ( lrc != 0 ) {
    /* error processing */
}

/* reference to the action information (pInfo) */

/* release the action information area (pInfo) */
Mp_FreeActionInfo(pInfo,nInfo) ;
```

Chapter 17 Exit Information

This chapter explains the exit information provided by Systemwalker Operation Manager.

17.1 Job Net Completed /Abended Exit

This section explains the exit program used when a job net registered on Jobscheduler ends in Completed, Pseudo-normal, Abended or Canceled.

[Windows]

If the job net ends in Completed or Pseudo-normal, the system searches for the normalexit.bat file first, and then the normalexit.exe file as the exit program. The exit program found will be used. If normalexit.bat is found, normalexit.exe will not be called up even if it exists.

If the job net ends in Abended or Canceled, the system searches for the jobschexit.bat file first, and then the jobschexit.exe file as the exit program. The exit program found will be used. If jobschexit.bat is found, jobschexit.exe will not be called up even if it exists.

If you want to use these exits to perform custom processing, create an exit program with the same name and store it in the following directory:

> *<Systemwalker installation directory>*\MpWalker.JM\bin

[UNIX]

If the job net ends in Completed or Pseudo-normal, the system searches for and calls the jobsch.exit.normal file as the exit program.

If the job net ends in Abended or Canceled, the system searches for and calls the jobsch.exit file as the exit program.

If you want to use these exits to perform custom processing, create an exit program with the same name and store it in the following directory:

Solaris/Linux:

> /opt/FJSVJOBSC/bin

HP-UX:

> /opt/FHPJOBSCH/bin

AIX:

> /usr/FAIXJOBSC/bin

Appropriate execution privileges are required to execute the exit program. If relevant privileges are not set, the exit program will not be called.

Parameters Passed to the Exit Program

The parameters passed to the exit program are as follows:

> **argc :**
>
>   Number of parameters (7) (Note1)
>
> **argv[0] :**
>
>   Exit program name
>
> **argv[1] :**
>
>   Name of project where the job net is registered

> **argv[2] :**
>
> Job net name
>
> **argv[3] :**
>
> Job net comment
>
> **argv[4] :**
>
> Maximum length (character string) of job completion code in the job net
>
> **argv[5] :**
>
> Number of jobs in job net (character string)
>
> **argv[6] :**
>
> Temporary file name
>
> **argv[7] :**
>
> Subsystem number (character string) (Note2)

Note1:

In Systemwalker Operation Manager EE, the number of parameters is eight.

Note2:

This parameter is used in Systemwalker Operation Manager EE.

Information on each job in the job net is stored in the temporary file as shown below. The job name, completion code, and status are stored for each job.

```
Job name
Completion code
Status
Job name
Completion code
Status
............
Job name
Completion code
Status
```

The following explanations describe each of the types of information shown above:

Job name:

Depending on execution attributes, the following contents are stored:

When the execution attribute is "Job Execution Control":

This is a job name. If the job name is omitted, the string registered as a command will be stored.

When the execution attribute is "Interstage":

This is a job workunit name.

When the execution attribute is "PowerAIM" [UNIX]:

This is a job workunit name.

When the execution attribute is "Normal" [UNIX]:

This is a command name.

Completion code:

This is a job completion code (numeric number between 0 and 256). Completion code for unexecuted job is 0.

Status:

The job status can be specified using one of the following seven strings:

normal_end:

Indicates completed.

pseudo-normal:

Indicates pseudo-normal completion.

abnormal_end:

Indicates abended.

canceled:

Indicates canceled.

not_executed:

Indicates the job was not started.

paused:

Indicates the job has been paused.

disabled:

Indicates the job has been disabled.

The temporary file is created in the following directories:

[Windows]

In the work directory under the database directory of Jobscheduler

The name of the completed exit file is created in NMLnnn.tmp (nnn is appended by the system) format and that of the abended exit file is created in NETnnn.tmp (nnn is appended by the system) format.

[Solaris, HP-UX]

/var/tmp directory

[AIX, Linux]

/tmp directory

Since the temporary file is not deleted on the Jobscheduler side, you must delete it from the exit program. If the exit is not registered, the temporary file is not created.

About the Job in Passed Status

If the execution condition of the following job is specified in Condition concerning the completion code of preceding job, the job that ends with the Passed status will be processed as follows:

- The job status (waiting /disabled/paused) before being passed will be passed as the job information.

- The completion code is 0.

A job that is waiting using the OR condition and is then canceled because of the OR condition of the following job will be processed as follows:

- The job status will be "canceled" regardless of whether the job was actually started.

- The completion code is 248 or 249.

<u>Cautions</u>

- Prevent time-consuming process because it may cause delay of execution for job nets and groups.

- The execution user at the exit is the logon account of Jobscheduler service. [Windows]

- The execution user at the exit is a root account. [UNIX]

- If the current job net is operated (for example, disabled or deleted) at the exit, the job net operation may fail. This is because the processing performance is considered and the termination of a job net is not guaranteed when called at the exit. If you need to operate a job net at its exit, retry it for a certain time period as its operation may fail.

Program Example

Below is an example program for the job net abended exit:

[Windows]

```
/* When a job net abnormal completion occurs, sends the following messages to "MANAGEMENT"
   server.
               Job net name is abended !! code = completion code*/
#include <stdio.h>
#include <process.h>

main(int argc, char *argv[])
{
    char cmdline[256];

    sprintf(cmdline,
            "NET SEND MANAGEMENT %s is abended !! code = %s",
            argv[2],    //job net name
            argv[4]);   //completion code

    system(cmdline);

    unlink(argv[6]);    // temp file is deleted

    exit(0);
}
```

[Solaris]

```
/*
  Example program for the job net abended exit (for UNIX), which outputs a job net
  abnormal completion message with a job net name and completion code to the console
*/

#include <syslog.h>
#include <stdio.h>

main(int argc, char **argv)
{
   openlog ("jobsch", LOG_CONS, LOG_USER);
   syslog (LOG_WARNING,
          "WARNING: jobnet abnormal end. project=%s, net_name=%s, net_comment=%s, net_code=
%s",
          argv[1], argv[2], argv[3], argv[4]);
   closelog();

   unlink (argv[6]);
   exit(0);
}
```

17.2 Job Net Abended Extended Exit

This section explains the exit program called when job nets end with Abended or Canceled.

[Windows]

If the job net ends in Abended or Canceled, the system searches for the jobschexitex.bat file first, and then the jobschexitex.exefile as the exit program. The exit program found in the search will be called up. If jobschexitex.bat is found, jobschexitex.exe will not be called up even if it exists.

If you want to use this exit to perform custom processing, create an exit program with the same name and store it in the following directory:

```
<Systemwalker installation directory>\MpWalker.JM\bin
```

[UNIX]

If the job net ends in Abended or Canceled, the system searches for and calls the jobschex.exit file as the exit program.

If you want to use this exit to perform custom processing, create an exit program with the same name and store it in the following directory:

Solaris/Linux:

```
/opt/FJSVJOBSC/bin
```

HP-UX:

```
/opt/FHPJOBSCH/bin
```

AIX:

```
/usr/FAIXJOBSC/bin
```

Appropriate execution privileges are required to execute the exit program. If relevant privileges are not set, the exit program will not be called.

The job net abended extended exit differs from the job net abended exit. This means if the job net of Job Execution Control attribute ends with Abended, the completion code passed to the exit program is the same with the value returned from Job Execution Control.

Parameters Passed to the Exit Program

The parameters passed to the exit program are as follows:

argc:

Number of parameters (20)

argv[0]:

Exit program name

argv[1]:

Name of project where the job net is registered

argv[2]:

Job net name

argv[3]:

Job net comment

argv[4]:

Name of group to which job net belongs

However, if it does not belong to any group, the parameter is set as ""

argv[5]:

Maximum value of job completion code in the job net

If it is the job net of Job Execution Control attribute, it may exceed 256

However, it is 256 if the job net has been canceled, and it becomes 239 if interrupted.

argv[6]:

Job net start time

The format is "year month/date hour:minute:second".

argv[7]:

Job net end time

The format is "year month/date hour:minute:second".

argv[8]:

Number of jobs in job net (character string)

argv[9]:

Temporary file name

argv[10]:

For SE version, set as "", and for EE version, set with subsystem number.

argv[11 to 19]:

Reserved ("")

Information on each job in the job net is stored in the temporary file as shown below. The job name, completion code, and status are stored for each job.

```
Job name
Completion code
Status
Job name
Completion code
Status
..........
Job name
Completion code
Status
```

The following explanations describe each of the types of information shown above:

Job name:

Depending on execution attributes, the following contents are stored:

When the execution attribute is "Job Execution Control":

This is a job name. If the job name is omitted, the string registered as a command will be stored.

When the execution attribute is "Interstage":

This is a job workunit name.

When the execution attribute is "PowerAIM" [UNIX]:

This is a job workunit name.

When the execution attribute is "Normal" [UNIX]:

This is a command name.

Completion code:

This is a job completion code. Completion code for unexecuted jobs is 0.

For a job net with the Job Execution Control attribute, completion code may exceed 256.

Status:

The job status can be specified using one of the following seven strings:

normal_end:

Indicates completed.

pseudo-normal:

Indicates pseudo-normal completion.

abnormal_end:

Indicates abended.

canceled:

Indicates canceled.

not_executed:

Indicates the job was not started.

paused:

Indicates the job has been paused.

disabled:

Indicates the job has been disabled.

The temporary file is created in the following directories:

[Windows]

In the work directory under the database directory of Jobscheduler

The name of the file is created in NEXnnn.tmp (nnn is appended by the system) format.

[Solaris, HP-UX]

/var/tmp directory

[AIX, Linux]

/tmp directory

Since the temporary file is not deleted on the Jobscheduler side, you must delete it from the exit program.

If the exit is not registered, the temporary file is not created.

About the Job in Passed Status

If the execution condition of the following job is specified in Condition concerning completion code of preceding job, the job that ends with the Passed status will be processed as follows:

- The job status (waiting /disabled/paused) before being passed, will be passed as the job information.

- The completion code is 0.

A job that is waiting using the OR condition and is then canceled because of the OR condition of the following job will be processed as follows:

- The job status will be "canceled" regardless of whether the job was actually started.

- The completion code is 248 or 249.

Exit Program Location

Windows	Systemwalker Operation Manager installation directory\MpWalker.JM\bin
Solaris Linux	/opt/FJSVJOBSC/bin

HP-UX	/opt/FHPJOBSCH/bin
AIX	/usr/FAIXJOBSC/bin

Cautions

- Prevent time-consuming process because it may cause delay of execution for job nets and groups.

- The execution user at the exit is the logon account of Jobscheduler service. [Windows]

- The execution user at the exit is a root account. [UNIX]

Program Example

Below is an example program for the job net abended extended exit:

[Windows]

```c
/* When a job net abnormal completion occurs, sends the following messages to "MANAGEMENT"
   server.
                Job net name (group name) is abended !! code = completion code*/
#include <stdio.h>
#include <process.h>

main(int argc, char *argv[])
{
    char cmdline[256];

    sprintf(cmdline,
                "NET SEND MANAGEMENT %s(%s) is abended !! code = %s",
                argv[2],      //job net name
                argv[4],      //group name
                argv[5]);     //completion code

    system(cmdline);

    unlink(argv[9]);    // temp file is deleted

    exit(0);
}
```

[UNIX]

```c
/*
  Example program for the job net abended extended exit (for UNIX) which outputs a job
  net abnormal completion message with a group name, job net name, and completion code
  on the console.
*/

#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <syslog.h>

main(int argc, char **argv)
{
    openlog ("jobsch", LOG_CONS, LOG_USER);
    syslog (LOG_WARNING,
        "WARNING: jobnet abnormal end. project=%s, group_name=%s, net_name=%s, net_comment=
%s,
 net_code=%s,
        argv[1], argv[4], argv[2], argv[3], argv[5]);
    closelog();
```

```
    unlink (argv[9]);
    exit(0);
}
```

17.3 Job End Exit

This section explains the exit program called when a job registered on Jobscheduler ends.

[Windows]

The system searches for the jobexit.bat file first, and then in jobexit.exe as the exit program. The exit program found in the search will be used. If jobexit.bat is found, jobsexit.exe will not be called up even if it exists.

If you want to use this exit to perform custom processing, create an exit program with the same name and store it in the following directory:

<*Systemwalker installation directory*>\MpWalker.JM\bin

[UNIX]

The system searches for and calls the jobsch.job.exit file as the exit program.

If you want to use this exit to perform custom processing, create an exit program with the same name and store it in the following directory:

Solaris/Linux:

/opt/FJSVJOBSC/bin

HP-UX:

/opt/FHPJOBSCH/bin

AIX:

/usr/FAIXJOBSC/bin

Appropriate execution privileges are required to execute the exit program. If relevant privileges are not set, the exit program will not be called.

Parameters Passed to the Exit Program

The parameters passed to the exit program are as follows:

argc :

Number of parameters (5) [Note1]

argv[0] :

Exit program name

argv[1] :

Name of project where the job net is registered

argv[2] :

Job net name

argv[3] :

Job net comment

argv[4] :

Temporary file name

argv[5] :

Subsystem number (character string) (Note2)

Note1:

In Systemwalker Operation Manager EE, the number of parameters is six.

Note2:

This parameter is used in Systemwalker Operation Manager EE.

In the temporary file, job information is stored as follows:

Job name
Completion code
Status

The following explanations describe each of the types of information shown above:

Job name:

Depending on execution attributes, the following contents are stored.

If a child job net is registered as a job, the name of the child job net will be stored.

If a linked job net is registered as a job, the name of the linked job net will be stored.

When the execution attribute is "Job Execution Control":

This is a job name. If the job name is omitted, the string registered as a command will be stored.

When the execution attribute is "Interstage":

This is a job workunit name.

When the execution attribute is "PowerAIM" [UNIX]:

This is a job workunit name.

When the execution attribute is "Normal" [UNIX]:

This is a command name.

Completion code

This is a job completion code (0 to 256 characters).

Status:

The job status is one of the following four strings.

normal_end:

Indicates completed.

pseudo-normal:

Indicates pseudo-normal completion.

abnormal_end:

Indicates abended.

canceled:

Indicates canceled.

The temporary file is created in the following directories:

[Windows]

In the work directory under the database directory of Jobscheduler

The name of the file is created in JOBnnn.tmp (nnn is appended by the system) format.

[Solaris, HP-UX]

/var/tmp directory

[AIX, Linux]

/tmp directory

Since the temporary file is not deleted on the Jobscheduler side, you must delete it from the exit program.

If the exit is not registered, the temporary file is not created.

About the Job in Passed Status

If the execution condition of the following job is specified in Condition concerning completion code of preceding job, the job end exit will not be activated on a job that ends with the Passed status.

A job that is waiting using the OR condition and is then canceled because of the OR condition of the following job will be processed as follows:

- The status of a job that is started and then canceled will be "canceled", and the completion code is 248 or 249.

- If a job is canceled without being started, its job end exit is not started.

Cautions

- Prevent time-consuming process because it may cause delay of execution for job nets and groups.

- The execution user at the exit is the logon account of Jobscheduler service. [Windows]

- The execution user at the exit is a root account. [UNIX]

Program Example

Below is an example program for the job end exit:

[Windows]

```
/* When a job ends, sends the following messages to "MANAGEMENT" server.
                Job name is ended !! code = completion code
*/
#include <stdio.h>
#include <string.h>
#include <process.h>

main(int argc, char *argv[])
{
    FILE *fp;
    char jobname[100];
    char jobcode[100];
    char cmdline[256];

    fp = fopen(argv[4], "r");
    fgets(jobname, 99, fp);
    jobname[strlen(jobname)] = '\0';
    fgets(jobcode, 99, fp);
    jobcode[strlen(jobcode)] = '\0';
    fclose(fp);

    sprintf(cmdline,
            "NET SEND MANAGEMENT %s is ended !! code = %s",
            jobname,    //job name
            jobcode);   //completion code
```

```
    system(cmdline);

    unlink(argv[4]);    // temp file is deleted
    exit(0);
}
```

[Solaris]

```
/*
  Example program for the job end exit (for UNIX) which outputs a job completion message
  with a job name and completion code on the console
*/

#include <syslog.h>
#include <stdio.h>

main(int argc, char **argv)
{
    FILE            *fp;
    char             jobname[100];
    char             jobcode[100];

    memset (jobname, 0x00, sizeof (jobname));
    memset (jobcode, 0x00, sizeof (jobcode));

    fp = fopen(argv[4], "r");
    fgets(jobname, sizeof (jobname), fp);
    jobname[strlen(jobname)-1] = '\0';
    fgets(jobcode, sizeof(jobcode), fp);
    jobcode[strlen(jobcode)-1] = '\0';
    fclose(fp);

    openlog ("jobsch", LOG_CONS, LOG_USER);
    syslog (LOG_WARNING,
        "WARNING: job end.  project=%s,  net_name=%s,  net_comment=%s,  job name=%s,
job_code=%s",
        argv[1], argv[2], argv[3], jobname, jobcode);
    closelog();

    unlink (argv[4]);
    exit(0);
}
```

17.4 Log File Switchover Exit

This section explains the exit program used when the Jobscheduler log files (jobdb1.log, jobdb2.log and jobdb3.log) are switched.

[Windows]

If the log file has changed, the system searches for the exit program in jobdbexit.bat first, and then in jobdbexit.exe. The exit program found will be used. If the jobdbexit.bat is found, jobdbexit.exe will not be called up even if it exists.

If you want to use this exit to perform custom processing, create an exit program with the same name and store it in the following directory:

> *<Systemwalker installation directory>*\MpWalker.JM\bin

[UNIX]

If the log file has changed, the system searches for and calls the jobdb.exit file as the exit program.

If you want to use this exit to perform custom processing, create an exit program with the same name and store it in the following directory:

Solaris/Linux:

```
/opt/FJSVJOBSC/bin
```

HP-UX:

```
/opt/FHPJOBSCH/bin
```

AIX:

```
/usr/FAIXJOBSC/bin
```

Appropriate execution privileges are required to execute the exit program. If relevant privileges are not set, the exit program will not be called.

Parameters Passed to the Exit Program

The parameters passed to the exit program are as follows:

argc :

Number of parameters (2) [Note1]

argv[0] :

Exit program name

argv[1] :

Log file name before switching

argv[2] :

Subsystem number (character string) [Note2]

Note1:

In Systemwalker Operation Manager EE, the number of parameters is three.

Note2:

This parameter is used in Systemwalker Operation Manager EE.

Cautions

- Prevent time-consuming process because it may cause delay of execution for job nets and groups.

- The execution user at the exit is the logon account of Jobscheduler service. [Windows]

- The execution user at the exit is a root account. [UNIX]

Program Example

Below is an example program for the log file switchover exit:

[Windows]

```
/*
  Generates a message event called DB-CHANGE
*/

#include <process.h>

main()
{
```

```
        system("jobschmsgevent DB-CHANGE");

        exit(0);
}
```

[Solaris]

```
/*
  Example program for the log file switchover exit (for UNIX) which outputs a log
  switchover message on the console
*/

#include <syslog.h>
#include <stdio.h>

main(int argc, char **argv)
{
    openlog("jobsch", LOG_CONS, LOG_USER);
    syslog(LOG_WARNING, "WARNING: Log file was switched. file=%s", argv[1]);
    closelog();

    exit(0);
}
```

17.5 Day Change Time Arrival Exit

This explains the exit program called up when the day change time set in the calendar function arrives.

[Windows]

If the day change time arrives, the system searches for the exit program in the timeexit.bat first and then in the timeexit.exe. The exit program found will be used. If the timeexit.bat is found, timeexit.exe will not be called up even if it exists.

If you want to use this exit to perform custom processing, create an exit program with the same name and store it in the following directory:

<Systemwalker installation directory>\MpWalker.JM\bin

[UNIX]

If the day change time arrives, the system searches for and calls the jobsch.time.exit file as the exit program.

If you want to use this exit to perform custom processing, create an exit program with the same name and store it in the following directory:

Solaris/Linux:

/opt/FJSVJOBSC/bin

HP-UX:

/opt/FHPJOBSCH/bin

AIX:

/usr/FAIXJOBSC/bin

Appropriate execution privileges are required to execute the exit program. If relevant privileges are not set, the exit program will not be called.

Parameters Passed to the Exit Program

The parameters passed to the exit program are shown below.

> **argc:**
>
> Number of parameters (1) (Note1)
>
> **argv[0]:**
>
> Exit program name
>
> **argv[1]:**
>
> Subsystem number (character string) (Note2)

Note1:

In Systemwalker Operation Manager EE, the number of parameters is two.

Note2:

This parameter is used in Systemwalker Operation Manager EE.

Cautions

- Prevent time-consuming process because it may cause delay of execution for job nets and groups.

- The execution user at the exit is the logon account of the Jobscheduler service. [Windows]

- The execution user at the exit is a root account. [UNIX]

Program Example

Below is an example program for the day change time arrival exit:

[Windows]

```
/*
  Generates a message event called JOBSCH-TIME
*/

#include <process.h>

main()
{
    system("jobschmsgevent JOBSCH-TIME");

    exit(0);
}
```

[Solaris]

```
/*
  Example program for the day change time arrival exit (for UNIX),
  which outputs a day change time arrival message on the console
*/
#include <syslog.h>
#include <stdio.h>

main()
{
    openlog("jobsch", LOG_CONS, LOG_USER);
    syslog(LOG_WARNING, "WARNING: Day change time has come.");
    closelog();

    exit(0);
}
```

17.6 Completed/Abended Shutdown Exit [Windows]

When the execution of the shutdown job net is completed, Jobscheduler calls up the following shutdown exit programs according to the execution result (Completed, Pseudo-normal, or Abended) of the shutdown job net.

If you want to run some processes in accordance with the result of the shutdown job net, create an exit program with the same name and store it in the following directory:

> *<Systemwalker installation directory>*\MpWalker.JM\bin

At the installation, batch files (jobschendexit.bat and jobschnoendexit.bat) are provided.

Case of Completed or Pseudo-Normal

The system searches for the jobschendexit.bat file first, and then jobschendexit.exe file as the exit program. The exit program found in the search will be used. If jobschendexit.bat is found, jobschendexit.exe will not be called up even if it exists.

The jobschendexit.bat file provided at the installation stops the services of Jobscheduler and Job Execution Control and calls up the queuing completion notification command (f3crheet.exe) with queuing completion notification "system."

In order to shut down your machine, define queuing completion notification "system" in the [Queuing Completion Notification Definition] dialog box.

See the *Systemwalker Operation Manager Online Help* for this dialog box.

Case of Abended

The system searches for the jobschnoendexit.bat file first, and then the jobschnoendexit.exe file as the exit program. The exit program found in the search will be used. If jobschnoendexit.bat is found, jobschnoendexit.exe will not be called up even if it exists.

In jobschnoendexit.bat provided at the installation, no processing will be executed.

Therefore, when the shutdown job net ends abnormally, the shutdown process will not be executed.

Cautions

- The execution user at the exit is the logon account of Jobscheduler service.

EE

- In case of operating multiple subsystems, a subsystem number is passed as an argument. Be sure to change the shutdown exit in matching with the operation type.

EE

Usage example in operating subsystem

In operating subsystem, the system should be shut down after terminating all of subsystems. An example of the shutdown exit is shown below.

The parameter to be passed to the exit program is %6 (subsystem number).

```
echo off

echo shutdown job net exit routine start

if "%6" == "0" goto zero
if not "%6" == "0" goto notzero

:zero
rem --- Systemwalker MpJobsch ---
net stop "Fujitsu MpWalker MpJobsch"
rem --- Systemwalker MpMjes ---
mjstop -sys %6
net stop "Fujitsu MpWalker MpMjes"
goto done

:notzero
```

```
rem --- Systemwalker MpJobsch ---
net stop "Fujitsu MpWalker MpJobsch%6"
rem --- Systemwalker MpMjes ---
mjstop -sys %6
net stop "Fujitsu MpWalker MpMjes%6"


:done
f3crheet.exe system%6
```

If you define the setting like above example, the queuing completion notification to be specified with the f3crheet command must be defined in the [Queuing Completion Notification Definition] dialog box beforehand. As a result, the shutdown process will be postponed until the queuing completion notification specified with the f3crheet command is output.

17.7 Job Net Start Delay Exit

This section explains the exit program used if the job net registered in Jobscheduler is not activated when the scheduled time arrives.

[Windows]

If the job net is not activated when the scheduled time arrives, the system searches for the startlateexit.bat file first, and then the startlateexit.exe file as the exit program. The exit program found will be used. If the startlateexit.bat file is found, startlateexit.exe will not be called up even if it exists.

For a job net whose start time is set in [Scheduled start time] with [Once in a day] checked in [Start only when message event has occurred], if no message event occurs when the scheduled startup time arrives for the job net or it is in the disabled status at the scheduled startup time, this exit program will be also called up.

If you want to use this exit to perform custom processing, create an exit program with the same name and store it in the following directory:

<*Systemwalker installation directory*>\MpWalker.JM\bin

[UNIX]

If the job net is not activated when the scheduled time arrives, the system searches for and calls the startlate.exit file as the exit program.

For a job net whose start time is set in [Scheduled start time] with [Once in a day] checked in [Start only when message event has occurred], if no message event occurs when the scheduled startup time arrives for the job net or it is in the disabled status at the scheduled startup time, this exit program will be also called up.

If you want to use this exit to perform custom processing, create an exit program with the same name and store it in the following directory:

Solaris/Linux:

/opt/FJSVJOBSC/bin

HP-UX:

/opt/FHPJOBSCH/bin

AIX:

/usr/FAIXJOBSC/bin

Appropriate execution privileges are required to execute the exit program. If relevant privileges are not set, the exit program will not be called.

Parameters Passed to the Exit Program

The parameters passed to the exit program are as follows:

---

argc :

Number of parameters (4) (Note1)

argv[0] :

Exit program name

argv[1] :

Name of project where the job net is registered

argv[2] :

Job net name

argv[3] :

Job net comment

argv[4] :

Subsystem number (character string) (Note2)

Note1:

In Systemwalker Operation Manager EE, the number of parameters is five.

Note2:

This parameter is used in Systemwalker Operation Manager EE.

Cautions

- Prevent time-consuming process because it may cause delay of execution for job nets and groups.

- The execution user at the exit is the logon account of the Jobscheduler service. [Windows]

- The execution user at the exit is a root account. [UNIX]

Program Example

Below is an example program for the job net start delay exit:

[Windows]

```
/* When the activation of the job net delays, sends the following messages to "MANAGEMENT"
   server.
             Job net name is delayed
*/
#include <stdio.h>
#include <process.h>

main(int argc, char *argv[])
{
        char cmdline[256];

        sprintf(cmdline,
                                "NET SEND MANAGEMENT %s is delayed",
                                argv[2]);         //job net name

        system(cmdline);

        exit (0);
}
```

```
/*
  Example program for the job net start delay exit (for UNIX), which outputs a job
  net start delay message with a project name and job net name to the console.
*/

#include <syslog.h>
#include <stdio.h>

main(int argc, char **argv)
{
    openlog("jobsch", LOG_CONS, LOG_USER);
    syslog(LOG_WARNING,
           "WARNING: %s %s is delayed.",
           argv[1], argv[2]);
    closelog();

    exit(0);
}
```

17.8 Job Net End Delay Exit

This explains the exit program used if the job net registered in Jobscheduler does not end when the estimated end time arrives.

[Windows]

If the job net does not end when the estimated end time arrives, the system searches the endlateexit.bat file first, and then in the endlateexit.exe file as the exit program. The exit program found will be used. If the endlateexit.bat file is found, then endlateexit.exe file will not be called up even if it exists.

If you want to use this exit to perform custom processing, create an exit program with the same name and store it in the following directory:

<Systemwalker installation directory>\MpWalker.JM\bin

[UNIX]

If the job net is not ended when the estimated end time arrives, the system searches for and calls the endlate.exit file as the exit program.

If you want to use this exit to perform custom processing, create an exit program with the same name and store it in the following directory:

Solaris/Linux:

/opt/FJSVJOBSC/bin

HP-UX:

/opt/FHPJOBSCH/bin

AIX:

/usr/FAIXJOBSC/bin

Appropriate execution privileges are required to execute the exit program. If relevant privileges are not set, the exit program will not be called.

Parameters Passed to the Exit Program

The parameters passed to the exit program are as follows:

argc :

    Number of parameters (5) (Note1)

argv[0] :

    Exit program name

argv[1] :

    Name of project where the job net is registered

argv[2] :

    Job net name

argv[3] :

    Job net comment

argv[4] :

    Job net status

argv[5] :

    Subsystem number (character string) (Note2)

Note1:

    In Systemwalker Operation Manager EE, the number of parameters is six.

Note2:

    This parameter is used in Systemwalker Operation Manager EE.

The character strings stored in the fifth parameter "job net status" are as follow:

not_executed:

    The job net is in Waiting.

executing:

    The job net is in Executing.

warning:

    The job net is in Warning.

Cautions

- Prevent time-consuming process because it may cause delay of execution for job nets and groups.

- The execution user at the exit is the logon account of Jobscheduler service. [Windows]

- The execution user at the exit is a root account. [UNIX]

Program Example

Below is an example program for the job net end delay exit:

[Windows]

```
/* When the job net end delays, sends the following messages to "MANAGEMENT" server
              Job net name is delayed
*/
#include <stdio.h>
#include <process.h>
```

```
main(int argc, char *argv[])
{
        char cmdline[256];

        sprintf(cmdline,
                      "NET SEND MANAGEMENT %s is delayed",
                      argv[2]);        //job net name

        system(cmdline);

        exit (0);
}
```

[Solaris]

```
/*
  Example program for the job net end delay exit (for UNIX) which outputs a job net end
  delay message with a project name and job net name on the console.
*/
#include <syslog.h>
#include <stdio.h>

main(int argc, char **argv)
{
    openlog("jobsch", LOG_CONS, LOG_USER);
    syslog(LOG_WARNING,
            "WARNING: %s %s is delayed.",
            argv[1], argv[2]);
    closelog();

    exit(0);
}
```

17.9 Unsupported Short Mail Message Conversion Exit [Windows]

When you want to send a message to a short mail, you must convert the message text into code.

If you are using a short mail that is not supported by Systemwalker Centric Manager, you will have to create an exit (DLL) for converting the message in advance.

Save the new DLL as "f3crhxpc.dll" in the following directory:

<Systemwalker installation directory>\MpWalker.JM\bin

Interface with Exit Program

Use the following format for calling up the function for converting the message.

```
int f3crhx_convert(char *Msg_Text,char *MsgNo,int MsgNoSize,char
*Vender)
```

Msg_Text:

This sets the text (character string for the message text to be converted) that is to be sent to the short mail.

MsgNo:

The storage area for the conversion result has been prepared. Specify it on exit. Convert the text stored in Msg_Text to a character string of dialable number and specify. Specify it in the range for the length set in MsgNoSize, and specify "\0" at the end of the character string.

Return Values

Use the function's return value to post the conversion result of the message.

Normal end:

Posts an integer above 0. (Sends short mail.)

Abnormal end:

Posts an integer less than 0. (Does not send short mail.)

Cautions

- When you create this DLL, you need to register your Short Mail company in the **Short Mail** sheet of the **Action Environment Setup** dialog box.

- If this DLL ends abnormally, Systemwalker service will stop.

- If the control is not returned from this DLL, you will not be able to send short mail.

Program Example

Below is an example program for the unsupported short mail message conversion exit:

```
int f3crhx_convert(char *MsgText,char *MsgNo,int MsgNoSize,char *Vender)
{
    int i ;
    int NoLen = 0 ;

    if ( strcmp(Vender,"ven01") != 0 ) {      //  unsupported company
        return -1 ;

    }

    //  message conversion process
    for ( i = 0 ; i < strlen(MsgText) ; i++ ) {
        /* store the conversion result in MsgNo */
        /* store the number of strings stored in MsgNo in NoLen */
    }
    MsgNo[NoLen] = '\0' ;                      //  setting of end code

    return 0 ;                                 //  normal end
}
```

17.10 On-Demand Job End Exit

This section explains the exit program called when an on-demand job ends.

[Windows]

The system searches for the exit program first in mjesexit.bat and then in mjesexit.exe, and calls the exit program found. If mjesexit.bat is found, mjesexit.exe will not be called even if it exists.

If you want to use this exit to perform custom processing, create an exit program with the same name and store it in the following directory:

<Systemwalker installation directory>\ MpWalker.JM\bin

[UNIX]

The system searches for and calls the exit program in the mjes.job.exit.

If you want to use this exit to perform custom processing, create an exit program with the same name and store it in the following directory:

Solaris/Linux:

```
/opt/FJSVMJS/usr/lib/mjes
```

HP-UX:

```
/opt/FHPMJS/usr/lib/mjes
```

AIX:

```
/opt/FAIXMJS/usr/lib/mjes
```

Appropriate execution privileges are required to execute the exit program. If relevant privileges are not set, the exit program will not be called.

Parameters Passed to the Exit Program

The parameters passed to the exit program are as follows:

```
argc : Number of parameters (9)
argv[0] : Exit program name
argv[1] : Job name
argv[2] : Job number (character string)
argv[3] : Host name of the execution server where the job was executed
argv[4] : Queue name where the job was executed
argv[5] : Comment specified in the job
argv[6] : End code of the job (character string)
argv[7] : File name of the standard output file of the job
argv[8] : File name of the standard error output file of the job
argv[9] : Sub-system number (character string) (Note 1)
```

Note 1

This parameter is used in Systemwalker Operation Manager EE. 0 is set in Systemwalker Operation Manager SE.

Cautions

- Under UNIX, the on-demand job end exit is called with job owner privileges. If the -cu option was not used to specify the client user name, the user that starts the qsub command becomes the job owner. Under Windows, the on-demand job end exit is called with the privileges of the service startup account.

- The exit program is started as a new process and its end is not waited for. Even if this exit process is not completed, the relevant job is seen to have been completed and the next job in the queue is started.

- Even if there is an error in the execution of the exit process, Systemwalker Operation Manager will do nothing.

- If an on-demand job end exit exists but is not called for some reason, a message will be output to the SYSLOG/event log.

Program Example

The following are some examples of on-demand job exit programs:

[Windows]

```
/*
    If the job completion code is other than "0", the following message
```

```
     is sent to the "MANAGEMENT" server:
       Job name is ended !! code = completion code
 */
#include <stdio.h>
#include <string.h>
#include <process.h>

main(int argc, char *argv[])
{
    char cmdline[256];
    int job_code = 0;

    job_code = atoi(argv[6]);

    if (job_code != 0) {
        sprintf(cmdline,
                "NET SEND MANAGEMENT %s is ended !! code = %s",
                argv[1],  //job name
                argv[6]); //completion code

        system(cmdline);
    }

    exit(0);
}
```

[Solaris]

```
/*
   If the job completion code is other than "0", a message is output
   to the console. The job name, job number, and the job
   completion code are output.
 */

#include <syslog.h>
#include <stdio.h>

main(int argc, char **argv)
{
    int job_code = 0;

    job_code = atoi(argv[6]);

    if (job_code != 0) {
        openlog("mjes.job.exit", LOG_CONS, LOG_USER);
        syslog(LOG_WARNING, "WARNING: job end. job_name=%s, job_number=%s,
                            job_code=%s", argv[1], argv[2], argv[6]);
        closelog();
    }
    exit(0);
}
```

Chapter 18 File Reference

This chapter explains the files provided by Systemwalker Operation Manager.

18.1 Log Files

This section explains the log files that are output by Systemwalker Operation Manager.

Log file types and content

In Systemwalker Operation Manager, the following log files are output:

Log files that are output by default	Log files that are not output by default
- Jobscheduler logs - Job execution history logs - Audit logs	- Operation record information logs

Each file is explained below.

Jobscheduler logs

Log content	Job net and job execution history (start time, end time, and completion code)
Log directory	Under the Jobscheduler database directories given below, depending on the operating system used: Windows: <*Systemwalker Operation Manager installation directory*>\MpWalker.JM\mpjobsch\jobdb Linux: /var/opt/FJSVJOBSC Solaris: /var/opt/FJSVJOBSC HP-UX: /opt/FHPJOBSCH/db AIX: /usr/FAIXJOBSC/db Note: In case of multi-subsystems, the logs are stored under the JOBDB*n* directory. [*n*: Subsystem number. A number from 1 to 9 is used.]
File name	jobdb1.log, jobdb2.log, and jobdb3.log
Log file size or retention period	1 - 99 MB (the default is 3 MB) Note: This can be configured in **Log file size** in the **Database** sheet of the **Define Jobscheduler Startup Parameters** window.
Number of files for cyclic use	There are three files (jobdb1.log, jobdb2.log, and jobdb3.log). The number of files cannot be changed. The log files are rotated cyclically according to the file size, as in jobdb1.log -> jobdb2.log -> jobdb3.log -> jobdb1.log. They are not retained based on the number of days.

Job execution history logs

Log content	Job submission, start, end, deletion date, time, and job completion code
Log directory	The default directory is the following: [Windows] <*Systemwalker Operation Manager installation directory*>\MpWalker.JM\mpmjessv\hist [UNIX] /var/spool/mjes/hist

	Note: This can be changed in the **Logging** sheet of the **Define Operating Information** window. Note: In case of multi-subsystems, the log of subsystem 0 is stored under the above-mentioned directory, while the logs of other subsystems from 1 to 9 are stored under the directory with the corresponding subsystem number *n*.
File name	*yyyymmdd*.log (<*date when the log file was created*>.log)
File format	This contains binary attributes, therefore the file cannot be referenced directly. Use the mjsview command to reference the file.
Log file size or retention period	1 - 31 days (the default is 7 days) Note: This can be configured in the **Logging** sheet of the **Define Operating Information** window.
Handling when the retention period is exceeded	Files that have exceeded the specified number of days are deleted.

Operation record information logs

(When configured in "Systemwalker Operation Manager Environment Setup" >> "Operation information" window)

Log type	Operation information about the executed job and job net
Log directory	No default directory Note: This can be configured in the **Logging** sheet of the **Define Operating Information** window. Note: In case of multi-subsystems, the log of subsystem 0 is stored under the configured directory, while the logs of other subsystems from 1 to 9 are stored under the directory with the corresponding subsystem number *n*.
File name	<*yyyymmdd*>.csv (<*date when the operation record information file was created*>.csv)
File format	CSV format
Log file size or retention period	1 - 31 days Note: This can be configured in the **Logging** sheet of the **Define Operating Information** window.
Handling when the retention period is exceeded	Files that have exceeded the specified number of days are deleted.

Audit logs

Log type	Record information for operations from the Systemwalker Operation Manager client
Log directory	The default directory is the following: [Windows] <*Systemwalker installation directory*>\MpWalker.JM\mpcmtool\audit [UNIX] /var/opt/FJSVftlo/audit Note: This can be changed using the mpsetlogsend_omgr command.
File name	mp_omgr_audit<*YYMMDD*>.log (*YYMMDD* is the date when the operation was recorded)
File format	Text (CSV) format

Log file size or retention period	1 - 99 days or "unlimit" (the default is 31 days) Note: This can be configured using the mpsetlogsend_omgr command.
Handling when the retention period is exceeded	Files that have exceeded the specified number of days are deleted. However, they are not deleted if "unlimit" has been specified.

18.1.1 Job Execution Control Operation Record Files

This section explains the operation record file output by Job Execution Control.

What is the operation record files?

The operation record file is the file in which the operation information about the executed jobs and job nets is logged in CSV format. To collect operation records, designate so in the [Logging] sheet of the [Define Operating Information] window.

The following three types of records are output in the CSV format to the operation record file.

- Service Record

- Job Record

- Step Record

For the information output to these records, see "18.1.1.1 Service Record", "18.1.1.2 Job Record", and "18.1.1.3 Step Record".

📙 Note

. . . .

Referencing Operation Record File

An attempt to write the operation record file being opened in an application such as Microsoft Excel when the Job Execution Control service is in execution may result in a failure.

. . . .

📙 Note

. . . .

Record outputs in network jobs

Record outputs in network jobs are shown below.

For JCL network jobs:

The execution host name is output to a step record of the step completion event.

For non-JCL network jobs:

The execution host name is output to a job record of the step completion event.

. . . .

📙 Note

. . . .

Queue name and job name

Following are notes on the information output to queue name and job name.

Queue name

The queue names output to the job record of a job submission event are identical to those output to the job record of a job completion event.

Job name

If the JCL is submitted from a job folder, the qsub command or an environment not under control of folders, a job file name without its path and extension is regarded as the job name. The job names written in jobstart statements are ignored.

∙∙∙

Output Conventions

The conventions for the operation record outputs are as follows.

- A line break is output for each record and one line is treated as one record.

- A comma "," is output to separate data.

- If the data type is string, it is enclosed in double quotes (""). When double quotes already exist within the string, two double quotes are output consecutively.

- If the data type is numeric, it is not enclosed in double quotes but is output as it is.

- Date and time data is output as strings.

- The maximum number of character strings for one operation record is limited up to 2048 bytes.

If the output data is date or time, actual examples are given.

18.1.1.1 Service Record

This record is output at the startup and stop of the Systemwalker MpMjes service.

No.	Data Item	Value (Fixed/Variable)	Output Data	Data Format
1	Record creation date	Variable	"01/01/2000"	String
2	Record creation time	Variable	"01:01:01"	String
3	Record type	Fixed	1	Numeric
4	Host name	Variable	"Host name"	String
5	Version information	Variable	"Version/level" (Note 1)	String
6	Startup or Stop	Variable	"START": Startup "STOP": Stop	String
7	Startup mode (Note 2)	Variable	"COLD": Initialization mode "WARM": Continuation mode "RECOVERY": Recovery mode	String
8	Operation mode (Note 2)	Variable	"CANCEL": Disable continuation mode "CONTINUE": Enable continuation mode	String

Note 1:

The current version/level name in use is output.

Note 2:

The startup and operation modes are output only when Startup or Stop is set to "START."

18.1.1.2 Job Record

The job record consists of a fixed part and a variable part. The fixed and variable parts are merged and output as one record.

The variable part changes according to the event type associated with the job. There are the following types of variable parts:

Job submission event:

Output when the job is submitted.

Job start event:

Output when the job execution starts.

Job completion event:

Output when the job execution completes.

Job delete event:

Output when the job is deleted from the server where Systemwalker Operation Manager was installed.

Job restart event:

Output after job start event is output when the job execution is requested to another host since the job execution request failed.

Fixed parts

No.	Data Item	Value (Fixed/Variable)	Output Data	Data Format
1	Record creation date	Variable	"01/01/2000"	String
2	Record creation time	Variable	"01:01:01"	String
3	Record type	Fixed	2	Numeric
4	Host name	Variable	"Host name"	String
5	Version information	Variable	"Version/level" (Note 1)	String
6	Job name	Variable	"Job name"	String
7	Job number	Variable	NN	Numeric
8	User name	Variable	"User name"	String
9	Event type	Variable	0: Job submission event 1: Job start event 2: Job completion event 3: Job delete event 4. Job restart event	Numeric
10	Submission source host name	Variable	"Submission source host name"	String
11	Job type	Variable	0: Local job (Note 2) 1: Network job (Note 2) 2: Load-sharing job (Note 3)	Numeric
12	Job attribute	Variable	0: General job 2: NJS job 3: GLOVIA job	Numeric

No.	Data Item	Value (Fixed/Variable)	Output Data	Data Format
			4 :GLOVIA-BO	
			7: Online job	

Note 1:

The current version/level name in use is output.

Note 2:

When a job is executed locally under distributed execution, "0: Local job" is output and when executed on another host, "1: Network job" is output.

Note 3:

Load-balancing of compatibility with older version is output only for Windows.

Variable parts: Job submission event

No.	Data Item	Value (Fixed/Variable)	Output Data	Data Format
1	Job submission date	Variable	"01/01/2000"	String
2	Job submission time	Variable	"01:01:00"	String
3	Client host name	Variable	"Client host name"	String
4	Queue name	Variable	"Queue name"	String
5	Command name (Note 1)	Variable	"Command argument"	String
6	Project name	Variable	"SampleProject"	String
7	Job net name	Variable	"Sample"	String

Note 1:

The name and argument of command submitted as a job is output as a command name.

Variable parts: Job start event

No.	Data Item	Value (Fixed/Variable)	Output Data	Data Format
1	Job start date	Variable	"01/01/2000"	String
2	Job start time	Variable	"01:01:01"	String

Variable parts: Job completion event

No.	Data Item	Value (Fixed/Variable)	Output Data	Data Format
1	Job completion date	Variable	"01/01/2000"	String
2	Job completion time	Variable	"01:01:02"	String
3	Execution host name	Variable	"Execution host name"	String
4	Job completion code (Note 1)	Variable	NN	Numeric

No.	Data Item	Value (Fixed/Variable)	Output Data	Data Format
5	Number of job steps	Variable	NN	Numeric
6	Queue name	Variable	"Queue name"	String
7	Elapsed time for execution (Note 2)	Variable	NN	Numeric
8	Time lapse limit (Notes 2 and 3)	Variable	NN	Numeric
9	Estimated processing time (Notes 2 and 3)	Variable	NN	Numeric
10	The ID of the first process of the job (Note 4)	Variable	NN	Numeric

Note 1:

When multiple steps have been executed, the largest completion code among those for the completed steps is output.

Note 2:

Output in seconds.

Note 3:

A 0 is output when it has not been specified when a queue was defined or a job was submitted.

Note 4:

-1 is output for network job and JCL.

Variable parts: Job delete event

No.	Data Item	Value (Fixed/Variable)	Output Data	Data Format
1	Job delete date	Variable	"01/01/2000"	String
2	Job delete time	Variable	"01:01:03"	String

Variable parts: Job restart event

No.	Data Item	Value (Fixed/Variable)	Output Data	Data Format
1	Job restart date	Variable	"01/01/2000"	String
2	Job restart time	Variable	"01:01:03"	String

18.1.1.3 Step Record

The step record consists of a fixed part and a variable part. The fixed and variable parts are merged and output as one record.

The variable part changes according to the event type associated with the step. There are the following types of variable parts:

Step start event:

Output when a JCL file step is started.

Step completion event:

Output when a JCL file step completes.

Fixed parts

No.	Data Item	Value (Fixed/Variable)	Output Data	Data Format
1	Record creation date	Variable	"01/01/2000"	String
2	Record creation time	Variable	"01:01:01"	String
3	Record type	Fixed	3	Numeric
4	Host name	Variable	"Host name"	String
5	Version information	Variable	"Version/level" (Note 1)	String
6	Job name	Variable	"Job name"	String
7	Job number	Variable	NN	Numeric
8	Event type	Variable	0: Step start event 1: Step completion event	Numeric
9	Step name	Variable	"Step name"	String
10	Step number	Variable	NN	Numeric

Note 1:

The current version/level name in use is output.

Variable parts: Step start event

No.	Data Item	Value (Fixed/Variable)	Output Data	Data Format
1	Step start date	Variable	"01/01/2000"	String
2	Step start time	Variable	"01:01:01"	String

Variable parts: Step completion event

No.	Data Item	Value (Fixed/Variable)	Output Data	Data Format
1	Step completion date	Variable	"01/01/2000"	String
2	Step completion time	Variable	"01:01:02"	String
3	Execution host name	Variable	"Execution host name"	String
4	Step completion code	Variable	NN	Numeric

18.1.2 Audit log Files

This section explains the audit log files provided by Systemwalker Operation Manager.

18.1.2.1 Audit Log Files

mp_omgr_auditYYMMDD.log

Description

This file records changes to definitions relating to Systemwalker Operation Manager jobs and histories of operations on jobs. This file can be viewed by only operating system administrators.

File Location

The default settings used when Systemwalker Operation Manager is installed are as shown in the following table.

See "Defining Audit Log Output" in the *Systemwalker Operation Manager Installation Guide* for how to change the settings.

Windows	*Systemwalker installation directory*\MpWalker.JM\mpcmtool\audit
Solaris Linux HP-UX AIX	/var/opt/FJSVftlo/audit

File Format

The following information is recorded as a single record per line. The file format is CSV format, and each item is separated by a comma.

Item	Maximum length (bytes)	Format	Remarks
Date	29	YYYY/MM/DD_HH:mm:SS.sss_[+|-]UTC	"_" indicates a single space. YYYY: Four-digit year MM: Two-digit month (01 to 12) DD: Two-digit day (01 to 31) HH: Two-digit hours (00 to 23) mm: Two-digit minutes (00 to 59) SS: Two-digit seconds (00 to 59) sss: Three-digit milliseconds (000 to 999) +|- UTC: Time difference as compared to UTC. The "+" or "-" will be followed by a four-digit number.
Operation location	1024	String indicating the location where the operation was performed. One of the following strings will be output. - IP address - Host name	
Execution host	1024	String indicating either the name of the host where the operation requested by a client was executed, or the name of the host where a command or API was executed	

Item	Maximum length (bytes)	Format	Remarks
Operator	255	The user ID used to log in to Systemwalker Operation Manager or the user ID used to log in to the operating system on the server. **[Windows]** The user name is output using either of the following formats: "*computer name|user ID*" or "*domain name|user ID*". Only the user ID is output when using the Systemwalker User Management function. Note that when multi-server monitoring is conducted using a multi-server monitoring client, the operator output to the monitored servers will differ depending on the user who has logged into the monitoring server. - When logged in as an administrator user: The user with administrator privileges who has logged onto the monitoring server will be output. - When logged in as a general user: The user connected to the monitored server will be output.	The string "server" will be output in the following situations - When policies are applied - When the Jobscheduler service/ daemon starts and stops - When the Operation Manager common infrastructure service/ daemon starts and stops - When the calendar service/ daemon starts and stops - When the automatic reflection function for calendars is used - When a server is turned off automatically with a power supply schedule - When calendar or system calendar information is updated when the year changes - When schedules or future schedules are executed - When schedule servers are added, edited or canceled - When virtual time settings for Master Schedule Management are made or canceled
Operation type	512	String that classifies the content of the operation that has been performed	The operation type will be output using the following format: `<operation>_<additional information>` See "List of Search Keywords for Audit Logs" in the *Systemwalker Operation Manager User's Guide* for details.
Operation target	4096	Information for identifying the operation target is output. If multiple items are output, each item will be separated by a semicolon.	See "List of Search Keywords for Audit Logs" in the *Systemwalker Operation Manager User's Guide* for details.
Operation content	4096	The content of the operation that has been performed is output.	If definitions have been changed using a command or API, or if job operations have been performed

Item	Maximum length (bytes)	Format	Remarks
		Example: "Stopped Job A"	using a command or API, the output will be as below. **Command execution:** Executed a command. (COMMAND =< command and parameter string>) **API execution:** Executed an API. (API =< API name>)
Execution result	1	B: BEGIN (operation started) S: SUCCESS (operation successful) F: FAIL (operation failed) C: CANCEL	"B" (for "BEGIN") is output when processing (for a command, etc.) will take a long time, in order to indicate that the processing has started. If a "BEGIN" log has been output, another log will be then output as a single line indicating the result of the processing (either "SUCCESS", "FAIL" or "CANCEL").
Component	256	The name of the component (function) operated on will be output.	The information that is output corresponds to function names as follows: MPCMTOOLO: Common tools MPFWSEC: ACL manager MPJMCAL: Calendar, Power Control MPJOBSCH: Jobscheduler MPJMSRV: Operation Manager common infrastructure MPMJES: Job Execution Control MPSTEM: Master Schedule Management MPNJSOSV: Task link
Additional information	1024	Supplementary detailed information may be output, such as the process ID when commands are executed, or the version of the client that has connected.	The following information is output in the authentication logs that are output as a pair with the log output when connected clients log in to the server: **If the connection is made from a client running V13.0.0 or V13.1.0:** CL_VERSION= V13.0 or V13.1 **If the connection is made from a client running V13.2.0 or later:** CL_VERSION=V13.2 or later
Reserved area	-	-	This area is not used.

[Additional information]

This is supplementary information for the operation type. The following table shows the main keywords.

Additional information	Description
CALENDAR	Calendar information
CARRYIEDOVERJOBNET	Carried over job net
DAEMON/SERVICE	Daemon/service
GROUP	Group
JOB	Job
JOBNET	Job net
LGMANAGE	Password management book
MESSAGE	Message event
OMGR_USER	Operation Manager user
PERMISSION	Access rights
PROJECT	Project
QUEUE	Queue
SCHEDULES	Schedule
SERVER	Server

Information

See "List of Search Keywords for Audit Logs" in the *Systemwalker Operation Manager User's Guide* for keywords to search audit logs.

Reference

mpsetlogsend_omgr

Examples

```
"2006/09/26 11:25:12.672
+0900","10.90.100.100","host1","user1","ADD_PROJECT","SUBSYSTEM=0;PROJECT=PRJ01;"," The
project is added.",S,"MPJOBSCH","PID=8045;",
"2006/09/26 11:30:41.835
+0900","10.90.100.100","host1","user1","ADD_JOBNET","SUBSYSTEM=0;PROJECT=PRJ01;JOBNET=JOBN
ET01;"," The job net is added.",S,"MPJOBSCH","PID=8045;",
"2006/09/26 11:30:45.390
+0900","10.90.100.100","host1","user1","START_JOBNET","SUBSYSTEM=0;PROJECT=PRJ01;JOBNET=JO
BNET01;"," The job net is started.",S,"MPJOBSCH","PID=8045;",
```

18.2 Message Event Occurrence History File

This section explains the history information file recorded when a message event occurs.

18.2.1 Message Event Occurrence History File

This file is recorded by Systemwalker Operation Manager when a message event occurs.

It is updated when the Jobscheduler service/daemon is running.

The maximum file size is 99 MB. Three files are rotated cyclically, so that when the file size of one file reaches 99 MB, the next file is used and the oldest file is discarded.

File name

```
msgevent.log
msgevent0.log
msgevent1.log
```

Description

Reference this file to check the occurrence of message events.

If a "Wait for Message Event" job is briefly stopped, for example because of system maintenance, the occurrence of message events that should have waited can be checked, so it is possible to determine whether or not the succeeding job should be executed.

File Location

Windows	*<Systemwalker installation directory>*\MpWalker.JM\mpjobsch \jobdb If using subsystem 1 to 9 in multi-subsystem operations: *<Systemwalker installation directory>*\MpWalker.JM\mpjobsch \jobdb*n* * Where "*n*" is the subsystem number
Solaris Linux	/var/opt/FJSVJOBSC If using subsystem 1 to 9 in multi-subsystem operations: /var/opt/FJSVJOBSC/JOBDB*n* * Where "*n*" is the subsystem number
HP-UX	/opt/FHPJOBSCH/db If using subsystem 1 to 9 in multi-subsystem operations: /opt/FHPJOBSCH/db/JOBDB*n* * Where "*n*" is the subsystem number
AIX	/usr/FAIXJOBSC/db If using subsystem 1 to 9 in multi-subsystem operations: /usr/FAIXJOBSC/db/JOBDB*n* * Where "*n*" is the subsystem number

File format

The following information is recorded as a single record per line. The file format is CSV format, and each item is separated by a comma.

Item	Size (bytes)	Output Data	Remarks
Date	19	Date when the message event occurred. YYYY MM/DD hh:mm:ss This item is always output.	YYYY: Four-digit year MM: Two-digit month (01 to 12) DD: Two-digit day (01 to 31)

Item	Size (bytes)	Output Data	Remarks
			hh: Two-digit hours (00 to 23) mm: Two-digit minutes (00 to 59) ss: Two-digit seconds (00 to 59)
Message event name	12	Name of the message event that occurred. This item is always output.	
Project name	50	Project name specified when the message event occurred. This data is not output if no project name was specified.	
Job net name	50	Job net name specified when the message event occurred. This item is not output if no job net name was specified.	
Variable parameter	200	Variable parameter specified when the message event occurred. This item is not output if no variable parameter was specified.	
Source name of the event log	64	Source name of the event log for a message event generated by the message table definition. This item is not output for anything other than a message event generated by the message table definition.	
Event ID of the event log	5	Event ID of the event log for a message event generated by the message table definition. This item is not output for anything other than a message event generated by the message table definition.	
Mail file name	256	Mail file name of the message event generated upon mail receipt. This is the name of the file containing the entire mail saved when the mail is received. This item is not output for anything other than a message event generated at mail receipt.	

Notes

Do not open this file in an application that opens it exclusively. If a message event is generated while this file is open exclusively, the message event generation process will become slow and message event information cannot be written.

Example

Record content when the message event "MESSAGE" is generated for the project name "TEST" and job net name "TEST01"

```
"2014 01/02 03:04:06","MESSAGE","TEST","TEST01",,,,
```

18.3 Job Net Variable Information Files

This section explains the job net variable information files.

18.3.1 Job Net Variable Information Files

Information about variables used during job execution and the result of importing the job net variables are stored in the job net variable information files. There are two types of job net variable information files:

- File containing the list of variables in effect and their values when the job net ended (.tvr extension)

- File containing the list of variables imported by each job in the job net and their values, and the variables in effect and their values during job execution (.tvh extension)

File name

[Windows]

<*project name*>_<*job net name*>_cccccccccccccccc.tvr
<*project name*>_<*job net name*>_cccccccccccccccc.tvh

* is a string

[UNIX]

<*project name*>_<*job net name*>_nn.tvr
<*project name*>_<*job net name*>_nn.tvh

* nn is a number

Description

Reference this file to check job net variables when, for instance, a job net or job has ended abnormally. You can check the following information:

- Variables imported by each job from its job output information

- Variables used by each job during job execution

- List of variables in effect when the job net ended

File Location

Windows	<*Systemwalker installation directory*>\MpWalker.JM\mpjobsch\jobdb\jobschtvr
	If using subsystem 1 to 9 in multi-subsystem operations: <*Systemwalker installation directory*>\MpWalker.JM\mpjobsch\jobdb*n*\jobschtvr * Where "*n*" is the subsystem number
Solaris Linux	/var/opt/FJSVJOBSC/jobschtvr If using subsystem 1 to 9 in multi-subsystem operations: /var/opt/FJSVJOBSC/JOBDB*n*/jobschtvr * Where "*n*" is the subsystem number
HP-UX	/opt/FHPJOBSCH/db/jobschtvr If using subsystem 1 to 9 in multi-subsystem operations: /opt/FHPJOBSCH/db/JOBDB*n*/jobschtvr * Where "*n*" is the subsystem number
AIX	/usr/FAIXJOBSC/db/jobschtvr

	If using subsystem 1 to 9 in multi-subsystem operations: /usr/FAIXJOBSC/db/JOBDB*n*/jobschtvr * Where "*n*" is the subsystem number

File format

These are plain text files, and can be viewed in a text editor.

.trv file

This file contains the list of job net variables in effect and their values when the job net ended.

If the job net is layered, the .tvr file for the job net in the 1st layer will contain the list of variables in effect and their values when a child job net in the 2nd layer or lower ended.

```
[TRANSFER_VARIABLE]        (1)
JOB.NAME=VALUE             (2)
JOB.A=B                    (2)
```

(1) Start of section containing the variables in effect when the job net ended. This is a
fixed string.
(2) Variable name and its value when the job net ended, using the format "<*variable
name*>=<*value*>".

.tvh file

This file contains the list of the variables imported by the jobs in the job net and their values, and the variables in effect and their value during job execution.

```
[VARIABLE_OUTPUT1]         (1)
JOB.NAME=VALUE             (2)
JOB.A=B                    (2)
[VARIABLE_INPUT2]          (3)
JOB.A=B                    (4)
[VARIABLE_OUTPUT2]         (1)
JOB.NAME=VALUE             (2)
JOB.A=C                    (2)
```

(1) Start of section containing the variables that a job imported from the job output
information.
    The string "VARIABLE_OUTPUT" is fixed. The number following this fixed string is managed
by the Jobscheduler.
    When the job is restarted, the section for the previous startup (or restart) is deleted
    and a new section is appended.
(2) Variable name and its value imported by the job from its job output information, using
the format "<*variable name*>=<*value*>".
(3) Start of section containing the variables used by a job during job execution.
    The string "VARIABLE_INPUT" is fixed. The number following this fixed string is managed
by the Jobscheduler.
    When the job is restarted, the section for the previous startup (or restart) is deleted
    and a new section is appended.
(4) Variable name and its value during job execution, using the format "<variable
name>=<value>".

Notes

- These files are deleted when a job net is started, and only the job net variable information from the most recent job net execution is stored.

- The following job net variables are not stored in these files:

  - Variable parameter (@.VPARAM@)

  - Project name (@.PROJECT@)

- Job net name (@.JOBNET@)

- Job name (@.JOBNAME@)

- Subsystem number (@.SUB_SYSTEM@)

- Master project name used with the Master Schedule Management function (@.MASTER_PROJECT@)

- Systemwalker Operation Manager updates this file when necessary while the job net is running. Do not directly access this file for a job net that is running. Doing so may cause the job net to end abnormally.

Examples

.tvr file

```
[TRANSFER_VARIABLE]
JOB.CF_FILENAME=/tmp/project.txt
JOB.CF_CHKTYPE=fwait
JOB.CF_FILESIZE=1024
JOB.CF_FILEYEAR=2014
JOB.CF_FILEDATE=0123
JOB.CF_FILETIME=1234
```

.tvh file

```
[VARIABLE_OUTPUT1]
JOB.CF_FILENAME=/tmp/project.txt
JOB.CF_CHKTYPE=fwait
JOB.CF_FILESIZE=1024
JOB.CF_FILEYEAR=2014
JOB.CF_FILEDATE=0123
JOB.CF_FILETIME=1234
[VARIABLE_INPUT2]
JOB.CF_FILENAME=/tmp/project.txt
JOB.CF_FILESIZE=1024
```

18.4 User Limit File [UNIX]

This chapter explains the user limit function file provided by Systemwalker Operation Manager.

18.4.1 User Limit Lists for Job Execution

In this section, you can find the setting method for creating a user limit list with an editor such as vi or Notepad to limit users for job execution.

File name

```
mjexuser
```

Description

In this file, enter the names of the users that will be allowed to execute jobs.

File Location

Solaris HP-UX AIX Linux	/etc/mjes

EE   Store user limit lists for job execution in the following locations if multi-subsystem operations are being performed:

/etc/mjes/mjes*n* (*n*: subsystem number)

File Format

```
<User name>
:
:
```

Parameters

<User name>

Enter one user name per line. These users will be allowed to execute jobs.

- User names are case sensitive.

- The number of users that can be listed in each file and the number of characters per line are as follows:

    - Number of users: 256 or less. Any additional users (the 257th user and onwards) will be invalid.

    - Number of characters per line: 1,024 bytes or less. If more than 1,024 bytes are specified, the entire line will be invalid.

- Only white space (spaces or tabs) can be entered in front of user names.

- Any white space or tab after user names is ignored.

- Blank lines are ignored.

- Use the hash "#" symbol to enter comments. Any text after the hash "#" symbol will be treated as a comment.

- If a blank file is created, no users will be able to execute jobs.

Cautions

- If there are any jobs that are re-executed when Systemwalker Operation Manager is restarted, these jobs will be re-executed even if their execution user is deleted from the user control list for job execution.

- Set up access rights for user limit lists for job execution to ensure security.

    Owner: root

    Mode: 600

- If user names in a user control list for job execution are duplicated as shown in the following example, "username1" and "username2" will be regarded as users that can execute jobs. However, duplicated users are counted twice in the count of the number of users that have been set, so in this example the number of users is three.

```
username1
username2
username1
```

Example

```
username1
username2
username3
```

18.5 Import/Export Execution Result Files

This section explains the execution result files that are output when definition information for Systemwalker Operation Manager is imported or exported via the GUI.

18.5.1 Import execution result file

File name

```
impinfo.txt
```

Description

The import execution result file contains the results of importing the specified job net definition CSV files or group definition CSV files.

File Location

Windo ws	*<Systemwalker Operation Manager client installation directory>*\MPWALKER.JM \mpjmc1\work/jmimpexp\

File Format

```
Subsystem number
Project name


Path containing name of job net definition CSV file or group definition CSV file
Import information messages


Path containing name of job net definition CSV file or group definition CSV file
Import information messages
              :
              :
```

Note: Import information messages can extend over multiple lines.

Parameters

The paths to the names of the job net definition CSV files or group definition CSV files

The names of the job net definition CSV files or group definition CSV files that have been imported are output to this parameter

Import information messages

The execution results of importing each of the specified job net definition CSV files or group definition CSV files are output to this parameter.

Cautions

Do not open this file with an application that will open it exclusively. If definitions are imported while this file is still opened exclusively by another application, then it will not possible to write import information.

Example

```
C:\temp\test1.csv
Mpjobsch:INFO:2300:Command jobschsetnet terminated normally.
```

18.5.2 Export execution result file

File name

```
expinfo.txt
```

Description

The export execution result file contains the results of exporting the specified job nets or groups.

File Location

Windows	*<Systemwalker Operation Manager client installation directory>*\MPWALKER.JM \mpjmc1\work\jmimpexp\

File Format

```
Subsystem number
Project name

Job net or group names
CSV file name (Path containing CSV file name)
Export information messages
            :
            :
Job net or group name
CSV file name (Path containing CSV file name)
Export information messages

Subsystem number
Project name

Job net or group name
CSV file name (Path containing CSV file name)
Export information messages
            :
            :
```

Note: Import information messages can extend over multiple lines.

Parameters

Project name

The name of the project that contains the job nets or groups being exported is output to this parameter.

Project name folder

The name of the folder (the name of the folder specified as the export destination plus the subsystem number and the project name) is output to this parameter.

Message

The result of creating the project name folder is output to this parameter.

Job net or group names

The names of the job nets or groups being exported is output to this parameter.

CSV file name

The name of the CSV file for the job nets or groups being exported is output to this parameter using the full path specification.

Export information messages

Export processing results for each of the specified job nets or groups is output to this parameter.

Cautions

Do not open this file with an application that will open it exclusively. If definitions are exported while this file is still opened exclusively by another application, then it will not possible to write export information.

Example:

```
Project name: ProjectA
Project name folder=C:\CSV\0\ProjectA
The project name folder has been created successfully.

Job net name=JNetA
CSV file name=C:\CSV\0\ProjectA\JNetA.CSV
Completed successfully
```

18.6 Cluster Operation Definition Files

This chapter explains the definition files for operating Systemwalker Operation Manager in a cluster system.

18.6.1 calcphost.def Calendar Destination Host Definition File

File name

calcphost.def

Description

The calendar destination host definition file is used to define the host where the calendar information is reflected if the automatic reflecting functions of the calendar are valid and if the calendar is changed.

File Location

Windows	*<Systemwalker Operation Manager installation directory>*\MPWALKER.JM\mpjmcal\etc
Solaris Linux	/var/opt/FJSVjmcal/etc
HP-UX	/opt/FHPjmcal/etc
AIX	/opt/FAIXjmcal/etc

File Format

The definition file must have the following standard format.

Specify a single destination host name or its IP address on each line, and enter a line feed code at the end of each line.

Item name	Maximum length (in bytes)	Comments
Host name	128	-
IP address	45	Specify the physical IP address in IPv4 or IPv6 format.

- Up to 10 host names or IP Addresses can be registered. If more than 10 host names or IP addresses are stored, the first 10 host names or IP addresses are made valid.

- The host name or IP address of the local node must also be registered.

- The file format version is written on the first line of the file. You cannot edit this first line.

Cautions

- This definition file does not exist when Systemwalker Operation Manager is installed. When the automatic reflecting functions of the calendar are made valid by the automatic calendar reflection setup command (calsetcluster), an initialized definition file is created. See "3.3 calsetcluster Calendar Auto Reflection Setup Command" for more information.

- This definition file is not delivered by policy distribution.

- Use the same definition file for all nodes where the calendar needs to be synchronized. When a reflection is requested by another host, it is rejected if the host is not included in the destination list of the local node.

- If multiple physical IP addresses are used, such as when multiple network cards are installed, specify the hostname that is obtained by executing the "hostname" in the OS command or the corresponding IP address for that hostname in this definition file.

Example

The following gives a definition example to automatically reflect the calendar information on the hosts having the host name of host1 and host2 and IP address of 192.0.2.0 for IPv4 format and 2001:db8::123:4567:89ab:cdef for IPv6 format if the calendar is changed.

```
@V11L10
#Specify host names or IP addresses on the subsequent lines.
host1
host2
192.0.2.0
2001:db8::123:4567:89ab:cdef
```

Line 1:

Exists in the initial status. It shows a version/level of the definition file. This line cannot be edited.

Line 2:

Exists in the initial status. This line can be edited.

If symbol "#" is written in column 1, this line is treated as a comment.

18.6.2 cluster_ip.csv Definition File

File name

cluster_ip.csv

Description

This file is made valid if created in the storage location of Systemwalker Operation Manager client.

A window for monitoring the job nets abnormally terminated is started from the Systemwalker Console by window linking with Systemwalker Centric Manager.

If the job net destination server is operating in a cluster configuration and if the relationships between logical IP address, physical IP address and subsystem number is defined in this definition file, the server of the current active system (identified by the logical IP address) is always connected and the job net monitoring window is started, independent from the execution server (identified by the physical IP address) during abnormal termination of job net.

File Location

Windows	<*Systemwalker Operation Manager installation directory*>\MpWalker.JM\mpjmcl\etc

File Format

Item name	Maximum length (in bytes)	Format of values
Logical IP address	15	IP address (xxx.xxx.xxx.xxx) (Notes 1, 2 and 3)
Subsystem number	1	0 to 9 (Notes 1 and 2)
Physical IP address	15	IP address (xxx.xxx.xxx.xxx) (Notes 1 and 3)

Note 1:

If the line begins with symbol "#", it is treated as a comment line.

Note 2:

If the logical address or subsystem number is the same as the above line, it may be omitted.

Note 3:

If the IP address is a single-digit value or a two-digit value, specify it without adding zero (0) to its beginning. For example, specify "10.3.15.5" rather than "010.003.015.005".

Caution

The cluster_ip.csv definition file is not backed up or restored by Systemwalker Operation Manager. You should manually back up or restore the file when necessary.

Examples

For 1:1 active node/standby node operation

```
#logical-IP-address, subsystem-number, physical-IP-address
172.23.73.10,0,172.23.73.15
,,172.23.73.16
```

For mutual standby operation

```
#logical-IP-address, subsystem-number, physical-IP-address
172.23.73.10,1,172.23.73.15
,,172.23.73.16
172.23.73.20,2,172.23.73.15
,,172.23.73.16
```

18.7 Web Server Port Definition File

This section explains the definition file for port numbers used for communication between the Systemwalker Operation Manager server and the Web console.

18.7.1 Web Server Port Definition File

File name

Any name (*.*)

Description

This file is used to output and update the port number settings for mpowebconfig (web server configuration update command).

Specify this file as the file name in the "-o *<file name>*" option or "-i *<file name>*" option of mpowebconfig (web server configuration update command).

File Location

Any location

File Format

Specify the port numbers to be assigned to the respective values that follow the attribute "name" of the element name "parameter". Note that the web server port definition file uses XML format. Items in italics may be edited.

```
<?xml version="1.0" encoding="UTF-8"?>
<web-server>
    <ahs>
        <parameter name="http.port">9900</parameter>
    </ahs>
    <fjjavaee>
        <domain>
            <jesetup>
                <parameter name="--domain.instancePort">9901</parameter>
                <parameter name="--http.ssl.port">9902</parameter>
                <parameter name="--domain.adminPort">9903</parameter>
                <parameter name="--jms.port">9904</parameter>
                <parameter name="--orb.listener.port">9905</parameter>
                <parameter name="--orb.mutualauth.port">9906</parameter>
                <parameter name="--orb.ssl.port">9907</parameter>
                <parameter name="--domain.jmxPort">9908</parameter>
            </jesetup>
            <setup/>
        </domain>
    </fjjavaee>
</web-server>
```

Parameters

http.port:

This definition is for the port number used for communication between the Systemwalker Operation Manager server and the Web console.

--domain.instancePort:
--http.ssl.port:
--domain.adminPort:
--jms.port:
--orb.listener.port:
--orb.mutualauth.port:
--orb.ssl.port:
--domain.jmxPort:

The above definitions are for the port numbers used for internal communication on the Systemwalker Operation Manager server.

See

mpowebconfig (web server configuration update command)

Chapter 19 Event Log [Windows]

This chapter explains the event logs output by individual functions provided by Systemwalker Operation Manager.

19.1 Event Log Contents

This section explains the source names of components which are output in the event log, and details of events. For more information about the meaning of messages output in the event log, see the *Systemwalker Operation Manager Message Guide*.

Source Name

Each component has a different source name. Accordingly, users can identify the source name from which component output the event.

Event Contents

The list below shows the contents of the event output by individual components by summarizing each component name corresponding to each source name, type, and event contents.

Component name	Source name	Type	Explanation
Calendar	MpJmCal	Client	This type of message will be output while a client request is being processed.
		Service	This type of message will be output while a service is being processed.
		API	This type of message will be output while an API is being processed.
Power Control/ Service and Application Execution	MpAosfP	Power control	This type of message will be output when the Power Control service is executing the power control functions.
		Service control	This type of messages concerns the control of services and applications that are output by Service/Application Execution services.
		MS-Mail transmission server	These messages are output by the MS Mail transmission server, which is used when transmitting MS Mail as an automatic action.
Jobscheduler	MpJobsch (Note 1)	Service	Service messages include information and errors that are output by Jobscheduler.
		Job net	This concerns the execution status of job nets. (Note 2)
		Command	Events output by the jobschmsgevent command.
Job Execution Control	MpMjes (Note 1)	Service	Messages in this type include information and errors that are output by Job Execution Control services.
		Initialization statement analysis	This type of message concerns mistakes and errors detected by the analysis of the initialization statement in the initialization file.
		Job	Warning message for jobs.
Event Monitoring/ Action Control	MpAosfB	–	This is a type of message output by Event Monitoring and Action Control.

Component name	Source name	Type	Explanation
Task Link	mpnjsosv	–	Task link messages include information and errors that are output by Task Link services.
Systemwalker Operation Manager Base	MpJmSrv	Service	Messages in this type include information and errors that are output by Systemwalker Operation Manager basic services.
ACL Manager	ACL Manager	Service	Information and error messages output by the ACL Manager service.

Note 1:

When operating multiple subsystems, the subsystem number (1 to 9) is appended to each source name. As with the subsystem number 0, the number is not appended.

Examples: MpJobsch2, MpMjes8, etc.

Note 2:

The **Advanced Settings for Event Output** window can be used to specify whether to output execution status events.

Chapter 20 Job Completion Codes

This chapter explains the job completion codes output by Systemwalker Operation Manager.

20.1 Jobscheduler Completion Code

The job completion code is set between 201 and 256 in Jobscheduler. The job completion codes returned by Jobscheduler and their meaning are explained below:

P Point

• •

Completion codes other than 201 to 256 and unused completion codes within the range of 201 to 256 are NOT set by the Operation Manager, but they are output by the job itself. That is, they are output by any of commands, batch files, shell scripts and others being registered in the command field of the job. Check the job itself such as its commands, batch files and shell scripts.

Also, make sure that the completion codes of commands, batch files, shell scripts and others to be registered as the job are other than 201 to 256. If a value within the range of 201 to 256 is used as the return value for the job itself, it will be difficult to distinguish whether the completion code was configured by the job or by the Jobscheduler. Note that if the return value of the job itself is 255 or greater, it will be handled as 255.

• •

Job Completion Codes Set by Jobscheduler and Their Meanings

[Windows]

Completion code	Meaning
201 to 231	The job has terminated abnormally.
232 to 238	Unused (but reserved for Jobscheduler)
239	Jobscheduler services have stopped during job execution. (Note 1)
240 to 241	Unused (but reserved for Jobscheduler)
242	Failed to copy and start job net.
243 to 244	The execution of the job (requested to Job Execution Control) has failed.
245	A variable parameter or a job definition variable contains an invalid character, or the standard output file name or the standard error output file name for the job is not specified in the full path.
246	Replacement of a variable parameter or a job definition variable resulted in the maximum string length being exceeded.
247	Job definition variable information failed to be read or written.
248 to 249	Unused (but reserved for Jobscheduler)
250	The specified job does not exist.
251 to 254	Unused (but reserved for Jobscheduler)
255	The job completion code is 256 or greater. (Note 3)
256	The job has been canceled.

[UNIX]

Completion code	Meaning
201 to 231	[Job Execution Control attributes] The job has terminated abnormally. [Normal attributes] Job terminated with a signal. The completion code is "200 + signal number". The completion codes 201, 213,218, 224, 226, and 227 will be ignored by Jobscheduler.
232 to 237	Unused (but reserved for Jobscheduler)
238	Setting job net administrator privileges (initgroups system call) failed.
239	Jobscheduler daemon stopped during job execution. (Note 1)
240 to 241	Unused (but reserved for Jobscheduler)
242	Failed to copy and start job net.
243 to 244	Failed to execute the job.
245	A variable parameter or a job definition variable contains an invalid character, or the standard output file name or the standard error output file name for the job is not specified in the full path.
246	Replacement of a variable parameter or a job definition variable resulted in the maximum string length being exceeded.
247	Job definition variable information failed to be read or written.
248 to 249	Unused (but reserved for Jobscheduler)
250	Failed to obtain the job net administrator information (getpwnam system call). The specified job might not have existed when the job execution attribute was other than Normal. (Note 2)
251	Setting job net administrator privileges (setgid system call) failed.
252	Setting job net administrator privileges (setuid system call) failed.
253	Failed to move to the specified directory (chdir system call).
254	Failed to start the job (execvp system call).
255	Failed to start the job (fork system call). If the job execution attribute is other than Normal, job completion code is 256 or greater. (Note 3)
256	The job has been canceled.

Note 1:

If a backup is made during job execution and if its information is restored, the job being executed is terminated abnormally with completion code 239.

Note 2:

If the job with the attribute other than Normal does not exist, an error message is output in the [Previous log] sheet of the [Monitor - Job] window.

Note 3:

If the actual completion code is 256 or greater, the following is shown in the [Job History] window.

```
... ended code=255.(Actual completion code)
```

20.2 Job Execution Control Completion Code

Depending on the type of job, the following values are set in the job completion code output by Job Execution Control:

Job type	Job completion code
Batch file [Windows]/Shell script [UNIX]	Return code of batch file [Windows]/shell script [UNIX]
Program	Return code of program
Job Control Language (JCL)	Maximum value for each step/completion code of the last step

Note that if the job execution is interrupted, a special value is assigned to the job completion code. The causes of job interruption and the job completion codes set when an interruption occurs are shown below:

[Windows/UNIX version]

Cause of job interruption	Completion code

- When a network job written in JCL was executed, the format of a network job control command (Note 1) was found illegal, or an internal processing error occurred in Job Execution Control on a network job.
 	
- When a job was submitted to a distributed execution queue and started at an execution server (a remote host which is different to a schedule server), one of the following errors occurred:
 	
  - The input file does not exist.
 	
  - The output file does not exist.
 	
- When the network job submission destination is a Windows server, it is possible that the desktop heap has been used up (Note 2).
	0x40000008

When a network job written in JCL was executed, the format of a control statement (Note 3) pertaining to a job step was found illegal.	0x40000012
- Network job submission processing resulted in an error on a remote server. - The execution request was rejected since a job with the request host specified was submitted to the distributed execution queue. - There is insufficient disk space on either the schedule server or the execution server.	0x40000016
When a network job was executed, a request from the schedule server to the execution server contained a function not supported by an earlier version of Systemwalker Operation Manager. This resulted in an error, which made it impossible to connect to the server running the earlier version that did not support the extended function.	0x40000020
An error occurred while receiving a network job from a remote server. Users are restricted by the user control list for job execution.	0x40000024

- A network job was unable to proceed due to a line failure or local server shutdown.
 	
- When a job was submitted to a distributed execution queue and started at an execution server (a remote host which is different to a schedule server), one of the following errors occurred:
 	
  - Errors related to the network occurred while transferring I/O files.
 	
  - System errors occurred while transferring I/O files.
 	
- There is insufficient disk space on either the schedule server or the execution server.
	0x40000132

The job was canceled by a user, the qdel command or the job termination API.	0x40000222
The job was canceled due to a job processing timeout (the maximum time limit reached).	0x40000322

Cause of job interruption	Completion code
When a JCL job was executed, the executable file specified in the job step did not exist.	0x40000806
[Windows] An error occurred during the user authentication to start the job as the job owner (Note 4). Or, an error occurred while authenticating the use on the remote server to process the distributed job or the network job. [UNIX] An error occurred while authenticating the use on the remote server to process the distributed job or the network job	0x40000913
An error that prevents the job from being continued occurred in the Job Execution Control processing.	0x40000999

Note 1:

The corresponding commands are listed below:

- connect

- send

- execute

- receive

- disconnect

Note 2:

In Windows Server 2003 or later, the event shown below will be output to the event log when the first desktop heap shortage occurs after starting the operating system.

Type: Warning
Source: Win32k
Event ID: 243
Description: Failed to allocate desktop heap.

Note 3:

The corresponding control statements are listed below:

- file

- exec

Note 4:

The cause of the error can be attributed to the following:

- Settings required for executing jobs with owner's privileges have not been defined.

- See the *Systemwalker Operation Manager Installation Guide* for required settings.

- A domain controller (PDC or BDC) has been stopped or restarted.

- The Windows Internet Naming Service (WINS) is not operating properly.

[Windows]

Cause of job interruption	Completion code
Illegal memory access by a user application.	0xC0000005

Cause of job interruption	Completion code
The user application attempted an integer divide by zero.	0xC0000094
Cannot find the DLL required to run the user application. (The DLL path is not set to the PATH environment variable.)	0xC0000135
Unable to initialize the DLL required to run the user application. The system may not have enough desktop heap memory to allocate (Note 1).	0xC0000142

Note 1:

In Windows Server 2003 or later, the event shown below will be output to the event log when the first desktop heap shortage occurs after starting the operating system.

```
Type: Warning
Source: Win32k
Event ID: 243
Description: Failed to allocate desktop heap.
```

If an online task is interrupted, the following job completion codes are set:

Meaning	Completion code
Online job ended normally.	0
Online job startup failed because Interstage has not been installed.	1
Online job startup failed because an Interstage error has occurred during initialization of online job startup environment.	2
Online job startup failed because an Interstage error has occurred during startup of online job.	3
Online job monitoring was interrupted because an Interstage error has occurred when the online job was being monitored.	4
Online job stop failed because an Interstage error has occurred when the online job was being stopped.	5
Online job was abnormally terminated.	6
Online job was forcibly stopped before it was started.	256
- A network job was unable to proceed due to a line failure or local server shutdown. - There is insufficient disk space on either the schedule server or the execution server.	0x40000132
Online job was forcibly terminated because the time lapse limit was exceeded or cut-off execution time arrived during the job execution.	0x40000322
Online job startup failed due to a system error.	0x40000999