FUJITSU

FUJITSU Software
Systemwalker Operation Manager

Technical Guide

UNIX/Windows(R)

J2X1-3180-16ENZ0(00)
May 2015

Preface

Purpose of This Document

This document describes the functions and usage of Systemwalker Operation Manager.

Systemwalker Operation Manager is a software product that provides integrated operation and management functions, such as automatic system operation, automatic job execution, control of the job execution environment, and monitoring, operation and automatic backup of jobs.

Systemwalker is the generic name for the family of operation management products for distributed systems provided by Fujitsu Limited.

Intended Readers

This document is intended for system administrators who are considering installing Systemwalker Operation Manager and for administrators who use Systemwalker Operation Manager to operate and manage distributed systems (consisting of servers and clients) or various systems within a network (LAN or WAN).

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

May 2015

Contents

Chapter 1 Function Overview

This chapter presents an overview of Systemwalker Operation Manager.

1.1 What is Systemwalker Operation Manager?

In today's system environment, where distributed systems are now being comprehensively adopted, client/server systems are spreading rapidly while still maintaining compatibility with conventional mission-critical systems. On the other hand, it is increasingly important to be able to operate increasingly diverse and complex systems easily and to manage these systems reliably.

To meet these challenges, Systemwalker Operation Manager automates a wide range of operation management tasks, from starting servers distributed over the network through to executing regular batch jobs on those servers. This significantly reduces the workload required for operation management.

Systemwalker Operation Manager comes in two editions: the Standard Edition, which provides the standard system specifications; and the Enterprise Edition, which supports large-scale systems and mission-critical jobs.

1.1.1 What Can Be Done with Systemwalker Operation Manager?

This section explains the general advantages of installing Systemwalker Operation Manager (points common to both the Standard Edition and the Enterprise Edition). For information about the advantages of installing the Enterprise Edition of Systemwalker Operation Manager, refer to "1.1.5 What Can Be Done with Systemwalker Operation Manager Enterprise Edition?"

Automating routine system operations and management tasks

By installing Systemwalker Operation Manager, the following system operations and management tasks can be automated.

- Automatic control of power on/power off, and shutdown/reboot

  Servers can be turned on and off automatically. Servers are turned on and off according to predefined time schedules. Also, servers can be turned on automatically when their clients start.

  This function can only be used with Windows x86, Solaris 32bit, and Linux x86 versions of this product.

  Note that automatic control of shutdown/reboot can be used with Windows x64/x86, Solaris 32bit, and Linux x86 versions.

- Automatic startup of services and applications

  When the server starts, services and applications are started up automatically in the preset sequence. Also, which services and applications are started can be changed for each day.

  Automatic service startup can only be used if the connection destination server is running the Windows version of this product.

- Automatic execution of regular batch jobs

  By registering the startup dates and times for regular batch jobs, batch jobs can be executed automatically according to predefined time schedules. Corrective action can be taken automatically without operator intervention when batch jobs fail.

- Automatic monitoring and automatic corrective actions for events [Windows version]

  By registering corrective actions (hereafter referred to as "actions") for certain events, these actions can be performed automatically when these events occur on the system.

- Automatic data backup [Windows version]

  Linking to ARCserve makes it possible to automatically back up file resources on servers and clients synchronously with job executions or system operations.

- Scheduling file compression and transfer

  Systemwalker Operation Manager provides commands for compressing and transferring files as Task Link commands. By registering these commands with the Jobscheduler, these jobs can be executed according to schedules.

Visually monitoring and operating jobs on-screen

By installing Systemwalker Operation Manager, the status of jobs can be displayed in real time using different colors for each status, making it possible to check the execution status of jobs at a glance. Jobs on multiple systems can also be monitored simultaneously.

Jobs can be operated while their status is being monitored.

Controlling job execution according to the system's operation and status

By installing Systemwalker Operation Manager, the following job execution items can be controlled.

- Controlling job execution environments

  The job execution environment, such as the job execution order and the number of jobs that can be executed simultaneously, can be adjusted to enable jobs to be executed efficiently.

- Distributed job execution

  Jobs can be distributed to servers with extra capacity to execute jobs simultaneously, rather than executing the jobs on a fixed server. This prevents load from concentrating on a particular server.

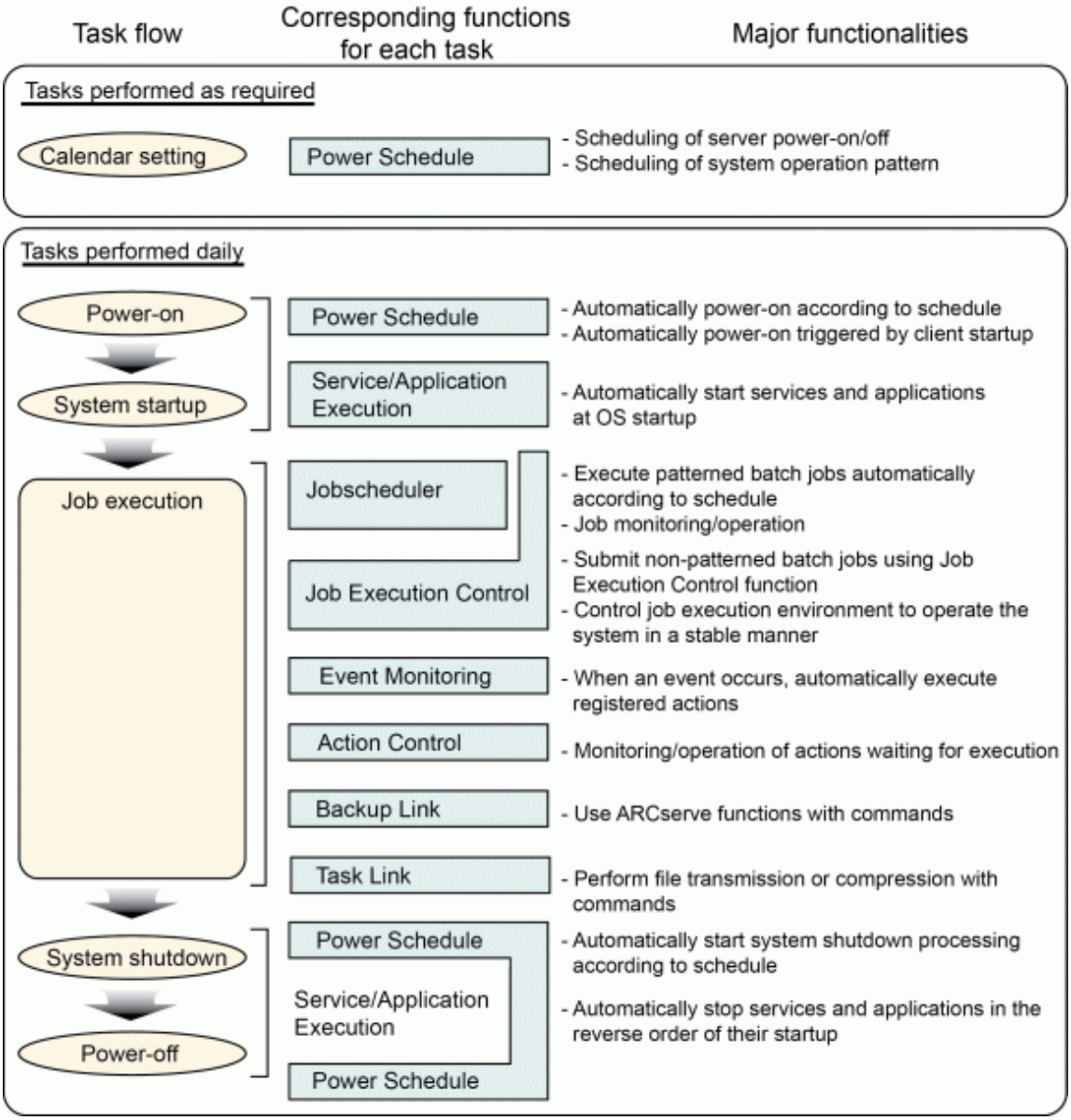
- Job execution on servers connected to the network

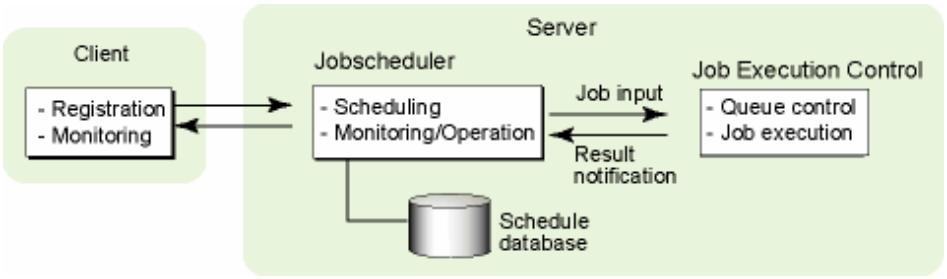
  Jobs can be executed on any server connected to the network.

- Control for clients connected to the network

  The power to clients connected to the network can be controlled, as can file transfer to and from these clients.

The following figure shows the relationship between the primary functions of Systemwalker Operation Manager and system operation management tasks, as well as showing an outline of batch job automation. Although the automation of batch jobs is performed using the Jobscheduler function, the Jobscheduler function and the Job Execution Control function are linked internally.





1.1.2 Roles of Systemwalker Operation Manager on Servers and Clients

Systemwalker Operation Manager consists of a server function and a client function. The server function is installed on servers that perform routine system operation management tasks. The client function is installed on the network PCs.

The following section provides an overview of the roles of the Systemwalker Operation Manager server and client functions.

Role of the server function

The following processing is performed on servers where the server function is installed:

- Automatic control of power on/power off, and shutdown/reboot (Note 1)
- Starting services and applications automatically when the server starts (Note 2)
- Automatic execution of regular batch jobs
- Automatic monitoring and automatic corrective actions for events [Windows version]
- Automatic data backup [Windows version]
- Controlling job execution environments
- Distributed job execution
- Job execution on servers connected to the network

Note 1)

Automatic power control is only available for Windows x86, Solaris 32bit, and Linux x86 versions.

Automatic control of shutdown/reboot is only available for Windows x64/x86, Solaris 32bit, and Linux x86 versions.

Note 2)

Automatic service startup is only available for the Windows version.

Role of the client function

The client function performs the following processing on connected Systemwalker Operation Manager servers. The client function can be accessed by logging on to the server function being registered or monitored, and can be operated using GUI screens.

- Registering and changing the various data required for the server function
- Monitoring and operating jobs and actions (Note 1)

Note 1)

Actions can only be monitored and operated if the connected server is running the Windows version.

Systemwalker Operation Manager supports the clients listed below. These clients can each be started by selecting **Start** or **Apps >> Systemwalker Operation Manager**.

- Systemwalker Operation Manager client

  This is the Systemwalker Operation Manager client. Select **Start** or **Apps** >> **Systemwalker Operation Manager** to start this client.

- Systemwalker Operation Manager environment setup client

  This client sets up the operating environment for Systemwalker Operation Manager. Select **Start** or **Apps** >> **Systemwalker Operation Manager** >> **Environment Setup** to start this client.

- Multi-server monitoring client

  This client monitors multiple servers. There is a Windows client that is used to register, modify, monitor, and perform operations for jobs, and a Web Console that is used to monitor and perform operations for jobs. Select **Start** or **Apps** >> **Systemwalker Operation Manager** >> **Multi-server Monitoring** to start the Windows client. Specify the URL in the browser to use the Web Console.

- Jobscheduler information print client

  This client prints various Jobscheduler information. Select **Start** or **Apps** >> **Systemwalker Operation Manager** >> **Jobscheduler Info Printout** to start this client.

EE

- Master Schedule Management environment setup client

  This client sets up the operating environment for the Master Schedule Management function. Select **Start** or **Apps** >> **Systemwalker Operation Manager** >> **Master Schedule Management Environment Setup** to start this client.

- Master Schedule Management monitor client

  This client monitors the status of the Master Schedule Management function. Select **Start** or **Apps** >> **Systemwalker Operation Manager** >> **Master Schedule Management** to start this client.

With Systemwalker Operation Manager, servers where the Systemwalker Operation Manager server function is installed are referred to as "Systemwalker Operation Manager servers."

1.1.3 Jobs Handled by Systemwalker Operation Manager

With Systemwalker Operation Manager, a sequence of scripts, commands and execution programs can be registered and executed. These scripts, commands and execution programs are referred to as "jobs." A sequence of jobs is referred to as a "job net."

Systemwalker Operation Manager can process the following jobs.

- Batch files [Windows version] and shell scripts [UNIX version]
- Execution programs (or commands)
- The Job Control Language (JCL) provided by Systemwalker Operation Manager
- Systemwalker scripts
- Systemwalker Operation Manager Task Link commands
- "Systemwalker for Systems Management Server" commands (for SMS commands) [Windows version]
- Interstage WorkUnits

  The following Interstage WorkUnits can be processed:

  - TD (Transaction Director) WorkUnits (Windows x86/Solaris 32bit/Linux x86)
  - EJB (Enterprise JavaBeans) WorkUnits (Windows x86/Solaris 32bit/Linux x86)
  - Utility WorkUnits (Solaris/Linux)
  - CORBA WorkUnits (Windows/Solaris/Linux)

  Note that the WorkUnits that can be handled vary depending on the operating system and the Interstage version and level. For more information, refer to "3.2 Software Resources".

- PowerAIM WorkUnits (only if the connection detection server is running UXP/DS) [UNIX version]

The Interstage WorkUnits and PowerAIM WorkUnits are jobs supported by other products. To link to Interstage refer to "2.16 Linking to Other Products".

All jobs handled by Systemwalker Operation Manager are basically executed by Job Execution Control. However, if the connection destination server is running the UNIX version, Job Execution Control may not be used for the following kinds of jobs.

Jobs executed by PowerAIM [UNIX version]

For PowerAIM WorkUnits provided by UXP/DS PowerAIM, the Jobscheduler links directly to PowerAIM. Queue Control and other Job Execution Control functions are disabled.

Jobs executed by the Jobscheduler [UNIX version]

The following jobs can be executed only by the Jobscheduler because of compatibility issues with earlier versions of the Jobscheduler.

- Shell scripts
- Commands
- Execution programs

Jobs and file extensions

The following table lists the jobs that can be handled by Systemwalker Operation Manager, the file extensions that are assigned when jobs are stored in files, and how each job is handled by Job Execution Control.

[Windows version]

Jobs	File extension	Recognition by Job Execution Control
Job Control Language (JCL)	filename.jcl	Recognized as the Job Control Language (JCL).
Systemwalker script	filename.swt	Recognized as a Systemwalker script.
Batch file	filename.bat filename.cmd	Recognized as a batch file.
Executable file	filename.exe filename.com	Recognized as an executable file.

[UNIX version]

Jobs	File extension	Recognition by Job Execution Control
Job Control Language (JCL)	filename.jcl	Recognized as the Job Control Language (JCL).
Systemwalker script	filename.swt	Recognized as a Systemwalker script.
Other jobs	Any extension (Note 1)	Recognized as a command if a binary file is used Recognized as a Shell script if a text file is used.

Note 1)

Files with the "inf" extension are created by Job Execution Control for job data control. Do not create shell script files with the "inf" extension.

Scheduled jobs and demand jobs

Jobs submitted from Systemwalker Operation Manager can be divided into the following two types:

- Scheduled jobs
- Demand jobs

Scheduled jobs can be executed automatically, and are monitored and operated by the Jobscheduler. Scheduled jobs are executed automatically by registering them in job nets.

Demand jobs are not scheduled, and are submitted separately when necessary. These jobs can be submitted directly, or they can be stored in a job folder and submitted when necessary. Once demand jobs have been stored in a job folder, jobs or commands need not be entered each time jobs are submitted.

Online jobs and batch jobs

Systemwalker Operation Manager can schedule all jobs, including batch jobs and online jobs.

Batch jobs are executed in the background process when their startup event, such as the preset time or message, occurs. Batch jobs (and job nets) end when the job processing that has been running ends (except for when jobs are cancelled due to their execution time expiring).

If job nets have Interstage attributes, their end times can be scheduled, as well as their start times. These jobs are referred to as "online jobs."

If online jobs are developed using Interstage, the start and end time of each WorkUnit can be automated by Systemwalker Operation Manager. Any combination of batch jobs can be scheduled before or after online jobs.

Jobs where windows are displayed

Using the Task Link client application startup commands makes it possible to execute applications that display windows on clients. This kind of application can be scheduled by registering client application startup commands as "jobs" in job nets.

Job net execution attributes

Before a scheduled job can be registered, a job net must be registered first. When the job net is registered, the function that will execute the job net can be selected by specifying "Job net execution attributes." These attributes are explained below.

Job Execution Control (JES) attribute

Specify this attribute when jobs are submitted to the Job Execution Control function.

Almost all jobs handled by Systemwalker Operation Manager can be registered by specifying the Job Execution Control attribute. Not only can many different types of jobs be registered with this attribute, the following kind of detailed execution control is also possible. For these kinds of operations, job nets must be registered with the Job Execution Control attribute.

- Managing jobs using queues, and controlling the execution priority of jobs and the number of jobs that can be executed concurrently.
- Executing jobs on other servers connected via the network.

For more information about using Job Execution Control, refer to "2.3 Job Execution Control".

Interstage (INTS) attribute

Specify this attribute to perform startup/shutdown control and monitoring for Interstage WorkUnits.

Interstage WorkUnits that have been registered with this attribute can be controlled via Job Execution Control. This means that Queue Control and other Job Execution Control functions can be used. Job nets with the Interstage attribute are restricted to job nets controlled by Interstage, and only Interstage WorkUnits can be registered with this attribute. Ordinary jobs cannot be registered with this attribute.

Note that WorkUnits that have been started using the Interstage *isstart* command cannot be managed by Systemwalker Operation Manager. WorkUnits started by Systemwalker Operation Manager are checked by Systemwalker Operation Manager every five minutes. Note that it may take up to five minutes to detect errors that occur.

PowerAIM (PAIM) attribute [UNIX version]

Specify this attribute to submit WorkUnits to PowerAIM for execution. To execute PowerAIM WorkUnits, job nets must be registered with the PowerAIM attribute. Only PowerAIM WorkUnits can be registered in job nets with the PowerAIM attribute.

Earlier version compatibility (NORMAL) attribute [UNIX version]

Specify the earlier version compatibility attribute to control job execution using the Jobscheduler. The jobs that are registered in job nets with this attribute must be able to be executed by the Jobscheduler alone.

Jobs that cannot be executed by Systemwalker Operation Manager

Systemwalker Operation Manager cannot execute the following jobs. Do not register these jobs with the Jobscheduler.

- Windows system commands (such as Notepad)

  However, these commands can be executed if the Task Link client application startup command is used with Windows systems.

- Interactive commands (such as edlin[Windows version], and more or pg [UNIX version])
- Commands or shell scripts that request data entry (such as format and backup)

  Jobs registered with the Jobscheduler are executed in background mode. Therefore, if a job that requests input is registered with the Jobscheduler, job execution is stopped when the job requests input and the job continues to be displayed in the Jobscheduler as "executing".
- Applications that are not processed correctly when their execution is requested by the AT command [Windows version]

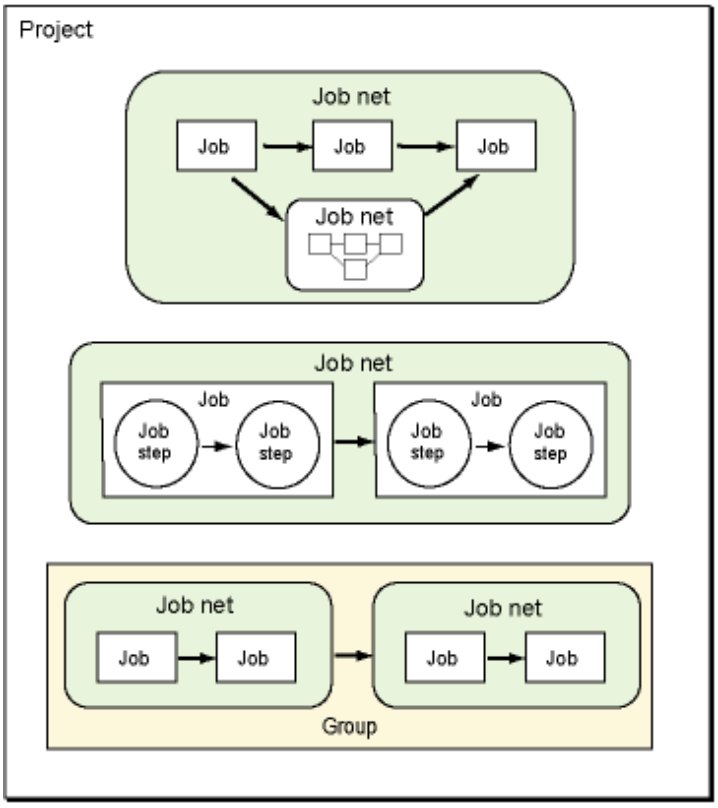
Note that commands that start resident programs do not terminate even after the command processing has completed. (The command is displayed as "Executing".)

If the connection destination server is running UNIX, the following jobs cannot be executed.

- Commands that require a control terminal (such as ps and passwd)
- Full-screen commands (such as vi and sysadm)
- Commands and shell scripts that run in the background
- Internal shell commands

1.1.4 Job Configuration in Systemwalker Operation Manager

With Systemwalker Operation Manager, jobs are regarded as being configured as follows.



Project

A "project" is the largest job category unit. Only users with administrator authority can create projects and register other users with the authority to update, operate and refer to the jobs and job nets in these projects.

Group

A "group" is a collection of related job nets that can be created when necessary. Groups can be used to control the execution sequence of job nets.

Job net

A "job net" is a group of related jobs. Start/stop control can be performed for each separate job net. Time schedules, event occurrence, mail reception and other triggers can be specified as startup conditions.

Job nets can be nested by registering job nets in other job nets. Up to five levels of job nets can be registered.

Job

The shell scripts, execution programs, and JCL (Job Control Language) scripts for job processing are referred to as "jobs." The priority of each job can be controlled. For information about the jobs Systemwalker Operation Manager can handle, refer to "1.1.3 Jobs Handled by Systemwalker Operation Manager".

Job step

A "job step" is the smallest unit of the processing procedures that make up JCL (Job Control Language) scripts.

Point

Job Control Language (JCL)

This language is useful for controlling jobs, as it can be used to easily describe the procedures for complicated jobs. It incorporates the concept of "job steps" and describes job procedures in a job-step structure.

EE

1.1.5 What Can Be Done with Systemwalker Operation Manager Enterprise Edition?

The following section explains what can be achieved by installing the Enterprise Edition of Systemwalker Operation Manager.

Large-scale batch job operations made possible

The differences with the Standard Edition are as follows:

- An unlimited number of job nets can be registered in each project. (The Standard Edition only allows up to 255 job nets to be registered.)
- Up to 255 job nets can be registered in a single group (compared with up to 50 job nets for the Standard Edition).
- No scheduling delays occur due to the registration of a large number of batch jobs with multi-subsystem operations.
- Operations for daily schedule management reduce the load of executing scheduled jobs by distributing the schedule for large numbers of batch jobs across multiple servers.

High reliability for mission-critical batch jobs

The differences with the Standard Edition are as follows:

- Jobs can be operated and tested on a single machine using multi-subsystem operations.
- Support for various types of cluster configurations delivers high availability for batch jobs.

Note

Although an unlimited number of job nets can be registered with the Enterprise Edition of Systemwalker Operation Manager, perform thorough testing before commencing actual operations, in order to ensure that job nets start on schedule without any problems. For information about performance tuning, refer to "2.2.4 Reference Information (Know-how, etc.)"

1.2 Operation Configurations for Systemwalker Operation Manager

This section explains the operation configuration for Systemwalker Operation Manager.

1.2.1 Basic Operations

Systemwalker Operation Manager can schedule and execute jobs using the following functions.

- Jobscheduler function

  This function manages job schedules, and also manages, controls and monitors each job.

- Job Execution Control function

  This function manages and controls the job execution environment, and also executes network jobs.

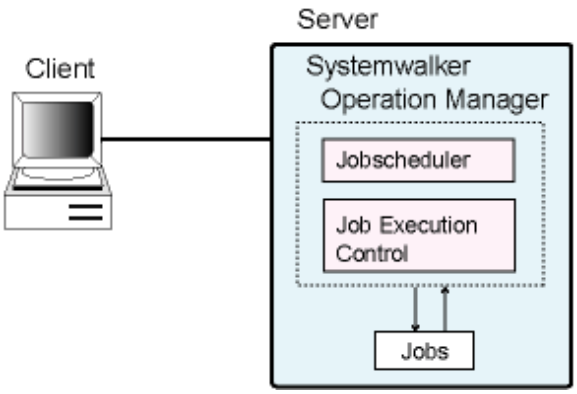
Jobs can be scheduled and executed using the following basic operation configurations.

- Executing jobs on a single server
- Scheduling and executing jobs on different servers

Each operation configuration is explained below.

Executing jobs on a single server

Jobs can be scheduled, managed, executed, monitored and controlled on a single server. The following figure illustrates an example of operation configuration where jobs are executed on a single server.

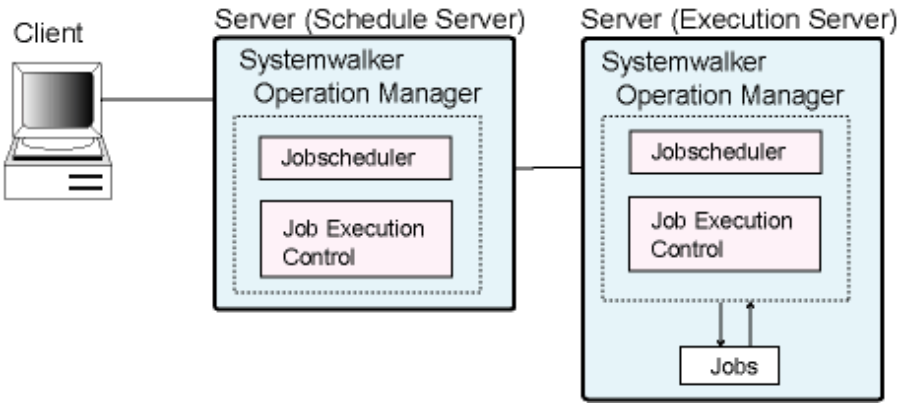


Scheduling and executing jobs on different servers

Using the Job Execution Control network job function, one server can be used solely for managing job schedules and another server can be used solely for executing jobs.

In this type of configuration, the server mainly used for managing job schedules, executing jobs automatically, monitoring and controlling jobs, and submitting network jobs is called the "schedule server." The server mainly used for setting up and controlling the job execution environment is called the "execution server." Systemwalker Operation Manager must be installed and set up on both the schedule server and the execution server.

The following figure illustrates an example of a configuration where jobs are scheduled and executed on different servers.



Schedule server:

The main roles of this Systemwalker Operation Manager server are to manage job schedules and control job execution. To use a server as a schedule server, set up calendars and job schedules on the server. When jobs are scheduled, a schedule information file is created.

Execution server:

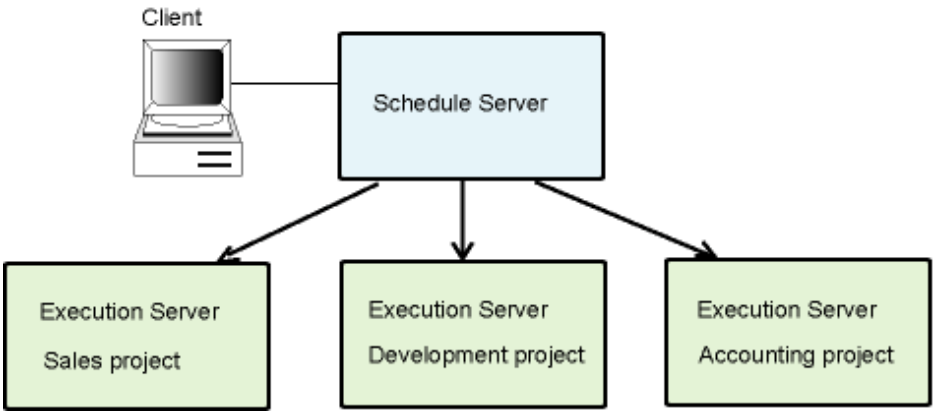
The main role of this Systemwalker Operation Manager server is to execute jobs. Network jobs can be submitted from the schedule server to the execution server using the Job Execution Control network job function.

Information

Schedule information files are files that store schedule information for jobs. The schedule information files are as follows:

- "project-name.dbz" files located under the database directory of the Jobscheduler
- "project-name.grz" files located under the database directory of the Jobscheduler

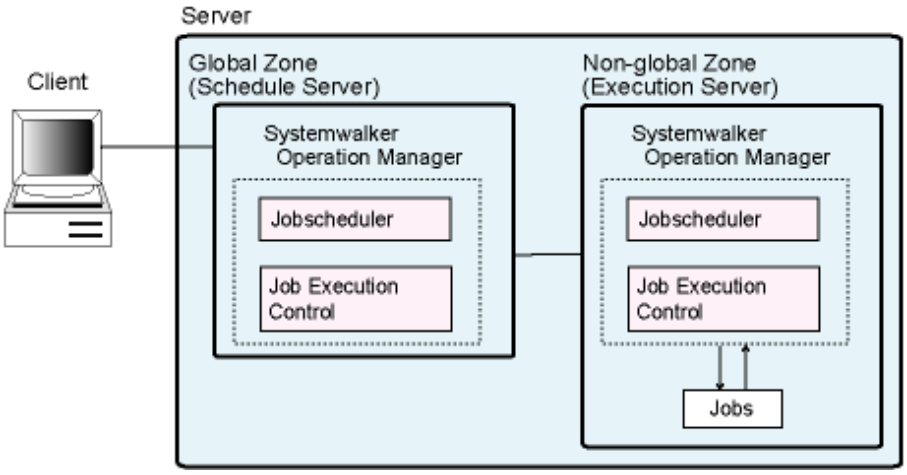
By linking to multiple execution servers, jobs for different projects can be executed on different servers, as shown in the following example.



Scheduling and executing jobs in different zones

This configuration uses the Zone function supported by Solaris 10 or later.

In this configuration, the Job Execution Control network job function is used so that the Global Zone is used solely for job scheduling and Non-global Zones are used solely for job execution.



The Zone function makes it possible to use zones like single servers. Multiple logical servers can be created on a single physical server.

Systemwalker Operation Manager operates by regarding a single zone as a single server. Using the logical IP addresses assigned to each zone, network jobs can be submitted as follows:

- From a zone to another zone
- From another server to a zone
- From a zone to another server

1.2.2 Multi-subsystem Operations

Multiple pairs of the Systemwalker Operation Manager Jobscheduler and Job Execution Control functions can be run simultaneously. Each pair is referred to as a "subsystem" and operations that run multiple subsystems are referred to as "multi-subsystem operations."

Up to 10 subsystems can run on a single machine

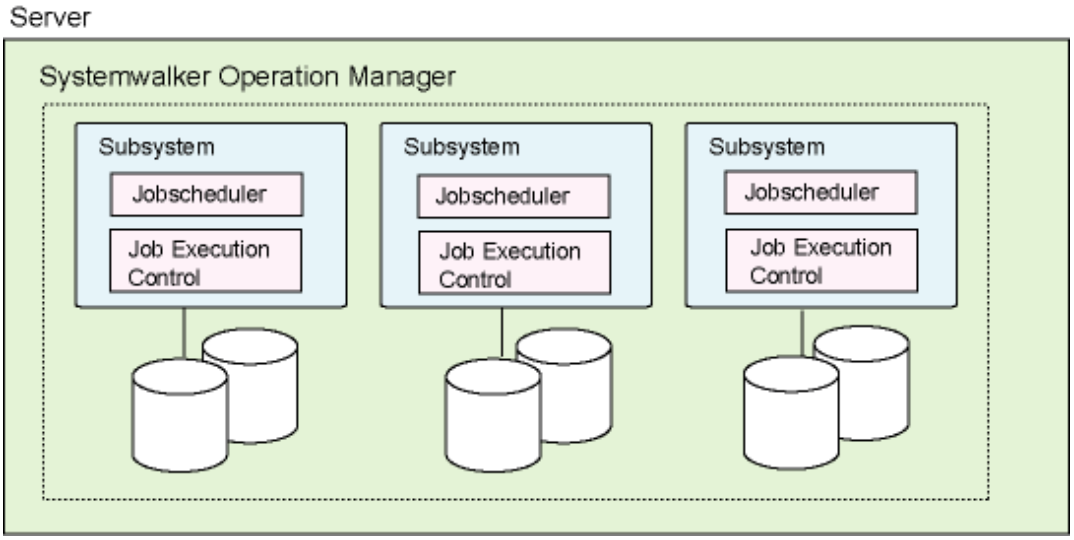
The following multi-subsystem operations are supported.

- Operating multi-subsystems on a single machine
- Operating multi-subsystems of schedule servers
- Operating schedule servers and execution servers as subsystems

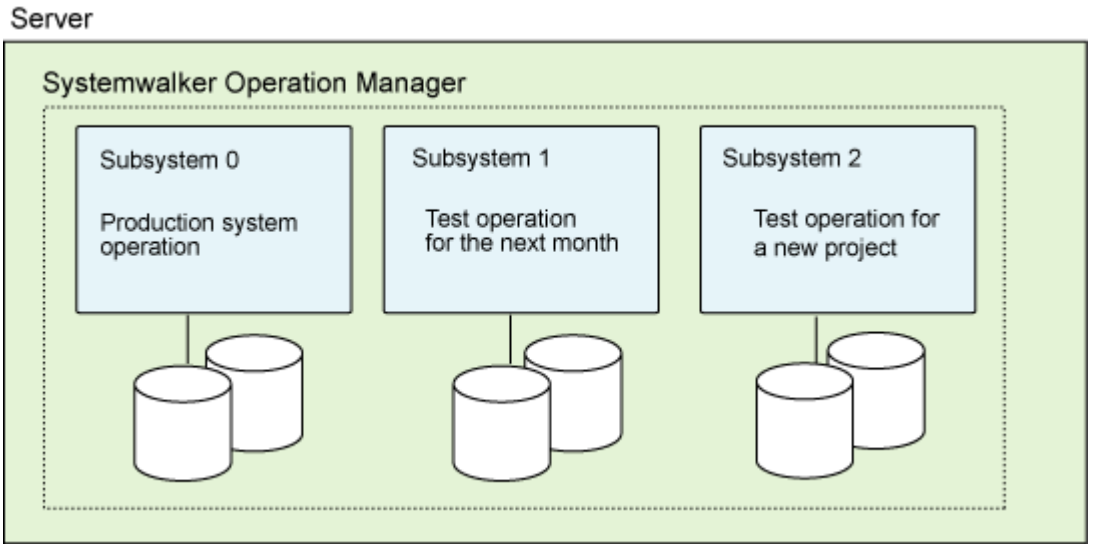
Each operation configuration is explained below.

Operating multi-subsystems on a single machine

The following figure illustrates multi-subsystem operations on a single machine.



If multi-subsystems operate on a single machine, subsystem 0 can be used for production system operations, subsystem 1 can be used for testing next month's data, and subsystem 2 can be used for testing new jobs, for example.

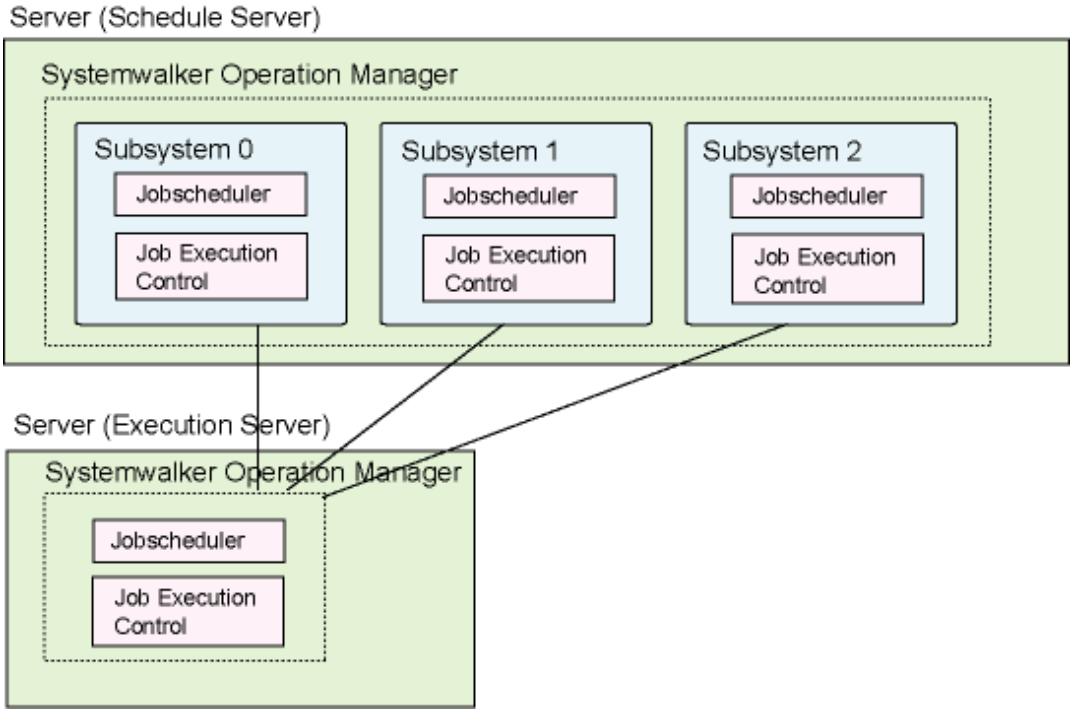


In this example, if jobs for test operations are actually executed, the resources for production system operations might be affected. If there is such a possibility, bypass execution of these jobs by disabling these jobs.

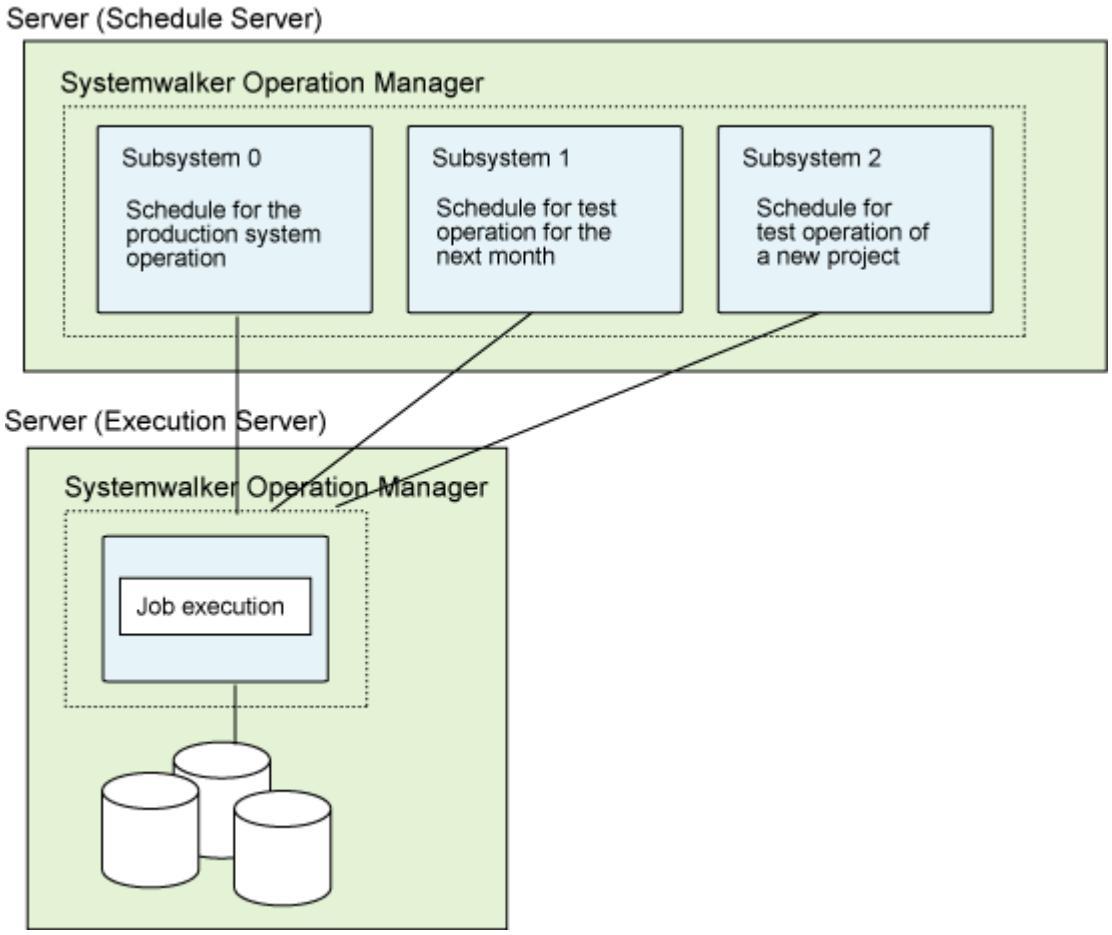
This makes it possible to test jobs that have been created without affecting production system operations.

Operating multi-subsystems of schedule servers

The following figure illustrates a configuration where multiple schedule servers alone are operated as subsystems.



Operating multiple subsystems consisting of only schedule servers makes it possible to perform test operations that link to network jobs, as shown in the following example.



Note

If the execution server is running one of the following versions of Systemwalker Operation Manager, it can receive jobs from only Subsystem 0:

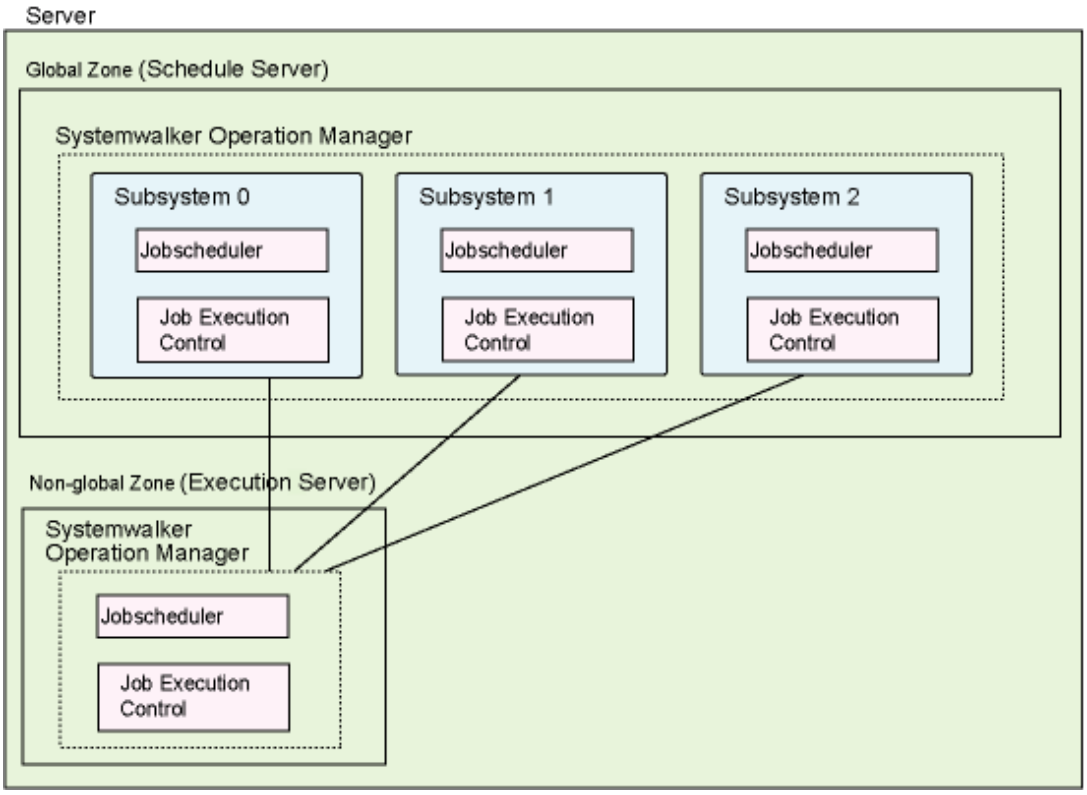
- Version V5.0L30 or before [Windows version]
- Version V5.0 or before [UNIX version]

Job execution requests can be sent from only Subsystem 0. Refer to "A.5 Executable Range of Network Jobs" for details.

Information

The Zone function supported by Solaris 10 or later makes similar operations possible with a single server, as shown below.

This configuration operates multiple subsystems in the Global Zone for the job schedule server, and uses a Non-global Zone as a job execution server.

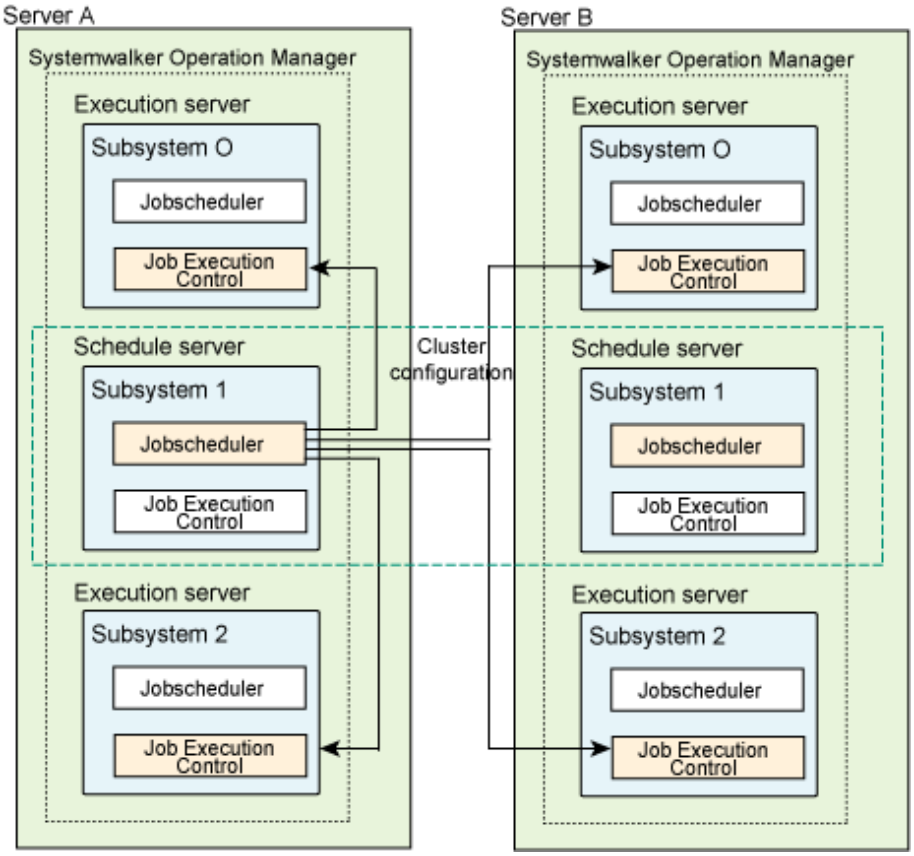


Systemwalker Operation Manager runs by regarding each zone as a single server. Even for multiple subsystem operations, network jobs can be submitted from a subsystem in a zone by assigning logical IP addresses to the zones and specifying logical port numbers for the multiple subsystems.

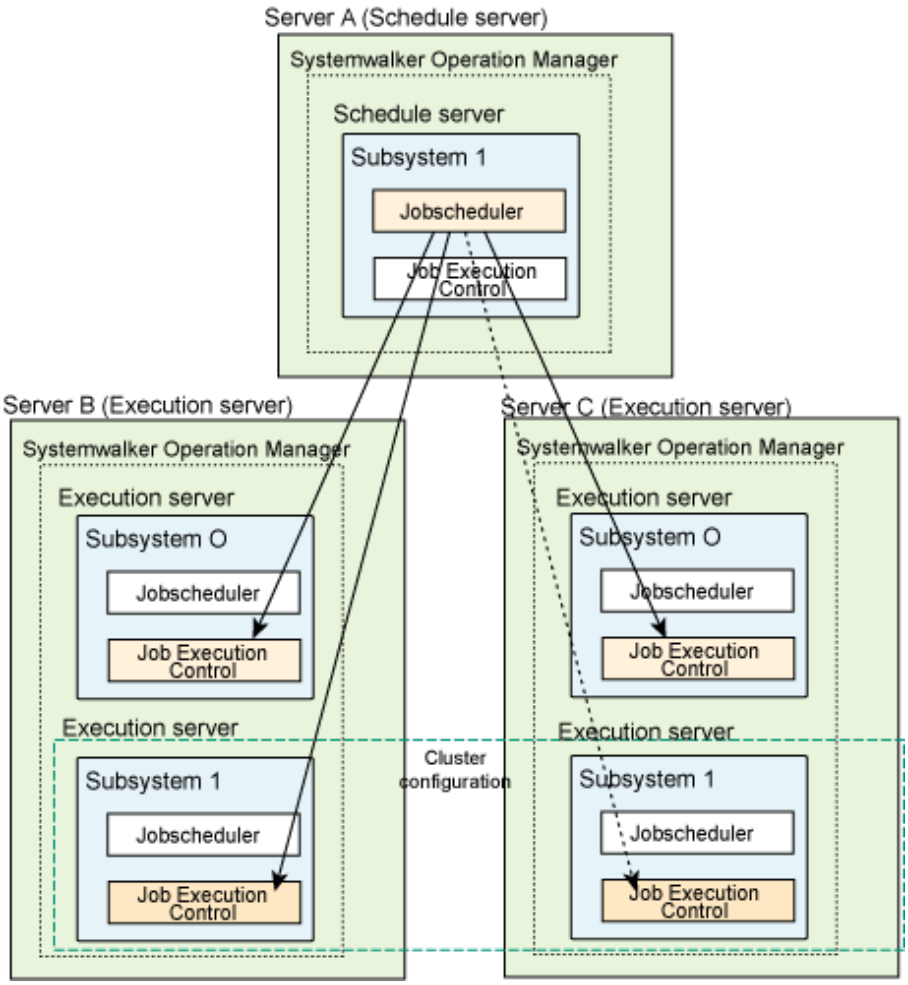
Operating schedule servers and execution servers as subsystems

The following figure illustrates a configuration where both schedule servers and execution servers are operated as multiple subsystems. Any desired subsystem can be operated as a schedule server or an execution server.

In the example in the following figure, Subsystem 1 on Server A and subsystem 1 on Server B are used as a schedule server in a cluster configuration, while subsystems 0 and 2 on Server A and subsystems 0 and 2 on Server B are used as execution servers.



In the example in the following figure, Subsystem 1 on Server A is used as a schedule server, while Subsystem 1 on Server B and Subsystem 1 on Server C are used as an execution server in a cluster configuration, and Subsystem 0 on Server B and Subsystem 0 on Server C are used as independent execution servers.



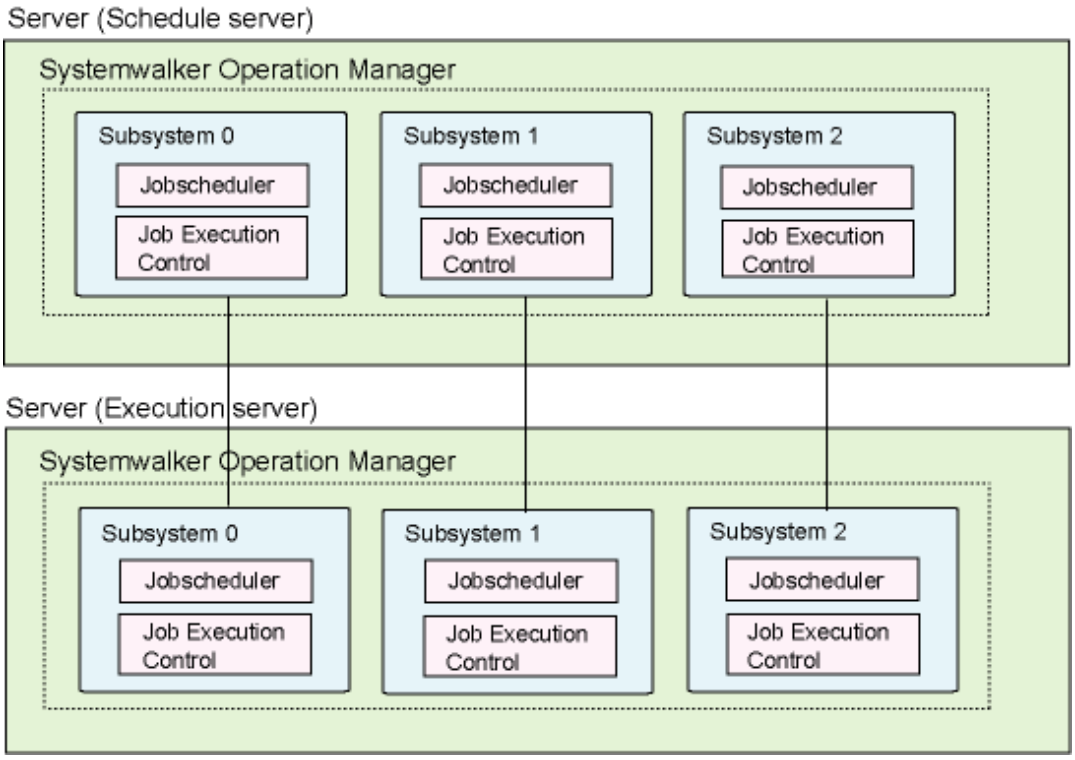
Operating schedule servers and execution servers as subsystems makes it possible to create separate job execution environments for each subsystem. For example, it is possible to set job multiplicity separately for each subsystem.

Note

If either the schedule server or the execution server is running V13.2.0 or earlier, or if the job to be executed is any of the following types, it is not possible to send a job execution request to any server:

- Demand jobs (jobs submitted from a job folder or from a non-folder management jobs)
- Jobs with Interstage attributes
- JCL (Job Control Language) jobs

When sending a job request from a schedule server to an execution server, send the request to the same subsystem number as the subsystem number on the server making the request, as shown in the following diagram.



1.2.3 Multi-server Monitoring Operations

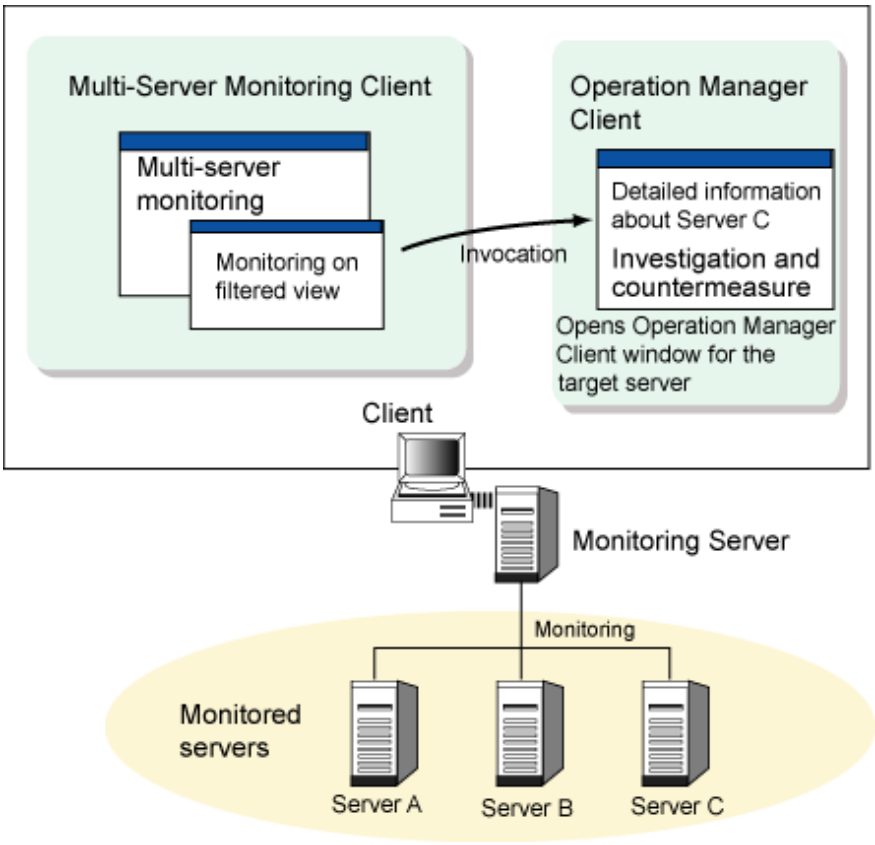
Using the multi-server monitoring client, the Jobscheduler services or daemons operating on multiple Systemwalker Operation Manager servers can be monitored. Certain job nets on monitored servers can be filtered and monitored or executed as a batch. It is also possible to investigate detailed information about the status of job nets on each monitored server. If action needs to be taken, it is possible to connect from the multi-server monitoring client to each monitored server and start the Systemwalker Operation Manager client.

From the client, the entire system can be monitored using the display in the multi-server monitoring client window. At the same time, investigations and actions can be performed from the Systemwalker Operation Manager client windows that are connected to each monitored server.

Multiple tree structures (referred to as "monitored host configuration names") of server groups and monitored servers can be registered.

Monitoring job nets on multiple servers using the multi-server monitoring client is referred to as "multi-server monitoring" or "multiple system monitoring."

The following figure illustrates multi-server monitoring operations.



Job nets for all users of monitored servers can be monitored, either for each separate server or for each separate server group. Administrators can also monitor environments that contain a mixture of Windows and UNIX servers.

EE When multi-subsystem operations are used, administrators can monitor each subsystem separately.

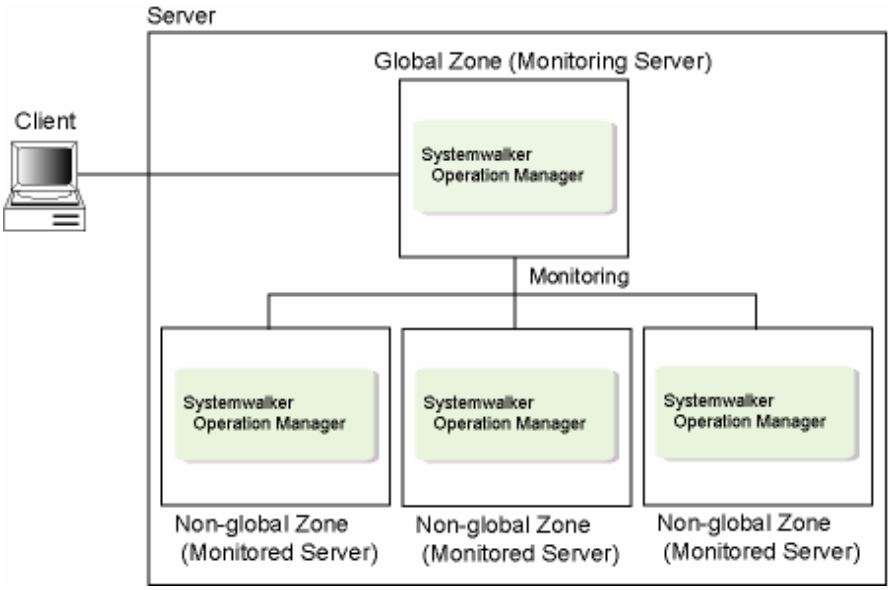
Whether jobs can be viewed or manipulated in multi-server monitoring depends on the privileges of the users on the monitored servers. Also, these operations are useful for small-scale systems on which only Systemwalker Operation Manager has been installed. For monitoring medium- or large-scale systems, Fujitsu recommends using Systemwalker Centric Manager. For information about monitoring using Systemwalker Centric Manager, refer to "2.11 Linking to Systemwalker Centric Manager."

For information about combining monitoring servers and monitored servers, refer to "A.2 Connection Support for Multi-server Monitoring."

Information

The Zone function supported by Solaris 10 or later allows the following operations to be performed.

Systemwalker Operation Manager operates by regarding each zone as a single server. Using the logical IP addresses that have been assigned to each zone, all of the Jobscheduler daemons running on multiple zones can be monitored together.



EE

1.2.4 Operations for Daily Schedule Management

During normal operations, a single database is used to control both the definition information for jobs and job nets and the schedule information for each system (or subsystem). Schedule information is updated each time the process date changes.

When the Systemwalker Operation Manager Master Schedule Management function (hereafter referred to as the "Master Schedule Management function") is enabled, schedule information is combined with the definition information for jobs and job nets, and managed as a single "master data" set for each system (or subsystem). The schedule information is created by distributing it across process dates based on the master data. By using it, an operation with only a specific date that is not a routine task can be set in advance and the original settings can be automatically restored after the operation has been performed.

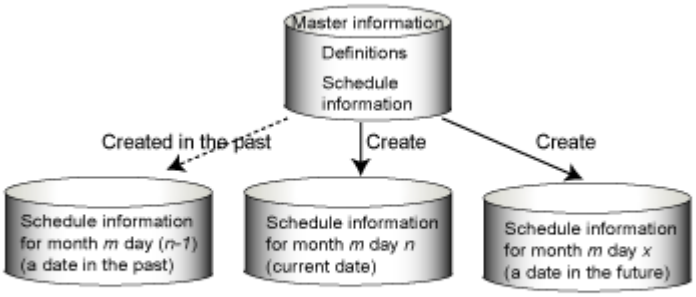
The Master Schedule Management function creates the schedule information for each process date by extracting only the data required for that process date from the master information. Accordingly, if, for example, there are a large number of job nets that are only executed once a month, the created schedule information will be smaller and have better execution performance.

Note also that in UNIX, the server load from schedule processes can be balanced by distributing it across schedule servers.

Normal operations

The schedule information is updated when the date is changed.

When the Master Schedule Management function is enabled



Schedule information is created for each operation day and depends on each operation day.

When the Systemwalker Operation Manager Master Schedule Management function is enabled, the following operations become possible.

- Daily schedule management

  Schedule information can be held for each process date. Past, present and future schedule information sets can be managed separately for different days.

- Daily schedule distribution [UNIX version]

  Schedule information can be distributed to up to five schedule servers and used for job scheduling. Definition information for jobs and job nets can be stored on the management server, and jobs can be executed on distributed servers based on the schedules stored on schedule servers.

- Operation changes

  If information for jobs and job nets changes significantly due to significant changes to the content of jobs, the modified definition information and schedule information can be registered in advance. When the release date arrives, the current master data can be swapped with the new master data automatically. Jobs and job nets can then be executed based on the new definition information and schedule information.

- Carried over job nets

  Job nets are managed for each process date. Job nets that span multiple process dates (referred to as "carried over job net") hold information such as which process date they started on, and their status when the process date changed. Users can specify how to handle job nets that span more than one process date, and can take actions if necessary. This makes it possible to control jobs taking process dates into account.

- Monitoring and operating schedule information

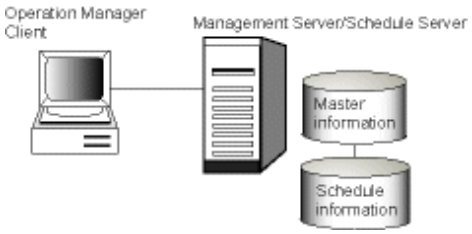
  The status of schedule information, which is managed separately for each process date, can be monitored via the GUI on the Operation Manager Client, and operations relating to schedules can also be performed. Job nets that span more than one process date can be monitored and operated from the GUI on the Operation Manager Client.

The following section explains operation configurations where the Master Schedule Management function is enabled, and is organized as follows.

- System operations controlled by a single server
- System operations controlled by multi-servers

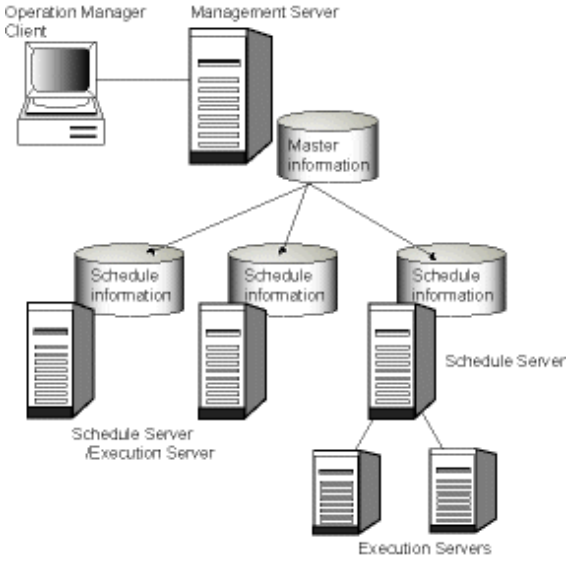
System operations controlled by a single server

In this configuration, a single server is used to manage and schedule jobs (acting as both a management server and a schedule server).



System operations controlled by multi-servers [UNIX version]

In this configuration, definition information is stored on the management server, and schedule information is distributed to schedule servers. Schedule information can be distributed to up to five schedule servers.



Point

Guidelines for using the Master Schedule Management function

The Master Schedule Management function should be used if:

- There are job nets whose execution conditions may not be met on the same process date or whose execution takes a long time, so that they often do not complete within the same process date.
- There are job nets that do not complete within the same process date, and the execution schedule for the next day needs to be controlled.
- A large number of jobs or job net definitions exist, but only a limited number of job net operations are performed per day, because there are a large number of job nets that are only executed once a month.

Note that the Master Schedule Management function should be used to distribute the load across schedule servers in cases such as the following:

- A large number of jobs or job net definitions exist, inhibiting the execution of other jobs on that process date.

As a guideline, a single subsystem can process up to 20,000 jobs or 2,000 job nets, and a single server can process up to 100,000 jobs or 10,000 job nets. If there are more jobs or job nets, the performance of the entire system can be improved by distributing to multiple schedule servers using the Master Schedule Management function.

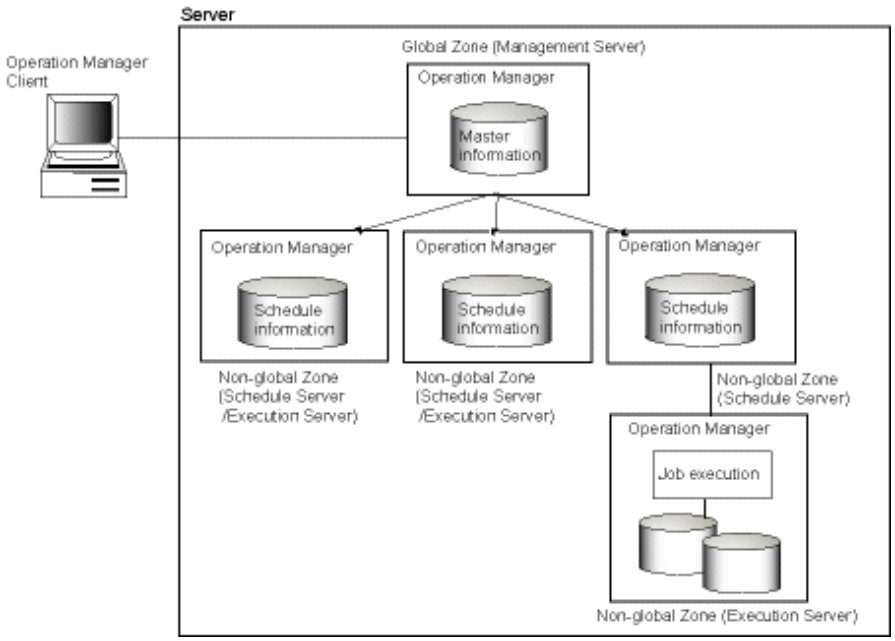
However, these process counts may vary depending on the system operating environment and operation requirements. When handling a large amount of jobs and job nets, perform thorough testing during the system operation design phase by referring to "Tuning of Performance" in the *Systemwalker Operation Manager User's Guide*.

Execute normal operations in cases such as the following:

- Jobs and job nets are completed before the end of the process date on which they are executed, and are rarely carried over until the next day to be executed.

Operation configurations with multiple zones

In this configuration, the Zone function supported by Solaris 10 or later is used to store definition information in the Global Zone and distribute schedule information to Non-global Zones.



Refer to the *Systemwalker Operation Manager User's Guide - Master Schedule Management* for information on how to operate the Master Schedule Management function.

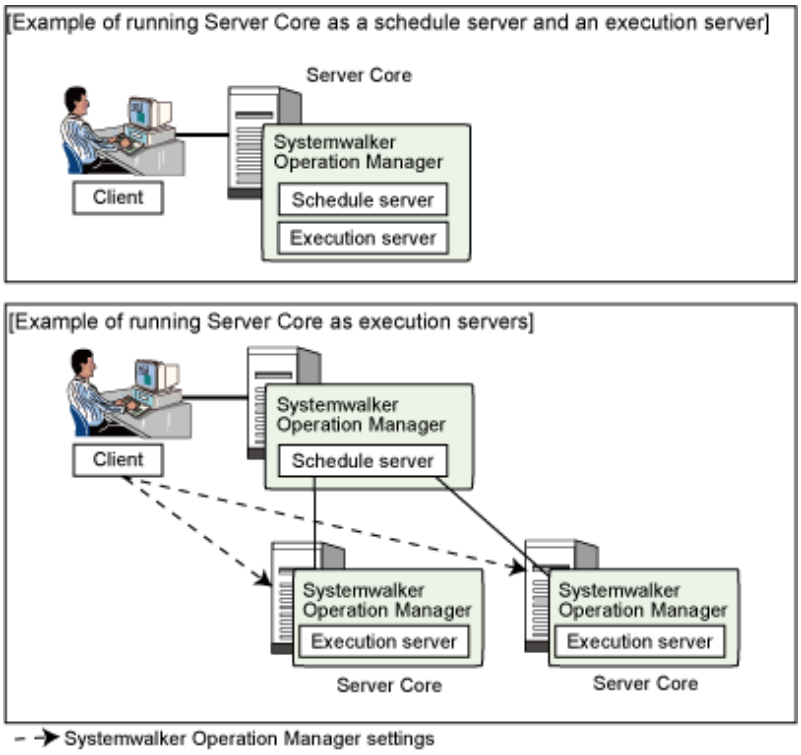
1.2.5 Systemwalker Operation Manager Operations in a Server Core Environment

Systemwalker Operation Manager can be installed in Server Core environments to perform job operations.

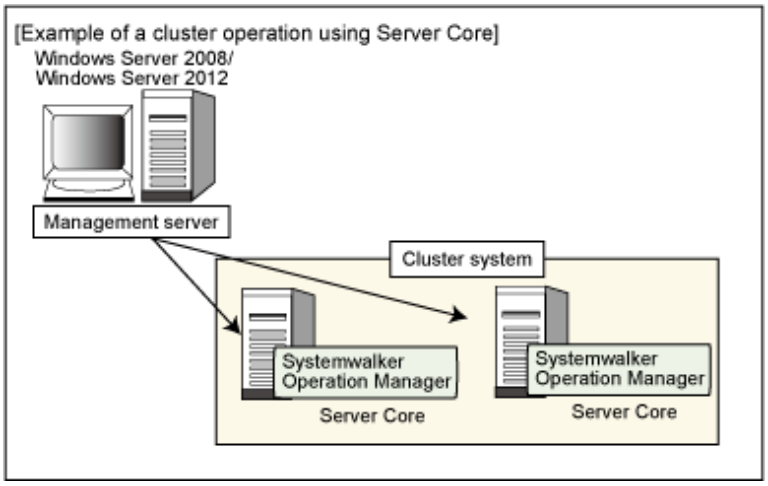
The following diagram shows a sample system configuration with Systemwalker Operation Manager installed in a Server Core environment to perform job operations.

The server is the only installation type that can be installed in a Server Core environment. Note that the optional **Client Functions of Systemwalker Operation Manager** and **Document** function cannot be used in a Server Core environment.

A Windows computer running the Systemwalker Operation Manager client should be deployed separately to set up a Systemwalker Operation Manager in a Server Core environment and to perform operations using a GUI.



To use a cluster configuration in the Server Core environment, deploy a separate computer to manage the Microsoft(R) Failover Clustering system.



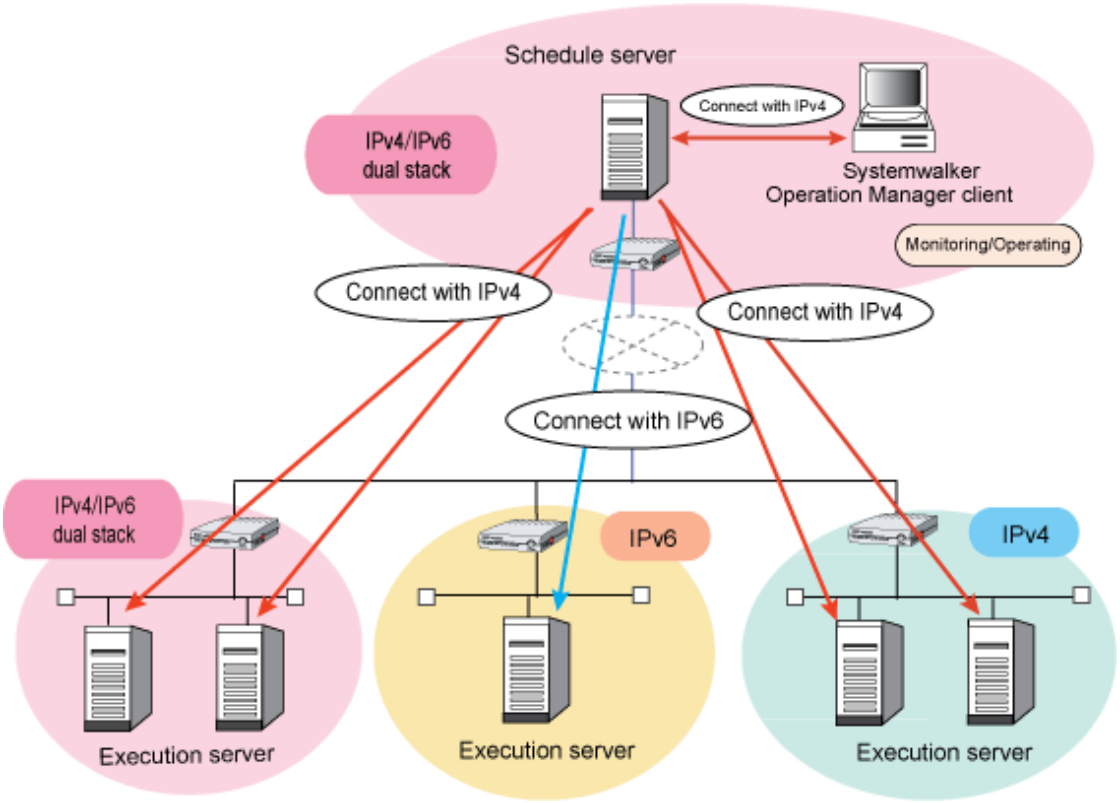
Note

Audio notification actions cannot be executed on servers in the Server Core.

1.2.6 Operations in the IPv6 Communications Environment

You can operate jobs by installing Systemwalker Operation Manager onto machines operating in the IPv4 or IPv6 communications environment. Systemwalker Operation Manager can even be used in environments that include a mix of IPv4 and IPv6.

The following diagram shows the system configuration connections in an environment that includes a mix of IPv4 and IPv6.



Communications environments in which Systemwalker Operation Manager can operate

Systemwalker Operation Manager can operate in the following environments. Also, even in environments that include a mix of communications environments on the network, you can operate jobs using Systemwalker Operation Manager.

- IPv4 single stack environment
- IPv6 single stack environment (*1)
- IPv4/IPv6 dual stack environment (*2)

*1: When operating in an IPv6 single stack environment, refer to "Operating in an IPv6 single stack environment" in "Notes for IPv6 environments".

*2: When operating in an IPv4/IPv6 dual stack environment, refer to "Operating in an IPv4/IPv6 dual stack environment" in "Notes for IPv6 environments".

IPv6 addresses that can be used

When operating jobs with Systemwalker Operation Manager, the types of IPv6 addresses that can be used are as follows:

- Global unicast addresses
- Unique local unicast addresses

IPv4/IPv6 address selection

Systemwalker Operation Manager prioritizes the use of IPv4 addresses in IPv4/IPv6 dual stack environments.

You can also connect to Systemwalker Operation Manager servers using version 13.4.1 or earlier in environments that mix IPv4 and IPv6 addresses. In this case, the IPv4 addresses are also used.

IPv6 address formats for input/output

The following IP address formats are supported for window input/output, command input/output, message output, and log/trace output when using IPv6 addresses in Systemwalker Operation Manager:

Format types	IPv6 address examples
Non-compression format (*1)	2001:0db8:0000:0000:0123:4567:89ab:cdef
Zero compression format (*1)	2001:0db8::0123:4567:89ab:cdef ("0000" compressed)
RFC 5952 compliant format	2001:db8::123:4567:89ab:cdef

*1: Uppercase/lowercase/mixed upper and lowercase alphabetic letters can be used.

Notes for IPv6 environments

This section provides notes for using Systemwalker Operation Manager in an IPv6 communications environment.

Operating in an IPv6 single stack environment

- When operating in an IPv6 single stack environment, do not uninstall IPv4 as below:
  [Windows]
  - IPv4 uninstallation command

    ```
    netsh interface ipv4 uninstall
    ```

  If you unintentionally uninstall IPv4, install IPv4 once again:
  - IPv4 installation command

    ```
    netsh interface ipv4 install
    ```

- You cannot build Web servers to be used in the Systemwalker Operation Manager Web Console in an IPv6 single stack environment.

  Therefore, use the Systemwalker Operation Manager client to monitor jobs using the Systemwalker Operation Manager server.

  You should operate with an IPv4/IPv6 dual stack environment when building a Web server.

Operating in an IPv4/IPv6 dual stack environment

When running a Systemwalker Operation Manager server in an IPv4/IPv6 dual stack environment, define both the IPv4 address and IPv6 address set for the local host name in the hosts file on the server.

Using Systemwalker User Management function/Systemwalker single sign-on

When using the Systemwalker User Management function or Systemwalker single sign-on, the communications environments for Systemwalker Operation Manager server and the Systemwalker Single Sign-On Server must both be IPv4/IPv6 dual stack environments.

You cannot use the Systemwalker User Management function and Systemwalker Single Sign-On in an IPv6 single stack environment.

Power Control function

You can use the Power Control function only when power control devices and power control software both support IPv6.

Setting the monitored host for multi-server monitoring

If the internet protocol versions of the respective monitored host definitions and the monitoring-permission host definitions between the monitoring server and the monitored server match, you can use it to monitor multiple Systemwalker Operation Manager servers.

Refer to "Define Monitoring Permission Host" in the *Systemwalker Operation Manager Installation Guide* for details.

Performing multi-server monitoring from a multi-server monitoring client of V13.3.1 or earlier

When there is a monitored server for an IPv6 communications environment or when a monitored server for an IPv4 communications environment and monitored server for an IPv6 communications environment coexist, all of these monitored servers cannot be monitored.

PONCLI.EXE/poncli power on command

You cannot use PONCLI.EXE or the poncli power on command in IPv6 single stack environments. Use it in an IPv4 single stack environment, or in an IPv4/IPv6 dual stack environment.

Executing network jobs from the cluster system configuration's schedule server

When executing a network job from the cluster system configuration's schedule server, define the logical IP address to be set to the schedule server, according to the communications environment of the execution server, as follows:

- When the execution server is in an IPv6 single stack environment:

  IPv6 addresses

- When the execution server is in an IPv4 single stack environment or an IPv4/IPv6 dual stack environment:

  IPv4 addresses

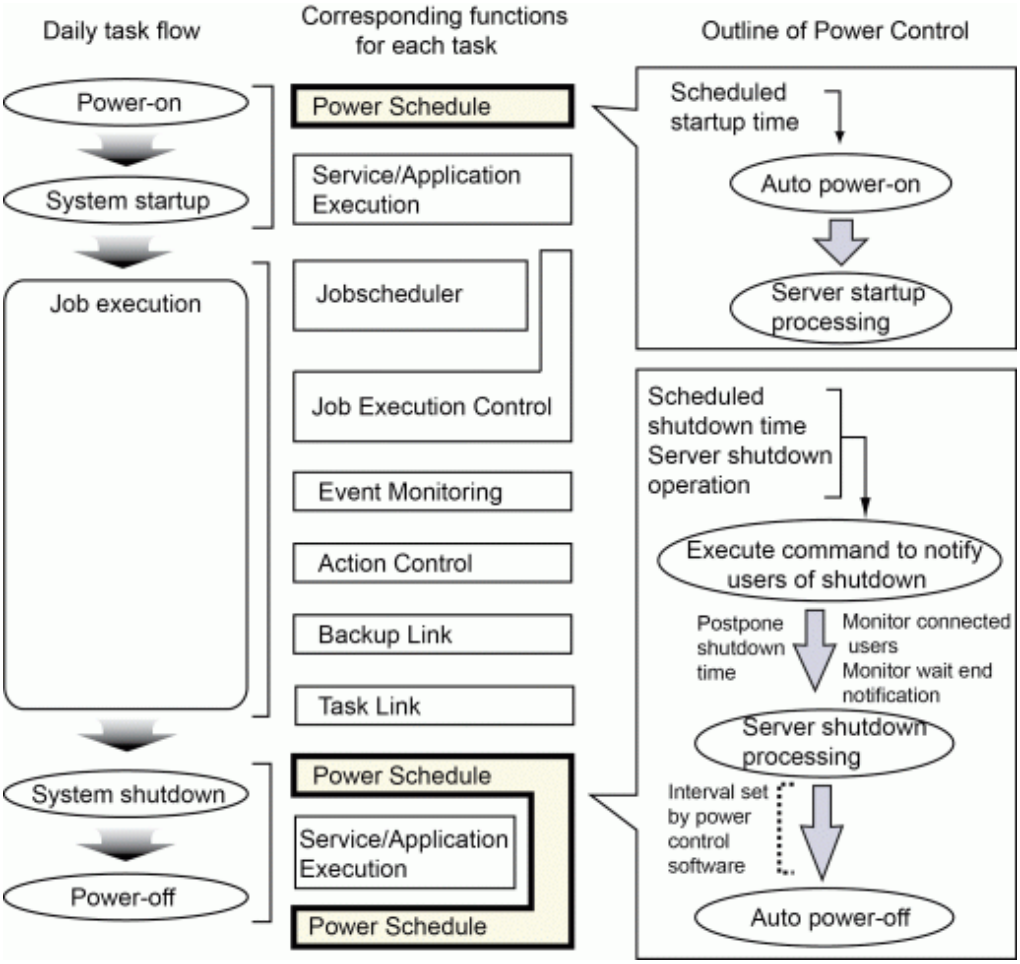
Note that you cannot operate an execution server in an IPv6 single stack environment if it is mixed with an IPv4 single stack environment or an IPv4/IPv6 dual stack environment execution server.

1.3 Outline of Systemwalker Operation Manager Functions

This section outlines Systemwalker Operation Manager functions.

1.3.1 Power Control

Servers can be turned on and off automatically, and also rebooted automatically using the Power Control function. The following figure outlines the Power Control function.



Automatic Power Control (turning the power on and off)

Servers can be turned on or off or rebooted automatically according to preset power schedules. Power schedules specify the dates and times when servers are turned on and off or rebooted.

If a server does not have a power control device (hardware), the power control function cannot be used. However, the operating system can be rebooted or shut down automatically.

Multiple servers can also be controlled simultaneously. Each server must have a power control device for simultaneous power control of multiple servers.

Delaying server shutdown processing

The shutdown monitoring option can be used to delay the server shutdown time that has been set up with the power schedule. For the Windows version, if a user is still connected to a server through a network connection when the server shutdown time has come, this function automatically delays the server shutdown process until the user disconnects from the network.

In addition, shutdown processing can be delayed until a shutdown instruction is given by a command.

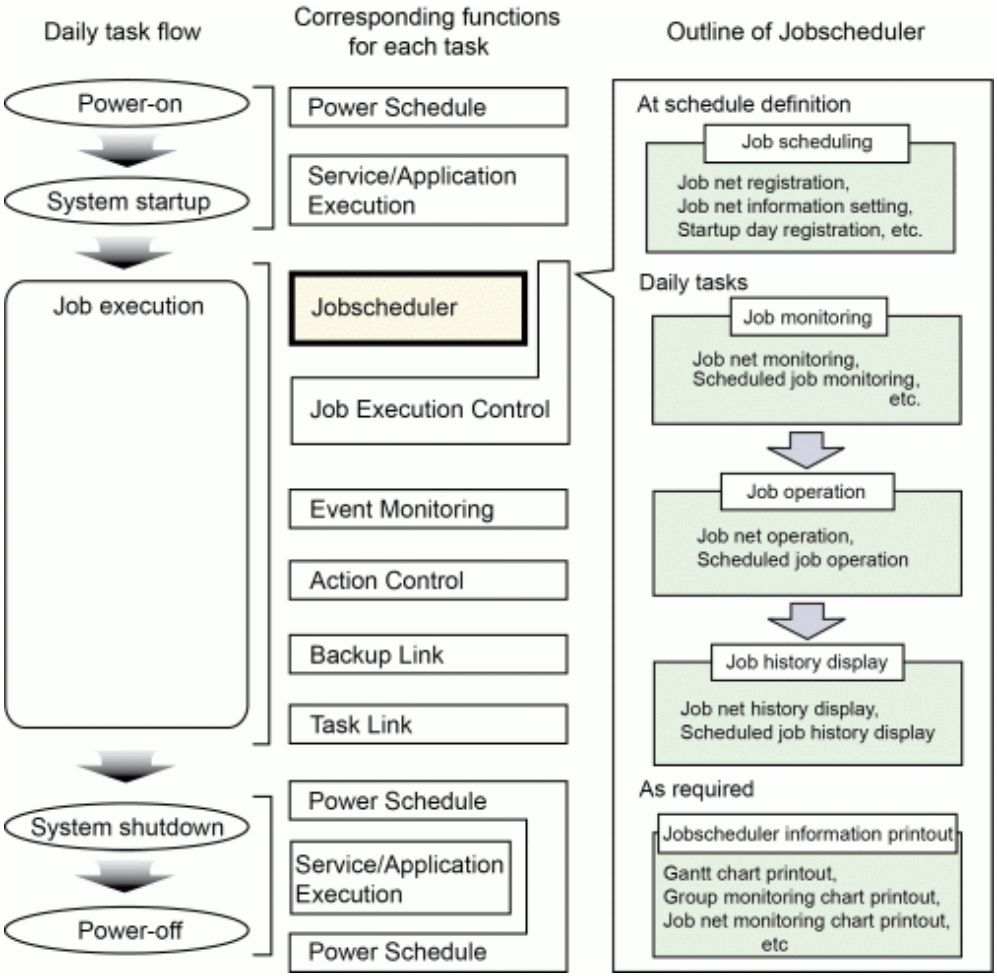
If a user is still connected to a server through a network connection, or if a shutdown command has not been issued when the delayed server shutdown time has come, this function forcibly stops any services or applications that are running before shutting the server down.

1.3.2 Job Scheduling

The Jobscheduler simplifies job operations for the entire system, including scheduling, monitoring and control. This section presents an overview of the following functions provided by the Jobscheduler:

- Automatic Job Execution

- Monitoring and Operating Jobs
- Displaying Job Histories
- Printing Jobscheduler Information



Automatic job execution

The Jobscheduler automatically executes jobs and job nets according to execution schedules that have been set, based on startup days that have been registered in advance. Each job is submitted to the Job Execution Control and then executed. Note that triggers such as event output, file transfer or mail reception can also be used to start jobs and job nets, as well as execution schedules based on startup days.

It is also possible to add recovery jobs to job nets, so that recovery actions can be executed without operator intervention if an error occurs with a job. Also, the completion code value of the preceding job can be used to select which subsequent job to branch to.

Monitoring and operating jobs

The current status of job nets and their schedule for the current day can be monitored graphically. The status of jobs and job nets is displayed in real time, with different colors allocated to each status, making it possible to check the progress of each job at a glance. The completion codes and output results of each job can also be looked up.

Jobs and job nets can be operated while their status is being monitored. An overview of the operations that are possible is give below.

As well as monitoring jobs and job nets on single systems, the Jobscheduler can also monitor jobs and job nets on multiple systems as a batch.

Cancel

This operation forcibly terminates a job or job net that is currently executing.

Start

This operation starts an urgent job or job net to be executed immediately.

Restart

This operation restarts a job or job net that has been cancelled or that has terminated abnormally.

The job net can be restarted from the job that terminated abnormally or from any other job in the job net.

By default, a completion code of "0" indicates that a job has terminated normally, and completion codes other than "0" indicate that a job has terminated abnormally. The values of the completion codes for distinguishing normal and abnormal completion can be changed if necessary.

Pause/Continue

Jobs and job nets can be paused temporarily. If a job is paused, the paused job (and all following jobs in the same job net) will not start until a "Continue" operation is made (canceling the paused status).

Disable/Enable

The execution of jobs and job nets can be skipped. Disabled jobs and job nets are not executed but enter a pseudo-normal status after the preceding jobs or job nets have been executed. As a result, the subsequent jobs or job nets are executed without being stopped. If a disabled job or job net is enabled before its turn to be executed arrives, it will be executed according to the normal schedule.

Confirm

This operation changes the status of job nets that have been canceled or that have terminated abnormally to "confirmed". If confirm operations are enabled, job nets that have been canceled or that have terminated abnormally will not start even if their execution conditions are satisfied and will not be scheduled even if the day change time arrives unless a "Confirm" operation is performed.

Confirm operations can be enabled or disabled. It is also possible to select whether job nets that have been canceled should be subject to confirm operations. If job nets that have been canceled are not subject to confirm operations, then only job nets that have terminated abnormally will be subject to confirm operations.

Reinstate

This operation cancels the execution records (such as "completed", "pseudo-normal", "abended", and "cancelled") of job nets that have already been executed during the current day and where the "Once in a Day" option has been specified with "Start only when message event occurred" in its execution conditions. The status of these job nets becomes "waiting to be executed" when they are reinstated, so that they can start again if their execution conditions are met by another message event during the current day.

Revoke

This operation cancels job nets that have been carried over. When a job net is revoked, a schedule for the current day is created and the status of the job net becomes "waiting to be executed".

Displaying job histories

Execution histories for jobs and job nets can be displayed.

Information such as the status, start time, end time, and completion code of jobs can be looked up using job execution histories. With job execution histories, the results of job execution can be viewed in message format.

Printing Jobscheduler information

The following Jobscheduler information can be printed.

- Gantt Charts:

  The execution status of past job nets and job nets for the current day can be printed out in the same format as displayed in the **Gantt Chart** window.

- Group and job net monitoring charts:

  Group monitoring charts can be printed out in the same flowchart format as displayed in the **Monitor Group** window.

  Also, job net monitoring charts can be printed out in the same flowchart format as displayed in the **Monitor Job Net** window.

- Group and job net lists

  Group information can be printed out in the same list format as displayed in the **Group Management** window.

  Also, job net information can be printed out in the same list format as displayed in the **Job Net Management** window.

- Job and Job net histories

  Job net histories can be printed out in the same format as displayed in the **Job Net History** window.

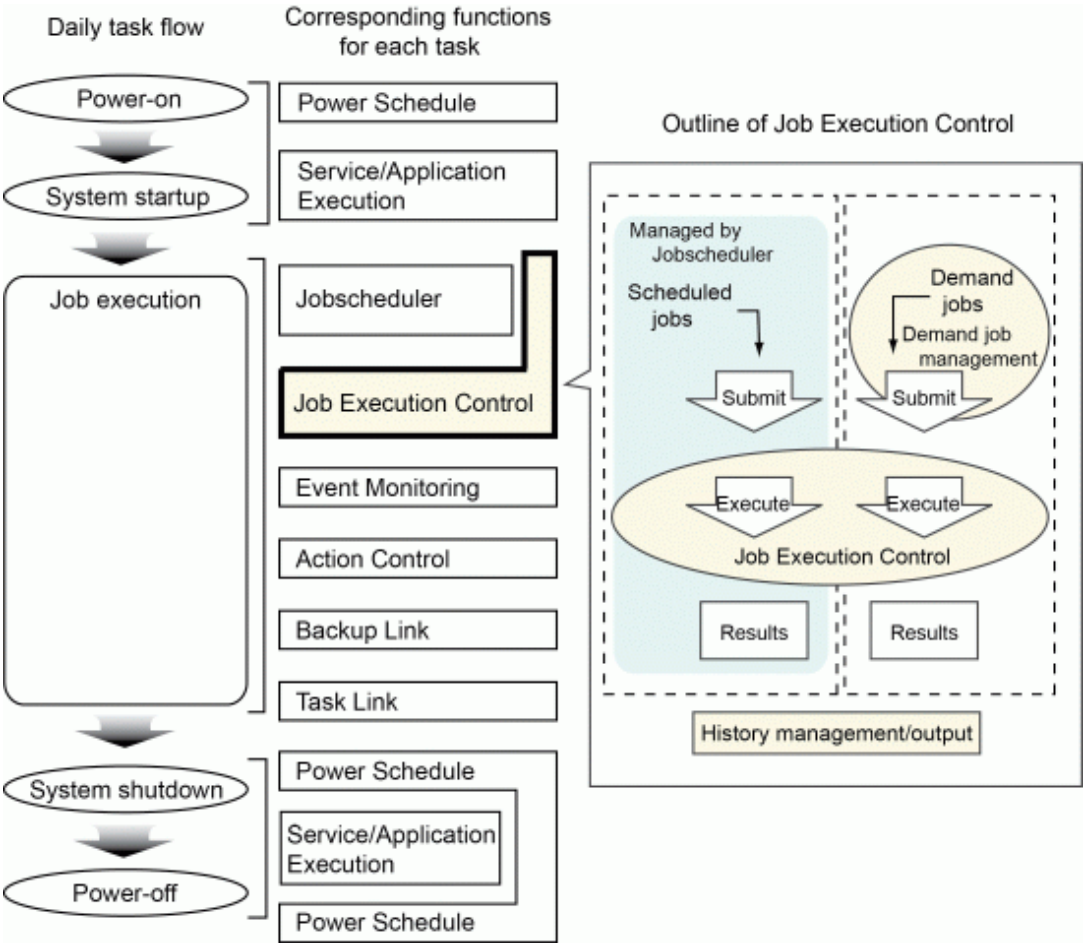
  Likewise, job histories can be printed out in the same format as displayed in the **Job History** window.

The information displayed on the screen can be filtered down to only the required items. Print items can be set up as necessary.

1.3.3 Job Execution Control

Job Execution Control efficiently controls jobs from execution through to completion, and manages and outputs job execution history information.

This function also manages demand jobs.



This section explains the Job Execution Control function, and is organized as follows:

- Controlling Job Execution Environments

- Job Execution Environments [UNIX version]
- Executing Network Jobs
- Output of Job Execution History

1.3.3.1 Controlling Job Execution Environments

The Job Execution Control function controls jobs using queues in order to manage job execution priorities, the number of jobs that can be executed concurrently, and the processing status of jobs.

When a job is submitted, it is placed in a queue and becomes subject to various execution controls. Job Execution Control processes jobs according to the following specifications.

Definitions in the Define Operating Information window:

The **Define Operating Information** window is used to make definitions about where (which of the servers where Systemwalker Operation Manager has been installed) queues should be created, and how these queues should be controlled. Although standard settings are already in place at installation time, these settings can be changed according to the operation mode. For more information about the **Define Operating Information** window, refer to the *Systemwalker Operation Manager Installation Guide* or the *Systemwalker Operation Manager Online Help*.

Multiple queues can be created. By creating queues according to specific attributes (such as departments, tasks, or execution time lengths), the efficiency of job throughput can be increased. For example, if separate queues are created for jobs with long execution times and for jobs with short execution times, then operating with rules about which queue jobs are allocated to can prevent situations where jobs with short execution times have to wait for hours for jobs with long execution times to finish.

The same definitions can also be made using an initialization file. For more information about initialization files, refer to the *Systemwalker Operation Manager Installation Guide*.

Specifications when jobs are submitted

Various specifications can be made regarding the execution methods for each job. Job Execution Control controls job execution according to these specifications.

The main controls performed by Job Execution Control are as follows:

Job Multiplicity for Systems

Job Execution Control limits the number of jobs that can be executed simultaneously throughout servers on which Systemwalker Operation Manager has been installed. If the number of jobs being processed has reached this limit, no more jobs can be executed even if the limits for each queue (that is, the job multiplicity for queues) have not been reached.

EE

For multi-subsystem operations, the number of jobs that can be executed concurrently can be limited for each separate subsystem.

Job multiplicity for queues

Job Execution Control limits the number of jobs that can be executed simultaneously for each individual queue. If the number of jobs being processed in each queue has reached this limit, no more jobs can be executed even if the limit for the entire server (that is, the job multiplicity for the system) has not been reached.

Maximum number of jobs that can be submitted to queues

Job Execution Control limits the maximum number of jobs that can be submitted to queues. This limit includes jobs currently being executed and jobs waiting for execution.

Job termination due to timeouts

Job Execution Control limits the time that jobs can be executed for each queue. Jobs cannot be executed for longer than this time. Jobs are forcibly terminated when they reach this limit.

Priority of jobs waiting to be executed

The priority of jobs waiting to be executed can be specified. Jobs with the highest priority are executed first.

Job execution priority

The priority of jobs that are being executed can be specified. Priorities are specified using a numerical value that indicates the priority for allocating CPU resources to the processes that run as jobs when jobs are executed.

If the connection destination server is running the Windows version, job execution priorities between 0 and 4 can be set. The higher the specified value, the higher the CPU allocation priority. The smaller the specified value, the lower the CPU allocation priority.

If the connection destination server is running the UNIX version, job execution priorities between -20 and 19 can be set. The specified value corresponds to the nice value used in UNIX systems. The smaller the specified value, the higher the CPU allocation priority. And the higher the specified value, the lower the CPU allocation priority.

If the connection destination server is running the UNIX version, values between 0 and 39 may be set depending on the screen where settings are entered. In this case, the higher the specified value, the higher the CPU allocation priority (and the smaller the specified value, the lower the CPU allocation priority).

For network jobs between different platforms (Windows and UNIX) the specified execution priority level will be converted, as shown below.

- Execution priority levels used when network jobs are sent from UNIX to Windows

	Source (UNIX)	Destination (Windows)
Execution priority level	-20 to -13, or 39 to 32	4
	-12 to -5, or 31 to 24	3
	-4 to 3, or 23 to 16	2
	4 to 11, or 15 to 8	1
	12 to 19, or 7 to 0	0

- Execution priority levels used when network jobs are sent from Windows to UNIX

	Source (Windows)	Destination (UNIX)
Execution priority level	4	-20, or 39
	3	-10, or 29
	2	0, or 19
	1	10, or 9
	0	19, or 0

The values of the job execution priority level are specified based on the priority levels shown below.

Priority level	Method used to specify the execution priority level
1	Execution priority level set using the source Jobscheduler, or qsub command -dp option
2	dfltdprty of queue that was set in the source operation information definition
3	dfltdprty of queue that was set in the destination operation information definition

Note

If 'continue' has been set as the continuous execution mode, then network jobs will continue if the schedule server system crashes. As a result, the job priority level that was reset by the schedule server system crash will not be reflected in the jobs that are currently being executed on the execution server.

If 'cancel' has been set as the continuous execution mode, network jobs will be forcibly terminated when the schedule server system crashes. For this reason, the job execution priority level that was reset after the system crash will become effective when these jobs are re-executed on the execution server.

Exclusive resource control

Job Execution Control specifies exclusive attributes so that multiple jobs that use the same resources are not executed simultaneously.

Excluding identical job names

With Job Execution Control, exclusive attributes can be specified so that multiple jobs with the same name are not executed simultaneously.

1.3.3.2 Job Execution Environments [UNIX Version]

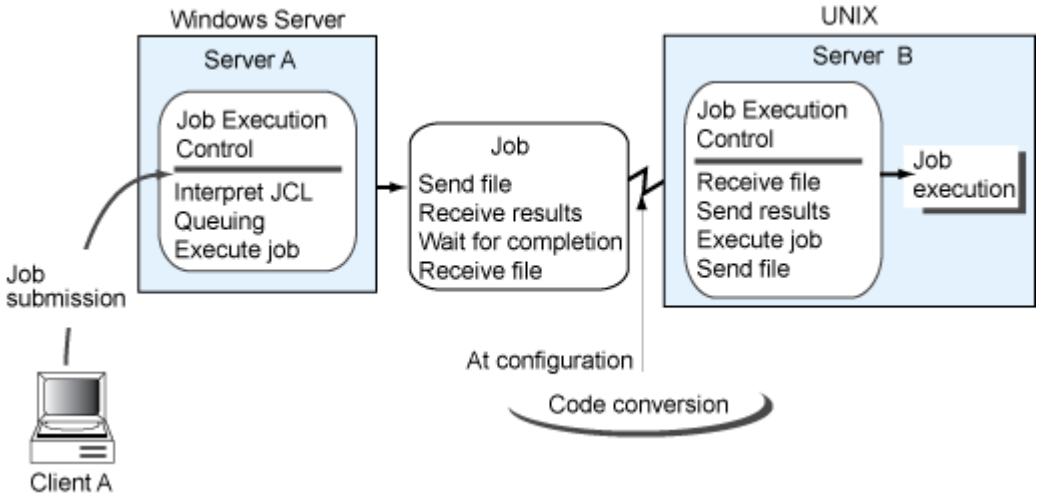
Job Execution Control starts a shell for interpreting shell scripts for jobs. The type of shell started is the login shell of the user who submitted the job (that is, the login shell defined in the password file).

When jobs are executed, the shell interprets shell scripts by adjusting the login environment (i.e., executing the ".profile", ".login", ".cshrc" and other startup files).

1.3.3.3 Executing Network Jobs

Systemwalker Operation Manager can execute jobs on any server connected to the network where Systemwalker Operation Manager has been installed. Jobs that are executed on arbitrary servers connected via the network are called "network jobs." The execution results of such jobs can be viewed from the server where the job was submitted. The following servers are supported as execution servers for network jobs: Windows Server, UXP/DS, Solaris, AIX, HP-UX, and Linux.

The following figure shows a representation of network job execution. In this example, client A submits a job to Windows server A, and requests UNIX server B to execute the job.



Information

Code conversion

If network jobs are executed using servers with different character encodings, job file codes can be converted using the Code Conversion function. This function can convert codes (including linefeed codes) for job files, standard output files, and standard error output files.

To perform code conversion, make definitions in the **Define Operating Information** window of the server where the job is submitted.

For information about making definitions with the **Define Operating Information** window, refer to the *Systemwalker Operation Manager Installation Guide*.

1.3.3.4 Output of Job Execution History

The following files can be output to servers as job execution information.

Log file:

This file contains the job execution history.

Operation performance data file:

This file contains information about the operational results of jobs.

These files are created every day and saved for the specified number of days.

1.3.4 Operation Dependent Functions

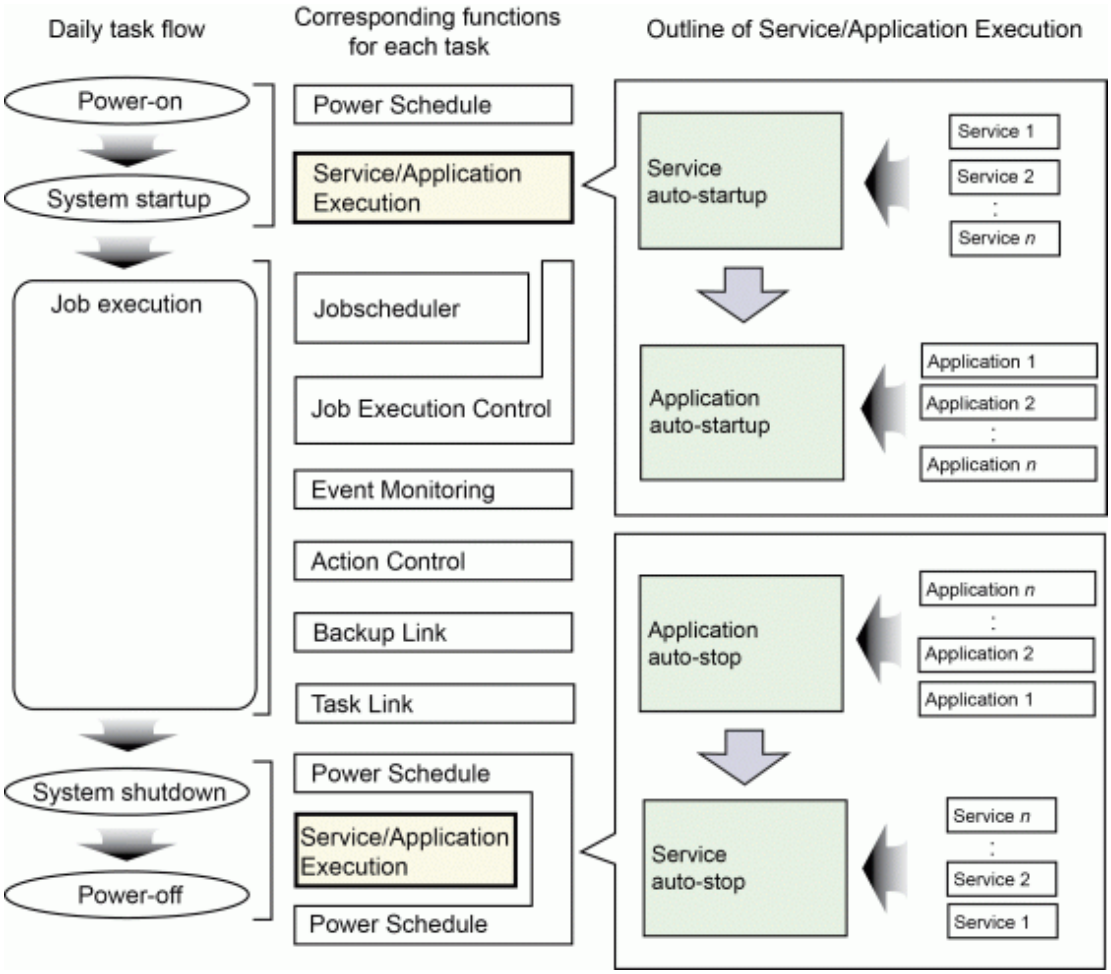
The following explains other operation dependent functions.

1.3.4.1 Services and Applications Startup

The Service/Application Execution function automatically creates a job environment when the operating system starts, using the following functions. This section presents an overview of these functions.

- Service Startup function [Windows version]

- Application Startup function



Service Startup function [Windows version]

This function starts services automatically according to a predefined service schedule when the server is turned on and the operating system starts up. Service schedules specify the services to be started and the sequence in which these services start.

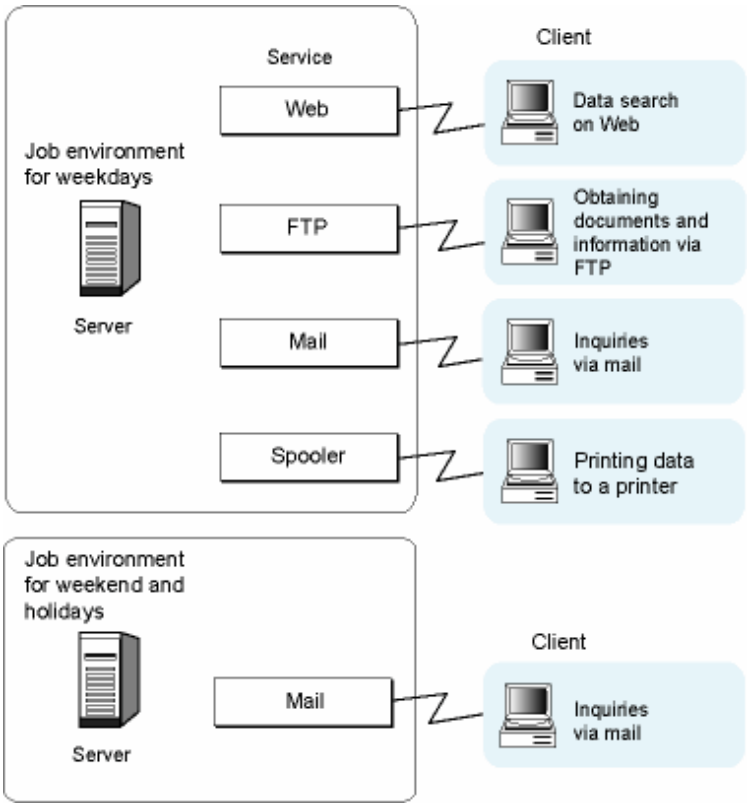
The services that are started by this function are terminated in the reverse procedure of startup before the server shuts down.

Application Startup function

This function starts applications automatically according to a predefined schedule after the services have started. Application schedules are information specifying the applications to be started and the sequence in which these applications start.

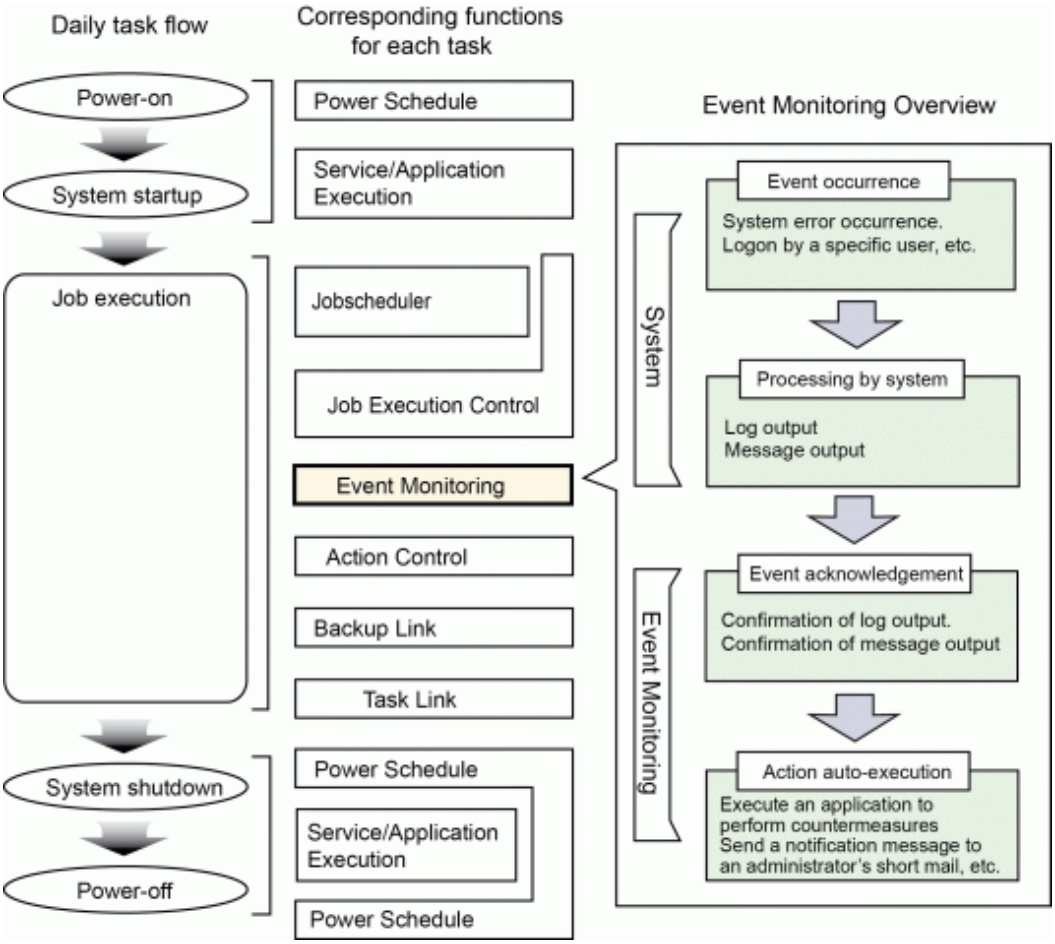
The activated applications are terminated in the reverse procedure of startup before the server shuts down.

The services and applications that are started can be changed for weekdays or weekends, or for specific days, making it possible to create different job environments for different days. The following example shows an operation with different job environments for weekdays and holidays.



1.3.4.2 Monitoring Events [Windows Version]

The Event Monitoring function makes it possible to perform actions (such as "sending Short Mails to the system administrator" and "starting an application to handle an event that has occurred") automatically without human intervention if an event such as "output of a message indicating the system failure" occurs. Which actions are executed automatically can be changed depending on weekdays or weekends, or certain times of day.



The following events can be monitored using the Event Monitoring function:

- Event log

  Messages that are output to the event log file.

  The format of these messages can be checked using the event viewer for Windows Server.

- Log file monitoring

  Additions to the text that is output to log files.

  The additional part is monitored as a message. These messages have the following format: "label-name + single-line-message".

- Monitoring messages

  The messages handled by Systemwalker Centric Manager (System Monitor).

  These are messages that are relayed from a system monitoring agent located at a lower level of logical hierarchy than the definition destination system. These messages can be monitored if Systemwalker Centric Manager has been installed.

  The format of these messages can be checked on the event list shown in the Systemwalker Centric Manager monitoring window.

The Event Monitoring function can perform the following automatic actions:

- Audio notification

  (Text-to-voice) message readout, WAV file playback, or beep sound

- E-mail transmission (including E-Mails)

  Notification of any document via e-mail

- Pop-up message display

  Notification of any message by pop-up message

- Short Mail notification

  Notification of any message by Short Mail

- SNMP trap transmission

  Issuing SNMP traps

- Application call

  Starting applications

- Message monitoring (if Systemwalker Centric Manager has been installed)

  Listing relevant events in the Systemwalker Centric Manager event monitoring window

- Event log output

  Outputting any message to the event log

- Remote commands (if Systemwalker Centric Manager has been installed)

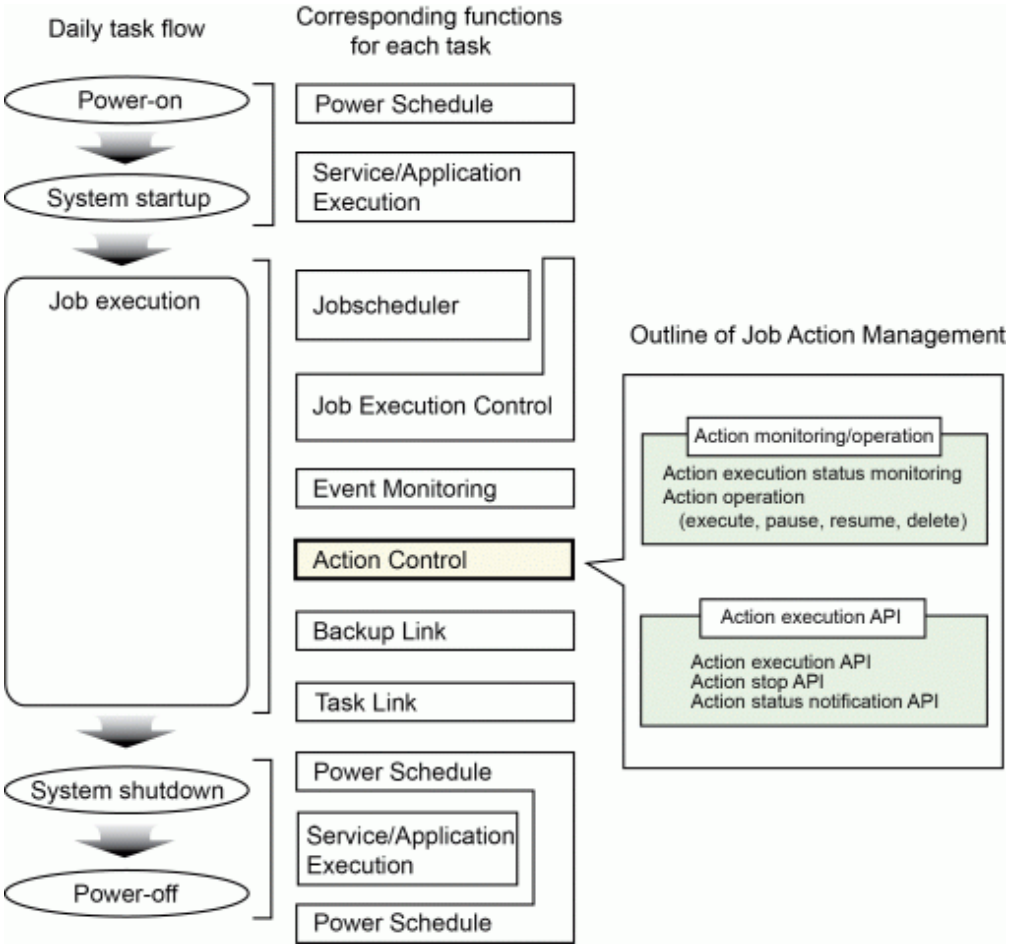
  Issuing remote commands via the Systemwalker Centric Manager System Monitoring function

1.3.4.3 Managing Actions [Windows Version]

The Action Control function can monitor and operate actions (such as audio notification, e-mail transmission, pop-up message notification, and Short Mail notification) that are executed automatically by the Event Monitoring function. This section outlines the following functions.

- Action Monitoring and Operation

- Action Execution API



Action monitoring and operations

The execution status of actions can be monitored using the Action Control function. It is possible to display only the type of actions that need to be monitored in the **Action control** window, so the execution status of actions can be monitored at a glance. Detailed information about each action can be displayed, as well as a list of actions.

In addition, actions can also be paused and deleted. If actions cannot be executed (because there are too many actions to be executed, for example), certain actions can be executed before other actions by changing the order of action execution.

Action execution API

The Action Control function provides APIs for executing actions.

For example, by registering a program created using these APIs with the Jobscheduler as a job, actions can be scheduled and the system administrator can be notified by Short Mail if the job terminates abnormally.

For more information about action execution APIs, refer to the *Systemwalker Operation Manager Reference Guide*.

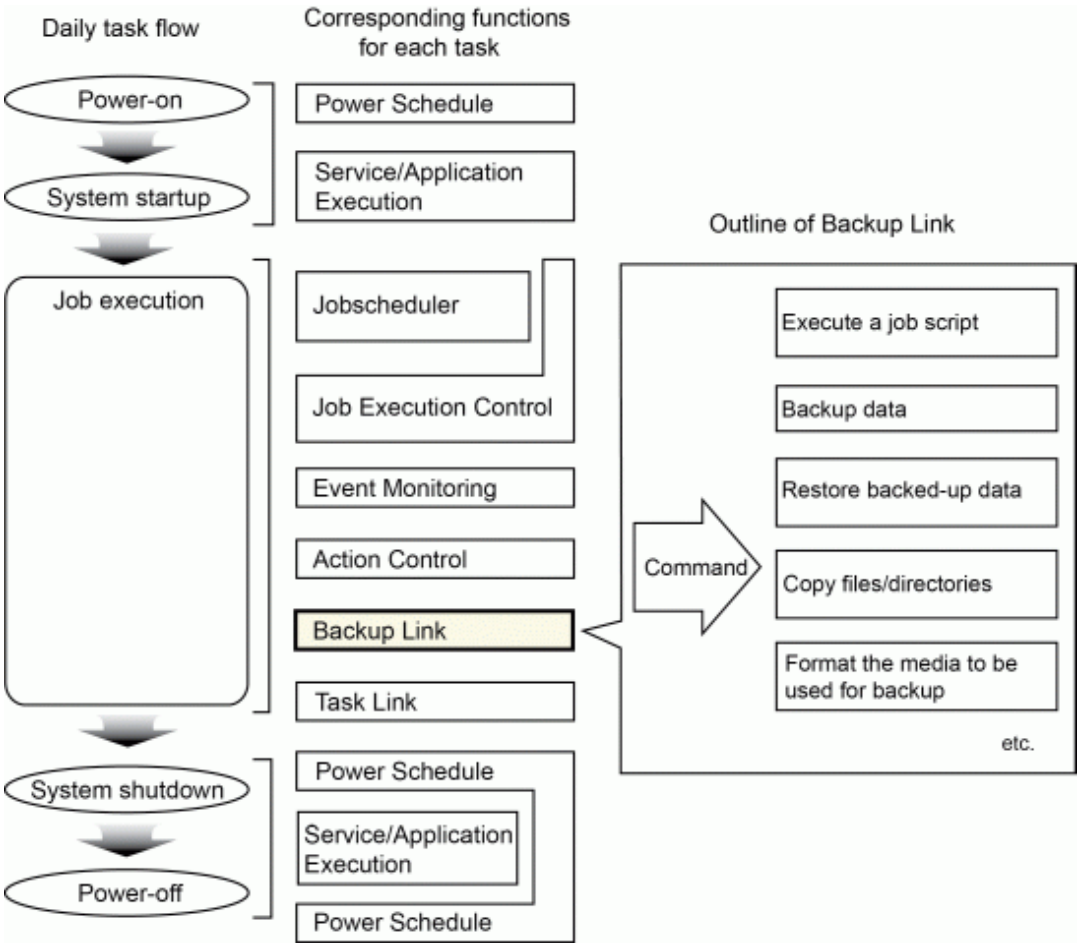
1.3.4.4 Backup Link [Windows Version]

Backup Link allows the ARCserver function (the program for automatic data backup) to be performed by issuing commands.

Backup Link makes it possible to do the following:

- Execute job scripts
- Back up data
- Restore backup data

- Copy files and directories
- Format the media to be used for backup

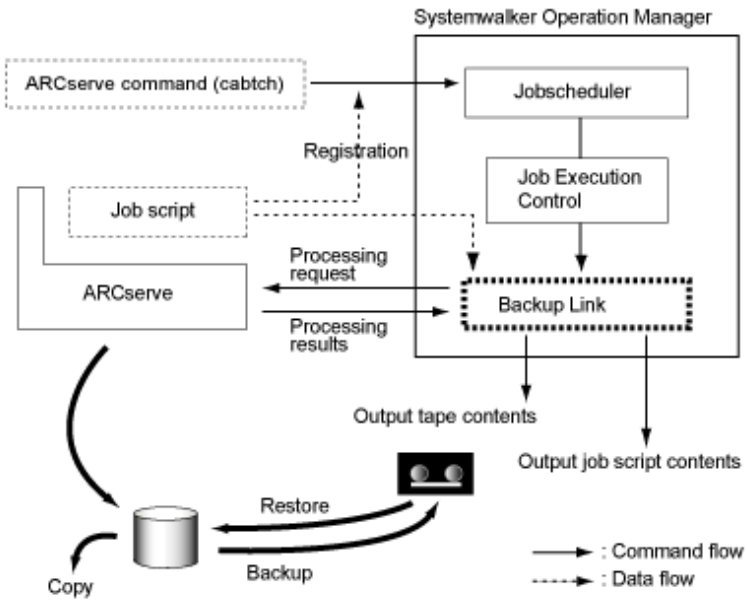


ARCserve is required to use Backup Link. Backup Link can be performed using the ARCserve command (*cabatch*).

For information about how to perform Backup Link, refer to the ARCserve manuals.

The following figure shows a representation of operations where the ARCserve command (*cabatch*) is registered with the Jobscheduler.

Operation flow



Note

Backup Link command (mpsubasx) is not available in ARCserve r11.5 or later.

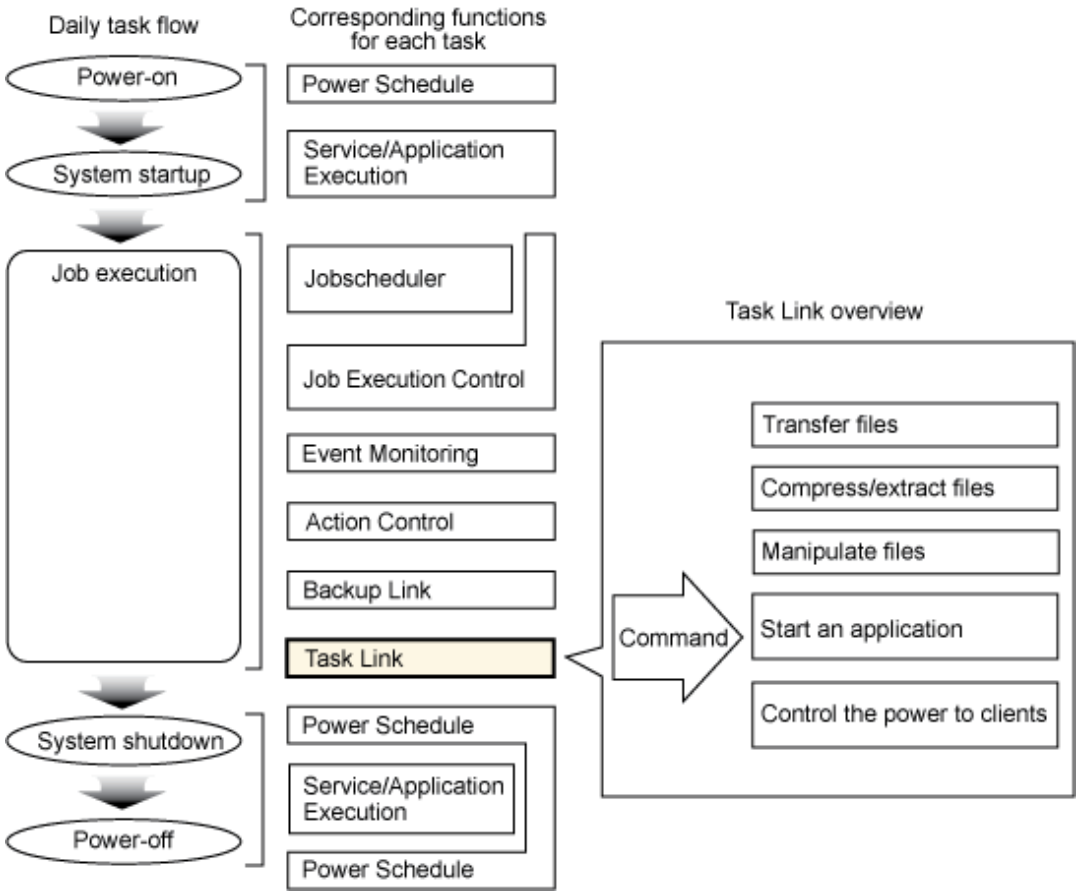
1.3.4.5 Task Link

The Task Link function enables job data to be exchanged between servers or between a server and clients by using commands.

The Task Link function makes it possible to do the following:

- Transfer files
- Compress and extract files
- Manipulate files
- Start an application

- Turn the power to clients on and off



Using the Task Link commands, the following processing (which has required JCL programs or batch files until now) can be simplified.

- Processing between servers
    1. Compress a data file on a server and transfer it to another server.
    2. Extract the file on the destination server and start an application.
    3. Return the results of processing back to the source server, and update the original data.
- Processing between a server and clients
    1. Turn the power to a client on.
    2. Compress a data file on a server and transfer it to the client.
    3. Extract the file on the destination client and start an application.
    4. Return the results of processing back to the source server, and update the original data.
    5. Turn the power to the client off after processing completes.

The basic operation mode of the Task Link function is to register Task Link commands with the Jobscheduler and then process these commands automatically.

Chapter 2 Operating Methods

This chapter presents an overview of the data required for operating Systemwalker Operation Manager.

2.1 Power Control

This section explains how to operate the Power Control function.

2.1.1 Basic Operations

The basic operation of the Power Control function is to define triggers for turning the power to servers on and off, or for rebooting servers, using the **Power Control Pattern** dialog box. Only users with Administrator privileges can create or modify these definitions. Definitions are applied automatically to days that have not been defined as holidays. To use different triggers (for rebooting servers and turning the power off and on) on different days, create separate definitions and apply each definition to different days.

When the defined time comes, the Power Control function links to a power control device and turns the power to the server on or off, or reboots the server.

The Power Control function is only supported by Windows. However, note that the Power Control function cannot be used in the Windows x64 version if linked to a power control device.

The basic operations for the Power Control function are explained below.

For information about how to operate the Power Control function, refer to the *Systemwalker Operation Manager User's Guide*.

(1) Setting up holidays

Days when the server does not operate can be set up as holidays in the calendar (SYSTEM_CALENDAR).

This step can also be done after steps (2) to (5) below have been performed.

(2) Setting up day change

Set up the day change time to match the server operation. The day change time is the time of day when the date changes.

(3) Setting up the power control method

Set up whether to control the power to the server. If the power is controlled, specify whether to control the power to the server individually, or to control the power to multiple servers as a batch.

(4) Setting up the power schedule

Set up the dates and times when the power to the server will be turned on or off, or when the server will be rebooted, in a way that matches the server operations.

(5) Setting up the completion monitoring options

Set up the completion monitoring options to control shutdown processing for the server, by postponing the server shutdown time that has been set in the power schedule, for example.

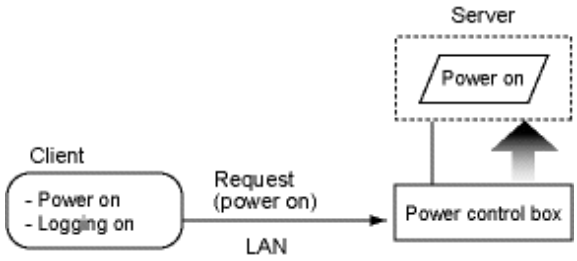
2.1.2 Additional Operations

Controlling the power to a server from a client

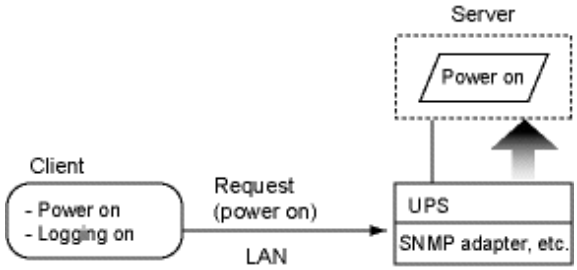
The power control function can automatically turn the power to a server on when the power to a client is turned on and a user logs in. The following figure shows an overview of this type of operation.

For information about the settings required, refer to the *Systemwalker Operation Manager Installation Guide*.

Using a power control box



Using a UPS



Controlling the power to multiple servers as a batch

When operating multiple servers (such as cluster operations), the power supply can be controlled as a batch by turning the power to servers on and off.

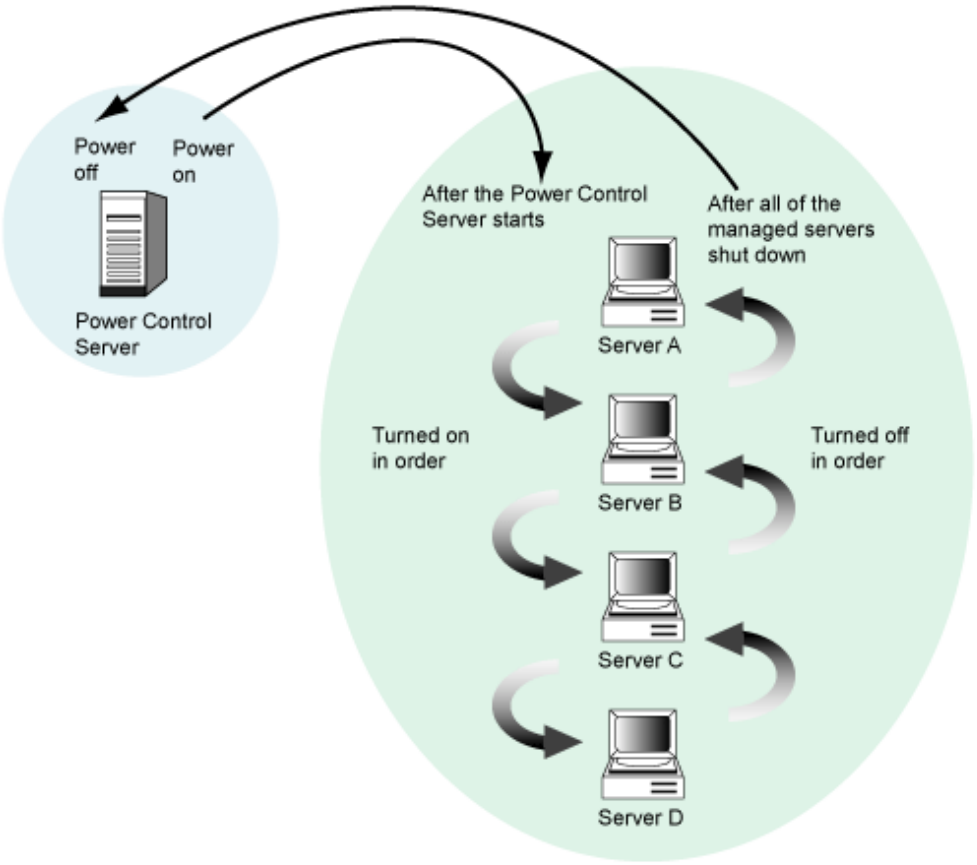
When the power is turned on, the server that controls the power starts first, and then the servers that are managed as a batch start in order. Conversely, when the power is turned off, the servers that are managed as a batch are turned off in order first, and then the server that controls the power is turned off. The interval between servers starting up and shutting down can also be specified. The following figure shows an overview.

For more information about setting up and operating power control, refer to the *Systemwalker Operation Manager User's Guide* or the *Systemwalker Operation Manager Online Help*.

Note

If there is an error with the server that performs batch power control, and it is not turned on normally, the servers that are managed as a batch are turned on automatically approximately 10 minutes after the scheduled time.

In this case, the servers might not be turned on in the preset sequence or at the preset startup intervals.



Using the end-of-wait completion notification command

By using the end-of-wait completion notification command, the server can be shut down after waiting for services and applications to complete. For example, the server can be shut down after waiting for the job net execution in the Jobscheduler to complete.

The waiting conditions for the completion of services and applications can be defined using the **Queuing Completion Notification Definition** dialog box. For information about the **Queuing Completion Notification Definition** dialog box, refer to the *Systemwalker Operation Manager Online Help*. For information about the End-of-wait Completion Notification command, refer to the *Systemwalker Operation Manager Reference Guide*.

Using calendar APIs

By creating applications using the following calendar APIs, custom processing can be performed using calendar data. For more information about these APIs, refer to the *Systemwalker Operation Manager Reference Guide*.

- Calendar Name List Acquisition API
- Calendar Registration API
- Calendar Update API
- Calendar Information Acquisition API
- Day Change Time Acquisition API

2.2 Jobscheduler

This section explains how to operate the Jobscheduler.

2.2.1 Basic Operations

This section presents an overview of the basic methods for operating the Jobscheduler.

For more information about how to operate the Jobscheduler, refer to the *Systemwalker Operation Manager User's Guide*.

(1) Registering projects

To schedule jobs, first register a project. A "project" is a collection of job nets and groups in the Jobscheduler, which have been organized according to the kinds of jobs that they contain.

(2) Registering job nets

Register job nets and their constituent jobs.

(3) Registering job net information

Register job net information, such as basic information, messages and startup days.

(4) Registering job net startup days

Register the days when job nets will start.

(5) Registering schedule patterns

Register patterns for the days when job nets will start.

(6) Monitoring the status of job nets and scheduled jobs

The status of job nets and scheduled jobs are monitored during operations.

(7) Operating job nets and scheduled jobs

Operations on job nets (cancel, start, restart, pause, continue, disable, enable, confirm, reinstate, revoke, and start with variable parameters) can be performed as necessary.

Operations on scheduled jobs (cancel, restart, pause, continue, disable, and enable) can be performed as necessary.

(8) Viewing histories of job nets and scheduled jobs

Histories of job nets and scheduled jobs can be displayed as necessary.

(9) Changing job schedule data

Job schedule data can be changed as necessary. The available options are:

- Adding, copying, and deleting job nets
- Changing job net information
- Temporarily changing the Job Net Startup Settings
- Adding, deleting, and connecting jobs
- Changing job information
- Adding, changing, and deleting schedule patterns

(10) Printing Jobscheduler information

Jobscheduler information can be printed. The following information can be printed:

- Gantt charts
- Group monitoring maps
- Job net monitoring maps
- Group lists
- Job net lists
- Job net histories
- Job histories

2.2.2 Additional Operations

Monitoring multiple servers

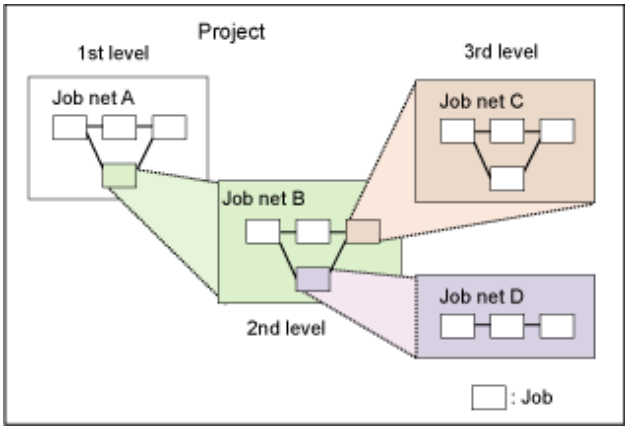
Job nets in projects under multiple servers can be monitored and operated according to the privileges of the login user. Job nets in mixed environments that contain both Windows and UNIX servers can also be monitored according to the privileges of the login user.

This type of operation is suitable for small-scale systems where only Systemwalker Centric Manager has been installed. For information about monitoring job nets on medium- to large-scale systems where Systemwalker Centric Manager has also been installed, refer to "2.11 Linking to Systemwalker Centric Manager".

Nesting job nets

A job net can be registered as a job in another job net. This kind of registration is referred to as "job net nesting." A job net where another job net has been registered is called a "parent job net," and a job net that is registered inside another job net is called a "child job net." For nested job nets, the depth of a particular level is represented as "the n^{th} level". Job nets can be nested up to five levels deep.

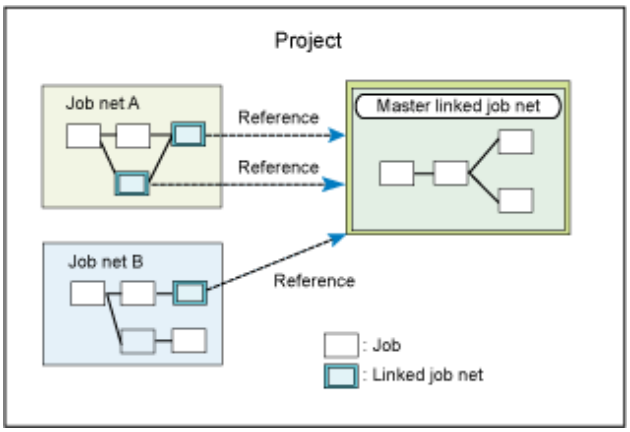
The following configuration diagram shows an example of a job net with three levels.



In this example, job net A is in the first level, job net B is in the second level, and job net C and job net D are in the third level. Job net A is the parent of job net B. Job net B is the parent of job nets C and D. Job net B is a child of job net A, and job nets C and D are children of job net B.

Nesting using linked job nets

Each job net can be used as a job by other job nets, a job net used as a job in multiple job nets is called a "linked job net". Linked job nets use another job net as definition information; job nets used to define linked job nets are called "master linked job nets". Multiple linked job nets can be used as a job in a parent job net. An example configuration is shown below.



In the above diagram, Job net A uses two linked job nets, which are defined by the master linked job net. Job net B uses one linked job net that is defined by the same master linked job net.

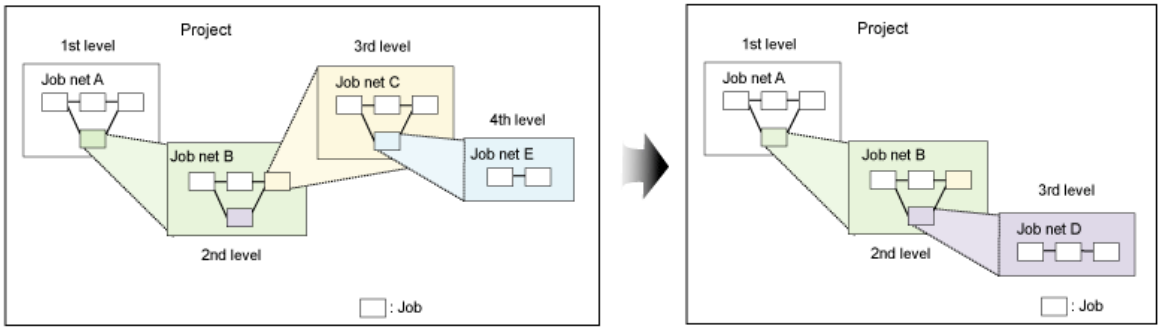
Setting up startup days for nested job nets

Startup days can be set up for child job nets in nested job nets. Additionally, by setting a startup day for the master linked job net, the startup day is also set for the linked job nets. By registering the child and linked job nets as jobs where the startup day has been set, the job net configurations are changed for each operation date automatically.

Refer to the *Systemwalker Operation Manager User's Guide* for information on the procedure for setting up startup days for child job nets and linked job nets.

The following configuration diagram shows an example of a job net with four levels.

In this example, the 1st day is set as the startup day for job net C and the 2nd day is set as the startup day for job net D.



Job net configuration for the 1st day

Job net configuration for the 2nd day

Executing a job net concurrently

You can execute the same job net concurrently by copying and starting multiple job nets from a single definition.

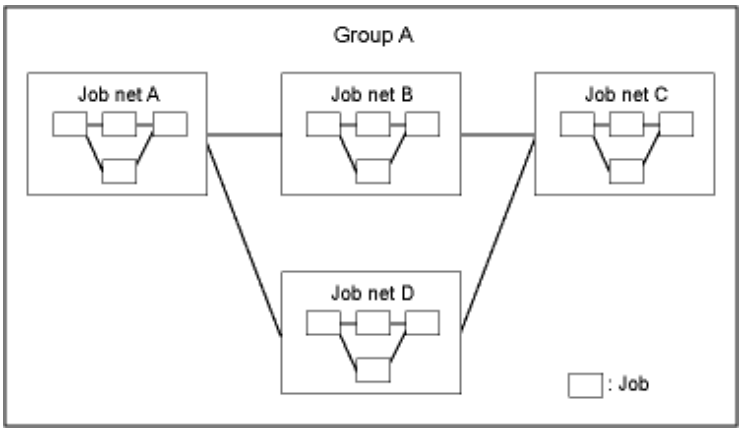
The number of job net definitions can be reduced because you no longer need to define multiple job nets containing similar processing. In addition, concurrent execution using copy and startup allows you to check past execution results because the execution results are not overwritten even when the same job net is executed multiple times.

Group management of job nets

By registering several job nets as a single group, all of the job nets can be started, monitored and operated as a group.

For information about the procedures for registering, monitoring and operating groups, refer to the *Systemwalker Operation Manager User's Guide*.

The following figure shows an example of a group configuration



Using event occurrence as an execution trigger

As well as using execution schedules based on startup days, job nets can also be started when events occur. For the Jobscheduler, events are referred to as "message events."

Message events can be generated using the **Jobschmsgevent** command.

The following examples show how to start job nets by generating message events using the **Jobschmsgevent** command.

- If there is a job net B that waits for the message event "Jobneta", this job net will be started if the following command is entered at the command prompt.

```
Jobschmsgevent Jobneta
```

  If "Jobschmsgevent.exe Jobneta" is defined as the last job in job net A, then job net B must wait until job net A has completed before it can commence.

- Executing the **Jobschmsgevent** command within a user application makes it possible to start any job net from the user application.

Information

Job net linkage using message events also has disadvantages, such as it being difficult to check the operating status of job nets, and the complexity of operations where message events need to be cleared. In many cases, equivalent operations can be achieved using the hierarchical job net operation function, so Fujitsu recommends using this function where possible.

It is also possible to specify execution conditions for job nets using AND/OR combinations of execution schedules based on startup days and execution triggers based on event occurrence.

For more information about the **Jobschmsgevent** command, refer to the *Systemwalker Operation Manager Reference Guide*.

Linking job nets between servers

Message events can be generated on other servers using the **Jobschmsgevent** command described above. In this way, job nets on other servers can be started as desired.

The following items are specified with the **Jobschmsgevent** command.

- The name of the message event to be generated
- The host name of the server on which the message event is to be generated, or the name of a file containing a list of hosts

Information

To generate message events on servers running different operating systems, use network jobs.

Starting job nets by specifying variable parameters

Job nets can be started by specifying variable parameters. "Variable parameters" are parameters that are passed to the job net in order to replace variables (@.VPARAM) that have been entered in the job definitions beforehand. For the jobs in the job net that receives these variable parameters, these variables are replaced with the variable parameters before the jobs start.

Defining a base job net in advance using pre-determined variables and then starting this base job net by specifying variable parameters eliminates the need to define a large number of different job nets, particularly in situations where there are multiple job nets that differ only by a parameter.

There are several methods for starting job nets by specifying variable parameters, as follows:

Starting job nets when a message event is notified

The **jobschmsgevent** command starts job nets by passing variable parameters at the same time as the notification that a message event has occurred.

Jobs are started by replacing variables that have been entered beforehand with the variable parameters that are received together with the message event notification.

Starting duplicate job nets when a message event is notified

When a job net is started by passing variable parameters when a message event is notified, it is possible to duplicate the job net to be started, and then start the duplicate job net. This allows multiple job nets (with different parameters) to be executed in parallel by attaching different suffixes, and makes it possible to execute a job net with different parameters without overwriting the previous execution results.

Starting job nets dynamically

It is possible to perform startup operations where variable parameters are specified dynamically, in situations such as when a job net is started from an operator instruction by specifying variable parameters, or when a job net is started manually by specifying variable parameters (as a recovery task when an error occurs, for example).

Refer to the *Systemwalker Operation Manager User's Guide* for information about the procedure for starting job nets by specifying variable parameters.

Simultaneously replacing job registration information using job definition variables

Jobs can be defined in advance using job definition variables (@*variable name*@), and the values of these job definition variables can be replaced as a batch when the job is executed. Information such as the environment definitions for the paths to job definitions can be easily replaced as a batch, making it easy to migrate assets between systems with different operation environments.

Refer to the *Systemwalker Operation Manager User's Guide* for more information.

Using a job net variable to pass information between jobs

Individually set information and information such as the file name or message event that triggered startup can be passed as a job net variable between jobs within the same job net. You can branch a subsequent process depending on the information received as the job net variable.

Refer to the *Systemwalker Operation Manager User's Guide* for details.

Importing and exporting definition information from the GUI

Definition information for jobs, job nets and groups can be imported and exported as CSV files from the Systemwalker Operation Manager client window.

The following CSV files can be imported and exported:

- Job net definition CSV files
- Group definition CSV files

Refer to "Jobscheduler Commands" in the *Systemwalker Operation Manager Reference Guide* for details on job net definition CSV files and group definition CSV files.

When importing definition information, it is possible to specify multiple CSV files or a directory containing CSV files. The import destination is a project. The CSV file determines how job/job net definition information and group definition information is imported into a project.

The target range of export includes job nets, parent job nets, child job nets, master linked job nets, groups, projects and subsystems. The definition information of jobs/job nets or groups within the target range is used to create a CSV file for each job net or group within the specified directory.

The definition information of the following Systemwalker Operation Manager server platforms and V/L can be imported or exported to/from Systemwalker Operation Manager clients running V13.3.0 or later:

- Windows: V11.0L10 or later

- Solaris: 11.0 or later
- Linux: V11.0L10 or later
- HP-UX: 11.0 or later
- AIX: 11.0 or later

Refer to "Importing and Exporting Definition Information from the GUI" in the *Systemwalker Operation Manager User's Guide* for more information.

Systemwalker Operation Manager users [UNIX version]

Using Systemwalker Operation Manager functions normally requires user authority for the operating system, such as logging in as a user that has been registered with the operating system. With the UNIX version, users of Systemwalker Operation Manager functions can be registered and managed on Systemwalker Operation Manager (using the Extended User Management function).

Using the Extended User Management function has the following advantages.

- The number of users on the operating system does not need to be increased unnecessarily in order to perform operations from clients.
- Administrator or non-administrator authority can be assigned to users registered with Systemwalker Operation Manager, so multiple administrators can be set up for Systemwalker Operation Manager, separate to the system administrators for the operating system. Also, detailed access authority for Systemwalker Operation Manager operations can be set up for administrators and non-administrators.
- Operating system users need to be registered for each server, but users registered using the Extended User Management function can be distributed to other servers using the Policy Extraction and Distribution function. The tedious task of registering users does not have to be performed separately for each server.

Users that are managed on Systemwalker Operation Manager using the Extended User Management function can use Systemwalker Operation Manager functions from clients.

Starting job nets when the server is turned off

It is possible to specify what to do when a job net cannot start at the scheduled starting time (because, for example, the power to the server is turned off at the time). Select the "Startup on power-on if power is off during scheduled execution" option when the job net information is registered.

For an overview of registering job net information, refer to the *Systemwalker Operation Manager User's Guide*.

Continuing jobs when the schedule server fails

Even if the system fails on the schedule server, network jobs that are executing on servers other than the schedule server can continue operating without having to be cancelled. When the schedule server is restarted, the status of all jobs that were executing on execution servers is checked, and the results are automatically reflected in the schedule information file. If there are any subsequent jobs that need to be executed, the relevant job net is restarted automatically and job execution continues.

Whether to continue network job operations when the schedule server fails is defined using a command that switches continuous execution mode on and off. For more information about how to make these definitions, refer to the *Systemwalker Operation Manager User's Guide*.

- The settings for enabling or disabling continuous execution mode must be the same on all linked servers, or else the following symptoms may occur:
  - When the schedule server terminates due to a system failure, "Executing" is displayed even when the job is terminated.
  - A job outputs an error message and terminates abnormally.
  - A network job is executed while it is already running.

- Even if continuous execution mode has been enabled, currently executing network jobs will terminate abnormally and will not continue executing if the execution server (a server other than the schedule server) fails. Continuous execution mode cannot be used to specify job continuation when the system fails on an execution server. Setting up continuous execution mode only enables jobs to continue when the system fails on the schedule server.

Duplicating execution servers for network jobs

If the execution server specified for a network job has failed or if the communication path has been interrupted, it is possible to request a secondary execution server to execute the network job.

When duplicating network job execution servers, define the primary and secondary execution servers when jobs are registered. For information about how to make these definitions, refer to the *Systemwalker Operation Manager User's Guide*.

Note

- Jobs will terminate abnormally if both the primary and secondary execution servers are down.
- Do not specify the local host name for the primary and secondary execution servers.

Operating in test mode

For multi-subsystem operations, the execution of future schedules can be checked in advance in Test mode using a subsystem other than the one where operations are currently taking place. Virtual times can be set up for particular subsystems without changing the time for the operating system, which makes it possible to check job execution on the subsystem where the virtual time has been set up. For information about how to define virtual times, refer to the *Systemwalker Operation Manager User's Guide*.

Although multi-subsystem operations are not supported in the Standard Edition, a virtual time can still be set up. Operations can be tested by setting up virtual times beforehand.

Temporarily changing job net startup settings

It is possible to temporarily change information (such as start times and estimated end times) that is set up as startup conditions when a job net is registered. During the period specified for this temporary change, the job net will be executed according to the new operation environment. When this specified period elapses, the settings return to that of the original operation environment.

For information about how to change startup settings temporarily, refer to the *Systemwalker Operation Manager User's Guide*.

Changing startup parameters

All startup parameters for the Jobscheduler are set to default values during installation, and can be changed when necessary.

For information about startup parameters and how to specify them, refer to the *Systemwalker Operation Manager Installation Guide*.

Using APIs and exits

The following APIs and exits can be used to customize the application environment to match user environments. For information about APIs and exits, refer to the *Systemwalker Operation Manager Reference Guide*.

- Job Net Manipulation API
- EE Job Net Manipulation API/EE
- Job Net Manipulation (Startup Parameter Setup) API
- EE Job Net Manipulation (Startup Parameter Setup) API/EE
- Group Manipulation API

EE - Group Manipulation API/EE

- Job Net Startup API [Windows version]

EE - Job Net Startup API/EE [Windows version]

- Job Net Startup Time Modification API

EE - Job Net Startup Time Modification API/EE

- Job Net Normal Termination/Abnormal Termination Exit
- Job Net Abnormal Termination Extended Exit
- Job Termination Exit
- Log File Switchover Exit
- Day Change Time Arrival Exit
- Normal/Abnormal Termination Shutdown Exit [Windows version]
- Job Net Activation Delay Exit
- Job Net Termination Delay Exit

Outputting various data

The following information managed by the Jobscheduler can be output to the standard output using the **Jobschprint** command. For more information about the **Jobschprint** command, refer to the *Systemwalker Operation Manager Reference Guide*.

- Current status of groups, job nets, and jobs
- Registration information for groups, job nets, and jobs
- Startup days data for job nets
- Job net operation schedules
- Job net history information
- Schedule pattern information
- Calendar information
- User information

Preventing operational mistakes for groups, job nets, and jobs

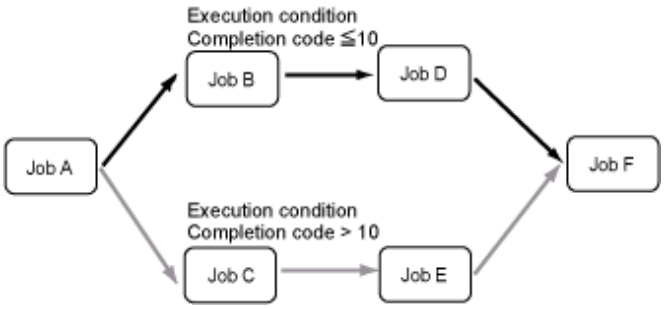
When operations are performed on groups, job nets or jobs, operational mistakes can be prevented by displaying a dialog box to confirm the operation. Operations are confirmed using the **Operation Confirmation** dialog box.

For information about how to set up the **Operation Confirmation** dialog box, refer to the *Systemwalker Operation Manager User's Guide*.

Switching between subsequent jobs according to completion codes

It is possible to select which subsequent job is executed according to the completion code of the preceding job.

The following figure shows an example where the execution conditions have been set up so that Job B starts if the completion code for Job A is 10 or less, but Job C starts if the completion code is greater than 10.



Refer to the *Systemwalker Operation Manager User's Guide* for more information.

2.2.3 Applications

Triggering execution when files are created, updated or deleted

Startup of jobs and job nets can be triggered when a certain file is created, updated, or deleted, or depending on whether a certain file exists. You can also specify a combination of these conditions and make jobs and job nets wait for any one of the specified conditions to be met. A wildcard character (*) can be used to specify the file names of files that will trigger execution.

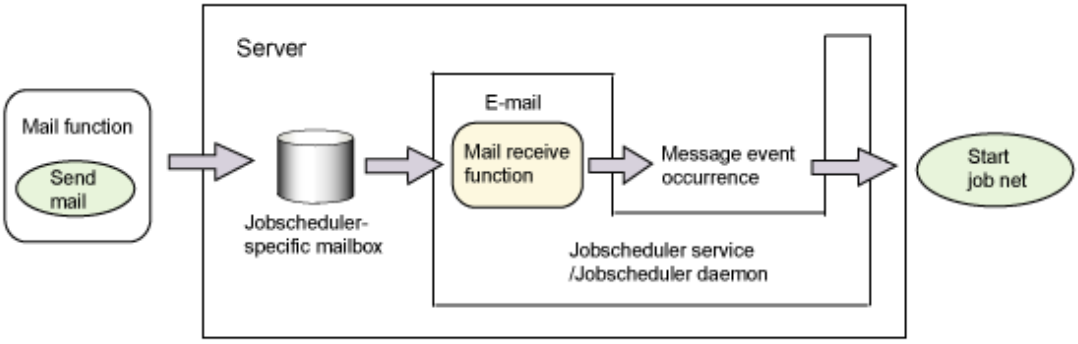
Refer to the *Systemwalker Operation Manager User's Guide* for information on how to start jobs and job nets triggered when files are created, deleted, or updated. Refer to the *Systemwalker Operation Manager Reference Guide* for information on the chkfile and jobschchkfile commands.

Triggering execution when mail is received

In addition to using execution schedules based on startup days, job net execution can also be triggered by the receipt of mail. To do this, mail waiting conditions must be set up by defining startup parameters. For more information about defining startup parameters, refer to the *Systemwalker Operation Manager Installation Guide*.

The type of mail referred here is "e-Mail". For information about the format of mails that can be received, refer to the *Systemwalker Operation Manager User's Guide*.

The following figure illustrates the flow for starting job nets triggered by the receipt of mail.



Triggering execution when events are output to event logs [Windows version]

Job net execution can be triggered by events being output to event logs by the operating system or other products. In order for the event logs being used as startup triggers to be recognized by the Jobscheduler, the "source name" and "event ID" in the event log must match the message event that is the execution condition for the job net. In order to associate "source names" and "event IDs" with message events, a "message table" must be created.

For information about how to create the message table, refer to the *Systemwalker Operation Manager Installation Guide*.

Coordinating job times

Job times can be coordinated within job nets. To coordinate job times within a job net, schedule the **Jobschchecktime** command (time wait command) as the preceding job of the job that needs to wait until a certain time before executing. If the **Jobschchecktime** command is registered as a job, the job does not end until the specified time comes. The subsequent job can then be executed after the specified time. It is also possible to specify the maximum waiting time, so that if jobs are started after the maximum waiting time, the job completion code will be different, which can be used to change subsequent processing.

For more information about the **Jobschchecktime** command, refer to the *Systemwalker Operation Manager Reference Guide*.

Switching between subsequent jobs by analyzing character strings in files

Each line of a specified file can be analyzed to check whether it contains a specified character string. To analyze the file, use the **Jobschchecklog** command (log file analysis command). Because the completion code of the job varies depending on the result of the analysis, the **Jobschchecklog** command can be scheduled as a job, and the value of its completion code can be used to switch between alternative subsequent jobs.

Regular expressions can be used to specify the character string. If the specified character string is found, the line including the character string can also be output to the standard output.

The specified file can be analyzed from beginning to end, or analysis can start from the middle of the file. To analyze a file from the middle of the file, specify a key to store ending information. Ending information is stored based on the file name, the user name and the key. Therefore, the same file can be analyzed by multiple users or multiple jobs starting from the end point if the file name, user name and key are unique.

For more information about the **Jobschchecklog** command, refer to the *Systemwalker Operation Manager Reference Guide*.

Monitoring scheduled startup times for job nets

It is possible to monitor whether job nets start according to the designated schedule. If a job net does not started at the scheduled startup time, a notification message can be output to an event log [Windows version] or the SYSLOG [UNIX version]. In this case, the start of the job net will be delayed, and so the progress can also be checked using a Gantt chart.

Monitoring scheduled startup times for job nets applies to the following kinds of situations:

- When the "Wait for startup time" option is used to start a job net when the conditions for a startup time and a message event are both met, and the message event has still not occurred when the schedule startup time arrives
- When a job net belongs to a group and the preceding job net has not completed but the scheduled started time has passed
- When a job net belongs to a group and the scheduled startup time has passed because the preceding job net has been canceled or has terminated abnormally
- When a job net belongs to a group and the scheduled startup time passes without the message event that it is waiting for occurring
- When the scheduled startup time for a job net has been defined with the **Start only when message event has occurred** option and the scheduled startup time passes without the message event that it is waiting for occurring

When a job net does not start even though the scheduled startup time has arrived, whether a notification message should be output can be defined using a startup parameter for the Jobscheduler service or daemon. For notes on startup parameters, including information about how to define them, refer to the *Systemwalker Operation Manager Installation Guide*.

If a job net has not started even though the scheduled startup time has arrived, custom processing can be executed using the "job net start delay exit." For more information about the "job net start delay exit," refer to the *Systemwalker Operation Manager Reference Guide*.

Monitoring scheduled end times for job nets

It is possible to monitor whether job nets have terminated according to the designated schedule. If a job net has not terminated at the scheduled end time, a notification message can be output to an event log [Windows version] or the SYSLOG [UNIX version]. If a job net has not terminated at the scheduled end time, this indicates that the status of the job net is either

"executing", "warning" or "waiting." If a job net is still executing after its scheduled end time has passed, the status of the job net is "delayed termination", and so the progress can also be checked using a Gantt chart.

The scheduled end time can be specified either as an absolute time or as a relative time (relative to the time the job net started). Define the scheduled end times for job nets when job net information is registered. For information about how to make these definitions, refer to the *Systemwalker Operation Manager User's Guide*.

When a job net does not terminate even though the scheduled end time has arrived, whether a notification message should be output can be defined using a startup parameter for the Jobscheduler service or daemon. For notes on startup parameters, including information about how to define them, refer to the *Systemwalker Operation Manager Installation Guide*.

If a job net has not terminated even though the scheduled end time has arrived, custom processing can be executed using the "job net end delay exit." For more information about the "job net end delay exit," refer to the *Systemwalker Operation Manager Reference Guide*.

Monitoring the estimated processing time for jobs

If the execution attribute for a job net is "Job Execution Control", a message can be output to the event log [Windows version] or SYSLOG [UNIX version] if the job does not end within the estimated processing time. If a job net is still executing after its estimated finishing time (calculated based on the estimated processing time for the specified job net) has passed, the status of the job net is "delayed termination", and so the progress can also be checked using a Gantt chart.

When a job net does not terminate even though the estimated end time has arrived, whether a notification message should be output can be defined using a startup parameter for the Jobscheduler service or daemon. For information about how to define startup parameters, refer to the *Systemwalker Operation Manager Installation Guide*.

Define the estimated processing time when jobs are registered. For information about how to make these definitions, refer to the *Systemwalker Operation Manager User's Guide*.

Turning off the power to the System when job nets terminate [Windows version]

The shutdown exit of the end processing job net provided by the Jobscheduler can be used to turn the power to the system off when a job net terminates.

Specifically, replace the f3crheet end-of-wait notification command in the shutdown exit of the job net that performs the shutdown processing with the shutdown command provided by the operating system.

If the power to the system is turned off by calling the shutdown command from the shutdown exit of the end processing job net, the power will be turned off as soon as the job net terminates. The system will shut down even if the server shutdown time that has been set up in the power schedule has not arrived yet. If the server shutdown time that has been set up in the power schedule arrives before the end processing job net runs, server shutdown processing will be performed even if job nets are still executing. When setting up the server shutdown time in the power schedule, sufficient time must be allowed before the time when the end processing job net runs.

For more information, refer to "Shutting Down the System at Optional Times [Windows version]" in the *Systemwalker Operation Manager User's Guide*.

Turning the system power off

In order to turn the system power off, the system must be equipped with a power control device (hardware).

Postponing system shutdown until job nets complete

System shutdown processing can be delayed until job nets have completed even if the shutdown time has arrived. This can be done by using the shutdown monitoring options (provided by the Power Control function) and the **f3crheet** end-of-wait notification command.

Specifically, set the shutdown monitoring options and register the **f3crheet** command as the last job to be started. This job must be registered so that it starts before the system shuts down.

After the shutdown processing completes, the system power will be turned off by the Power Control function.

For an explanation of the shutdown monitoring options and information about how to set them up, refer to the *Systemwalker Operation Manager User's Guide*. Also, for information about the **f3crheet** end-of-wait notification command, refer to the *Systemwalker Operation Manager Reference Guide*.

Note

Turning the system power off

In order to turn the system power off, the system must be equipped with a power control device (hardware).

Permitting access to projects

With the Jobscheduler, access rights can be set to specify which users are permitted to have access to which projects. For information about setting up access rights, refer to the *Systemwalker Operation Manager User's Guide*.

2.2.4 Reference Information (Know-how, etc.)

Registering startup days

Sometimes there is a specific pattern for the days when a batch file or shell script that performs a job is started. Examples include jobs that are performed every Monday or on a fixed date each year. When registering this kind of job (or a job net), first register the startup day pattern (schedule pattern) and then specify this pattern as the startup days for the job net. Alternatively, register startup day patterns for job nets on a "yearly" basis or a "monthly" basis.

To set up startup days using patterns such as "the *n*th day of the month" or "*n* business days before end of the month", register the startup day pattern using a "business day" basis.

For jobs with no fixed pattern for the days when they start, startup days can be registered for each job net using the **Startup Days** window. Alternatively, jobs can be started as desired using the **Select Job** window, without registering startup days.

To set up the same startup days as for another job net, the other job net can be specified as the base job net.

How to make job nets finish by a certain time

Sometimes it is desirable to have a job net finish executing by a certain time. In this kind of situation, scheduled end time of the job net can be worked out by starting the job net and then using the **Gantt Chart** window. Then, the startup time can be set up by calculating back from this scheduled end time. Note, however, the time displayed is an estimate based on the previous execution time, so allow sufficient time when setting up the startup time.

The startup time for a job net only becomes relevant when startup days have been registered.

How to delete job output files

It is possible to delete job output files that are created when jobs registered in a job net with the "Job Execution Control" attribute are executed. It is also possible to save these output files without deleting them.

The files to be deleted are as follows:

- Jobname."o"Jobnumber (Standard output files)
- Jobname."e"Jobnumber (Standard error output files)
- Jobname."l"Jobnumber (Job list files)

The startup parameters for the Jobscheduler service or daemon can be used to define whether to delete job output files. For notes on startup parameters, including information about how to define them, refer to the *Systemwalker Operation Manager Installation Guide*.

EE

Registering large numbers of job nets

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 registered in a single project, job nets may no longer be able to start on schedule due to performance problems.

When registering a large number of job nets, perform thorough testing to ensure that job nets start on schedule without any problems before commencing production operations. Refer to "Tuning of Performance" in the *Systemwalker Operation Manager User's Guide* for information about performance testing. If there are any performance problems, divide the project into several smaller projects.

If the system environment allows for multiple CPUs or multiple I/O controllers, system performance can be improved by using multiple subsystems.

Environment variables

Before starting jobs, the Jobscheduler assigns values to the environment variables listed below. Use these environment variables in situations such as when the processing for subsequent jobs needs to be changed depending on the type or completion code of the preceding job.

With network jobs, however, the environment variables listed below are not inherited.

Information

For network jobs, if the schedule server and execution server are both running Systemwalker OperationMGR V10.0L20/10.1 or later, the environment variables that have been set in the "Environment variables" section of the **Detail information** sheet in the **Add/Change/Monitor - Job** window are inherited by network Jobs. (If both the schedule and execution servers are not running Systemwalker OperationMGR V10.0L20/10.1 or later, these environment variables are not inherited).

USERNAME environment variable [Windows version] and USER environment variable [UNIX version]

These variables store the owner of the project.

PWD environment variable [UNIX version]

This variable stores the directory name of the job. If the directory name is omitted, this variable stores the name of the home directory for the project owner.

HOME environment variable [UNIX version]

This variable stores the name of the home directory for the project owner.

LOGNAME environment variable [UNIX version]

This variable stores the name of the project owner.

MAIL environment variable [UNIX version]

This variable stores "/usr/mail/project-owner-name". In AIX, this variable stores "/usr/spool/mail/project-owner-name". In the Linux versions, this variable stores "/var/spool/mail/project-owner-name".

USERDOMAIN environment variable

This variable stores the domain name for the project owner if the connection destination server is a Windows server. If no domain name is specified, the value of this environment variable is omitted. If the connection destination server is a UNIX server, the value of this environment variable is omitted.

EE

JOBSCH_SUB_SYSTEM environment variable

This variable stores the subsystem number.

JOBSCH_PROJECT environment variable

This variable stores the name of the project where the job net has been registered.

JOBSCH_JOBNET environment variable

This variable stores the name of the job net.

JOBSCH_JOBNAME environment variable

If the job name that is passed to Job Execution Control has been registered in the job registration information, this variable stores that job name. If a job name has not been registered, this variable stores information about the execution file that

has been registered in the "Command" field of the **Add/Change - Job** window (including the **qsub** command option and execution file parameters).

JOBSCH_JOBNET_DATE environment variable

This variable stores a value indicating the date of the configuration with which the job net has been started. The value is stored using the "yyyymmdd" date format.

JOBSCH_ENVIRONMENT environment variable

This variable stores the "BATCH" character string.

LANG environment variable [UNIX version]

This variable stores "C". Set this environment variable if it is required by a startup shell (".login", ".cshrc" or ".profile") for the job execution user.

In addition, values are set for the following environment variables when preceding jobs exist. However, if there is more than one preceding job, the data for the job that triggered the startup of the current job (the preceding job that was executed most recently) is set.

JOBSCH_PRE_JOBNAME environment variable

If the job name that is passed to Job Execution Control has been registered in the registration information for the preceding job, this variable stores that job name. If a job name has not been registered, this variable stores information about the execution file that was registered in the "Command" field of the **Add/Change - Job** window (including the **qsub** command option and execution file parameters).

JOBSCH_PRE_JOBCODE environment variable

This variable stores the completion code for the preceding Job.

JOBSCH_ROOT_JOBNET environment variable

If the job net is nested, this variable stores the name of the job net in the first layer. If the job net is not nested, this variable stores the job net name (the value will be the same as the JOBSCH_JOBNET environment variable).

2.3 Job Execution Control

This section explains how to operate the Job Execution Control function.

2.3.1 Basic Operations

This section outlines the basic operation procedures for the Job Execution Control function.

1) Designing and setting up the job environment

Design the job environment so that a reliable and efficient job execution environment can be created. The design content can be set up using either the **Define Operating Information** window or an initialization file.

For more information, refer to the *Systemwalker Operation Manager Installation Guide* or the *Systemwalker Operation Manager Online Help*.

2) Executing jobs

Jobs can be executed once they have been created. If job execution is automated, job submission is mainly managed by the Jobscheduler. However, the Job Execution Control function is used when jobs are submitted without being scheduled.

Jobs submitted by the Jobscheduler are called "scheduled jobs," and jobs submitted by Job Execution Control are called the "demand jobs." For information about executing demand jobs, refer to the *Systemwalker Operation Manager User's Guide*.

3) Checking the status of jobs and queues

The status of scheduled jobs can be viewed in the Jobscheduler window, and the status of demand jobs can be viewed in the **Job Execution Control** window.

If an execution delay occurs with either a scheduled job or a demand job, check the operational status of queues.

4) Operating jobs and queues

If job execution does not proceed smoothly, perform operations (such as changing execution queues, changing job priorities, suspending execution, or deleting jobs) as necessary.

For information about job and queue operations, refer to the *Systemwalker Operation Manager User's Guide*.

5) Saving job execution histories and operation records

The following files can be saved as history information about job execution.

EE

For multi-subsystem operations, this file can be collected for each subsystem.

- Log (execution history) files
- Achievement record files

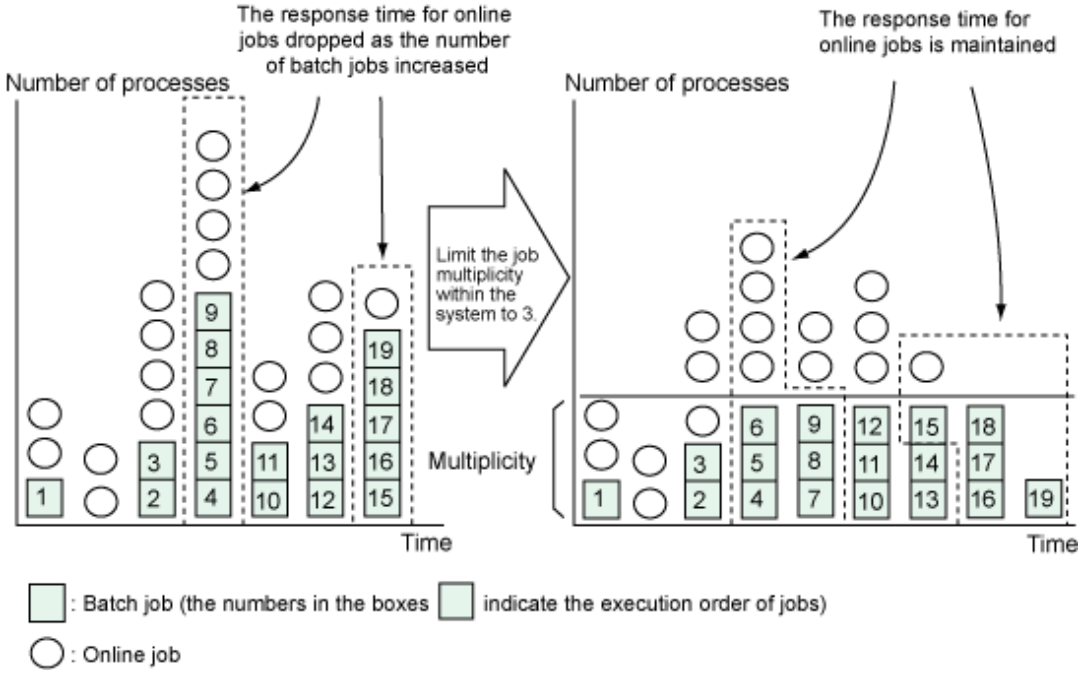
These files can be used to analyze the job submission status on servers where Systemwalker Operation Manager has been installed. To collect these files, make specifications in the **Logging** sheet of the **Define Operating Information** window.

For more information about log files and achievement record files, refer to the *Systemwalker Operation Manager User's Guide* and the *Systemwalker Operation Manager Reference Guide*.

Information

Example of how to create a job execution environment (with a limited number of executable jobs)

As an example of how to create an efficient job execution environment, the following figure shows the advantages of limiting job multiplicity within the system. Specifically, limiting multiplicity makes it possible to prevent increases or decreases in the number of batch jobs submitted from influencing the response times for online jobs.



2.3.2 Additional Operations

Selecting job execution rights [Windows version]

Jobs submitted by Systemwalker Operation Manager are executed using the authority of the logon account that started the Job Execution Control service on the server. This execution right can be changed to the authority of the user ID that actually submitted the job if the following settings are performed.

- Select "Execute jobs under the respective job owner's authority" in the **Options** sheet of the **Define Operating Information** window.
- In the **Define Job Owner's Information** window, set the password for the user ID that will submit jobs.
- Assign "logon as a batch job" authority to the user ID that will submit jobs.

For more information, refer to the *Systemwalker Operation Manager Installation Guide*.

Submitting network jobs

Any server connected to the network can be requested to execute jobs. The execution results for these jobs can be checked from the server where they were submitted. It is also possible to specify retry processing for jobs submitted in this way, in order to prepare for situations where a problem occurs with the communication line during execution. If a network job is submitted while communications with the server that requested the job are not possible, error detection may be delayed as a result of retrying communication.

To prevent network jobs from being submitted to the local server from unintended servers, "trust host definitions" can be made. With these definitions, specify which servers the local server is allowed to accept network jobs from.

Executing distributed jobs

Jobs can be executed by distributing them to multiple specified servers (using the distributed execution function).

Jobs that are executed on multiple distributed servers using the distributed execution function are referred to as "distributed execution jobs." Distributed execution jobs are executed on the server with the lowest multiplicity level, which is calculated as the number of jobs currently executing divided by the execution multiplicity. The distributed job multiplicity can be changed and distributed execution destination hosts can be added or deleted during operations.

The distributed execution function is an extension of the "Load Balancer function" that was provided for Windows versions of Systemwalker Operation Manager V10.0L21 or earlier. To use the Load Balancer function in order to maintain compatibility with systems running V10.0L21 or earlier, select the **Enable the Load Balancer function of the previous version** option in the **Backward compatibility** sheet of the **Define Operating Information** window. For more information about how to use the Load Balancer function, refer to the manuals for the appropriate version (V10.0L21 or earlier).

Submitting and operating jobs from servers

Demand jobs can be submitted and operated using client windows, but the same kind of operations can also be performed using commands on servers. This makes it possible to submit and operate jobs from servers using batch files, shell scripts, applications, and so on.

Stopping queues in recovery mode

When a system is restarted after stopping due to a system failure or an interruption to the power supply (including interruptions during actual operations), the Job Execution Control service or daemon resumes operation with queues activated, in order to preserve any jobs that were executing before the stoppage (Recovery Mode).

However, it is sometimes desirable to take the necessary corrective actions before operations recommence, by checking the status of jobs and the servers where Systemwalker Operation Manger has been installed. In this case, select the **Stop all queues when started in the recovery** option in the **Options** sheet of the **Define Operating Information** window. Selecting this option changes the status of jobs to "waiting" (i.e., stops queues) even after the Job Execution Control service has restarted. Jobs will not be executed until the queues are started. After the necessary actions have been taken, recommence operations by restarting the queues.

Stopping all queues when the Job Execution Control service/daemon starts

During routine maintenance (such as system hardware maintenance), it may be desirable to prevent jobs from starting when the Job Execution Control service or daemon next starts, at least until the status of the system has been checked. To do this, select the **Stop all queues when starting the service** option in the **Options** sheet of the **Define Operating Information** window. Selecting this option changes the status of jobs to "waiting" (i.e., stops queues), stopping jobs from executing until the queue is restarted manually. After the necessary actions have been taken, recommence operations by restarting the queues.

Batch output of job results

Using JCL scripts, all job results can be grouped together as a single job and output as a batch. This prevents jobs output results from being mixed up with the output results for other jobs.

Specify batch output for job results using the "jobstart" control statement. Specifying this control statement makes it possible to perform the following processing on batch output jobs.

- Saving the batch output job in case an output failure occurs or the results need to be output again
- Placing the output of batch jobs on hold when job execution results need to be checked as soon as job execution completes

Batch output jobs that have been placed on hold can be operated on from the monitoring window (including allowing the job to output results, viewing results and deleting the job).

For more information about JCL control statements, refer to the *Systemwalker Operation Manager Reference Guide*.

Outputting job report lists

Using JCL scripts, report lists about job processing can be output. Specify report list output using the "Joblst" operand of the "Jobstart" JCL control statement. The delimiter output for job steps can be defined by selecting the **Output job step delimiters in the standard output file** option in the **Options** sheet of the **Define Operating Information** window.

The following items are output.

- Job start date and time
- Job step start date and time
- Job step end date and time, and completion code
- Job end date and time, and completion code
- Information relating to network jobs
- Job step delimiters (step name and completion code)

For more information about outputting report lists and the "Jobstart" control statement, refer to the *Systemwalker Operation Manager Reference Guide*.

2.3.3 Applications

Not creating jobs that have to wait for execution

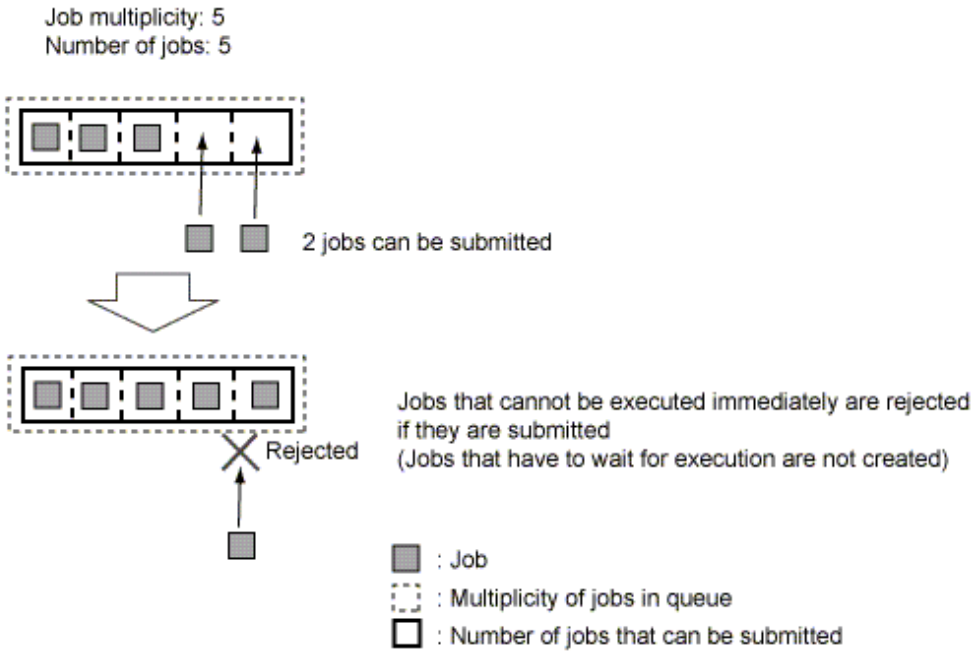
If the initialization file is defined using the **Create/Edit Queues** window so that the same limits are specified for both the multiplicity level of jobs in a queue and the number of jobs that can be submitted to the queue, then the number of jobs submitted cannot exceed the multiplicity limit.

If a suitable limit value is specified for multiplicity (taking the load on the system into account), then it is possible to operate in such a way that jobs that cannot be executed immediately will be rejected if they are submitted, and so jobs that have to wait for execution will not be created.

The multiplicity of jobs in a queue is defined using **Limit the number of jobs to execute simultaneously (Max Execution Jobs)** in the **Create/Edit Queues** window that is called from the **Define Operating Information** window. The number of jobs that can be submitted is defined using **Limit the number of jobs to submit (Number of jobs)** in the same window.

The following figure shows a representation of this kind of job queuing operation.

Example of when the following settings are made in the **Define Operating Information** window



Controlling job execution priority

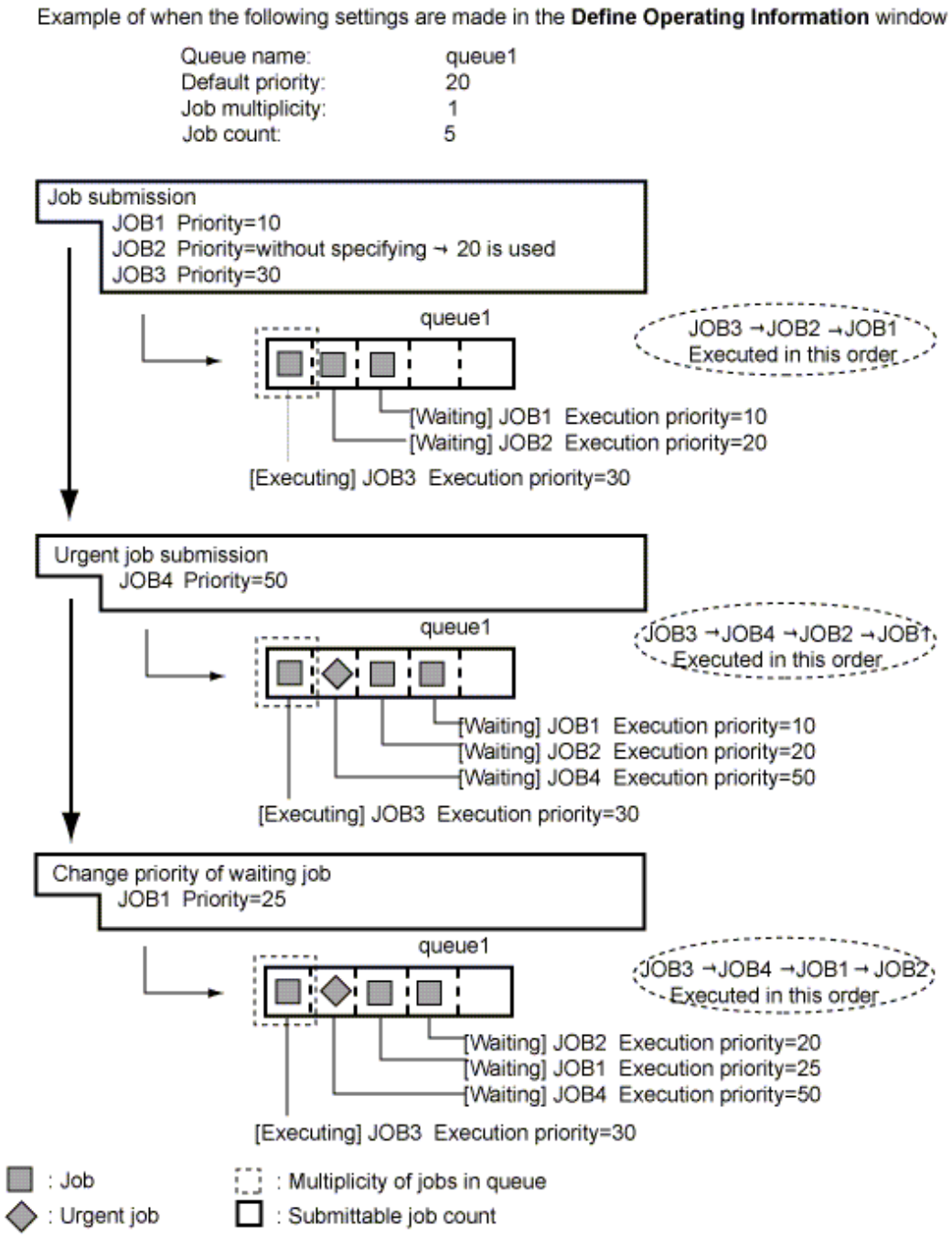
The execution priority of jobs can be specified when they are submitted. If this specification is omitted, the default priority for the queue (the value specified for "Specify default job priority (Default Job Priority)" in the **Create/Edit Queues** window) will be used.

The job with the highest priority is executed first. Jobs with higher priority are still executed first even if they are submitted later.

If this arrangement is being used and a job that requires immediate execution needs to be submitted, specify a priority that is higher than that of other waiting jobs.

It is also possible to dynamically change the priority of any or all waiting jobs.

The following example shows a representation of an operation where an urgent job is submitted and execution priorities are changed dynamically.



Executing jobs by exclusively occupying resources

You can specify any name and either the shared or exclusive attribute for a job resource. In addition, as implicit resources, all jobs have "host name" and "execution queue name" as shared attributes. You can use this mechanism to exclusively occupy the server on which Systemwalker Operation Manager is installed (so that no other job is being executed) and then execute jobs, as is done with the backup job in the following example:

Example: Execute a backup job

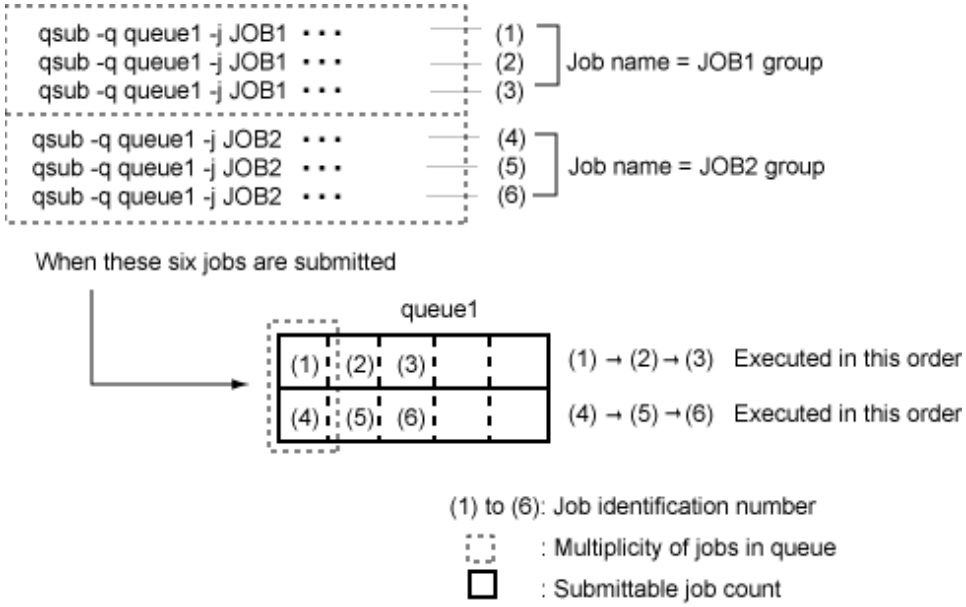
1. Set **Resource Used** to the host name and **Type** to **Exclusive**, and then submit the job.
2. The backup job waits for all jobs that are running to end.
3. The backup job starts after all running jobs have ended.
4. Jobs submitted during or after step 1 wait until the backup job has ended.

If exclusive attributes are specified for the jobs with the same name

If exclusive attributes are specified for jobs with the same name, the jobs with the same name will not be executed simultaneously. All but one of these jobs will be queued as "jobs waiting to be executed" and then executed sequentially. The execution sequence of these jobs depends on their priorities and resource specifications. If no priorities or resources have been specified, the jobs will be executed in the order in which they were submitted.

This specification is effective when a number of related jobs are executed in order.

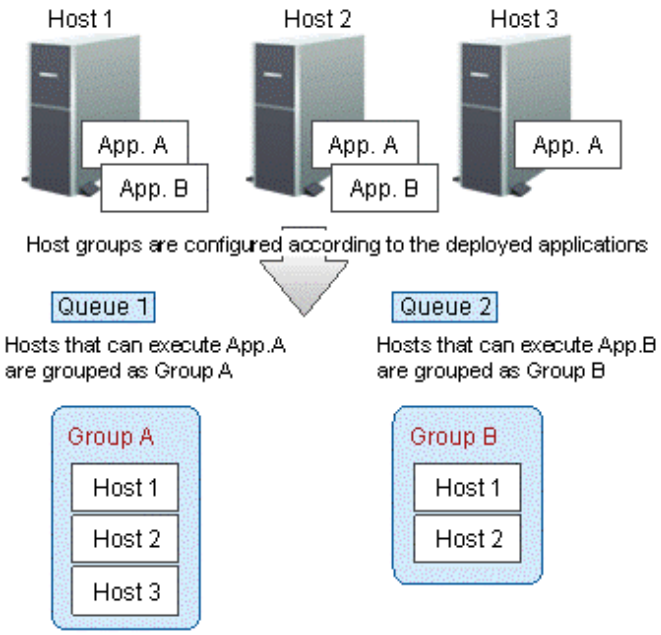
The following figure shows a representation of operations where jobs with two identical names are submitted and exclusive attributes have been specified for jobs with the same name.



Calculation processing is distributed to multiple machines and executed to enable load leveling

The execution rate can be leveled using the Distributed Execution function, even if the deployed status of the calculation application has not been integrated.

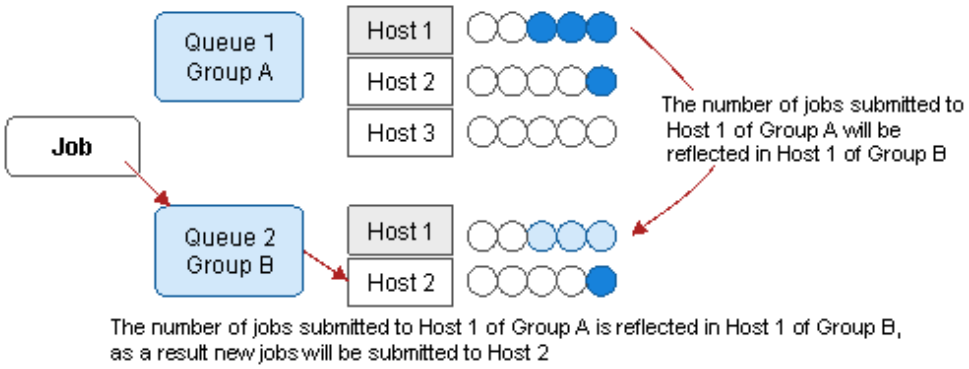
To level the execution rate, firstly define the execution server as an identical host group according to the condition, such as the deployed status of the application, and associate it with the queue.



To effectively distribute execution evenly, consider the server load between the queue and the host groups, and configure the settings so that the number of jobs between host groups is shared.

To configure the settings to share the number of jobs between host groups, select **Share the number of job entries to same execution servers between host groups.** on the **Backward compatibility** sheet of the **Define Operating Information** window.

If the job is submitted to a queue associated with an application used by the user, then the number of jobs in the queue will be reflected in all the execution servers with the same name, even if the server is in another group. As a result, jobs can be executed on servers with a lower load.



2.3.4 Reference Information

If the system stops during job execution

This section explains how jobs are handled (processed) if the system stops due to a system failure or an interruption to the power supply (including interruptions during actual operations) while a job is executing.

Demand jobs

Submitted jobs are not cleared but preserved even if the system stops.

Jobs are processed as explained below depending on their status when the system stopped.

Jobs that were waiting to be executed:

Jobs are saved and then executed when the Job Execution Control service or daemon is restarted.

Jobs that were executing

Jobs that were set (*1) to re-execute are saved and then re-executed from the beginning when the Job Execution Control service or daemon is restarted.

Jobs that were not set to re-execute are not re-executed because their queuing information is cleared. If the system stops on the execution server while network jobs or distributed execution jobs are executing, these jobs are not re-executed because their queuing information is cleared. This is regardless of the status of the jobs when the system stops and whether re-execution has been specified. This prevents the jobs from being executed twice, which would be the result if they were set for re-execution and the submitting server resubmitted the jobs as a recovery action. Take measures such as resubmitting the jobs from the submitting server after the system has restarted.

*1: Specifying re-execution

To re-execute demand jobs when the system stops, in the **Edit Job Information/Submit** dialog box, click the **Additional Information** tab and specify **Re-execute Job**.

When using commands to execute demand jobs, you can control re-execution by specifying or omitting the -nr option in the qsub command. Specifying the -nr option prohibits the re-execution of jobs. You must take care when not specifying the -nr option because operations will permit the re-execution of jobs.

Refer to the *Systemwalker Operation Manager Reference Guide* for information on the qsub command.

Scheduled jobs

If the system stops, queuing information is cleared and jobs are not re-executed. This is to prevent subsequent jobs from being re-executed because the job net has not completed processing. After the system has restarted, the only operations that should be performed are recovery operations for the job net, such as restarting it.

If the system stops on the execution server for network jobs or distributed execution jobs, their queuing information is cleared by the execution server in the same way as for demand jobs, so they are not re-executed.

Information

If the system stops on the schedule server (submitting server) for network jobs and distributed execution jobs, these jobs can continue to be executed. Refer to "Continuing Job Operations at Schedule Server System Down" in the *Systemwalker Operation Manager User's Guide* for information on how to define continuous execution.

On a cluster system, jobs can be taken over by the standby node if failover occurs. Refer to "Jobs Taken Over in the Cluster System" in the *Systemwalker Operation Manager Cluster Setup Guide* for information on the settings required for takeover of jobs.

Environment variables

Before starting jobs, the Job Execution Control function assigns values to the environment variables listed below. Use these environment variables in situations such as when the processing for subsequent jobs needs to be changed depending on the type or completion code of the proceeding job. The environment variables present when the Operation Manager is started are also inherited. However for network jobs, environment variables on the submitting server are not inherited.

JC_SUBSYSID environment variable

This variable stores the subsystem number.

JC_CHOST environment variable

This variable stores the host name of the client. This variable is set when jobs are submitted from the **Edit Job Information/Submit** dialog box or window, or from the **Select/Submit Jobs** window.

JC_CUSER environment variable

This variable stores the name of the job owner.

JC_COMMENT environment variable

This variable stores the job comment.

JC_EXHOST environment variable

This variable stores the name of the execution host.

JC_JOBID environment variable

This variable stores the job number.

JC_JOBNAME environment variable

This variable stores the name of the job.

JC_QUEUE environment variable

This variable stores the name of the queue.

QSUB_WORKDIR environment variable

This variable stores the name of the job submission directory.

JC_TRANSFILEDIR environment variable

If I/O file transfer has been performed for network jobs and distributed execution jobs, the directory where the files (sent or received on the execution server) have been stored is set in the environment variable of the job process on the execution server.

JC_SCHEDULESERVER environment variable

In network jobs and distributed execution jobs, the schedule server host name is set in the environment variable of the job process on the execution server.

Job numbers

Job numbers 1 to 99999 are used cyclically.

Job Execution Control can manage up to 99999 jobs.

2.4 Starting Services and Applications

Note

The Service/Application Execution function can only be used if the connection destination server is running Windows.

The basic operation of the Service/Application Execution function is to use the **Service/Application Execution Pattern Definition** dialog box to define the type of services and applications to be started when the power to the server is turned on, and the startup order of these services and applications. When the power to the server is turned on, these services and applications are started according to the defined startup order, which makes it possible to automatically create a job environment that is specific to the particular server. These services and applications can be changed for weekdays or weekends/holidays or for specific days, which makes it possible to create different job environments for different days.

The following section explains the basic operation of the Service/Application Execution function. For more information about how to set up and operate this function, refer to the *Systemwalker Operation Manager User's Guide*.

(1) Setting up holidays

To have different server operations for weekdays and weekends/holidays, set up the dates that will be holidays in the calendar.

(2) Registering the standard schedule

Set up the startup order of services and applications that are started on weekdays.

(3) Registering the holiday schedule

Set up the startup order of services and applications that are started on weekends or holidays.

(4) Registering user-specific definition patterns

For days where the server needs to be operated using a pattern that is different from the standard schedule or the holiday schedule, set up a user-specific definition pattern.

(5) Confirming the startup type of services

Confirm the startup type of the services for which startup patterns have been set up.

2.5 Monitoring Events [Windows Version]

This section explains how to operate the event monitoring function.

2.5.1 Basic Operations

The basic operation for event monitoring is to use the **Monitored Event Table** window to define events to be monitored and actions that are automatically executed when monitored events occur. When monitoring events occur, actions will be executed automatically according to the definitions that have been set up.

The following section explains the basic operations of the Event Monitoring function. For more information about how to define and operate this function, refer to the *Systemwalker Operation Manager User's Guide*.

(1) Defining the action execution environment

Define the environment for executing actions.

(2) Registering the monitored events

Make settings for the events for which actions will be automatically executed.

(3) Registering execution actions

Make settings for the actions that will be executed automatically when monitored events occur.

2.5.2 Additional Operations

Separate action execution conditions can be defined for each event being monitored. These execution conditions combine definitions for weekdays, weekends/holidays and times. By defining action execution conditions, different actions can be executed for weekdays or weekends/holidays, or different times of day. For example, system administrators can be notified of events by e-mail during the day and by short mail at night.

For an overview of these operations, refer to the *Systemwalker Operation Manager User's Guide*.

2.6 Managing Actions [Windows Version]

This section explains how to operate the Action Control function.

2.6.1 Basic Operations

The basic operation for Action Control is to use the **Action control** window to check and manipulate the execution status of actions when a large number of events have occurred and there are a large number of actions waiting to be executed. The execution order of actions can be changed so that certain actions are executed before others. The following kinds of operations can be performed on actions:

Execute

This operation sets the execution waiting priority (for the action selected in the **Action control** window) to the highest priority, so that the action is executed first.

Pause

This operation temporarily suspends the action selected in the **Action control** window. The action will not be executed until it is resumed.

Resume

This operation releases the pause on the action selected in the **Action control** window.

Remove

This operation deletes the action selected in the **Action control** window. Audio notifications can be deleted even if execution is in progress. Audio notifications that are executing will be stopped as soon as the "Remove" operation is executed.

For an overview of how to monitor and operate actions, refer to the *Systemwalker Operation Manager User's Guide*.

2.6.2 Additional Operations

The Action Control function provides APIs for executing and managing actions. If a program is created using these APIs, actions can be executed from the program. Adding this program (created using the action control APIs) to the Jobscheduler as a job makes it possible to perform operations such as scheduling actions, or notifying the system administrator by short mail when jobs terminate abnormally.

The following APIs are provided by the Action Control function.

- Action execution API
  - Audio notification

    This API notifies events by reading messages out loud (text-to-voice), playing WAV files or sounding beeps.
  - Audio pause

    This API stops audio notification.
  - Pop-up message display

    This API sends pop-up messages to domains, hosts or users.
  - E-Mail transmission

    This API sends emails.
  - Short mail transmission

    This API sends short mail.
- Action stop API

  This API stops the execution of actions that have already been requested to the server.
- Action status notification API
  - Action status notification API

    This API notifies the current status of actions.
  - Action data space release API

    This API releases the data space that has been allocated by the action status notification API.

For more information about these APIs, refer to the *Systemwalker Operation Manager Reference Guide*.

2.7 Backup Link [Windows Version]

This section provides information about Backup Link.

2.7.1 Basic Operations

Systemwalker Operation Manager can take backups automatically by linking to ARCserve.

Use the ARCserve command (**cabatch**) to take backups by linking to ARCserve.

For more information about how to use this command, refer to the following ARCserve manuals.

- "*BrightStor ARCserve Backup for Windows Administrator Guide*":
  The "cabatch" item in the "Using Command Line Utilities" section

For BrightStor(R) ARCserve(R) Backup r11.1 for Windows - Japanese or later versions, also refer to the following document.

- "*BrightStor ARCserve Backup for Windows Getting Started*"

Note

The Backup Link command (mpsubasx) cannot be used in ARCserve r11.5 or later.

2.8 Task Link

This section explains Task Link operations.

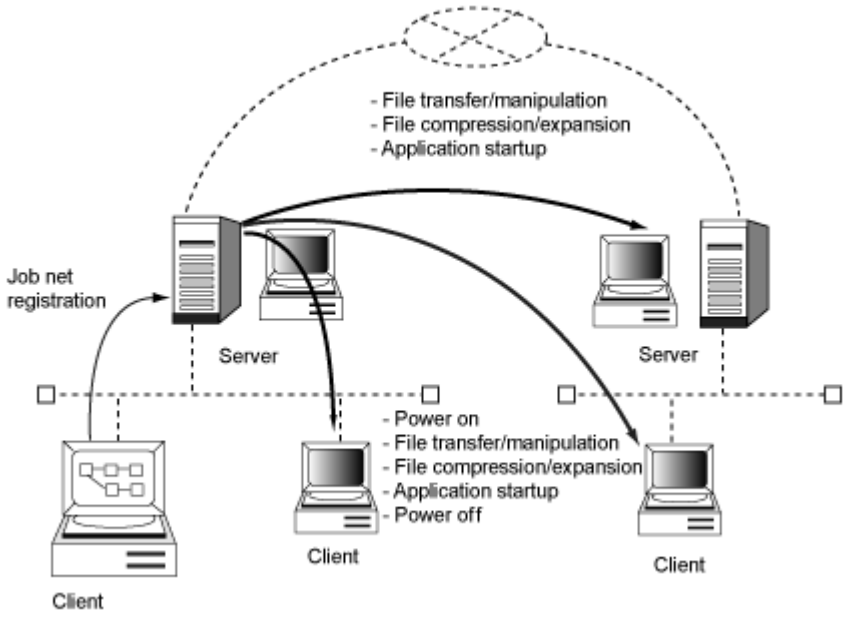
What is Task Link?

Tasks such as exchanging data between servers (or between servers and clients) can be performed by linking jobs. Jobs can be linked to match the type of the tasks being performed.

For example, a sequence of tasks can be linked as follows: a data file can be compressed on one server and then transferred to another server where it is expanded and an application is started. The processing results can then be sent back to the original server so that the data can be updated. Business systems that have previously been created using JCL and batch jobs can now be created using simple schedule operations with the Jobscheduler.

Also, a sequence of task processing, from turning on the power to the client through to turning the power off after processing has completed, can be automated using only schedule operations on the server.

The following figure shows a representation of Task Link.



Additional operations

The Task Link function can link tasks either between servers or between servers and clients.

Hereafter, task linking between servers is referred to as "Server Task Link," and task linking between servers and clients is referred to as "Client Task Link."

Server Task Link includes the following functions:

- File control
- Application startup

Client Task Link includes the following functions:

- File control
- Application startup
- Client power control

File control

The following section explains how to control files (including file transfer, file operations, compression and expansion).

File transfer

This function transfers files either between servers or between servers and clients. Between servers, files can be transferred to local files, including directories connected using NFS. Files on FTP servers can also be transferred, and files can be transferred even if the servers are running different operating systems. Files can also be transferred between servers and clients running different operating systems.

If files are only being transferred, files can be transferred to servers where Systemwalker Operation Manager has not been installed.

To use this function, register the server file transfer command or the client file transfer command as a job in a job net.

File operation

This function performs file operations (such as file deletion, file renaming, directory creation and directory deletion).

These operations can be performed on remote files as well as files on local servers and clients.

For remote file operations only, files on remote servers where Systemwalker Operation Manager has not been installed can be manipulated.

To use this function, register the server file operation command or the client file operation command as a job in a job net.

File compression and expansion

This function compresses and expands local files.

This function can be used to compress a file on one server and then expand it on another server. This makes it possible to reduce both file transfer time and the load on the network.

To use this function, register the server file compression/expansion command or the client file compression/expansion command as a job in a job net.

Application startup

This function starts applications on servers or clients.

If an application is started on a server, this function does not wait for the application to complete. However, if an application is started on a client, this function can wait for the application to complete.

To use this function, register the server application startup command or the client application startup command as a job in a job net.

Note

With server application startup, applications that require entry from the keyboard cannot be started. Do not register this kind of application.

Client power control

The client power control function allows the following operations.

- Power-on
- Power-off

Power-on

This function turns the power to clients on. Clients at distant places can be turned on automatically.

Because this function uses the Wakeup on LAN power-on mechanism, only clients that support Wakeup on a LAN can be turned on.

In TCP/IP environments, the power to clients on other subnets can also be turned on.

To use this function, register the client power-on command in a job net.

Power-off

This function turns the power to clients off if Client Task Link is running on the client. Clients at distant places can be turned off automatically.

The power can be turned off for clients that support turning the power off from Windows.

To use this function, register the client power-off command as a job in a job net.

Basic operations

This section outlines the basic operations for Task Link.

For Task Link to run, the commands corresponding to the various functions are provided as Task Link commands.

Register Task Link commands as jobs in job nets. Tasks that are registered in job nets are referred to as Task Link jobs.

When Task Link commands are registered in a job net as jobs, they can be monitored and operated in the same way as other scheduled jobs.

The following section provides an overview of registering, monitoring, and operating Task Link jobs.

Registering Task Link jobs

Task Link jobs are registered by selecting the icon corresponding to a Task Link command from the **New/Change Job Net** window for job nets with the Job Execution Control attribute.

For more information about how to register jobs, refer to the *Systemwalker Operation Manager User's Guide* and the *Systemwalker Operation Manager Online Help*. Also, for more information about Task Link commands, including syntax rules, refer to the *Systemwalker Operation Manager Reference Guide*.

Note

Notes on registering Task Link jobs

- When a Task Link icon is selected without using the Command Wizard, do not specify execution programs other than Task Link commands in the **Specify Command Name** field of the **Add/Change - Job** window. When an icon other than a Task Link icon is selected, do not specify a Task Link command in the **Specify Command Name** field of the **Add/Change - Job** window.
- Task Link jobs cannot be specified in the following situations.
  - When connecting to a server running SystemWalker/OperationMGR V4.0L20 or earlier [Windows version]
  - When connecting to a server running SystemWalker/OperationMGR V5.0 or earlier [UNIX version]
  - When adding jobs to a job net with an execution attribute other than Job Execution Control

Monitoring Task Link jobs

Either the **Monitor Job Net** window or the **Job List** window can be used to monitor Task Link jobs.

For more information about monitoring Task Link jobs, refer to the *Systemwalker Operation Manager User's Guide* and the *Systemwalker Operation Manager Online Help*.

Operating Task Link jobs

Operations on Task Link jobs (cancel, restart, pause, continue, disable, and enable) can be performed as required.

For more information about how to perform these operations, refer to the *Systemwalker Operation Manager User's Guide*.

2.9 Systemwalker Scripts

This section explains Systemwalker script operations.

2.9.1 What Are Systemwalker Scripts?

Systemwalker scripts use a version of the Tool Command Language/Tool Kit (Tcl/Tk) that has been extended for Systemwalker.

Systemwalker scripts consist of the language (extended for Systemwalker Operation Manager) and samples (provided for Systemwalker Centric Manager).

The extended language for Systemwalker Operation Manager can only run in Systemwalker Operation Manager execution environments. Similarly, the samples for Systemwalker Centric Manager can only run in Systemwalker Centric Manager execution environments.

This manual explains the Systemwalker scripts provided for Systemwalker Operation Manager.

For information about the Systemwalker sample scripts provided for Systemwalker Centric Manager, refer to the documentation for Systemwalker Centric Manager.

Features of Systemwalker scripts

Systemwalker scripts are based on the Tool Command Language/Tool Kit (Tcl/Tk) and are not dependent on the operating platform.

Until now, creating jobs required knowledge of the different job construction languages for each platform, and jobs needed to be created separately for each platform. However, Systemwalker scripts can be used on any platform, so jobs can be created more efficiently and resources can be shared.

What Systemwalker scripts can do

Systemwalker scripts can be executed in the same way as other job files.

Systemwalker scripts make it possible to create the following kinds of jobs.

- Jobs that manipulate character strings
- Jobs that execute subsequent processes by evaluating complex conditions. (For example, jobs that decide which job to execute next based on the presence of a file and information in the file, in addition to the completion code of the preceding job.)

2.9.2 Basic Operations

The following figure shows the basic operation flow for Systemwalker scripts.

Creating Systemwalker scripts

Create a Systemwalker script.

Using a text editor such as notepad or "vi," create a Systemwalker script by following the syntax rules for Systemwalker scripts, and save the script with the "swt" file extension. Before saving the script, insert calls to the trace extension command at certain key points in the script so that it can be debugged. Fujitsu recommends setting calls to the trace extension command in all routes.

For information about creating scripts, refer to the *Systemwalker Operation Manager Reference Guide*.

Testing and debugging scripts

Use the **swotclsh** command (the script execution command) to run the script that has been created alone. The trace extension commands that were added when the script was created will output information to a trace file, making it possible to check whether the script is running as expected and make corrections if necessary.

Operating jobs with Systemwalker Operation Manager

Systemwalker scripts can be executed as Systemwalker Operation Manager jobs.

Systemwalker scripts can be used as scheduled jobs or demand jobs, in the same way as batch files and JCL scripts.

Using scripts as scheduled jobs

Register Systemwalker scripts as jobs in job nets. In this case, the job net must have the Job Execution Control attribute. Systemwalker scripts cannot be used within job nets with other attributes.

Once a script has been registered as a job, it can be operated in the same way as for other Systemwalker Operation Manager jobs with the Job Execution Control attribute. For information about how to define job nets and perform other Systemwalker Operation Manager operations (such as monitoring), refer to the *Systemwalker Operation Manager User's Guide* and the *Systemwalker Operation Manager Online Help*.

Using scripts as demand jobs

Systemwalker scripts can be submitted as demand jobs. Systemwalker scripts can also be registered in job folders and executed in the same way as batch files and JCL scripts.

The procedure for submitting Systemwalker scripts as demand jobs is the same as for batch files and JCL scripts. For more information, refer to the *Systemwalker Operation Manager User's Guide* and the *Systemwalker Operation Manager Online Help*.

2.9.3 Debugging Systemwalker Scripts

Although Systemwalker scripts can be run without having to be compiled, script errors cannot be detected until the section with the error is executed. To avoid problems with production operations using Systemwalker Operation Manager, scripts need to be tested and debugged after they are created, and any errors need to be removed.

The following kinds of errors may occur when scripts are created.

- Simple typos
- Syntax errors
- Logical errors

  Examples of logical errors include conditional expressions that are syntactically correct, but that do not evaluate in the way that was intended, or the values for necessary variables may not be set as a result of the way that conditions branch.

- Poor consideration of execution environments

  Examples of this kind of error include situations where files that should always exist do not exist, or where startup programs output unexpected information.

Trace levels

To make debugging easier, add calls to the trace extension command to the scripts in advance.

An optional trace level can be specified with the trace extension command. For information about how to set up trace levels, refer to the *Systemwalker Operation Manager Reference Guide*.

Changing the trace level changes which trace information is collected. There are two trace levels. Fujitsu recommends using these trace levels as follows:

Trace level 1: Used to output information during actual operations

With this trace level, information such as startup data and processing results is output as "operation logs."

Detailed information provided when errors are detected is output as "error logs."

Trace level 2: Used to output information during testing and debugging

With this trace level, "route check" information showing how many routes the scripts have executed (process branching points and program startup points) is output.

Detailed information (such as the content of files that have been read and variables in loop processing) is also output as "internal script information".

Testing and debugging procedure

The following procedure is used for testing and debugging Systemwalker scripts.

1. Script creation process

   Add calls to the trace extension command for debugging the script.

   - Use Level-1 tracing to collect operation logs and error logs.
   - Use Level-2 tracing to collect route check and internal script information.

2. Testing and debugging process

   1. Execute the script using the test execution command (**swotclsh**). Note the following points.

      - If there is any processing that cannot (or should not) actually be executed during the test (such as starting an application that cannot or should not be started), comment out the processing or replace the application with a dummy program for the test.
      - Collect and look up route checking traces during the test, and make sure that the script passes through the correct route.
      - Test all routes, including error processing routes, and conditional branches of scripts one by one. (This is because scripts do not detect errors such as typing mistakes with variable names unless the route with the error is actually executed.)

   2. If the script does not run correctly, locate and remove errors by checking the traces that have been collected and the error messages output by "Tcl/Tk" (if errors have occurred).

3. After job operations have commenced

   Fujitsu recommends always collecting the minimum information required for debugging, as potential problems may appear after operations have commenced. If an error occurs with script behavior, save and investigate the trace information (which is always being collected). If necessary, collect more detailed information by increasing the trace level.

2.10 User Management

Systemwalker Operation Manager users can be managed by one of the following methods:

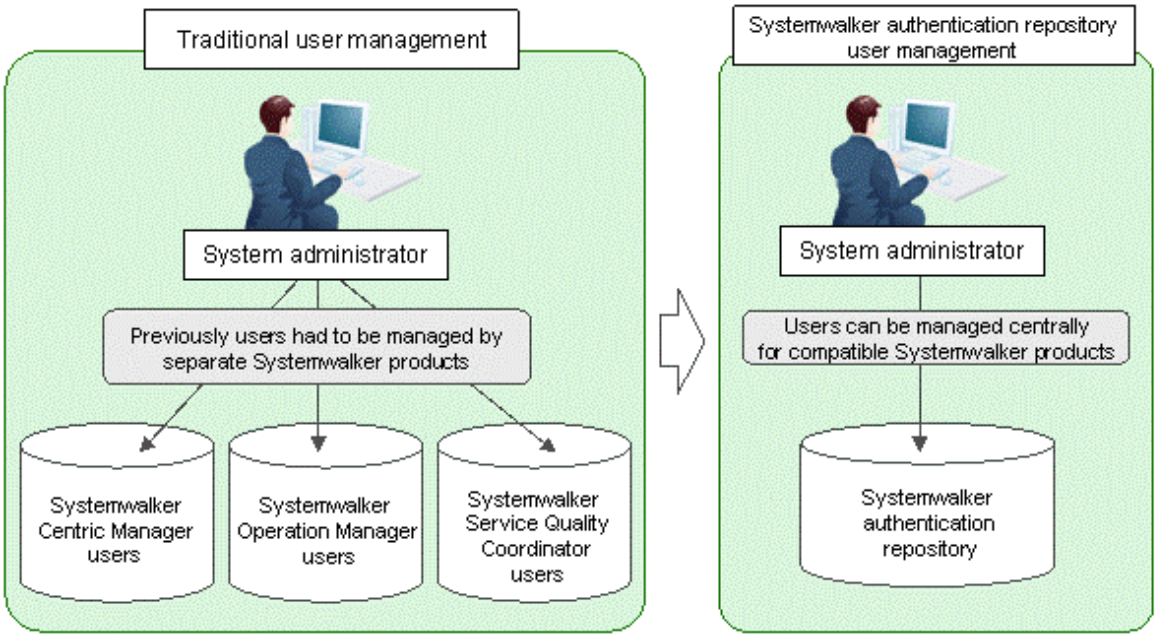
- OS user management
- Systemwalker Extended User Management

- Systemwalker authentication repository

Systemwalker authentication repository

The Systemwalker authentication repository is a directory that centralizes Systemwalker user information, and can be shared between Systemwalker Operation Manager and other Systemwalker products. It is configured using either Interstage Directory Service or Active Directory. Refer to the *Systemwalker User's Guide - Systemwalker User Management and Single Sign-on* for more information.

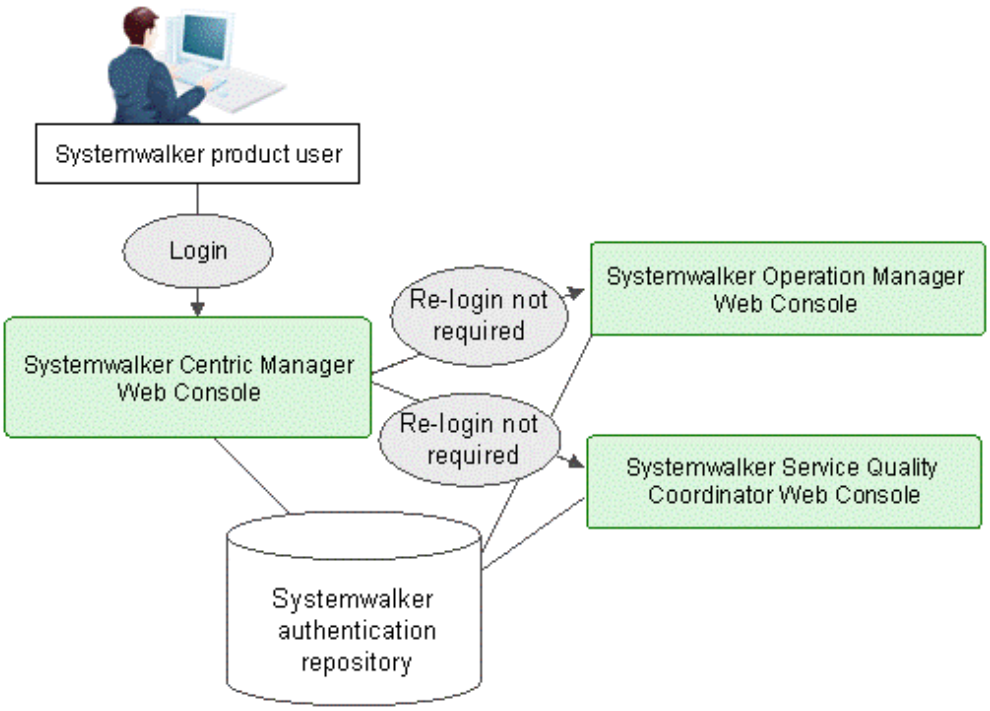
Using the Systemwalker authentication repository makes it possible to centrally manage user IDs instead of through each separate Systemwalker product, as shown below:



By using a Systemwalker user ID managed by the Systemwalker authentication repository, it is possible to log into the Systemwalker Operation Manager Web Console or Windows client and use the various Systemwalker Operation Manager functions.

Systemwalker Single Sign-On

When the Systemwalker authentication repository is used, the use of a Systemwalker Single Sign-On between Systemwalker products can be created. For example, after logging into the Systemwalker Operation Manager Web Console, the user ID and password would no longer be required for the Systemwalker Operation Manager Web Console and the Systemwalker Service Quality Coordinator Web Console, as shown below:



Single Sign-On linked with ServerView Operations Manager

If you have built a cloud environment with ServerView Resource Orchestrator installed, you can link to ServerView Operations Manager and use the following function:

- Single Sign-On

If the ServerView Resource Orchestrator console is running, this function can be used without the need to log in again to the Systemwalker Operation Manager Web console. This facilitates efficient operations in the cloud environment. This function is supported by servers running Windows or Linux.

Note

- The Systemwalker authentication repository and the Systemwalker Single Sign-On function cannot be used simultaneously with the Single Sign-On function that is linked with ServerView Operations Manager.
- If using Active Directory to manage OS users, do not use a domain user with the same name as a user that already exists on the computer where Systemwalker Operation Manager will be installed.

Refer to "Definitions when Linking with ServerView Operations Manager to Perform Single Sign-On" in the *Systemwalker Operation Manager Installation Guide* for details.

Refer to "Systemwalker Operation Manager users [UNIX version]" in "2.2.2 Additional Operations" for an overview of the Systemwalker Extended User Management function.

2.11 Linking to Systemwalker Centric Manager

This section explains how Systemwalker Operation Manager can link to Systemwalker Centric Manager. For more information (such as procedures for setting up the environment and other notes), refer to the *Systemwalker Operation Manager Installation Guide*.

Monitoring job net execution status

Linking to Systemwalker Centric Manager makes it possible to monitor the execution status of job nets in medium- and large-scale systems from Systemwalker Centric Manager.

There are three monitoring methods, as shown below.

With each of these methods, the execution status of job nets can be monitored in the system monitoring window of Systemwalker Centric Manager.

- Monitoring when job nets terminate abnormally or restart

  If a Systemwalker Operation Manager job net terminates abnormally, the job net is displayed in the **System Monitoring** window of Systemwalker Centric Manager. Information about the job net that terminated can be sent using the mail notification action for event monitoring.

  When the job net that has terminated abnormally is restarted or confirmed, its execution status automatically changes to "resolved" in the System Monitoring window of Systemwalker Centric Manager.

- Monitoring by displaying the monitoring window for job nets that have terminated abnormally

  By selecting a job net abnormal termination event from the event list in the **System Monitoring** window of Systemwalker Centric Manager, the monitoring window for the job net can be displayed directly without opening the **Systemwalker Operation Manager** window.

  This procedure only works if all Systemwalker Operation Manager servers and clients and all Systemwalker Centric Manager Operation Management Servers and Operation Management Clients are running V12.0L10/12.1 or later.

  If the version of the Systemwalker Operation Manager server, Systemwalker Centric Manager Operation Management Server, and Systemwalker Centric Manager Operation Management Client is V13.4.0 or later, the Web Console monitoring window will be displayed.

  If any version of the Systemwalker Operation Manager server, Systemwalker Centric Manager Operation Management Server, or the Systemwalker Centric Manager Operation Management Client is V13.3.1 or earlier and the Systemwalker Operation Manager client function has been installed, then the Windows client **Monitor Job Net** window will be displayed.

  If the version of the Systemwalker Operation Manager server, Systemwalker Centric Manager Operation Management Server is V13.4.0 or later, users in the Systemwalker authentication directory can be used.

- Monitoring by outputting job net execution histories to the event log or SYSLOG

  The execution status of job nets that have been output to the event log or SYSLOG is displayed in the System Monitoring window of Systemwalker Centric Manager. Information about the job net that terminated can be sent using the mail notification action for event monitoring.

  The job net execution statuses to be monitored can be selected from the following list using environment settings. The level shown in parentheses indicates the level of each event log message (Windows version) or SYSLOG message (UNIX version).

  - Started (information level or "INFO" level)
  - Completed (information level or "INFO" level)
  - Abended ("Resolved" is not shown automatically after restart.) (Error level or "ERR" level)
  - Skipped (Warning level or "WARNING" level)
  - Canceled (Error level or "ERROR" level)
  - Closed (Error level or "ERROR" level)
  - Refused (Warning level or "WARNING" level)
  - Pseudo-normal (information level or "INFO" level)

  For information about how to set up the environment, refer to the *Systemwalker Operation Manager Installation Guide*.

Centrally managing audit log files

Systemwalker Operation Manager can manage audit logs centrally on the Operation Management Server by linking to Systemwalker Centric Manager and using the Systemwalker Centric Manager Audit Log Management function to collect records of the operations performed on Systemwalker Operation Manager servers.

Security can be improved by using the Audit Log Management function in the following kinds of situations:

- To manage audit logs centrally in order to find errors and illegal operations on all servers in the entire system, not only on a particular system.
- To find clues for resolving problems when system errors or illegal operations occur, by accumulating audit logs collected from servers.

To use the Systemwalker Centric Manager Audit Log Management function, specify the -y option of the *mpsetlogsend_omgr* command. For more information about this command, refer to "mpsetlogsend_omgr command" in the *Systemwalker Operation Manager Reference Guide*.

Also, refer to "Defining Audit Log Output" in the *Systemwalker Operation Manager Installation Guide* for information about changing the settings for the Systemwalker Centric Manager Audit Log Management function.

For more information about the Systemwalker Centric Manager Audit Log Management function, refer to the *Systemwalker Centric Manager Solution Guide - Security*.

2.12 Creating Existing Environments on Other Servers

Systemwalker Operation Manager can extract the following information from a server that is already operating and distribute it to another server.

- Environment definition information for Systemwalker Operation Manager
- Registration information for Systemwalker Operation Manager

This information is collectively referred to as "policy data".

By extracting and distributing policy data, the same operating environment as a server that is already operating can be set up on other servers.

However, only users with administration authority (that is, system administrators) can perform this operation.

Policy data can be extracted and distributed for each individual function provided by Systemwalker Operation Manager, but it is not possible to extract only a part of one of these functions (such as only particular calendars or particular exits).

The following section shows the details of the two types of policy data that can be extracted and distributed using Systemwalker Operation Manager.

Environment definition information for Systemwalker Operation Manager

- Monitored host information
- Execution environment information for actions defined using Action Control [Windows version]
- Information relating to Jobscheduler definitions, such as startup parameters, message tables [Windows version], and monitoring permission hosts
- Information relating to Job Execution Control definitions, such as operation information, job owners [Windows version], and trust hosts
- User information defined in user control lists for job execution [UNIX version]
- EE Information relating to environment settings for the Master Schedule Management function

Registration information for Systemwalker Operation Manager

- Operation Manager user information [UNIX version]
- Calendar information and power control information
- Services and application startup information (*1)
- Information about monitored events and execution actions [Windows version]
- Information relating to schedules, schedule patterns, and exit files registered with the Jobscheduler
- Job folder information

EE - Master information for the Master Schedule Management function

*1 The service startup function can only be used when the connection destination server is running Windows.

For more information about extracting and distributing policy information, refer to the *Systemwalker Operation Manager Installation Guide*.

EE Refer to the *Systemwalker Operation Manager User's Guide - Master Schedule Management* for information on the method that is used to extract/distribute Master Schedule Management function policy information.

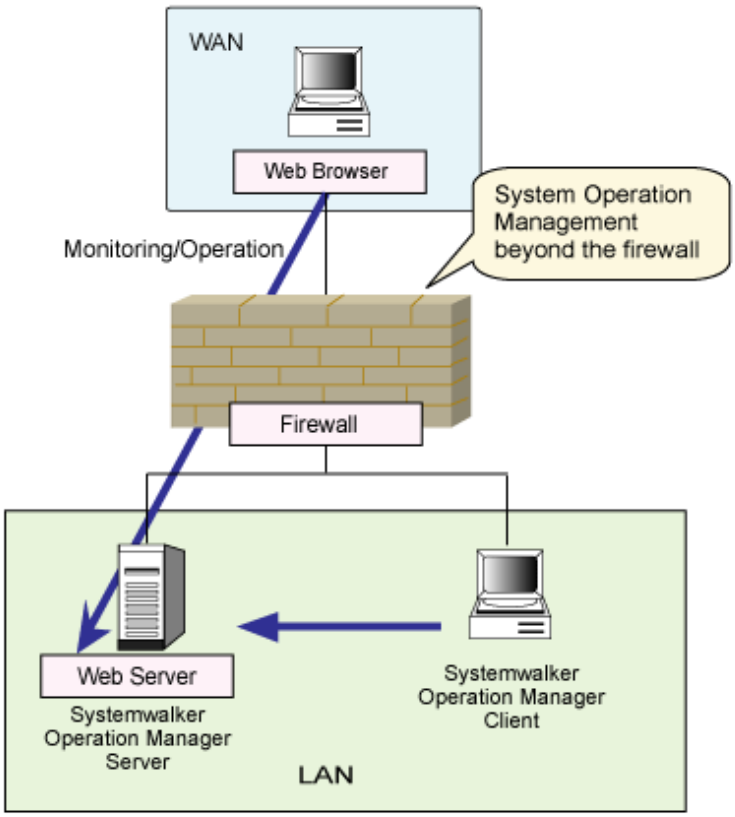
2.13 Monitoring and Operating Jobs Across Firewalls

When a network configuration joins up LANs in multiple locations using a WAN (in a corporate network, for example), firewalls are installed at the contact between the WAN and LANs to improve security by controlling unnecessary communications.

With this kind of network configuration, the networks inside each LAN need to be able to be administered from outside the firewalls.

Operations in firewall environments

When Systemwalker Operation Manager is used in a firewall environment, Systemwalker Operation Manager servers in the firewall environment can be monitored and operated using a web browser, as shown in the following configuration diagram. In this configuration, each Systemwalker Operation Manager server in the firewall environment should also be a Web server.



Note

Jobs should be registered from a Systemwalker Operation Manager client inside each LAN.

For information about the functions available from the web browser, refer to "2.15 Using Systemwalker from a Web Browser". For more information about the environment definitions required for monitoring, refer to the *Systemwalker Operation Manager Installation Guide*.

2.14 Monitoring and Operating Jobs from Outside NAT Environments

Increasing numbers of IDCs (Internet Data Centers) are attempting to improve corporate security by using the address translation function of NAT routers to hide internal network addresses in customer environments so that these addresses are not disclosed to the public.

In environments that use NAT routers like this, internal networks, which are hidden by the address translation function, need to be administered from a network on the other side of the NAT router.

Configurations that use NAT

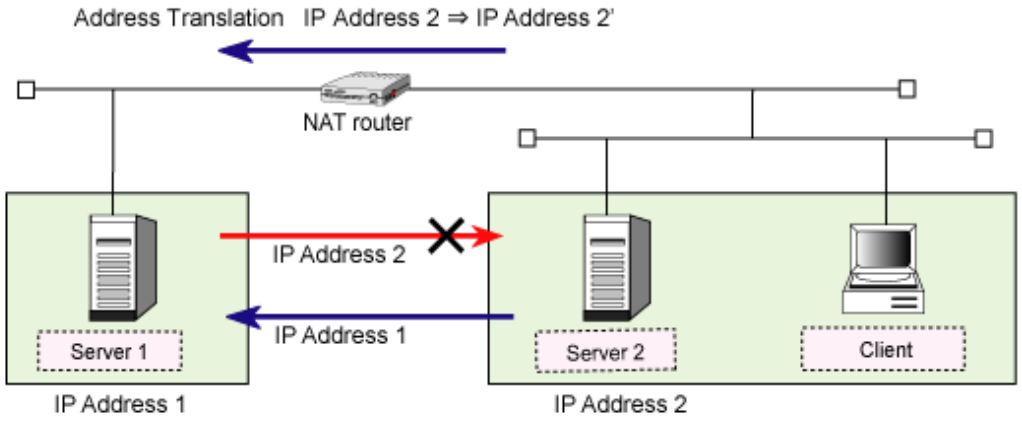
NAT has both a static translation function that maintains static correspondence between hidden IP addresses and converted IP addresses, and a dynamic translation function that associates hidden IP addresses dynamically with converted IP addresses.

Systemwalker Operation Manager only supports 1:1 static address translation.

With 1:1 static address translation, 1:1 static correspondence is maintained between the IP addresses hidden by NAT and the IP addresses converted by the address translation function.

Systemwalker Operation Manager can manage networks and jobs in the following basic NAT environments.

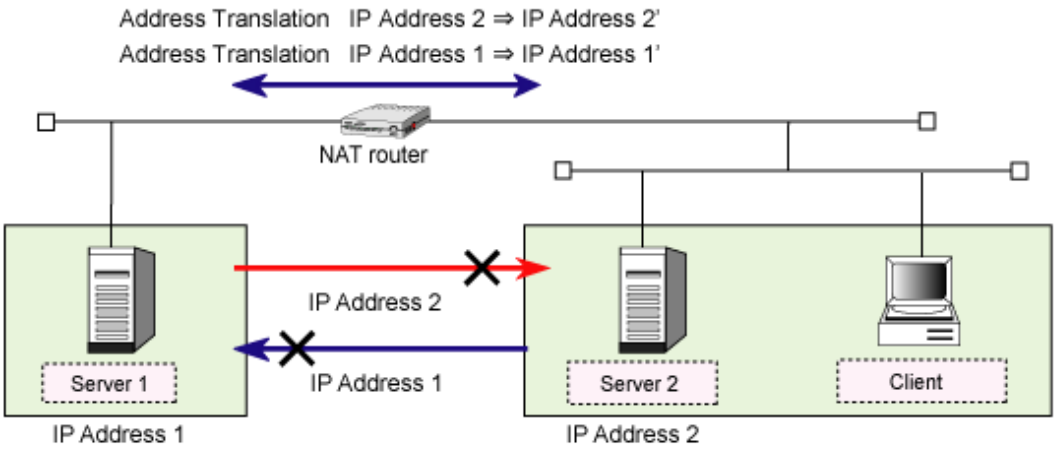
NAT configuration 1 (Address translation where addresses in the NAT environment are hidden)



If Server 1 has *IP address 1* and Server 2 (in a NAT environment) has *IP address 2*, the address translation function of the NAT router hides *IP address 2* of Server 2 in the NAT environment from Server 1 so that, from the viewpoint of Server 1, the IP address of Server 2 appears as *IP address 2'*, which is actually the result of conversion by the address translation function.

However, the IP address of Server 1 is not hidden from Server 2 and its clients in the NAT environment so that, from the viewpoint of Server 2, the IP address of Server 1 appears as *IP address 1*.

NAT configuration 2 (Address translation where the addresses of both the NAT environment and Server 1 are hidden)

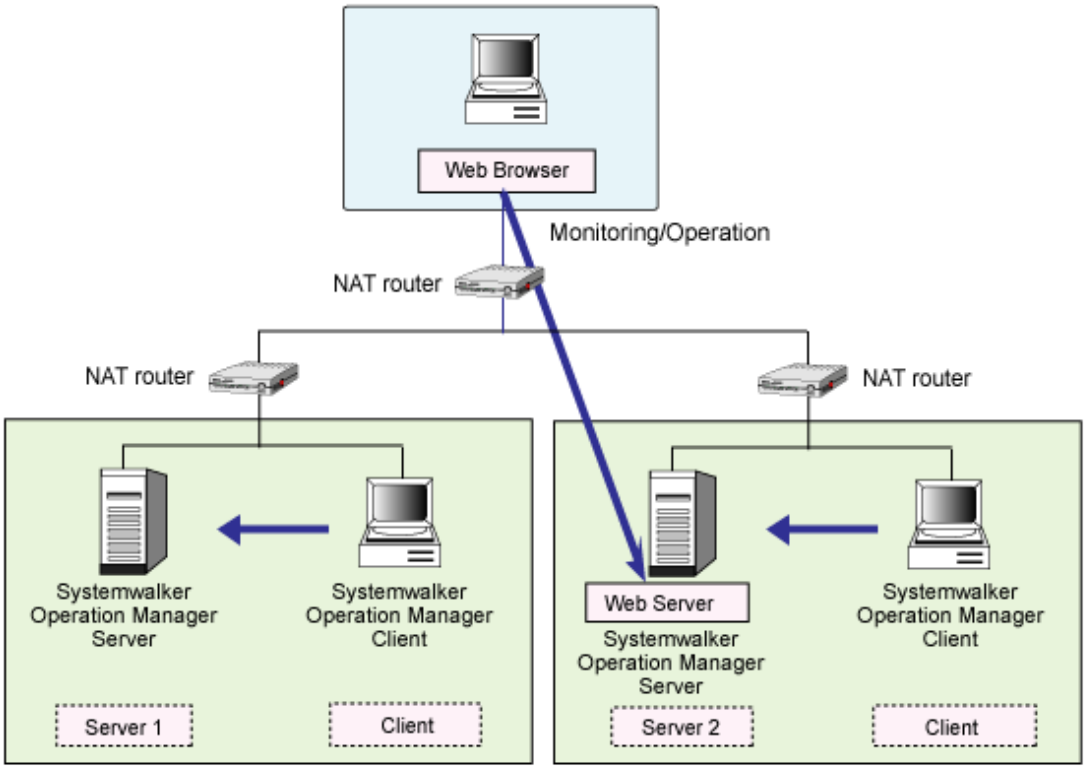


If Server 1 has *IP address 1* and Server 2 (in a NAT environment) has *IP address 2*, the address translation function of the NAT router hides *IP address 2* of Server 2 in the NAT environment from Server 1 so that, from the viewpoint of Server 1, the IP address of Server 2 appears as *IP address 2',* which is actually the result of conversion by the address translation function.

In addition, the address translation function also hides *IP address 1* of Server 1 from Server 2 and its clients in the NAT environment so that, from the viewpoint of Server 2, the IP address of Server 1 appears as *IP address 1'*, which is actually the result of conversion by the address translation function.

Operation in NAT environments

When Systemwalker Operation Manager is used in a NAT environment, Systemwalker Operation Manager servers can be monitored and operated from a web browser, as shown in the following configuration diagram. In this configuration, each Systemwalker Operation Manager server in the NAT environment should also be a Web server.



Note

Jobs should be registered from a Systemwalker Operation Manager client inside each LAN.

For information about the functions available from the web browser, refer to "2.15 Using Systemwalker from a Web Browser". For more information about the environment definitions required for monitoring, refer to the *Systemwalker Operation Manager Installation Guide*.

2.15 Using Systemwalker from a Web Browser

With Systemwalker Operation Manager, functions related to monitoring and operating the Jobscheduler and Job Execution Control can be used from a Web browser by using a service that controls communication with the Web GUI.

This section explains the following:

- List of functions that can be used from a Web browser
- URLs for using functions from a Web browser

For information about the operation environments required for using a Web browser, refer to "3.2 Software Resources".

List of functions that can be used from a Web browser

With Systemwalker Operation Manager, the following functions can be used from a Web browser.

Jobscheduler

- Looking up lists of the job nets registered in groups
- Looking up lists of the jobs registered in job nets
- Looking up a list of the message events that have been registered as job net startup conditions
- Displaying Gantt charts
- Looking up the number of job nets with each type of status
- Multi-server monitoring
- Operating groups and job nets (cancel, startup, restart, pause, continue, disable, enable, and confirm)
- Looking up a list of all registered groups and job nets
- Looking up job output information
- Looking up job net and job history information
- Displaying and editing memos

Job Execution Control

- Displaying a list of job statuses
- Operating jobs
- Displaying detailed job information
- Displaying summaries of job folder, listing job files, and submitting jobs
- Listing the status of all queues on connected servers
- Looking up detailed queue information
- Operating queues (pausing, starting, and temporarily changing definitions).

URLs for using functions from a Web browser

To use Systemwalker Operation Manager Web Console from a Web browser, specify the following URLs.

http://Systemwalker Operation Manager server IP address:port number /Systemwalker-omgr/login.op

To use https, specify the following:

https://Systemwalker Operation Manager server IP address:port number /Systemwalker-omgr/login.op

2.16 Linking to Other Products

This section presents an overview of linking Systemwalker Operation Manager to other products.

2.16.1 Linking to Interstage

Note

This function is supported by Windows, Solaris, and Linux versions.

Interstage defines execution environments for each operational unit in "WorkUnits". Systemwalker Operation Manager can link to Interstage by registering WorkUnits that have been defined using Interstage as jobs in job nets. By starting a registered WorkUnit, online business processes can be performed, and online business operations can be automated, monitored and operated.

The following WorkUnits can be registered as jobs. Note that which WorkUnits can be handled may vary depending on the version and level of Interstage. For more information, refer to "3.2 Software Resources".

- TD (Transaction Director) WorkUnits (Windows x86/Solaris 32-bit/Linux x86)
- EJB (Enterprise JavaBeans) WorkUnits (Windows x86/Solaris 32-bit/Linux x86)
- Utility WorkUnits [UNIX version] (Solaris/Linux)
- CORBA WorkUnits (Windows/Solaris/Linux)

When WorkUnits are registered and executed, related batch jobs can also be executed using groups. At the same time, exclusive control can also be performed between WorkUnits and batch jobs. WorkUnits must be registered in job nets with the Interstage attribute.

For more information about Interstage and how to define WorkUnits, refer to the Interstage manuals.

For information about registering, monitoring and operating Interstage jobs, refer to the *Systemwalker Operation Manager User's Guide*.

Chapter 3 Operating Environment

This chapter explains the system environment required for operating Systemwalker Operation Manager.

3.1 Hardware Resources

This section explains the hardware resources required to install and operate Systemwalker Operation Manager.

Hardware resources required for installing Systemwalker Operation Manager

This section explains the hardware resources required for installing Systemwalker Operation Manager.

Disk capacity

The disk capacity required for installing Systemwalker Operation Manager is shown in the tables below.

The disk where the Windows system has been installed is referred to as the "system directory," and the disk where Systemwalker Operation Manager has been installed is referred to as the "installation directory."

[Windows x86]

Additional functions	Areas used	Server	Client
No additional functions selected (default installation)	System directory	30 MB or more	15 MB or more
	Installation directory	660 MB or more	395 MB or more
	Total	690 MB or more	410 MB or more
All additional functions selected	System directory	30 MB or more	15 MB or more
	Installation directory	680 MB or more	400 MB or more
	Total	710 MB or more	415 MB or more

[Windows x64]

Additional functions	Areas used	Server	Client
No additional functions selected (default installation)	System directory	45 MB or more	35 MB or more
	Installation directory	700 MB or more	375 MB or more
	Total	745 MB or more	410 MB or more
All additional functions selected	System directory	45 MB or more	35 MB or more
	Installation directory	720 MB or more	380 MB or more
	Total	765 MB or more	415 MB or more

[Solaris 32bit version]

Additional functions	Areas used	Server	Client
All additional functions selected	/(root) (Note)	1 MB or more	---
	/opt	480 MB or more	---
	/etc	25 MB or more	---
	/var	150 MB or more	---
	System directory	---	35 MB or more

Additional functions	Areas used	Server	Client
	Installation directory	---	380 MB or more
	Total	656 MB or more	415 MB or more

Note)

If there is not /opt or /etc or /var directory, it takes each capacity additionally in /(root) directory.

[Solaris 64bit version]

Additional functions	Areas used	Server	Client
All additional functions selected	/(root) (Note)	1 MB or more	---
	/opt	480 MB or more	---
	/etc	25 MB or more	---
	/var	150 MB or more	---
	System directory	---	35 MB or more
	Installation directory	---	380 MB or more
	Total	656 MB or more	415 MB or more

Note)

If there is not /opt or /etc or /var directory, it takes each capacity additionally in /(root) directory.

[Linux x86]

Additional functions	Areas used	Server	Client
All additional functions selected	/opt	580 MB or more	---
	/etc	30 MB or more	---
	/var	110 MB or more	---
	System directory	---	35 MB or more
	Installation directory	---	380 MB or more
	Total	720 MB or more	415 MB or more

[Linux x64]

Additional functions	Areas used	Server	Client
All additional functions selected	/opt	580 MB or more	---
	/etc	30 MB or more	---
	/var	180 MB or more	---
	System directory	---	35 MB or more
	Installation directory	---	380 MB or more
	Total	790 MB or more	415 MB or more

Hardware resources required for operating Systemwalker Operation Manager

In addition to the hardware resources required for installing Systemwalker Operation Manager, it is also necessary to estimate the following memory and disk requirements for operating Systemwalker Operation Manager.

- Memory requirements
- Disk space requirements for different operation configurations

Memory requirements

The following table lists the amount of memory that is required to run Systemwalker Operation Manager.

[Windows x86]

Server	Client
320 MB or more	60 MB or more

[Windows x64]

Server	Client
420 MB or more	60 MB or more

[Solaris 32bit version]

Server	Client
400 MB or more	60 MB or more

[Solaris 64bit version]

Server	Client
480 MB or more	60 MB or more

[Linux x86]

Server	Client
230 MB or more	60 MB or more

[Linux x64]

Server	Client
390 MB or more	60 MB or more

[HP-UX]

Server	Client
90 MB or more	50 MB or more

[AIX version]

Server	Client
20 MB or more	50 MB or more

Disk space requirements for different operation configurations

The disk space shown below is required to operate Systemwalker Operation Manager, depending on the scale of operation. Estimate the disk space required and allocate more than enough space when installing Systemwalker Operation Manager, by referring to the tables below.

Note that Systemwalker Operation Manager files are stored in the following location. (With the exception of the files for the Jobscheduler, Job Execution Control, Task Link and Systemwalker trace information)

[Windows version] *Installation directory*

[UNIX Version] /var

<Calendar>

Files required for operations	Required disk space (bytes)
Calendar file	<*number of calendars*> x 14 KB + 94 KB
Power control file	5 KB
Log file (Note)	6 MB

Note)

Operation log files for calendar and power control

<Starting services and applications>

Files required for operations	Required disk space (bytes)
Information file	2 KB

Note)

The service startup function cannot be used with UNIX systems.

<Jobscheduler>

Files required for operations	Required disk space (bytes)
Basic capacity	13 MB
Schedule information file	(512 x <*number of projects*>) x 2 + (10000 x <*total number of job nets*> + 2200 x <*total number of jobs*>) x 2 (Note 1)
Log file (Note 2)	<*value set in Jobscheduler startup parameter (1 to 99 MB)*> x 3
Job net history file	10000 x <*number of job nets*> x <1 to 24 generations>
Standard output and error output files for each job	Total space required for all registered jobs that send output to the standard output and the error output
Message event history file	99 MB x 3

Note 1)

To operate groups, add the following value to the value shown in the table above:
(1000 x <*total number of groups*> + 27400 x <*total number of job nets belonging to groups*>) x 2

Note 2)

Log files for job net execution histories

Remarks 1:

If job nets are nested, child job nets are registered as jobs in their parent job nets. This means that jobs need to be counted, as well as job nets.
Each file is stored in the following location.

[Windows version] Installation directory

[Solaris version/Linux version] /var

[HP-UX version/AIX version] /opt

Remarks 2:

Both jobs and job nets need to be counted because linked job nets are registered as jobs in parent job nets. If the referenced master linked job net is also nested, then the linked job net will also be nested and all job nets and jobs registered under the nested linked job net will need to be counted.

<Job Execution Control>

Files required for operations	Required disk space (bytes)
Basic capacity [Windows version]	14 MB
Basic capacity [UNIX version]	15 MB
Spool	256 bytes x <*q*> + <*n*> x (4 KB + <*j*> + <*o*> + <*e*>) q: Number of queues n: Maximum number of jobs that can exist concurrently in the system (the total number of jobs that are "waiting", "on hold", "active", and "output on hold") j: Size of job file o: Size of standard output file e: Size of standard error output file
Trace file [Windows version] (Note)	500 KB x 202
Trace file [UNIX version] (Note)	5 MB x 2 x 12

Note)

The trace file is used for cyclic control of trace information within the calculated disk capacity.

Remarks:

To collect log files and operation record files, there needs to be enough space to store them.
For information on how to make these estimates, refer to the *Systemwalker Operation Manager Installation Guide*.

Each file is stored in the following location.

[Windows version] Installation directory

[Solaris version/Linux version] /var

[HP-UX version/AIX version] /opt

<Event Monitoring [Windows version]>

Files required for operations	Required disk space (bytes)
Definition file	<*number of event lines defined*> x 2 KB (Note)

Note)

If complex definitions and long parameters are specified, the required capacity may exceed this limit.
In this case, double the estimate for the space required.

<Task Link>

Files required for operations	Required disk space (bytes)
Trace file [Windows version]	6.2 MB
Host information definition file	350 x <*number of registration hosts*>
Password management file	200 x <*number of registration users*>

Remarks:

Each file is stored in the following location.

[Windows version] Installation directory

[UNIX version] /opt

<Systemwalker trace information>

Files required for operations	Required disk space (bytes)
Trace file [Windows version]	Server: 190.0 MB or more (Up to 380.0 MB) Client: 1.0 MB or more (Up to 2.0 MB)
Trace file [UNIX version]	Server: 5.0 MB or more (Up to 10.0 MB)

Remarks:

Each file is stored in the following location.

[Windows version] System drive

[UNIX version] /var

<Audit log output>

Files required for the operations	Required disk space (bytes)
Audit log files	<*number of operations per day (Note 1)*> x <*average output size per record (Note 2)*> x <*number of storage days (Note 3)*>

Note 1)

This is the total number of login operations, definition change operations and job/queue operations per day. This includes operations resulting from commands or APIs, as well as operations performed from clients. A large number of operations will be performed on maintenance days and at initial installation time, so use the highest likely value for the estimate. (Assume around 1,000 operations per day, depending on the system being installed.)

Note 2)

This is the average length of one line of messages output to audit logs. (This value depends on the user names and host names that are being used, but assume around 500 bytes.)

Note 3)

This is the number of days that audit logs will be held. The default value is 31 days. Because audit log files are stored for only the specified number of days, it is recommended that audit log files be backed up periodically if necessary. Refer to the *Systemwalker Operation Manager Reference Guide* for more information about the mpsetlogsend_omgr command.

EE

<Master Schedule Management>

Management server

Files required for operations	Required disk space (bytes)
Basic capacity	(10000 x <*number of distribution job nets*> + 2200 x <*number of jobs*>) x 2 (Note 4)
Master schedule management status file (Note 1)	(100 + <*number of schedule servers*> x 200) x <*number of schedule records (Note 5)*> + 40 x <*number of schedule records (Note 5)*>
Carried over control information file (Note 2)	(420 x <*number of carried over job nets*> x 2) + (115 x <*number of job nets not applied that day*>)
Trace file (Note 3)	(30 MB x <*number of subsystems*>) + 30 MB

Schedule server [UNIX version]

Files required for operations	Required disk space (bytes)
Basic capacity	(10000 x <*number of distribution job nets*> + 2200 x <*number of jobs*>) x 2 (Note 4)
Master schedule management status file (Note 1)	40 x <*number of schedule records (Note 5)*>
Carried over control information file (Note 2)	(420 x <*number of carried over job nets*> x 2) + (115 x <*number of job nets unapplied to the subject day*>)
Trace file (Note 3)	(30 MB x <*number of subsystems*>) + 30 MB

Remarks:

When the management server and schedule server are run on the same machine, the required disk space will be the same as if only the management server was used.

Note 1)

The master schedule management status file is used to store the distribution status of the schedule information. The following files are master schedule management status files:
- "stemmanager.db" and "start_end.log" in the master schedule management database directory.

Note 2)

The carried over control information file stores the carried-over job net information and so on.
These files are as follows:
- "carry_st.lst", "carry.lst" and "process date(yyyymmdd).dat" files, located in the database directory for master schedule management.

Note 3)

The trace file is used to cyclically control trace information within the determined disk capacity.

Note 4)

The following value must be added for group operations:
(1000 x <*total number of groups*> + 27400 x <*total number of job nets belonging to groups*>) x 2

Note 5)

The number of schedule records is the number of days calculated with the following formula:
<*number of days for schedule status management*> + <*number of days in future schedules*>
where <*number of days in future schedules*> means "the number of days in advance that schedules are registered".

Hardware requirements for server power control

A power control device and supporting software are required in order to control the power to servers. This function is only available when the operating system is Windows x86.

The power control device and software are not required for operations that only involve shutting down and rebooting.

Note

- For Solaris servers, power control is not available for a Non-global Zone. If Systemwalker Operation Manager is installed on a Global Zone, power control is available for the Global Zone.
- The power control function is not supported if multiple operating systems are running with a VM function or on a blade server.

Hardware and software requirements for server power control

Server OS	Provider	Power control device/UPS	Software
Windows x86	APC (Fujitsu OEM)	SmartUPS, and one of the following network adapters for SmartUPS - Web/SNMP Management Card (AP9606) - Network Management Card EX (AP9617)	PowerChute(R) *plus* V5.2 (Note) PowerChute(R) Business Edition Version 9.0.1 (Note)

Note)

Web/SNMP Management Card (AP9606) or Network Management Card EX (AP9617) is required to start the server from the client, or to use the Batch Power Control function. The PowerChute(R) Plus V5.2 software can be integrated at this time, however the PowerChute(R) Business Edition cannot be used.

If the server is not started by a client, the Web/SNMP Management Card (AP9606) and the Network Management Card EX (AP9617) are not required.

The software to be installed may vary depending on the software used and the client/server type, as shown in the following table.

Software used	Server/client type	Software to be installed
PowerChute(R) *plus* V5.2	Server	UPSSLEEP.EXE (Note) (Provided with PowerChute(R) *plus*)
PowerChute(R) Business Edition Version 9.0.1	Server	UPSSLEEP.EXE (Note) (Provided with PowerChute(R) Business Edition Version)

Remark:

Clients do not need to be set up if power control is only performed according to schedules for each server.

Note)

Set up the path to the directory where UPSSLEEP.EXE is installed in the PATH system environment variable.

Some of the functions of the power control software for "Systemwalker Enabled" authorized products may not be available depending on the software. Also, some functions may need special care when used. For more information, refer to the Systemwalker website and the manuals for the software.

Hardware requirements for Event Monitoring [Windows version]

The following hardware is required in order to send short mail notifications:

- An NTT DoCoMo mobile phone
  A phone that supports Short Mail is required. Refer to the NTT DoCoMo home page for information about Short Mail-compatible phones.

The following hardware device is required for voice notification.

- WAVE audio card (Note)

Note)

It may not be possible to install an audio card on some models.

Hardware required for Task Link

The following hardware devices are required to use various Task Link functions.

- Client power-on
  - A device that supports Wakeup on LAN
  - A LAN card that supports Wakeup on LAN

  Turning the power on via Wakeup on LAN must be enabled at the BIOS level.
- Client power-off
  - A device that supports either APM (Advanced Power Management) or ACPI (Advanced Configuration & Power Interface)

  Turning the power off from Windows must be enabled.

3.2 Software Resources

This section explains the software resources required to install Systemwalker Operation Manager.

3.2.1 Operating Systems

The software resources that are required to install Systemwalker Operation Manager are explained in the table below.

Windows version

Installation type	Operating systems	Remarks (Service Pack information/Patch number)
Server	Microsoft(R) Windows Server(R) 2008 Standard(x86)	SP None/2
	Microsoft(R) Windows Server(R) 2008 Enterprise(x86)	SP None/2
	Microsoft(R) Windows Server(R) 2008 Datacenter(x86)	SP None/2 (Note 1)

Installation type	Operating systems	Remarks (Service Pack information/Patch number)
	Microsoft(R) Windows Server(R) 2008 Standard without Hyper-V(TM) (x86)	SP None/2
	Microsoft(R) Windows Server(R) 2008 Enterprise without Hyper-V(TM) (x86)	SP None/2
	Microsoft(R) Windows Server(R) 2008 Datacenter without Hyper-V(TM) (x86)	SP None/2 (Note 1)
	Microsoft(R) Windows Server(R) 2008 Foundation(x64)	SP None/2
	Microsoft(R) Windows Server(R) 2008 Standard(x64)	SP None/2
	Microsoft(R) Windows Server(R) 2008 Enterprise(x64)	SP None/2
	Microsoft(R) Windows Server(R) 2008 Datacenter(x64)	SP None/2 (Note 1)
	Microsoft(R) Windows Server(R) 2008 Standard without Hyper-V(TM)(x64)	SP None/2
	Microsoft(R) Windows Server(R) 2008 Enterprise without Hyper-V(TM)(x64)	SP None/2
	Microsoft(R) Windows Server(R) 2008 Datacenter without Hyper-V(TM)(x64)	SP None/2 (Note 1)
	Microsoft(R) Windows Server(R) 2008 R2 Foundation(x64)	SP None/1
	Microsoft(R) Windows Server(R) 2008 R2 Standard(x64)	SP None/1
	Microsoft(R) Windows Server(R) 2008 R2 Enterprise(x64)	SP None/1
	Microsoft(R) Windows Server(R) 2008 R2 Datacenter(x64)	SP None/1 (Note 1)
	Microsoft(R) Windows Server(R) 2012 Foundation(x64)	SP None
	Microsoft(R) Windows Server(R) 2012 Standard(x64)	SP None
	Microsoft(R) Windows Server(R) 2012 Datacenter(x64)	SP None
	Microsoft(R) Windows Server(R) 2012 R2 Foundation(x64)	SP None
	Microsoft(R) Windows Server(R) 2012 R2 Standard(x64)	SP None
	Microsoft(R) Windows Server(R) 2012 R2 Datacenter(x64)	SP None
Client	Windows Vista(R) Home Basic (x86)	SP None/1/2
	Windows Vista(R) Home Premium (x86)	SP None/1/2

Installation type	Operating systems	Remarks (Service Pack information/Patch number)
	Windows Vista(R) Business (x86)	SP None/1/2
	Windows Vista(R) Enterprise (x86)	SP None/1/2
	Windows Vista(R) Ultimate (x86)	SP None/1/2
	Windows Vista(R) Home Basic (x64)	SP None/1/2
	Windows Vista(R) Home Premium (x64)	SP None/1/2
	Windows Vista(R) Business (x64)	SP None/1/2
	Windows Vista(R) Enterprise (x64)	SP None/1/2
	Windows Vista(R) Ultimate (x64)	SP None/1/2
	Windows(R) 7 Home Premium (x86)	SP None/1
	Windows(R) 7 Professional (x86)	SP None/1
	Windows(R) 7 Enterprise (x86)	SP None/1
	Windows(R) 7 Ultimate (x86)	SP None/1
	Windows(R) 7 Home Premium (x64)	SP None/1
	Windows(R) 7 Professional (x64)	SP None/1
	Windows(R) 7 Enterprise (x64)	SP None/1
	Windows(R) 7 Ultimate (x64)	SP None/1
	Windows(R) 8(x86)	SP None
	Windows(R) 8 Pro(x86)	SP None
	Windows(R) 8 Enterprise(x86)	SP None
	Windows(R) 8(x64)	SP None
	Windows(R) 8 Pro(x64)	SP None
	Windows(R) 8 Enterprise(x64)	SP None
	Windows(R) 8.1(x86)	SP None
	Windows(R) 8.1 Pro(x86)	SP None
	Windows(R) 8.1 Enterprise(x86)	SP None

Installation type	Operating systems	Remarks (Service Pack information/Patch number)
	Windows(R) 8.1(x64)	SP None
	Windows(R) 8.1 Pro(x64)	SP None
	Windows(R) 8.1 Enterprise(x64)	SP None

SP: Service Pack

Note 1:

Systemwalker Operation Manager Enterprise Edition only.

For operating systems where the server function is running, the Systemwalker Operation Manager client can be used only when the server function of Systemwalker Operation Manager is also installed.

Solaris version

Installation type	Operating systems	Remarks (Service Pack information/Patch number)
Server	Oracle Solaris 11	
	Oracle Solaris 10	
	Oracle Solaris 9	(Note 1)
Client	The same as for Windows version clients	

Note 1)

Solaris 32-bit version only.

Patches are required for each operating system. For information about the patches required, refer to the Release Note.

Linux

Installation type	Operating systems	Remarks (Service Pack information/patch number)
Server	Red Hat Enterprise Linux 5.0 (for x86)	
	Red Hat Enterprise Linux 5.1 (for x86)	
	Red Hat Enterprise Linux 5.2 (for x86)	
	Red Hat Enterprise Linux 5.3 (for x86)	
	Red Hat Enterprise Linux 5.4 (for x86)	
	Red Hat Enterprise Linux 5.5 (for x86)	
	Red Hat Enterprise Linux 5.6 (for x86)	
	Red Hat Enterprise Linux 5.7 (for x86)	
	Red Hat Enterprise Linux 5.8 (for x86)	
	Red Hat Enterprise Linux 5.9(for x86)	
	Red Hat Enterprise Linux 5.10(for x86)	

Installation type	Operating systems	Remarks (Service Pack information/patch number)
	Red Hat Enterprise Linux 6.0 (for x86)	
	Red Hat Enterprise Linux 6.1 (for x86)	
	Red Hat Enterprise Linux 6.2 (for x86)	
	Red Hat Enterprise Linux 6.3 (for x86)	
	Red Hat Enterprise Linux 6.4(for x86)	
	Red Hat Enterprise Linux 6.5(for x86)	
	Red Hat Enterprise Linux 5.0 (for Intel64)	
	Red Hat Enterprise Linux 5.1 (for Intel64)	
	Red Hat Enterprise Linux 5.2 (for Intel64)	
	Red Hat Enterprise Linux 5.3 (for Intel64)	
	Red Hat Enterprise Linux 5.4 (for Intel64)	
	Red Hat Enterprise Linux 5.5 (for Intel64)	
	Red Hat Enterprise Linux 5.6 (for Intel64)	
	Red Hat Enterprise Linux 5.7 (for Intel64)	
	Red Hat Enterprise Linux 5.8 (for Intel64)	
	Red Hat Enterprise Linux 5.9 (for Intel64)	
	Red Hat Enterprise Linux 5.10 (for Intel64)	
	Red Hat Enterprise Linux 6.0 (for Intel64)	
	Red Hat Enterprise Linux 6.1 (for Intel64)	
	Red Hat Enterprise Linux 6.2 (for Intel64)	
	Red Hat Enterprise Linux 6.3 (for Intel64)	
	Red Hat Enterprise Linux 6.4 (for Intel64)	
	Red Hat Enterprise Linux 6.5 (for Intel64)	
	Red Hat Enterprise Linux 7.0 (for Intel64)	(Note 1)
Client	Same as Windows client.	

Note 1)

There are points to note regarding shutdown. Refer to the following description in *Systemwalker Operation Manager Release Information* for details.

- "Migrating from V13.3.0/V13.3.1 to V13.8.0" - "Incompatibilities relating to stopping daemons at shutdown in Red Hat Enterprise Linux 7 [Linux]"

Information

Environments where the SELinux (Security-Enhanced Linux) function is enabled

Systemwalker Operation Manager supports the SELinux function in Red Hat Enterprise Linux 6.2 (for x86/for Intel64) or later.

3.2.2 Related Software

This section provides an overview of the software related to Systemwalker Operation Manager.

Redistributable package [Windows version]

The redistributable package "Microsoft Visual C++ 2005 Redistributable" is required software for Systemwalker Operation Manager.

If the "Microsoft Visual C++ 2005 Redistributable" package does not exist, it will be installed automatically when Systemwalker Operation Manager is installed.

Do not uninstall the "Microsoft Visual C++ 2005 Redistributable" package if Systemwalker Operation Manager is already installed.

Web Console encrypted communication

The certificate issued by the Certificate Authority is required for encrypted communications (SSL: Secure Socket Layer).

Certificates or CRLs are supported when they are issued by one of the following:

- VeriSign, Inc.

  Supports Secure Site and Secure Site with EV Certificates (EV SSL Certificates).

- Cybertrust, Inc.

  Supports SureServer for SSL Certificates.

Systemwalker User Management

To use the Systemwalker User Management function, one of the following programs is required to set up the Systemwalker authentication repository. Refer to the *Systemwalker User's Guide - Systemwalker User Management and Single Sign-on* for more information.

- Interstage Directory Service

  Use the Interstage Directory Service included with either of the following:

  - Interstage Application Server Enterprise Edition V9.2 or later included with Systemwalker Centric Manager V13.4.0
  - Interstage Application Server Standard-J Edition/Enterprise Edition V8.0.0 or later

  Point

  In IPv6 communication environments

  Use the Interstage Directory Service included with either of the following products when performing operations in an IPv6 communication environment:

  - Systemwalker Centric Manager V13.4.0 or later
  - Interstage Application Server Standard-J Edition/Enterprise Edition V9.1.0 or later

- Active Directory

  If Active Directory is used as the Systemwalker authentication repository, the following operating systems are supported:

  - Windows Server 2008 (including Server Core)
  - Windows Server 2012 (including Server Core)

  Note

  Multi-domain environments are not supported if Active Directory is used as the Systemwalker authentication repository.

Point

When Active Directory is used to set up the Systemwalker authentication repository, Interstage Directory Service is still required in order to use single sign-on from the **Systemwalker Web Console**, which is done by linking with Systemwalker Centric Manager.

To use the Interstage Directory Service as the Systemwalker authentication repository, an electronic certificate will be required in order to set up the SSL communication environment (encrypted communication and authentication of the Systemwalker authentication repository server). Electronic certificates are supported when they are issued by one of the following:

- VeriSign, Inc.
- Cybertrust, Inc.

As an alternative, a simple Systemwalker certificate can be used, which does not cause any problems for encrypted communication, however since it is not issued by a Certificate Authority it cannot be used for server authentication.

Systemwalker Single Sign-On

One of the following products is required in order to use the Systemwalker Single Sign-On function:

- Systemwalker Centric Manager V13.4.0 or later
- Interstage Application Server Standard-J Edition/Enterprise Edition V9.0.0 or later

Point

Use the following products when Systemwalker Operation Manager is installed together with Systemwalker Single Sign-On Server or Interstage Application Server:

- Systemwalker Single Sign-On Server for Systemwalker Centric Manager V13.6.1 or later
- Interstage Application Server Standard-J Edition/Enterprise Edition V11.0.0 or later

Point

One of the following products is required when performing operations in an IPv6 communication environment and when Active Directory is used as the Systemwalker authentication repository for storing user information:

- Systemwalker Centric Manager V13.4.0 or later
- Interstage Application Server Standard-J Edition/Enterprise Edition V9.1.0 or later

Any of the following Web browsers can be used with the Systemwalker Single Sign-On function:

- Internet Explorer 8/9/10/11

Point

If using Internet Explorer 10/11, use one of the following products as the Systemwalker Single Sign-On Server:

- Systemwalker Single Sign-On Server for Systemwalker Centric Manager V13.6.1 or later
- Interstage Application Server Standard-J Edition/Enterprise Edition V11.0.0 or later

Note

If a Systemwalker product and the Interstage Application Servers shown below co-exist, then the Systemwalker User Management function and Systemwalker Single Sign-On cannot be used. An error will occur when the environment is set.

- Earlier than Interstage Application Server Enterprise Edition V6.0
- Earlier than Interstage Application Server Standard Edition V6.0
- Earlier than Interstage Application Server Web-J Edition V7.0
- Earlier than Interstage Application Server Plus V7.0

To use the Interstage Directory Service as the Systemwalker authentication repository, an electronic certificate will be required to set up the SSL communication environment (encrypted communication and authentication of the Systemwalker authentication repository server). Electronic certificates are supported when they are issued by one of the following:

- VeriSign, Inc.
- Cybertrust, Inc.

As an alternative, a simple Systemwalker certificate can be used, which does not cause any problems for encrypted communication, however since it is not issued by a Certificate Authority it cannot be used for server authentication.

Single Sign-On linked with ServerView Operations Manager

The following product is required in order to use the Single Sign-On function linked with ServerView Operations Manager:

- ServerView Resource Orchestrator Cloud Edition V3.1.1 or later

Also, ServerView Operations Manager must be built and Single Sign-On linkage must be configured for the above product by using ServerView Operations Manager.

If using the Single Sign-On function with ServerView Operations Manager, the ServerView Operations Manager built during the installation of ServerView Resource Orchestrator Cloud Edition must be used as the authentication infrastructure.

Only those ServerView Operations Manager versions and directory services can be used, which are compliant with the ServerView Resource Orchestrator Cloud Edition. However, only a single domain environment is supported when Active Directory is used as the directory service.

Additionally, only those Web browsers can be used with the Single Sign-On function linked with ServerView Operations Manager, which are compliant with the ServerView Resource Orchestrator Cloud Edition management client. However, Web browsers that are not supported by Systemwalker Operation Manager cannot be used.

Backup software [Windows version]

The following table lists the communication software required for backup linking.

Installation type	Backup software
Server	CA ARCserve Backup r12.5 for Windows
	CA ARCserve Backup r15 for Windows
	CA ARCserve Backup r16 for Windows
	CA ARCserve Backup r16.5 for Windows
Client	---

EE

Cluster systems

The following cluster systems are supported by Systemwalker Operation Manager.

OS type	Cluster system
Windows Server 2012	Microsoft(R) Fail Over Clustering included in:

OS type	Cluster system
	- Microsoft(R) Windows Server(R) 2012 Standard(x64) - Microsoft(R) Windows Server(R) 2012 Datacenter(x64 - Microsoft(R) Windows Server(R) 2012 R2 Standard(x64) - Microsoft(R) Windows Server(R) 2012 R2 Datacenter(x64)
Windows Server 2008	Microsoft(R) Fail Over Clustering included in: - Microsoft(R) Windows Server(R) 2008 Enterprise (x86) - Microsoft(R) Windows Server(R) 2008 Datacenter (x86) - Microsoft(R) Windows Server(R) 2008 Enterprise (x64) - Microsoft(R) Windows Server(R) 2008 Datacenter (x64) - Microsoft(R) Windows Server(R) 2008 R2 Enterprise (x64) Microsoft(R) Windows Server(R) 2008 R2 Datacenter (x64)
Solaris 11	PRIMECLUSTER Enterprise Edition 4.3A10/4.3A20 PRIMECLUSTER HA Server 4.3A10/4.3A20 PRIMECLUSTER Clustering Base 4.3A10/4.3A20 Oracle Solaris Cluster 4.1/4.2
Solaris 10	PRIMECLUSTER Enterprise Edition 4.1A40/4.2A00/4.3A10/4.3A20 PRIMECLUSTER HA Server 4.1A40/4.2A00/4.3A10/4.3A20 PRIMECLUSTER Clustering Base 4.1A40/4.2A00/4.3A10/4.3A20 Oracle Solaris Cluster 3.3
Solaris 9	PRIMECLUSTER Enterprise Edition 4.1A40/4.2A00 PRIMECLUSTER HA Server 4.1A40/4.2A00 PRIMECLUSTER Clustering Base 4.1A40/4.2A00 Sun Cluster 3.2
Linux(x86)	PRIMECLUSTER Enterprise Edition 4.2A30/4.3A00/4.3A10/4.3A20/4.3A30/4.3A40 PRIMECLUSTER HA Server 4.2A30/4.3A00/4.3A10/4.3A20/4.3A30/4.3A40 PRIMECLUSTER Clustering Base 4.2A30/4.3A00/4.3A10/4.3A20/4.3A40
Linux(Intel64)	PRIMECLUSTER Enterprise Edition 4.2A30/4.3A00/4.3A10/4.3A20/4.3A30 PRIMECLUSTER HA Server 4.2A30/4.3A00/4.3A10/4.3A20/4.3A30/4.3A40 PRIMECLUSTER Clustering Base 4.2A30/4.3A00/4.3A10/4.3A20/4.3A30/4.3A40

VM operations

Supported software

Systemwalker Operation Manager supports the following Virtual Machine (VM) software:

- Linux virtual machine function
- VMware(R) Infrastructure 3
- VMware vSphere(R) 4

- VMware vSphere(R) 5.0
- Windows Server 2008 Hyper-V
- Microsoft Hyper-V Server 2008 R2
- Windows Server 2012 Hyper-V
- Microsoft Hyper-V Server 2012
- Solaris Containers (Solaris Zone)
- Oracle VM Server for SPARC

Software requirements for power control

Refer to "Hardware requirements for server power control".

Software required for completion monitoring for power control

To perform completion monitoring for the power control function (sending popup messages or emails), Systemwalker Centric Manager V5.0 or later is required.

Using e-mail

Email software is required to use e-mail.

Using audio notification

To use audio notification, one of the following products with audio synthesis engines (compatible with Microsoft Speed API (SAPI) 5.x) must be installed on a client running a 32-bit Windows(R) operating system. (Note)

- The standard audio synthesis engine provided with Microsoft(R) Office XP or Microsoft(R) Office 2003

  Note)

  Audio notification cannot run on 64-bit Windows(R) operating systems.

To use audio notification, the voice synthesis engine that is installed must use the same language as Systemwalker.

The gender of the audio notification voice should also match that of Systemwalker.

Example:

When the Systemwalker language is an English female voice

Audio synthesis engine: English - Female

Using Systemwalker from a Web browser

Systemwalker Operation Manager monitoring and operation functions can be used from the following Web browsers:

Web browser

OS type	Supported Web browser
Windows Server 2008 Windows(R) 7 Windows Vista(R)	Internet Explorer 8/9
Windows Server 2012 R2 Windows Server 2012 Windows Server 2008 R2 (Service Pack 1) Windows(R) 8.1 Windows(R) 8 Windows(R) 7 (Service Pack 1)	Internet Explorer 10/11

Linking to Interstage

Systemwalker Operation Manager can link to Interstage Application Server.

If Interstage Application Server to be linked is the 32-bit version, also use the 32-bit version of Systemwalker Operation Manager. Likewise, if Interstage Application Server to be linked is the 64-bit version, also use the 64-bit version of Systemwalker Operation Manager.

The following table lists the Interstage versions and WorkUnits that can be linked.

WorkUnit	Interstage Application Server Versions				
	V2 [Windows] [UNIX]	V3 [Windows] [UNIX]	V4 [Windows] [UNIX]	V5 - V 8 [Windows] [UNIX]	V9 - V11 [Windows] [UNIX]
TD (Note 1)	Yes	Yes	Yes	Yes	Yes
EJB (Note 1)	No	Yes	Yes	Yes	No
UTY (Note 2)	No	No	Yes	Yes	Yes
CORBA (Note 3)	No	No	No	Yes	Yes

```
TD: Transaction Director
EJB: Enterprise JavaBeans
UTY: Utility Work Unit
CORBA: CORBA Work Unit

Yes: Can be linked.
No: Cannot be linked.
```

Note 1)

Windows x86, Solaris 32bit, and Linux x86 versions can be linked, and Windows x64, Solaris 64bit and Linux x64 versions cannot be linked.

Note 2)

Can be linked for Solaris and Linux versions.

Note 3)

Windows, Solaris, and Linux versions can be linked.

Linking to PowerAIM [UNIX version]

Systemwalker Operation Manager can link to PowerAIM V10L31 or later.

Chapter 4 Security

This chapter explains the security functions of Systemwalker Operation Manager.

4.1 Security Guidelines

This section explains the security guidelines for Systemwalker Operation Manager.

Positions and roles of Systemwalker Operation Manager users

To use Systemwalker Operation Manager securely, there must be a clear distinction between the role of administrators and the roles of other users.

A system administrator must be a trustworthy person who would not perform any inappropriate operations.

The following table shows the positions and roles of the different Systemwalker Operation Manager users.

OS user	Systemwalker Operation Manager user	Description	Role
System administrator	System administrator	A person who has administrator privileges for the server devices that make up the business system (either a member of the Administrators group (for Windows) or a superuser (for UNIX))	Registers and deletes operations administrators and operations staff, and also performs the role of operations administrators. When necessary, the system administrator can perform the duties of both operations administrator and operations staff.
	Operations administrator	A person who manages operations staff and overall business operations	Sets the range of responsibilities of operations staff in a business system. When necessary, an operations administrator can also perform the duties of operations staff.
General user	Operations staff	A person responsible for operation duties	Performs operation duties in a business system within the range permitted by an operations administrator.

Note

This chapter uses the user classification that is shown in the "Systemwalker Operation Manager user" column in this table.

However, other Systemwalker Operation Manager manuals basically use the user classification that is shown in the "OS user" column in this table. When the Extended User Management function is used, an operations administrator refers to an Operation Manager user (i.e., a user registered with and managed under Systemwalker Operation Manager by the Extended User Management function) with administrator privileges, and operations staff refers to an Operation Manager user who does not have administrator privileges. Also, when the Systemwalker authentication repository is used, operations

administrators are called system administrators (in Windows, this is the built-in Administrator account, and in UNIX it is the superuser), and operations staff are called Systemwalker users.

4.1.1 What is Security?

The growth of computer networks, most notably the Internet, has made it possible to access corporate networks from not only within the corporation itself, but from a variety of other locations as well. While this has made life more convenient, it has also meant that security problems such as unauthorized access and information leaks have become much more serious.

Security means "safety" and "protection from dangers and risks that can threaten that safety". The following table lists some of the major security threats.

Classification		Concrete example
Accidental or incidental threats	Natural disasters	Earthquakes, lightning, fire, etc.
	Faults and breakdowns	Hardware and software faults, etc.
	Human error	Operational errors, incorrect connections, etc.
Intentional threats	Malicious action by third parties	Unauthorized access, impersonation, etc.
	Malicious action by related parties	Password leaks, disclosure of customer information, etc.

4.1.2 Security Requirements

The following three requirements must be met in order to guarantee security:

- Confidentiality

  It must not be possible to view any information that is not allowed to be viewed.

- Integrity

  Information must always be maintained in a complete form and not be altered without authorization, either intentionally or unintentionally.

- Availability

  The system must operate normally and be available at any time.

Threats can arise when these requirements are not met.

Failure to meet requirement	Specific example
Loss of confidentiality	Password disclosure
Loss of integrity	Planned execution of jobs is obstructed if definition data that is used for management is altered by someone without authorization.
Loss of availability	Executed jobs terminate abnormally at unscheduled times.

4.1.3 Security Measures

Threats arise if security requirements are not met. Therefore, to maintain security, it is necessary to implement measures that ensure security requirements are satisfied at all times.

[Measures to ensure confidentiality and integrity]

User authentication

One way to prevent unauthorized access to or illegal operations on a server is user authentication.

Operating system authentication controls the users who can access Systemwalker Operation Manager. System administrators, operations administrators and operations staff are identified and authenticated, and if identification or authentication fails, access to Systemwalker Operation Manager is denied.

If the Extended User Management function [UNIX version] is used, it is also possible to register Operation Manager users who can access Systemwalker Operation Manager independently of the users registered with the operating system. This enables user management to be conducted totally within Systemwalker Operation Manager itself, which can improve the security of operations.

Also, when the Systemwalker authentication repository is used, Systemwalker product's user management can be centralized, resulting in more secure operations for the entire system, as well as a reduction of the administrative duties of the system administrator. Also, if single sign-on can use the Systemwalker authentication repository, then operations staff can use multiple Systemwalker products securely with a single login.

Access control

To prevent unauthorized access to systems and important assets, it is necessary to control which users can access which assets.

Two effective ways to prevent unauthorized access are to keep the number of users who can access a system to a minimum, and to ensure that system administrators and operations administrators give careful consideration before deciding on the range of information that operations staff can access.

Controlling access to projects

This security measure sets the users that are permitted to access projects and the rights of these users.

To prevent a user performing operations that are outside his or her duties, it is necessary to correctly assign the authority appropriate to user's role.

Controlling access to Systemwalker Operation Manager directories and files

This security measure restricts which users are permitted to submit demand jobs, start job nets with Job Execution Control attributes, and use Jobscheduler commands.

Execution user restrictions

One effective way to prevent unauthorized operations is to only permit jobs to be executed by their intended users. This will ensure that no unintended operations are conducted.

Define precisely which users have permission to run jobs beforehand.

Audit log

Using audit logs to regularly check for signs of unauthorized access is an effective way to detect or prevent suspicious user behavior and unauthorized access. Damage can be kept to a minimum by sensing suspicious operations and using audit logs to track and deal with the user behavior that caused those operations. To maintain the security of operations, the system administrator must monitor the audit logs that are output. Fujitsu recommends enabling output of audit logs.

In addition to the security measures above, which can be implemented using Systemwalker Operation Manager functions, it is also important to set up and implement information security policies for the organization so that appropriate education is provided to Operation Manager users. Security measures must be comprehensive, and should include operation rules as well as function-related measures.

Some of the security measures that are not related to Systemwalker Operation Manager functions are described below.

Physical protection and protection of the network environment

Physically protecting the hardware devices and recording media associated with Systemwalker Operation Manager is an extremely effective security measure. Concrete examples include the following:

- Installing hardware devices in locked rooms that are only accessible to operations administrators
- Recording all movement in and out of rooms and checking for suspicious persons (The same measures apply to people entering and leaving buildings.)
- Installing hardware devices in buildings with earthquake-resistant facilities and protecting hardware from fire damage

It is also important to protect the data traffic on networks. An effective security measure when using Systemwalker Operation Manager from a Web browser is to set up the security functions for the Web server, and to protect communications using SSL.

Protection via operational measures

Operational measures must also be implemented to protect the hardware devices, recording media, and other assets that are used with Systemwalker Operation Manager. A concrete example is as follows:

- Do not leave the terminal while logged into Systemwalker Operation Manager.

ID and password leaks may result in unauthorized accesses, such as intrusion, impersonation, mail viewing and data alteration. To improve security, it is important to maintain the confidentiality of IDs and passwords at all times.

Granting IDs and passwords

IDs and passwords must be allocated properly.

- Provide user IDs that have the minimum authority level required for each task.
- Promptly delete user IDs that are no longer required.

Setting passwords

Avoid using passwords that can be easily guessed, such as words that can be found in dictionaries, names, birthdays and telephone numbers. A strong password should be as long as possible (at least 5 characters) and include upper-case characters, lower-case characters, numerals and symbols to make it difficult to guess.

Managing passwords

Passwords must be correctly managed to prevent them from becoming known to third parties. In concrete terms, this means implementing the following measures:

- Do not write down passwords on paper or other material.
- Change passwords regularly.
- Do not allow passwords to be seen by third parties when they are being input.

The following rules must also be observed:

- System administrator passwords must not be made known to anyone but the system administrator.
- Operations administrator passwords must not be made known to anyone but system administrators and operations administrators.
- Operations staff or general user passwords must not be made known to other operations staff or general users.

User education

Educating system users in security matters is vital to ensure that users are aware of security measures and to maintain and improve the security level of the entire organization. User education is a fundamental way of preventing a variety of threats, such as information losses and information leaks. In order to raise security-related awareness and improve the ability to handle risks, user education must be provided on an ongoing basis.

In concrete terms, this means the following:

- A responsible party in the organization must select a reliable person to act as a system administrator and instruct that person to manage his or her passwords to prevent them from being known to third parties. The system administrator

must select reliable persons to act as operations administrators, and then instruct operations administrators, operations staff, and general users to manage their passwords to prevent them from becoming known to others. Adherence to these rules must be enforced.

- If more than one person uses Systemwalker Operation Manager on the same machine, an account must be created for each user and each user should only use his or her own user ID.

[Measures to satisfy availability requirements]

Managing completed jobs

One measure to ensure that jobs run as planned is to re-execute jobs that terminate abnormally. If a scheduled job fails to run as planned and terminates abnormally halfway through the operation for some reason, it should be restarted manually or automatically.

Operation design

When designing an operation, the resources needed for the operation should be carefully estimated so that the operation proceeds smoothly. (Refer to "Tuning of Performance" in the *Systemwalker Operation Manager User's Guide* for more information.)

When a large number of job nets and jobs are to be registered and operated, performance should be carefully and thoroughly tested in advance.

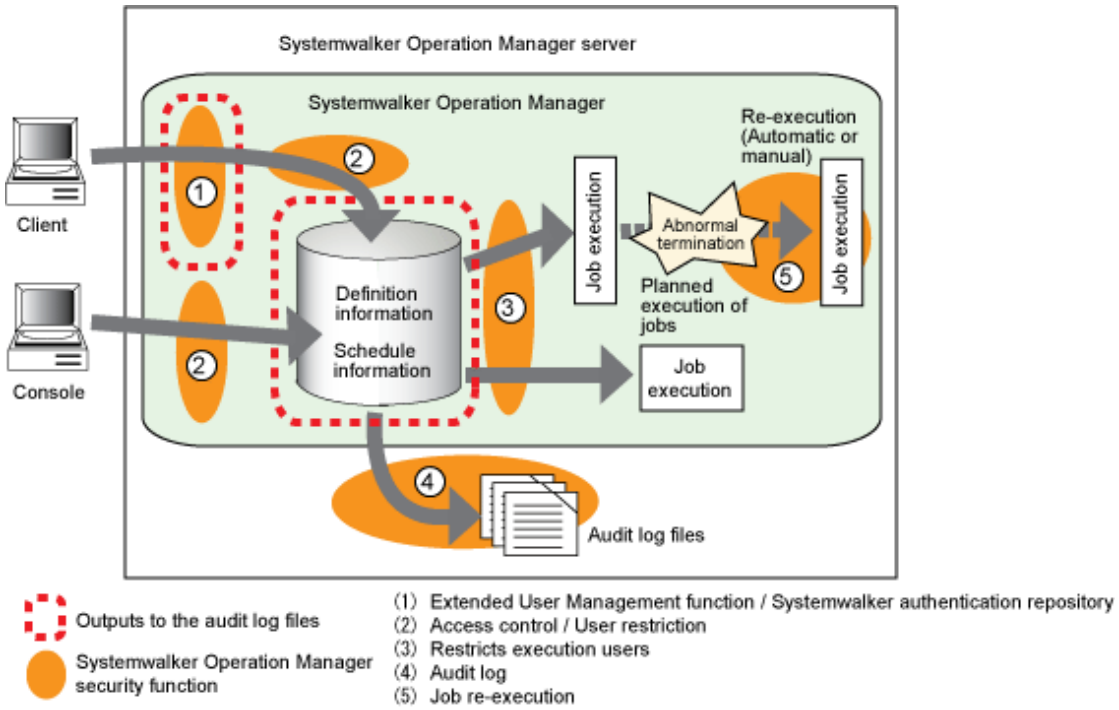
Backups

Backups are an important way of enabling information assets to be restored. Copies of files can be created, and spare servers and disk devices can be used to protect valuable information assets from becoming damaged or lost.

4.2 Systemwalker Operation Manager Security Functions

The security functions provided by Systemwalker Operation Manager are mainly intended to prevent obstruction to the planned execution of jobs (loss of integrity) referred to in "4.1.2 Security Requirements".

The following diagram shows the security functions provided by Systemwalker Operation Manager.



The functions provided by Systemwalker Operation Manager are described in more detail in the following sections.

4.2.1 Extended User Management Function [UNIX Version]

This function can manage the users (Operation Manager users) who can access Systemwalker Operation Manager. These users are separate from the users registered with the operating system. This function enables user authentication management to be performed on Systemwalker Operation Manager without increasing the number of users registered with the operating system unnecessarily.

This function makes it possible to specify managers for running Systemwalker Operation Manager, as distinct from the system administrators registered with the operating system, and also enables the access rights of managers (operations administrators) and non-managers (operations staff) to be specified in fine detail.

The Extended User Management function permits the system administrator to perform the following settings:

- Enable and disable the Extended User Management function
- Register Operation Manager users (operations administrators/operations staff), and view, delete and change users after they have been registered

  User names and passwords are specified at registration time.
- Change the passwords for Operation Manager users that have been registered.

The Extended User Management function provides the following security-related benefits:

- Because the Extended User Management function enables operations administrators to be set independently of the system administrators registered with the operating system, it can prevent the risk of incorrect operations being performed by system administrators for the operating system, who are otherwise permitted to perform any operations on servers.
- The Extended User Management function prevents Systemwalker Operation Manager users from logging into servers directly, thereby preventing illegal operations from being performed on servers.
- When you set a different password for a Systemwalker Operation Manager user from the operating system user's password, and administrator privileges are given to the Systemwalker Operation Manager user, only administrator operations specific to Systemwalker Operation Manager will be possible.

If the Extended User Management function is disabled, users who can access Systemwalker Operation Manager will be controlled based on operating system authentication.

Refer to "Users (When using Extended User Management function) [UNIX version]" in the *Systemwalker Operation Manager Installation Guide* for more information.

Refer also to "Using the Users on Systemwalker Operation Manager [UNIX version]" in "2.2.2 Additional Operations" for an overview of this function.

4.2.2 Systemwalker Authentication Repository

By using the Systemwalker authentication repository, Systemwalker product's user management can be centralized, resulting in more secure operations for the entire system, as well as a reduction of the administrative duties of the system administrator. Also, if single sign-on can use the Systemwalker authentication repository, then operations staff can seamlessly and securely use multiple Systemwalker products with a single login.

When the Systemwalker authentication repository is used, the system administrator is able to configure the following:

- Enable or disable the Systemwalker authentication repository (the Systemwalker authentication repository settings at the connection destination).
- Register users (operations administrator or operations staff) on the Systemwalker authentication repository, and reference, delete, or modify users after they have been registered.

  Specify the user name and password at the same time as the user is registered.
- Change the user password after it has been registered in the Systemwalker authentication repository.

By centrally managing users with the Systemwalker authentication repository, the following effects will be seen from a security perspective:

- Because users do not need to be registered on the server in order to only use Systemwalker Operation Manager, illegal operations to the server can be prevented.
- More secure operations are possible for the entire system.

If the Extended User Management function is enabled, it can be used to control which users can access Systemwalker Operation Manager. If the Extended User Management function is disabled and the Systemwalker authentication repository is not used (the Systemwalker authentication repository settings at the connection destination have not been set), then the operating system's user authentication can be used to control access to Systemwalker Operation Manager.

Refer to "Defining Users (When Using a Systemwalker Authentication Repository)" in the *Systemwalker Operation Manager Installation Guide* for more information.

Refer to "2.10 User Management" for an overview.

4.2.3 Access Control

Access control sets user rights and manages access through user identification and authentication. It is important to organize and manage the access rights that are granted to users according to their roles in the system.

Appropriate access control can provide the following security-related benefits:

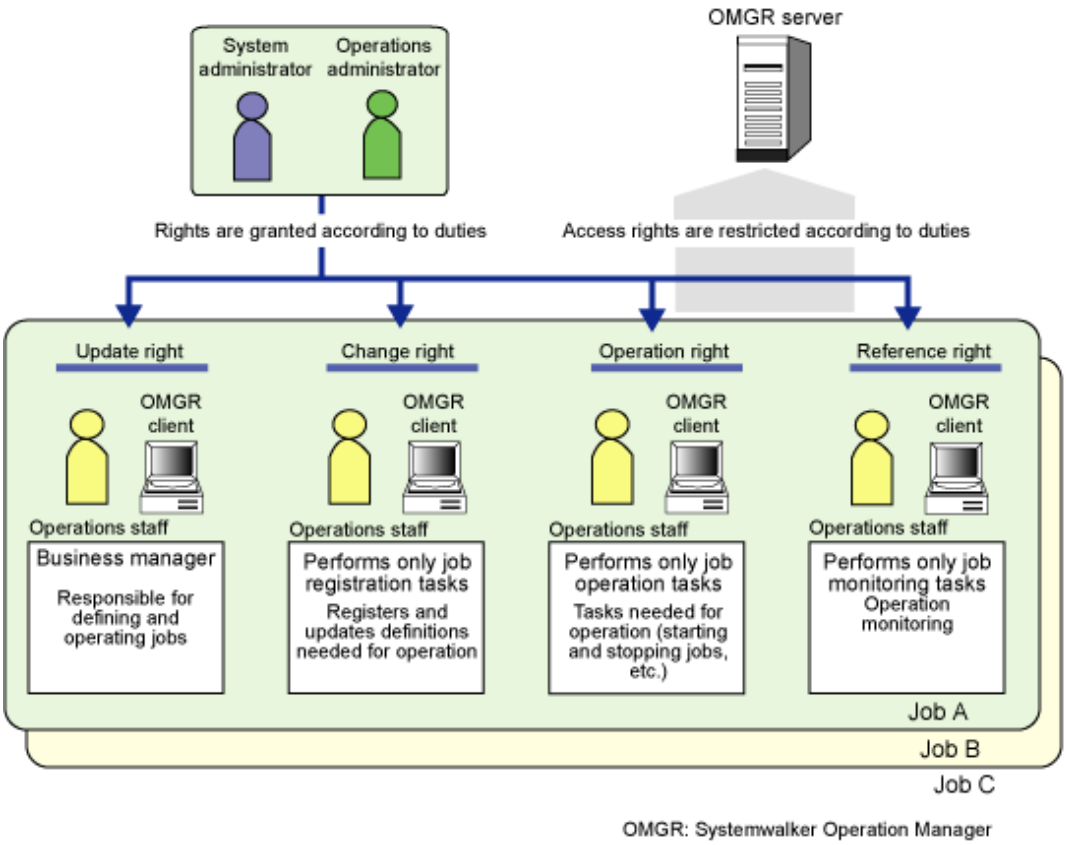
- Preventing access by users that do not have the proper authority, and preventing damaging actions, such as unauthorized deletion or alteration of data
- Preventing incorrect operations from being performed by users exceeding their authority

Controlling access to projects

System administrators and operations administrators can restrict the role (authority) of operations staff by setting up separate access rights for each individual project for operations staff.

Because operations staff are only granted the minimum authority necessary to do their jobs, this method eliminates problems caused by users performing incorrect operations outside their scope of responsibility or authority, and ensures that system operations are conducted more securely.

For example, in the situation shown in the following diagram, users are only granted authority proportional to their duties. Users in charge of job design are only granted change rights and users in charge of job operations are only granted operation rights. This prevents problems caused by users overstepping their authority whether through action or inaction. It prevents, for example, a user with change rights in charge of job design from accidentally performing a job operation.



The following table lists the different user types and roles that can be specified for projects.

User type	Role
System administrator Operations administrator	The operations administrator who can set up various operating environments for Systemwalker Operation Manager, register and delete projects, and set access rights to projects (This administrator is given update rights by default.)
Operations staff with update rights (General user)	The operations staff who can update, register, operate, and look up groups, job nets, and jobs within permitted projects
Operations staff with change rights (General user)	The operations staff who can only register and look up groups, job nets, and jobs within permitted projects
Operations staff with operation rights (General user)	The operations staff who can only operate and look up groups, job nets, and jobs within permitted projects
Operations staff with reference rights (General user)	The operations staff who can only look up groups, job nets, and jobs within permitted projects

Refer to "Setting Access Rights for Projects" in the *Systemwalker Operation Manager User's Guide* for more information about how to set up roles.

Refer also to "Usage Restrictions Based on Access Rights" in the *Systemwalker Operation Manager Installation Guide* for a list of the menu items, operations, commands and APIs that can be used by different user types.

Information

When a user is granted multiple access rights

In the following cases, the most powerful access right (update right > change right or operation right > reference right) is valid:

- When a user and the group to which that user belongs have been assigned different access rights
- When the operating system user that executed a command or API corresponds to multiple Operation Manager users (operations administrators or operations staff) and different access rights have been set up [UNIX version]

Note that in the above cases, if change rights have been set up on the one hand and operation rights have been set up on the other, then both rights are enabled.

Controlling access to Systemwalker Operation Manager directories and files

The users permitted to access directories and files relating to Systemwalker Operation Manager can be restricted as follows:

Systemwalker Operation Manager [Windows version]

System administrators, operations administrators, and users belonging to the swadmin group

Systemwalker Operation Manager [UNIX version]

File and directory owners and users belonging to the swadmin group

To restrict users as shown above, select the Operation Manager user restrictions option in the **Define Operation Manager Shared Parameter** window.

The system administrator should register all Systemwalker Operation Manager users with the swadmin group.

Note that if the above option is selected, use of the following functions will be restricted to system administrators, operations administrators, and users belonging to the swadmin group:

- Starting demand jobs
- Starting job nets that have the Job Execution Control attribute
- Jobscheduler commands

The output destinations of audit log files must be set individually. Refer to "Define user restrictions" in the *Systemwalker Operation Manager Installation Guide* for more information.

4.2.4 Restricting Execution Users

It is possible to restrict the users who are permitted to execute jobs. Register only those users who are needed to execute jobs.

Execution user restriction provides the following security-related benefit:

- Preventing jobs being executed by unexpected users

[Windows version]

Specify **Execute jobs under the respective job owner's authority** in the **Options** sheet of the **Define Operating Information** window, then register the users permitted to execute jobs in the **Define Job Owner's Information** window.

Whenever a user marked as "Undefined" in the **Define Job Owner's Information** window attempts to run a job, that job will terminate abnormally.

Refer to "Defining the System Operating Information" and "Defining the Job Owner Information [Windows version]" in the *Systemwalker Operation Manager Installation Guide* for more information.

[UNIX version]

Users permitted to execute jobs must be registered beforehand with the user control list for job execution. If a job is submitted with the execution user name of an unregistered user, that job will generate a submission error and execution of that job will be denied.

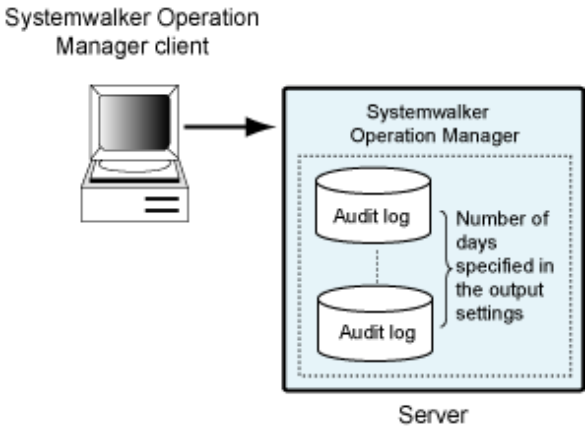
Refer to "Defining the user control list for job execution [UNIX version]" in the *Systemwalker Operation Manager Installation Guide* for more information.

Refer also to "Job Execution Privileges" in the *Systemwalker Operation Manager Installation Guide* for more information about job execution users.

4.2.5 Audit Log Output

Records of the operations performed on Systemwalker Operation Manager can be output to audit logs. Audit logs are set up so that they are output by default when Systemwalker Operation Manager is installed. Audit logs can be used to verify what operations were performed, when and where those operations were performed, and by whom.

Audit logs can be stored at an output destination on a Systemwalker Operation Manager server for a specified number of days only. Operations on clients are also recorded in audit log files on the server.



The following is a list of the main security-related operation records that are output to audit log files:

- Records of authenticated logins to Systemwalker Operation Manager
- Operations administrator and operations staff registrations
- Operations administrator and operations staff password changes
- Project additions and updates
- Service/daemon startup records
- A record of changes to definition information relating to job operations

Note that a history of job execution results is output to Jobscheduler log files (jobdb1.log/jobdb2.log/jobdb3.log). The content of these log files can be checked using the following history display windows. Refer to "Browsing Job History" in the *Systemwalker Operation Manager User's Guide* for more information.

- **Job Net History** window
- **Job History** window

Using audit logs provides the following security-related benefits:

- It is possible to monitor and investigate which users performed unauthorized operations.
- Regular log checks can reveal suspicious operations and access attempts (such as repeated login failures).

The information contained in audit log files can be used in a variety of fault isolation procedures because it can be used to investigate problems caused by incorrect operations as well as malicious behavior.

When Systemwalker Operation Manager is linked to Systemwalker Centric Manager, Systemwalker Centric Manager's audit log management function can be used to gather audit logs on an Operation Management Server where they can be centrally managed.

Refer to "Defining Audit Log Output" in the *Systemwalker Operation Manager Installation Guide* and "Analyzing Audit Logs" in the *Systemwalker Operation Manager User's Guide* for more information about audit log output.

Refer also to the *Systemwalker Operation Manager Reference Guide* for more information about the information that is output to audit log files.

4.2.6 Job Re-execution

Abnormal job termination is one of the main reasons why scheduled jobs fail to run as planned. One way to ensure that jobs run as planned is to restart from any jobs that terminate abnormally.

Systemwalker Operation Manager makes it possible to register an executable file for handling problems (referred to in this manual as a recovery job) as a scheduled job and to run this recovery job automatically whenever an executing job terminates abnormally. If the recovery job terminates normally, the original job that terminated abnormally can be restarted without the need for intervention by operations staff.

Even if a recovery job is not registered, jobs that underwent an abnormal or forced termination can be restarted manually.

The ability to restart jobs provides the following security-related benefit:

- The effects of unexpected interruptions to schedules such as abnormal job terminations can be kept to a minimum, thereby helping to ensure that jobs run as planned.

Refer to the following sources for more information:

Registering a recovery job:

Refer to "Recovery Jobs" in the *Systemwalker Operation Manager User's Guide* for more information. The operations for changing or deleting recovery jobs are the same as the operations for normal jobs.

Performing a manual restart:

Refer to "Operating Scheduled Jobs" in the *Systemwalker Operation Manager User's Guide* for more information. To perform more detailed job recovery operations, refer to "Recovering Scheduled Jobs" and "Job Recovery Operations and Actions" in the *Systemwalker Operation Manager User's Guide*.

Systemwalker Operation Manager also provides the following monitoring windows as a way of checking the status of jobs:

Gantt Chart window:

This window displays the status of all job nets running for the current day, and the job net execution schedule for the current day. A chart showing historical information can be displayed by specifying a past date.

Job Net Management window:

This window displays a list of the job nets that can be viewed by users connected to clients.

Job List window:

This window displays information relating to a job net and a list of the jobs registered with the job net.

Job net list window:

This window displays information relating to a group and a list of the job nets registered with the group.

Refer to "Monitoring Jobs" in the *Systemwalker Operation Manager User's Guide* for more information about how to check the status of jobs, and "Status and Operations of Jobs, Job Nets and Groups" in the *Systemwalker Operation Manager User's Guide* for more information about the statuses of jobs and job nets.

4.3 Web Console Encrypted Communication

In Systemwalker Operation Manager, encrypted communication (SSL: Secure Socket Layer) is used for communication between the Web server and the Web Console. This section explains the certificate and key management environment required to use SSL.

Certificates and private keys

To use SSL, the Certificate Authority (CA) certificate, site certificate, and the corresponding private keys are required. Also, a CRL (Certificate Revocation List) is used to check the certificate validity.

A certificate and CRL that conforms to either X.509 or RFC2459, and uses the RSA cipher algorithm to generate keys, can be used.

- CA certificates

  This is the certificate of the CA itself. This certificate guarantees certificates issued by the CA.

  The CA can issue a certificate to a subordinate CA. In this case, the CA's certificate and the subordinate CA's certificate are both called CA certificates. However the subordinate CA's certificate is also called an intermediate CA certificate.

- Site certificates

  This is a certificate issued by the CA that guarantees the identify of a server. It contains information related to the server and the CA. The site certificate must be used in combination with the CA's certificate. A certificate's validity period is contained in the certificate itself, and cannot be used once it has expired. The certificate must be updated and a new one obtained before it expires. Refer to "Updating Certificates (When Certificates Expire" in the *Systemwalker Operation Manager Installation Guide* for more information.

- The private key that corresponds to the site certificate

  This is the key that is paired with the public key contained in the site certificate.

  Note

  If the private key is lost, the corresponding site certificate cannot be used. For this reason, it must always be backed up.

- CRL (Certificate Revocation List)

  The CA issues the CRL, which includes a list of invalid certificates that were issued by that CA. Examples of events that will expire or invalidate a certificate are the theft of a private key or the loss of user credentials.

  If this is used in SSL communication, it will be referenced when the destination server's certificate is checked for validity.

  The CRL is issued periodically, and is released to each Web server or directory server that is managed by the CA. The release method is different depending on the CA system, so check with the CA. Note that the release location might be described in the certificate.

Certificate Authority (CA)

A CA is required to obtain a certificate.

In certificate and key management environments, certificates and CRLs are supported when they are issued by one of the following:

- VeriSign, Inc.

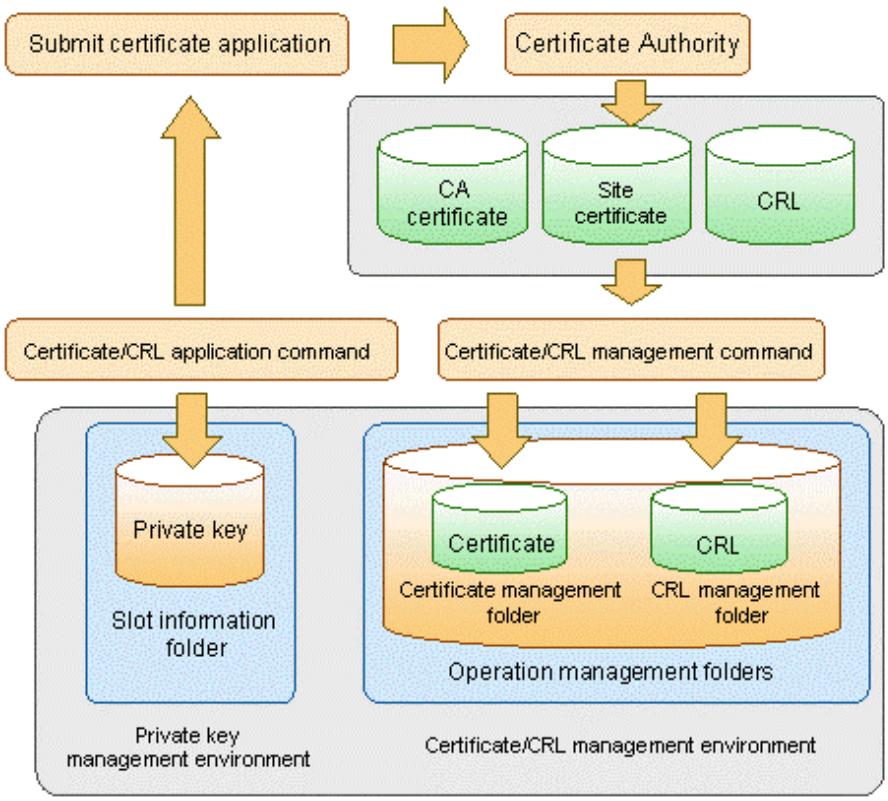
  Supports the Secure Site and Secure Site with EV (EV SSL Certificates).

- Cybertrust, Inc.

  Supports SureServer for SSL Certificates.

Image of the certificate/key management environment

An image of the certificate and key management environment is shown below:



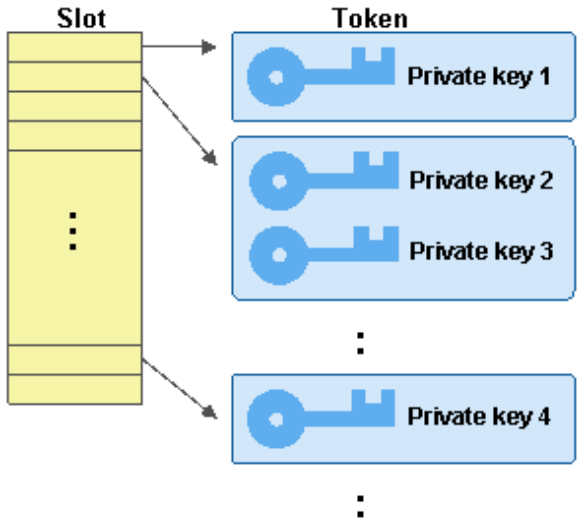
Method used to manage private keys

In private key management, the private key is handled according to the concept of slot and token.

Slot is an abstraction for a physical device which can load encryption equipment, and a token is an abstraction for the encryption equipment.

Only one token can be assigned to a slot, however multiple private keys can be registered in a token.

The relationship between the slots, tokens, and private keys is shown below:



The slot password is required for operations to process the slot information. SO-PIN, or the user PIN, is required for operations to process the token information. The respective passwords are set when the slots and tokens are generated. Note that SO-PIN is only a setting, and is not used in normal operations.

The user PIN is the information required for authentication when the private key in the token is accessed (when the private key is generated using the cmmakecsr command). Since the user PIN exists in each token, if multiple private keys have been registered in one token, information about the private keys can be accessed using a single user PIN.

The relationship between the password and PIN for the slot and token is shown in the table below:

Type	Number	Main use
Slot password	One per slot	Generation of the token
SO-PIN	One per token	-
User PIN	One per token	Private key access (cmmakecsr)

Appendix A Compatibility with Earlier Versions

This appendix explains the compatibility of this version of Systemwalker Operation Manager with earlier versions.

The following table shows the correspondences between the Systemwalker Operation Manager version/levels for each platform, as well as the editions provided.

Systemwalker Operation Manager version/levels

Windows x86 version	Windows x64 version	Windows for Itanium version	Solaris 32bit version	Solaris 64bit version	Linux x86 version	Linux x64 version	Linux for Itanium version	HP-UX version	AIX version	Generic UNIX version
V4.0L10	-	-	-	-	-	-	-	-	-	V4.0
V4.0L20	-	-	4.1	-	-	-	-	4.1 (SE)	4.1 (SE)	V4.1
-	-	-	5.0 (SE/EE)	-	-	-	-	-	-	V5.0
V5.0L20 (SE/EE)	-	-	-	-	-	-	-	-	-	V5.1
V5.0L30 (SE/EE)	-	-	5.2 (SE/EE)	-	5.2 (SE)	-	-	-	-	V5.2
V10.0L10 (SE/EE)	-	-	10.0 (SE/EE)	-	V10.0L10 (SE)	-	-	10.0 (SE)	10.0 (SE)	V10.0
V10.0L20 (SE/EE) V10.0L21 (SE/EE)	-	-	10.1 (SE/EE)	-	-	-	-	-	-	V10.1
V11.0L10 (SE/EE)	-	-	11.0 (SE/EE)	-	V11.0L10 (SE/EE)	-	-	-	-	V11.0
-	-	-	-	-	-	-	V12.0.L10 (EE)	-	-	V12.0
-	-	-	12.1 (SE/EE)	-	-	-	-	-	-	V12.1
V13.2.0 (SE/EE)	-	V13.2.0 (SE/EE)	V13.2.0 (SE/EE)	-	V13.2.0 (SE/EE)	-	V13.2.0 (SE/EE)	V13.2.0 (SE/EE)	V13.2.0 (SE/EE)	V13.2.0

Windows x86 version	Windows x64 version	Windows for Itanium version	Solaris 32bit version	Solaris 64bit version	Linux x86 version	Linux x64 version	Linux for Itanium version	HP-UX version	AIX version	Generic UNIX version
V13.3.0 (SE/EE)	-	V13.3.0 (SE/EE)	V13.3.0 (SE/EE)	-	V13.3.0 (SE/EE)	-	V13.3.0 (SE/EE)	V13.3.0 (SE/EE)	V13.3.0 (SE/EE)-	V13.3.0
V13.3.1 (SE/EE)	-	V13.3.1 (SE/EE)	V13.3.1 (SE/EE)	-	V13.3.1 (SE/EE)	-	V13.3.1 (SE/EE)	V13.3.1 (SE/EE)	V13.3.1 (SE/EE)	V13.3.1
V13.8.0 (SE/EE)	V13.8.0 (SE/EE)	-	V13.8.0 (SE/EE)	V13.8.0 (SE/EE)	V13.8.0 (SE/EE)	V13.8.0 (SE/EE)	-	-	-	V13.8.0

A.1 Support for Client / Server Connections

This section explains the support range between the current version of Systemwalker Operation Manager and earlier versions when clients and servers are connected.

Note

Note the following points when connecting a newer version client to an older version server:

- Functions that are not provided by the server cannot be used.

A.1.1 Connection between Windows Version Clients and Windows Version Servers

The following table shows which Windows version Systemwalker Operation Manager clients can connect to which Windows version Systemwalker Operation Manager servers.

Systemwalker Operation Manager connection availability

(Server versions: V11.0L10 - V13.8.0)

		Windows version server		
		V11.0L10	V13.0.0 V13.2.0 V13.3.0 V13.3.1	V13.8.0
Client	V11.0L10	Yes	No	No
	V13.0.0 V13.2.0 V13.3.0 V13.3.1	Yes	Yes (Note)	No
	V13.8.0	Yes	Yes	Yes

Yes: Can connectNo: Cannot connect

Note: Connection availability between servers and environment setup clients

The table below shows the connection relationships between environment setup clients and servers in V13.0.0 - V13.3.1:

		Server	
		V13.0.0 V13.2.0	V13.3.0 V13.3.1
Environment setup client	V13.0.0 V13.2.0	Yes	No
	V13.3.0 V13.3.1	Yes	Yes

EE

Refer to the *Systemwalker Operation Manager User's Guide - Master Schedule Management* for information on the connection range of the client and the server for the Master Schedule Management function.

A.1.2 Connection between UNIX Version Clients and UNIX Version Servers

The following table shows which UNIX version Systemwalker Operation Manager clients can connect to which UNIX version Systemwalker Operation Manager servers.

Systemwalker Operation Manager connection availability

(Server versions:V11.0 - V13.8.0)

		UNIX version server			
		V11.0	V12.0 V12.1	V13.0.0 V13.2.0 V13.3.0 V13.3.1	V13.8.0
Client	V11.0	Yes	No	No	No
	V12.0 V12.1	Yes	Yes	No	No
	V13.0.0 V13.2.0 V13.3.0 V13.3.1	Yes	Yes	Yes (Note)	No
	V13.8.0	Yes	Yes	Yes	Yes

Yes: Can connectNo: Cannot connect

Refer to the table at the beginning of this appendix for information about correspondences between the Systemwalker Operation Manager versions for each operating system and the version names in the table above.

Note: Connection availability between servers and environment setup clients

Refer to "Note: Connection availability between servers and environment setup clients" in "A.1.1 Connection between Windows Version Clients and Windows Version Servers" for details on the connection relationships between environment setup clients and servers in V13.0.0 - V13.3.1.

EE

Refer to the *Systemwalker Operation Manager User's Guide - Master Schedule Management* for information on the connection range of the client and the server for the Master Schedule Management function.

A.1.3 Connection between Windows Version Clients and UNIX Version Servers

The following table shows which Windows version Systemwalker Operation Manager clients can connect to which UNIX version Systemwalker Operation Manager servers.

Systemwalker Operation Manager connection availability

(Server versions:V11.0 - V13.8.0)

		UNIX version server			
		V11.0	**V12.0 V12.1**	**V13.0.0 V13.2.0 V13.3.0 V13.3.1**	**V13.8.0**
Client	**V11.0L10**	Yes	No	No	No
	V12.0L10 V12.0L11	Yes	Yes	No	No
	V13.0.0 V13.2.0 V13.3.0 V13.3.1	Yes	Yes	Yes (Note)	No
	V13.8.0	Yes	Yes	Yes	Yes

Yes: Can connectNo: Cannot connect

Refer to the table at the beginning of this appendix for information about correspondences between the Systemwalker Operation Manager versions for each operating system and the version names in the table above.

Note: Connection availability between servers and environment setup clients

Refer to "Note: Connection availability between servers and environment setup clients" in "A.1.1 Connection between Windows Version Clients and Windows Version Servers" for details on the connection relationships between environment setup clients and servers in V13.0.0 - V13.3.1.

EE

Refer to the *Systemwalker Operation Manager User's Guide - Master Schedule Management* for information on the connection range of the client and the server for the Master Schedule Management function.

A.1.4 Connection between UNIX Version Clients and Windows Version Servers

The following table shows which UNIX version Systemwalker Operation Manager clients can connect to which Windows version Systemwalker Operation Manager servers.

Systemwalker Operation Manager connection availability

(Server versions:V11.0L10 - V13.8.0)

		Windows version server			
		V11.0L10	**V12.0L10 V12.0L11**	**V13.0.0 V13.2.0 V13.3.0 V13.3.1**	**V13.8.0**
Client	**V11.0**	Yes	No	No	No
	V12.0 V12.1	Yes	Yes	No	No
	V13.0.0 V13.2.0 V13.3.0 V13.3.1	Yes	Yes	Yes (Note)	No

		Windows version server			
		V11.0L10	V12.0L10 V12.0L11	V13.0.0 V13.2.0 V13.3.0 V13.3.1	V13.8.0
	V13.8.0	Yes	Yes	Yes	Yes

Yes: Can connectNo: Cannot connect

Refer to the table at the beginning of this appendix for information about correspondences between the Systemwalker Operation Manager versions for each operating system and the version names in the table above.

Note: Connection availability between servers and environment setup clients

Refer to "Note: Connection availability between servers and environment setup clients" in "A.1.1 Connection between Windows Version Clients and Windows Version Servers" for details on the connection relationships between environment setup clients and servers in V13.0.0 - V13.3.1.

EE

Refer to the *Systemwalker Operation Manager User's Guide - Master Schedule Management* for information on the connection range of the client and the server for the Master Schedule Management function.

A.2 Connection Support for Multi-server Monitoring

This section explains which versions can be monitored when servers with different versions are monitored using a multi-server monitoring client.

The following table shows the range of supported connections between different versions of Systemwalker Operation Manager.

For information about the correspondence between the version numbers in the table below and the Systemwalker Operation Manager versions for each operating system, refer to the correspondence table at the start of this appendix.

Monitored server versions: V11.0L10 - V13.8.0

			Monitored server		
			V11.0L10/ V11.0 - V13.2.0	V13.3.0 V13.3.1	V13.8.0
Monitoring server	**Login as the system administrator**	**V11.0L10/ V11.0 - V13.2.0**	Yes	Yes	Yes (*4)
		V13.3.0 V13.3.1	Yes	Yes	Yes (*4)
		V13.8.0	Yes	Yes	Yes
	Login as a general user	**V13.3.0 V13.3.1**	No (*1)	Yes	Yes (*2,4)
		V13.8.0	No (*1)	Yes	Yes (*2)
	Login using a Systemwalker user ID	**V13.8.0**	No (*3)	No (*3)	Yes (*3)

Yes:

Can connect (and subsystem monitoring is also possible).

No:

Cannot connect.

***1:**

If a general user logs in to the monitoring server and attempts to connect to a monitored server of V13.2.0 or earlier, the information cannot be obtained since there is no function for authenticating general users in monitored servers of V13.2.0 or earlier. This monitored server will be treated as "access denied" in the host summary window. Also, the job net information will not be obtainable using the **Job Nets Management** window.

***2:**

If a general user logs in to the monitoring server and attempts to connect to a monitored server that uses the Systemwalker User Management function, the information cannot be obtained since general users cannot be authenticated in monitored servers. This monitored server will be treated as "access denied" in the host summary window. Also, the job net information will not be obtainable using the **Job Nets Management** window.

***3:**

If a user with a Systemwalker user ID logs in to the monitoring server and attempts to connect to a monitored server that does not use the Systemwalker User Management function, the information cannot be obtained since Systemwalker user IDs cannot be authenticated in monitored servers. This monitored server will be treated as "access denied" in the host summary window. Also, the job net information will not be obtainable using the **Job Nets Management** window.

***4:**

If the monitoring targets are only the servers for an IPv6 communication environment or a mix of servers for an IPv4 communication environment and for an IPv6 communication environment, all of the monitoring target servers, including those for the IPv4 communication environment, cannot be monitored.

If you log on from a multi-server monitoring client, a message stating that host summary information cannot be found is displayed, and the multi-server monitoring client terminates.

Connection between multi-server monitoring clients and monitoring servers

Connecting a V12.0L10/V12.1 client to a monitoring server running V11.0L10/V11.0 or earlier

Definitions for multiple monitoring hosts cannot be registered or selected.

When monitoring V13.8.0 or later servers from a V13.3.1 or earlier client

If the monitoring targets are only the servers for an IPv6 communication environment or a mix of servers for an IPv4 communication environment and for an IPv6 communication environment, all of the monitoring target servers, including those for the IPv4 communication environment, cannot be monitored.

If you log on from a multi-server monitoring client, a message stating that host summary information cannot be found is displayed, and the multi-server monitoring client terminates.

A.3 Support for Extracting and Distributing Policy Data

This section explains the scope of support for extracting and distributing policy data.

Support for different platforms

- Policy information extracted using the Windows version can only be distributed to systems running the Windows version. These policies cannot be distributed to systems running the UNIX version.
- Policy information extracted using the UNIX version (Solaris, HP-UX or AIX) can only be distributed to systems running the UNIX version (Solaris, HP-UX or AIX).
- Policy information extracted using the Linux version or the Linux for Itanium version can only be distributed to systems running the Linux version or the Linux for Itanium version. These policies cannot be distributed to other platforms.

Support for different editions

The following extraction source/distribution destination combinations can be used with Systemwalker/OperationMGR 5.0 or later.

- Extraction source: Standard Edition; Distribution destination: Standard Edition or Enterprise Edition
- Extraction source: Enterprise Edition; Distribution destination: Enterprise Edition

Policy information collected using the Enterprise Edition cannot be distributed to Standard Edition systems.

Support for different versions

The following table shows the range of support for different versions.

Distribution destination versions:V11.0L10 - V13.8.0

		Distribution destination					
		V11.0L10 V11.0	**V12.0L10 V12.0L11 V12.0 V12.1**	**V13.0.0**	**V13.2.0**	**V13.3.0 V13.3.1**	**V13.8.0**
Extraction source	**V11.0L10 V11.0**	Yes	Yes (*1)	Yes (*1)	Yes (*1)	Yes (*1)	Yes (*1,2)
	V12.0L10 V12.0L11 V12.0 V12.1	No	Yes (*3)	Yes (*3)	Yes (*3)	Yes (*3)	Yes (*3)
	V13.0.0	No	No	Yes (*3)	Yes (*3)	Yes (*3)	Yes (*3)
	V13.2.0	No	No	No	Yes (*3)	Yes (*3)	Yes (*3)
	V13.3.0 V13.3.1	No	No	No	No	Yes (*3)	Yes (*3)
	V13.8.0	No	No	No	No	No	Yes (*3)

Yes: Available No: Unavailable

Refer to the table at the beginning of this appendix for information about correspondences between the Systemwalker Operation Manager versions for each operating system and the version names in the table above.

EE

Refer to the *Systemwalker Operation Manager User's Guide - Master Schedule Management* for information on the support range for the extraction and distribution of the Master Schedule Management function policy information.

***1:**

To distribute policy information for **Schedule DB/schedule pattern** to the server where the Extended User Management function is enabled, be sure to disable the Extended User Management function at the destination server before distributing the policy information. [UNIX version]

***2:**

To distribute the **Schedule DB/schedule pattern** policy information to a server in which the Systemwalker authentication repository is enabled, disable the Systemwalker authentication repository on the destination first.

***3:**

To distribute the **Schedule DB/schedule pattern** policy information, first make sure that the Extended User Management function or the Systemwalker authentication repository usage status on the source and destination match.

A.4 Support for Operation Information Definitions

This section explains support for operation information definitions in earlier versions.

Node name definition for cluster operations

- For four-node clusters, this setting can only be made with Systemwalker Operation Manager V11.0L10 or later. [Windows version]
- For four-node clusters, this setting can only be made with Systemwalker Operation Manager V11.0 or later. [UNIX version]

EE

Defining logical IP addresses for cluster operation

- This setting can only be made with Systemwalker Operation Manager V11.0L10 or later. [Windows version]
- This setting can only be made with Systemwalker Operation Manager V11.0 or later. [UNIX version]

Job object mode

- Job object mode can only be set for V13.3.0 or later of the Windows or Windows for Itanium versions of Systemwalker Operation Manager.

A.5 Executable Range of Network Jobs

This section explains the range of network jobs that can be executed when the schedule server (where the network jobs are submitted) uses a version that is different from the version used by the execution server (where the network jobs are executed).

Basically, network jobs can be submitted to servers running any version.

However, this does not apply if the multi-subsystem operations provided by the Enterprise Edition of Systemwalker Operation Manager are used. The tables below show the combinations where network jobs can be executed.

<table>
<tr><th colspan="3" rowspan="3"></th><th colspan="4">Destination of request</th></tr>
<tr><th colspan="2">V10.0L10/5.2 to V13.2.0 (*2)</th><th colspan="2">V13.3.0 and later</th></tr>
<tr><th>No multi-sub system operation/ Subsystem 0</th><th>Sub systems 1 to 9</th><th>No multi-subsystem operation/ Subsystem 0</th><th>Sub systems 1 to 9</th></tr>
<tr><td rowspan="4">Source of request</td><td rowspan="2">V10.0L10/ 5.2 to V13.2.0 (*2)</td><td>No multi-subsystem operation/Sub system 0</td><td>A</td><td>C</td><td>A</td><td>C</td></tr>
<tr><td>Subsystems 1 to 9</td><td>C(*1)</td><td>A</td><td>C(*1)</td><td>A</td></tr>
<tr><td rowspan="2">V13.3.0 and later</td><td>No multi-subsystem operation/Sub system 0</td><td>A</td><td>C</td><td colspan="2" rowspan="2">B</td></tr>
<tr><td>Subsystems 1 to 9</td><td>C(*1)</td><td>A</td></tr>
</table>

A: Can send an execution request to the same subsystem number
("No multi-subsystem operation" is handled in the same way as when "Subsystem 0" is specified)

B: Can send an execution request to any subsystem

C: Cannot send an execution request

***1:**

Can send an execution request when no port number is specified by the requesting source (the schedule server)

***2:**

This does not apply to AIX versions of Systemwalker Operation Manager 11.0 and earlier.

Note

When submitting network jobs from the schedule server in a new version to the execution server in an old version, the character limit for setting items must follow the one used in the old version.

Refer to "Upgrading Notes and Incompatible Items" in the *Systemwalker Operation Manager Upgrade Guide* for compatibilities for the character limit of setting items.

A.6 Support for Monitored Servers and Monitoring Servers Using the Web Console

This section explains the compatibility between monitored servers and monitoring servers that connect using the Web Console (Web browser), when the server versions do not match. Refer also to "Defining Users" in the *Systemwalker Operation Manager Installation Guide*.

Note

Note the following points when connecting from a monitoring server running a newer version to a monitored server running an older version:

- Functions that are not provided by the monitored server cannot be used.

Any platform where Systemwalker Operation Manager has been installed can be registered as a monitored server.

The compatibility between monitored servers and monitoring servers that connect using the Web Console is shown in the table below.

Monitored server versions: V11.0L10 - V12

		Monitored server		
		V11.0L10 11.0	**V12.0L10 12.0**	**V12.0L11 12.1**
Server connected via Web Console	**V11.0L10 11.0**	Yes	No	No
	V12.0L10 12.0	Yes	Yes	No
	V12.0L11 12.1	Yes	Yes	Yes
	V13.0.0	Yes	Yes	Yes
	V13.2.0	Yes	Yes	Yes
	V13.3.0 V13.3.1	Yes	Yes	Yes
	V13.8.0	Yes	Yes	Yes

Yes: Can connect No: Cannot connect

Monitored server versions:V13.0.0 to V13.8.0

		Monitored server			
		V13.0.0	**V13.2.0**	**V13.3.0 V13.3.1**	**V13.8.0**
Server connected via Web Console	**V11.0L10 11.0**	No	No	No	No
	V12.0L10 12.0	No	No	No	No
	V12.0L11 12.1	No	No	No	No
	V13.0.0	Yes	No	No	No
	V13.2.0	Yes	Yes	No	No
	V13.3.0 V13.3.1	Yes	Yes	Yes	No
	V13.8.0	Yes	Yes	Yes	Yes

Yes: Can connect No: Cannot connect

A.7 Systemwalker Operation Manager Client that is Called from Systemwalker Centric Manager

When Systemwalker Operation Manager is called from Systemwalker Centric Manager, the client used will be different depending on which Systemwalker Operation Manager and Systemwalker Centric Manager versions are used.

The Systemwalker Operation Manager client that is called from Systemwalker Centric Manager is shown in the table below.

			Systemwalker Operation Manager client (to be called)	
			V13.3.1 or earlier	V13.8.0 or later
Systemwalker Centric Manager (Caller)	V13.3.1 or earlier	The Systemwalker Operation Manager client function is installed	Windows client	Windows client
		The Systemwalker Operation Manager client function is not installed	No	No
	V13.8.0 or later	The Systemwalker Operation Manager client function is installed	Windows client	Web Console
		The Systemwalker Operation Manager client function is not installed	No	Web Console

No: Neither the Windows client nor the Web Console can be called.

Appendix B Limit Values

This appendix explains the limit values for Systemwalker Operation Manager.

B.1 Limit Values for Operations

This section explains the limit values for operations.

Number of servers that can be monitored by the Multi-server Monitoring function and by the Web Console

Up to 62 servers can be monitored using the Multi-server Monitoring function. If 63 or more servers are monitored, the 63rd and subsequent servers will be displayed as "Not running" or "Requesting" in the **Multi-server Monitoring** window. A single subsystem is counted as a single server.

Number of clients that can be connected to the Systemwalker Operation Manager server

Up to 62 clients can connect to the Systemwalker Operation Manager server (Note). The maximum number of connected clients can be limited to no more than 62 clients by using the **Use function2** tab in the **Define Jobscheduler Startup Parameters** window. However, the number of clients cannot be limited if they are connecting to a Systemwalker OperationMGR server running V10.0L21 [Windows version] or V10.1 [UNIX version] or earlier.

Note) This limit may vary depending on the operating system.

The following systems are counted as "connected clients":

- Systems that have been connected to the Jobscheduler server by selecting **Jobscheduler** from the **job selection** pane of the Systemwalker Operation Manager window on a client
- Systems that have been connected to the Jobscheduler server using the Jobscheduler information print client
- Servers that are monitored using the Multi-server Monitoring client

Number of message events that can be accumulated for job nets

Up to 255 message events can be accumulated for each job net.

If the message event count is decremented or cleared due to the job net starting, message events can be accumulated from the decremented or cleared level until the count reaches 255.

B.2 Limit Values for Job Scheduling

This section explains the limit values for job scheduling.

Number of projects that can be registered with the Jobscheduler

Up to 1000 projects can be registered with the Jobscheduler.

Number of job nets that can be registered per project

The number of job nets that can be registered per project is as follows.

Standard Edition:

Up to 255 job nets

Enterprise Edition:

No limit

Note

Although there is no limit to the number of job nets that can be registered when the Enterprise Edition of Systemwalker Operation Manager is used, conduct thorough performance tests to make sure that job nets start on schedule without any problems. Refer to "Tuning of Performance" of the *Systemwalker Operation Manager User's Guide* for details.

Number of job nets that can be registered per group

The number of job nets that can be registered per group is as follows.

Standard Edition:

Up to 50 job nets

Enterprise Edition:

Up to 255 job nets

Number of jobs that can be registered per job net

Up to 255 jobs (including recovery jobs) can be registered per job net. However, only one job can be registered per job net with the Interstage attribute.

In a child or linked job net registered in a job net is treated as a job. A child or linked job net in a job net is counted as one job, and up to 255 jobs (including child and linked job nets) can be registered per job net.

Number of message events that can be registered per job net

Up to 70 message events can be registered for each job net.

Number of linked job nets that reference a master linked job net

Up to 100 linked job nets can reference a single master linked job net. When a parent job net uses multiple linked job nets that reference the same master linked job net, they are counted as separate job nets. Linked job nets copied using 'copy and start' are not counted towards the maximum number of linked job nets that can be registered.

Note that there are no limits for the following:

- Number of master linked job nets
- Number of linked job nets in one project

B.3 Limit Values for Job Execution Control

This section explains the restrictions for Job Execution Control.

Number of queues that can be created per system

Up to 64 queues can be created per system.

Number of jobs that can run concurrently per system (Multiplicity limit)

The maximum number of jobs that can be executed concurrently per system (system multiplicity) is 99 for the Windows version or 999 for the UNIX version.

Number of jobs that can be submitted per system

The maximum number of jobs that can be submitted per system is 99999, the maximum value for job numbers. This number may be further limited according to the number of jobs that can be submitted to queues.

Number of job files that can be displayed per job folder

Up to 4096 job files can be displayed per job folder on Systemwalker Operation Manager clients.

Number of job folders that can be created per system

Up to 255 job folders can be created per system.

Number of resources that can be created per system

Up to 9999 resources can be created per system.

Number of execution servers that can be defined in the host group

Up to 100 execution servers can be defined in a single host group.

B.4 Limit Values for the Systemwalker Operation Manager Web Console

This section explains the limit values for the Systemwalker Operation Manager Web Console.

Number of subsystems that can be monitored

Up to 62 subsystems can be monitored out of all the monitored hosts.

Appendix C List of Functional Differences for Each Operating System

The functions provided by Systemwalker Operation Manager vary depending on the operating systems where they run. The following table shows the differences in functions between operating systems.

Major classification	Function	Windows x86 version x64 version		Solaris 32bit version 64bit version		HP-UX version		AIX version		Linux x86 version x64 version	
		SE	EE	SE	EE	SE	EE	SE	EE	SE	EE
Power Control	Turning server power on and off	S (x86 version)	S (x86 version)	N	N	N	N	N	N	N	N
Extended User Management		N	N	S	S	S	S	S	S	S	S
Systemwalker User Management		S	S	S	S	N	N	N	N	S	S
Calendar		S	S	S	S	S	S	S	S	S	S
Jobscheduler (*1)		S	S	S	S	S	S	S	S	S	S
Job Execution Control (*2)		S	S	S	S	S	S	S	S	S	S
Master Schedule Management	Daily schedule management	N	S	N	S	N	S	N	S	N	S
	Operation change	N	S	N	S	N	S	N	S	N	S
	Carried over job net	N	S	N	S	N	S	N	S	N	S
	Daily schedule distribution	N	N	N	S	N	S	N	S	N	S
	Multi-server operation	N	N	N	S	N	S	N	S	N	S
Service/ Application Execution	Service execution	S	S	N	N	N	N	N	N	N	N
	Application execution	S	S	S	S	S	S	S	S	S	S
Event Monitoring		S	S	N	N	N	N	N	N	N	N

Major classification	Function	Windows x86 version x64 version		Solaris 32bit version 64bit version		HP-UX version		AIX version		Linux x86 version x64 version	
		SE	EE	SE	EE	SE	EE	SE	EE	SE	EE
Action Control		S	S	N	N	N	N	N	N	N	N
Backup Link		S	S	N	N	N	N	N	N	N	N
Task Link		S	S	S	S	S	S	S	S	S	S
Systemwalker Script		S	S	S	S	S	S	S	S	S	S
Monitoring and Operating from a Web Browser		S	S	S	S	P (*7)	P (*7)	P (*7)	P (*7)	S	S
Systemwalker Single Sign-On		S	S	S	S	N	N	N	N	S	S
ServerView Single Sign-On		S	S	N	N	N	N	N	N	S	S
Linking to Interstage (*3)		S	S	S	S	N	N	N	N	S	S
API	Calendar APIs	S (*8)	S (*8)	S	S	N	N	N	N	N	N
	Jobscheduler APIs (*4)	S (*8)	S (*8)	S	S	N	N	N	N	S (*9)	S (*9)
	Job Execution Control APIs (*4)	S (*8)	S (*8)	S	S	N	N	N	N	S (*9)	S (*9)
	Action Control APIs	S (*8)	S (*8)	N	N	N	N	N	N	N	N
Policy Operation	Policy distribution (*5)	S	S	S	S	S	S	S	S	S	S
Security	Audit log output	S	S	S	S	S	S	S	S	S	S
Large-Scale Systems	Large-scale operations	N	S	N	S	N	S	N	S	N	S
	Subsystems	N	S	N	S	N	S	N	S	N	S
High Reliability	Cluster system	N	S	N	S	N	S	N	S	N	S

Major classifi-cation	Function	Windows x86 version x64 version		Solaris 32bit version 64bit version		HP-UX version		AIX version		Linux x86 version x64 version	
		SE	EE	SE	EE	SE	EE	SE	EE	SE	EE
	applicati on (*6)										
Trouble shooting	Mainten ance Informat ion Collecti on Tool	S	S	S	S	S	S	S	S	S	S
IPv6	IPv6	S	S	S	S	S	S	S	S	S	S

S: Supported.

P: Partially supported.

N: Not supported.

***1:**

Define Message Table is only available in Windows versions.

***2:**

Define Job Owner's Information and the Backward Compatibility Load Balancer function are only available in Windows versions.

***3:**

The WorkUnits that can be registered vary for each operating system. Refer to "Linking to Interstage" for details.

***4:**

Available APIs vary for each operating system. Refer to the *Systemwalker Operation Manager Reference Guide* for details.

***5:**

The target operating systems and editions for policy distribution vary depending on the operating system and edition from where the policy is extracted. Refer to "A.3 Support for Extracting and Distributing Policy Data" for details.

***6:**

The following cluster systems are supported:

- Solaris: PRIMECLUSTER, Sun Cluster, and Oracle Solaris Cluster
- Windows: Microsoft(R) Fail Over Clustering
- Linux: PRIMECLUSTER
- HP-UX: HP ServiceGuard
- AIX: PowerHA

***7:**

Enable operation and monitoring by registering as a Windows, Solaris, or Linux monitoring target.

***8:**

These APIs have been created for 64-bit operating systems, so create 64-bit applications when using these APIs with Windows x64 version.

***9:**

These APIs have been created for 64-bit operating systems, so create 64-bit applications when using these APIs with Linux x64 version.