Fujitsu Software
Technical Computing Suite V4.0L20

Job Operation Software
Administrator's Guide
for System Management

J2UL-2455-02ENZ0(10)
March 2024

Preface

Purpose of This Manual

This manual describes the settings and operation method of the system management function for the "Job Operation Software" included in the Technical Computing Suite.

Intended Readers

This manual is intended for system administrators who operate and manage systems where the Job Operation Software is installed.

The manual assumes readers have the following knowledge:

- Basic Linux knowledge
- Overall understanding of the Job Operation Software, obtained from the "Job Operation Software Overview."

The administrator who operates jobs must read the "Job Operation Software Administrator's Guide for Job Management."

To maintain the system or to perform a troubleshooting, read the "Job Operation Software Administrator's Guide for Maintenance ″and the "Job Operation Software Troubleshooting."

Organization of This Manual

This manual is organized as follows.

Chapter 1 What is the System Management Function?

This chapter provides the list of the system management functions.

Chapter 2 Overview of System Management Function

This chapter describes the overview of the system management functions.

Chapter 3 Details of the System Management Function

This chapter describes the details of the system management functions.

Notation Used in This Manual

Representation of Units

The following table lists the prefixes representing units in this manual. Basically, disk size is represented as a power of 10, and memory size is represented as a power of 2. Be careful about specifying sizes when displaying or entering commands.

Prefix	Value	Prefix	Value
K (kilo)	10^3	Ki (kibi)	2^{10}
M (mega)	10^6	Mi (mebi)	2^{20}
G (giga)	10^9	Gi (gibi)	2^{30}
T (tera)	10^{12}	Ti (tebi)	2^{40}
P (peta)	10^{15}	Pi (pebi)	2^{50}

Notation of Model Names

In this manual, the computer that based on Fujitsu A64FX CPU is abbreviated as "FX server", and FUJITSU server PRIMERGY as "PRIMERGY server" (or simply "PRIMERGY").

Also, specifications of some of the functions described in the manual are different depending on the target model. In the description of such a function, the target model is represented by its abbreviation as follows:

[FX]: The description applies to FX servers.
[PG]: The description applies to PRIMERGY servers.

Administrators

The Job Operation Software has different types of administrator: system administrator, cluster administrator, and job operation administrator. Unless otherwise noted, the descriptions in this manual apply to functions for system administrators and cluster administrators. For this reason, the term "administrator" usually means a system administrator and a cluster administrator.

Prompt in Command Input Example

# means that the command must be executed with administrator privileges (superuser) for the OS.

Path Names of Commands

In the examples of operations, commands in the /bin, /usr/bin, /sbin, or /usr/sbin directory may not be represented by absolute paths.

Symbols in This Manual

This manual uses the following symbols.

Note

The Note symbol indicates an item requiring special care. Be sure to read these items.

See

The See symbol indicates the reference source of detailed information.

Information

The Information symbol indicates a reference note related to the Job Operation Software.

Trademarks

- Linux(R) is the registered trademark of Linus Torvalds in the U.S. and other countries.
- Red Hat and Red Hat Enterprise Linux are trademarks of Red Hat, Inc. in the U.S. and other countries.
- Microsoft, Windows are either registered trademarks or trademarks of Microsoft Corporation in the U.S. and other countries.
- Other company and product names in this manual are trademarks or registered trademarks of their respective owners.

Export Controls

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

Date of Publication and Version

Version	Manual Code
March 2024, Version 2.10	J2UL-2455-02ENZ0(10)
September 2023, Version 2.9	J2UL-2455-02ENZ0(09)
March 2023, Version 2.8	J2UL-2455-02ENZ0(08)
March 2022, Version 2.7	J2UL-2455-02ENZ0(07)
August 2021, Version 2.6	J2UL-2455-02ENZ0(06)
March 2021, Version 2.5	J2UL-2455-02ENZ0(05)
January 2021, Version 2.4	J2UL-2455-02ENZ0(04)

Version	Manual Code
December 2020, Version 2.3	J2UL-2455-02ENZ0(03)
September 2020, Version 2.2	J2UL-2455-02ENZ0(02)
June 2020, Version 2.1	J2UL-2455-02ENZ0(01)
March 2020, Second version	J2UL-2455-02ENZ0(00)
January 2020, First version	J2UL-2455-01ENZ0(00)

Copyright

Update history

Changes	Location	Version
Changed the description of getting packages for Logstash and Filebeat.	3.6.4.1	2.10
Added 200 Gb/sec (2X NDR) on the Standard Value of the Performance Value of the IC Service.	3.3.4.2	2.9
Added a note about setting item hostname.	3.8.3.1.2 3.8.3.2.2 3.8.8.4 3.8.8.6	2.8
Added instructions if node state is "Failed" after completing a backup or restore.	3.9.2.6 3.9.3.6	
Added 200 Gb/sec (4X HDR) on the Standard Value of the Performance Value of the IC Service.	3.3.4.2	2.7
Corrected an error in IP addresses.	3.8.3.2.3	
Added 100 Gb/sec (2X HDR) on the Standard Value of the Performance Value of the IC Service.	3.3.4.2	2.6
Added a note about the --force-all option of the papwrctl command.	3.2.2 3.2.3.1	2.5
Added a note about displaying the update elapsed time of operation status.	3.3.2.3	
Added instructions for updating the module definition files required to update RHEL 8 OS packages.	3.8.2.6	
Added the information about using a system management node, a compute cluster management node, and a login node on one node.	3.3.1.1 3.6.4.1	2.4
Added instructions for using OS versions RHEL8 for peripherals.	3.1.1.4 3.3.3.3 3.5.4.3 3.8.1.3 3.8.3.1.1 3.8.8.5	2.3
Added a note for maintenance operations.	3.2.2 3.4.3.1 3.4.3.3	2.2
Improved pachecker description.	3.7.2	
Added instructions for configuration to monitor only specified port of Dual Port IB.	3.3.4.2	2.1

Changes	Location	Version
Added a description of behavior when the specified range of nodes for the paclstmgr --recover option includes nodes that are not in soft maintenance mode.	3.4.3.3	
Added a note about operating the service of the Job Operation Software.	3.4.4	
Corrected location of the node serving as both the compute node and the global I/O node.	3.8.3.2.3	
Renamed initial RAM disk (initrd) file.	3.8.4	
Added packages Import to rootfs of PRIMERGY server by default.	3.8.5.5	
Changed the description of packages that can be included in the initrd used for OS installation.	3.8.7	
Corrected errors.	-	
The communication path of a job can now be dynamically changed when a Tofu interconnect link goes down.	3.3.1.2 3.3.2.2	2
The look has changed according to product upgrades.	-	

Contents

Chapter 1 What is the System Management Function?

The computational performance ability of computer systems to scientifically and technically calculate can be enhanced by applying parallel processing using many nodes. The following requirements are required for such a large-scale system.

- A lot of the hardware and software available can be used as one system.
- Even if a failure occurred in a part of the system, the system as a whole can continue to operate normally.
- The system can be divided into several parts to provide an environment where they do not interfere with each other, as needed.

The Job Operation Software has the "system management function" to manage the hardware and software that configure the system in order to realize the system as described above and to efficiently and safely operate the system.

The system management function can be classified as follows.

Table 1.1 System Management Function

Function	Function Overview
Configuration management function	Manages the configuration information of the nodes and networks in the system.
System control function	Controls to start or stop a node in the system.
System monitoring function	Monitors the hardware or software status.
System maintenance function	Supports hardware maintenance and software maintenance.
Operation support function	Supports the operation of the system administrator (batch command execution, etc.).
Log management function	Collects and manages the system logs.
Software environment check function	Checks the Job Operation Software status in the system.
Installation function	Supports the system installation (OS installation, settings, etc.).
Backup and restoration function	Backs up the system disk as a disk image and restores from the disk image.

1.1 Preconditions of the System Management Function

This chapter describes the preconditions to run the system management function.
The system administrator needs to refer to this chapter so as not to affect the operation of the system management function when setting a service used in the system or changing a port number.

1.1.1 Services and Port Numbers Used by the System Management Function

The table below lists the services used by the system management function and the port number used by each service.

Table 1.2 List of Port Numbers Used by Services

Node Type	Service	Protocol	Port Number	Remarks
System management node	dhcpd	UDP	67 68	
	xinetd	UDP	69	This port is used by TFTP.
	httpd	TCP	80	
	chronyd	UDP	123 323	Time synchronization
	sshd	TCP	22	The port can be changed.
	-	UDP	623	For IPMI communication
	pxinstd	TCP	30405	This port is used by the installation function.
	pxmonitor_masterd	TCP	30400	This port is used by the system monitoring function.

Node Type	Service	Protocol	Port Number	Remarks
		UDP	30400	This port is used by the system monitoring function.
	pxmonitor_slaved	TCP	30402	This port is used by the system monitoring function.
		UDP	30402	This port is used by the system monitoring function.
	logstash	TCP	15044	This port is used by the regular log collection function (the port can be changed).
Storage cluster management node	chronyd	UDP	323	Time synchronization (client function only)
	sshd	TCP	22	
	-	UDP	623	For IPMI communication
	pxmonitor_subd	TCP	30401	This port is used by the system monitoring function.
		UDP	30401	This port is used by the system monitoring function.
	pxmonitor_slaved	TCP	30402	This port is used by the system monitoring function.
		UDP	30402	This port is used by the system monitoring function.
	logstash	TCP	15044	This port is used by the regular log collection function (the port can be changed).
Compute cluster management node	chronyd	UDP	323	Time synchronization (client function only)
	sshd	TCP	22	
	-	UDP	623	For IPMI communication
	pxmonitor_subd	TCP	30401	This port is used by the system monitoring function.
		UDP	30401	This port is used by the system monitoring function.
	pxmonitor_slaved	TCP	30402	This port is used by the system monitoring function.
		UDP	30402	This port is used by the system monitoring function.
	logstash	TCP	15044	This port is used by the regular log collection function (the port can be changed).
Compute cluster sub management node	xinetd	UDP	69	This port is used by TFTP.
	httpd	TCP	80	
	chronyd	UDP	123 323	Time synchronization
	sshd	TCP	22	The port can be changed.
	pxinstd	TCP	30405	This port is used by the installation function.
	-	UDP	623	For IPMI communication

Node Type	Service	Protocol	Port Number	Remarks
	pxmonitor_subd	TCP	30401	This port is used by the system monitoring function.
		UDP	30401	This port is used by the system monitoring function.
	pxmonitor_slaved	TCP	30402	This port is used by the system monitoring function.
		UDP	30402	This port is used by the system monitoring function.
	logstash	TCP	15044	This port is used by the regular log collection function (the port can be changed).
	filebeat	TCP	5044	This port is used by the regular log collection function (the port can be changed).
Boot I/O node	dhcpd	UDP	67 68	
	xinetd	UDP	69	This port is used by TFTP.
	nfs	TCP	2049	When NFSv4 is used.
	chronyd	UDP	123 323	Time synchronization
	sshd	TCP	22	The port can be changed.
	-	UDP	623	For IPMI communication
	pxmonitor_subd	TCP	30401	This port is used by the system monitoring function.
		UDP	30401	This port is used by the system monitoring function.
	pxmonitor_slaved	TCP	30402	This port is used by the system monitoring function.
		UDP	30402	This port is used by the system monitoring function.
	filebeat	TCP	5044	This port is used by the regular log collection function (the port can be changed).
Other node types	chronyd	UDP	323	Time synchronization (client function only)
	sshd	TCP	22	
	-	UDP	623	For IPMI communication
	pxmonitor_slaved	TCP	30402	This port is used by the system monitoring function.
		UDP	30402	This port is used by the system monitoring function.

Chapter 2 Overview of System Management Function

This chapter describes the overview of each system management function.

2.1 Overview of Configuration Management Function

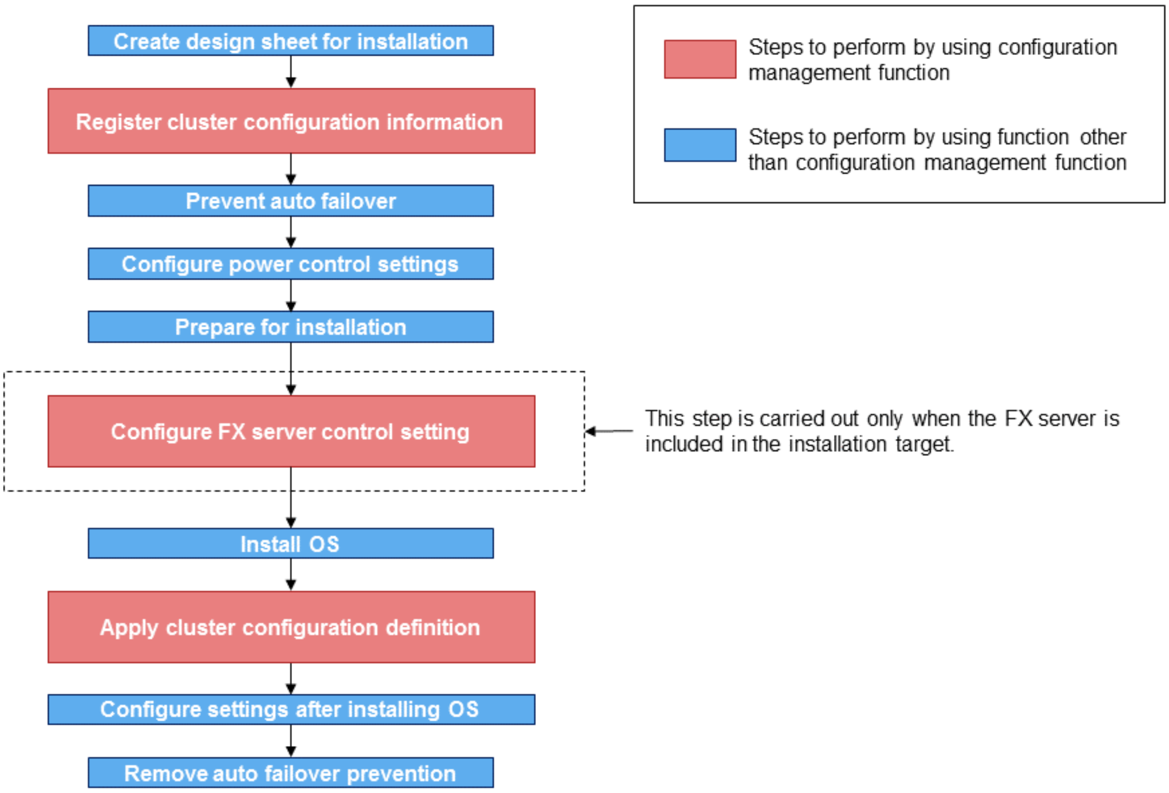
The system administrator uses the configuration management function to systematically manage the entire system.
When installing the system or changing the configuration, the system administrator sets the installed nodes to the necessary settings to enable the Job Operation Software functions, such as grouping information assignment and the initial settings of the Job Operation Software.

This section uses the cluster configuration at the system installation as an example in order to demonstrate the installation steps where the configuration management function is used.

See

This section does not cover the installation steps where the configuration management function is not used. For details on how to install the system for the first time or how to change the configuration, see the "Job Operation Software Setup Guide."

Figure 2.1 Steps that Use the Configuration Management Function When Installing the System


Create design sheet for installation
Register cluster configuration information
Prevent auto failover
Configure power control settings
Prepare for installation
Configure FX server control setting
Install OS
Apply cluster configuration definition
Configure settings after installing OS
Remove auto failover prevention
Steps to perform by using configuration management function
Steps to perform by using function other than configuration management function
This step is carried out only when the FX server is included in the installation target.

The following describes the settings that use the configuration management function.

- Registration of cluster configuration information

  The system administrator uses the configuration management function to divide the nodes that configure the system into groups based on the clusters and node groups, and assigns an identifier, such as a node ID, to each group. Then the system administrator can use the system monitoring function and the batch operation function.
  In addition, the system administrator can define devices, such as nodes, disk devices, and network switches, as configuration information in order to control the power using the system control function.

- FX server control setting

  The system administrator needs to configure the power control setting and the Tofu coordinates for FX server if FX server is included in the system. The system administrator can batch configure these settings using the configuration management function.
  Once the settings are configured, the system administrator can check the FX server power control and the power status.

- Applying the cluster configuration definition

  The cluster configuration information registered to the system management node needs to be applied to the nodes in the system.
  The system administrator uses the configuration management function to apply the cluster configuration defined in the system management node to the entire system. By doing so, the Job Operation Software function is activated in each node in the cluster. Then the system administrator can check the operation status of the Job Operation Software service running on each node.

See

For the procedure to make various settings of configuration management function, see "3.1 Details of the Configuration Management Function."

2.1.1 Overview of Registering the Cluster Configuration Information

The system administrator registers node information, such as node grouping information and IP addresses, to the system management node using the --create option of the pxsystemadm command. The system administrator also configures the initial setting of the Job Operation Software.

Once the node information is registered or the initial setting of the Job Operation Software is configured, the system monitoring function is activated on the system management node. Then the system administrator can check the cluster configuration with the pashowclst command and batch operate multiple nodes with the pmexe command.
The pxsystemadm command uses the following file as input information.

- Node information definition file for FX server: node_ft_*nn*.conf(*)

  When registering the FX server configuration information, the system administrator creates a "design sheet for installation" of the installation function on a Windows PC, and creates node_ft_*nn*.conf, which is the node information definition file for FX server, to describe the Tofu coordinates and IP address of FX server.
  After that, the system administrator transfers the created files to the active system management node in order to use them to register FX server information and install an OS. For details on the files, see "3.8.8.6 Node Information Definition File for FX server."

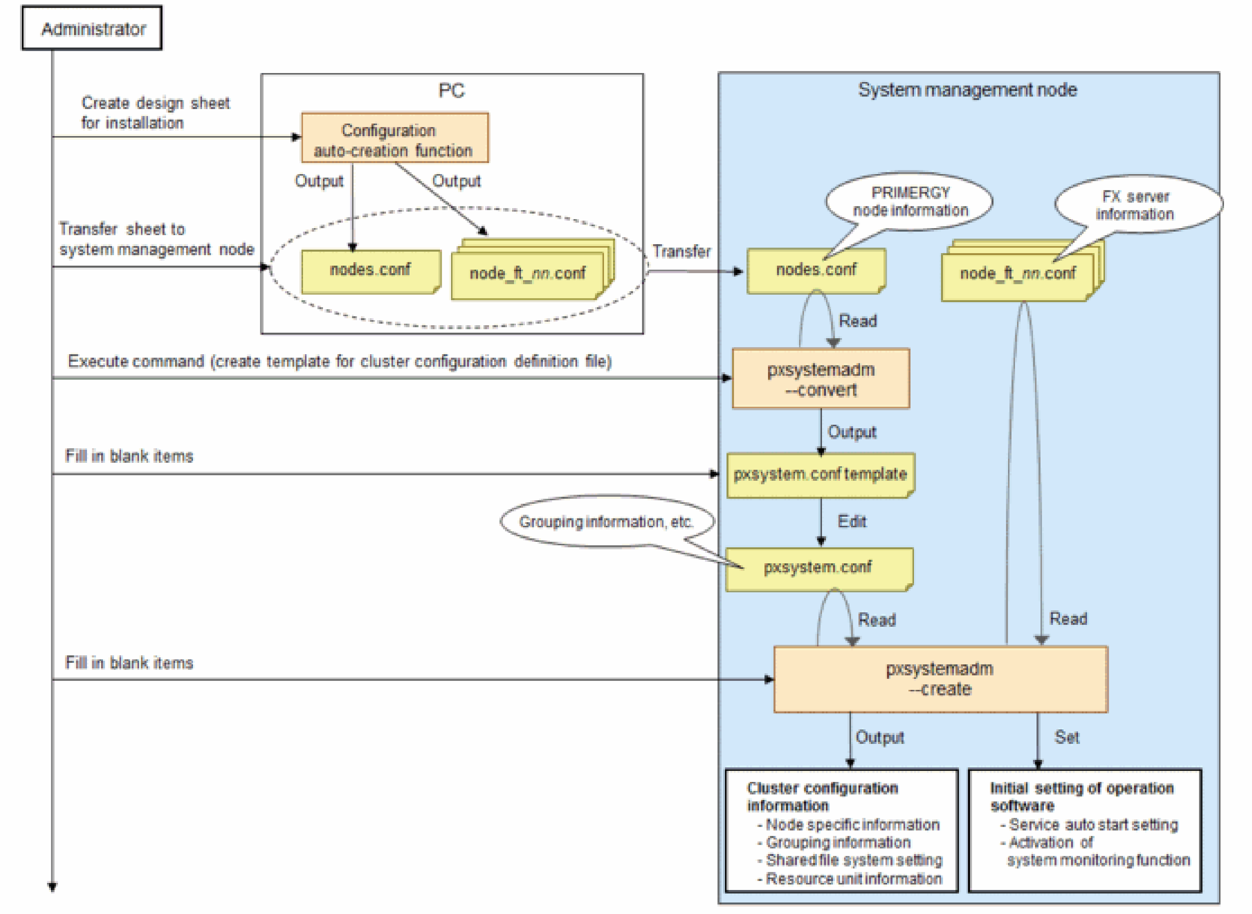
  (*) The node information definition file for FX server is divided by 27-rack unit (648 BoB = 10368 node units) when the file is output from the design sheet. "*nn*" indicates the sequential number of a divided file, starting from 01.

- Cluster configuration definition file: pxsystem.conf

  The system administrator writes information to divide the nodes in the system into clusters and node groups, and the node specific information, including IP address, to the pxsystem.conf file.
  The system administrator can create a pxsystem.conf template using nodes.conf, which is the output file of a design sheet for installation, and the --convert option of the pxsystemadm command.

Information, such as the IP addresses of the nodes, is described in the template beforehand. The system administrator can easily create a pxsystem.conf file by filling in the blank items in the template.

Figure 2.2 Conceptual Diagram of Registering the Cluster Configuration Information


Administrator
PC
System management node
Create design sheet for installation
Configuration auto-creation function
Output
Output
Transfer sheet to system management node
nodes.conf
node_ft_nn.conf
Transfer
PRIMERGY node information
FX server information
nodes.conf
node_ft_nn.conf
Read
Execute command (create template for cluster configuration definition file)
pxsystemadm --convert
Output
Fill in blank items
pxsystem.conf template
Grouping information, etc.
Edit
pxsystem.conf
Read
Read
Fill in blank items
pxsystemadm --create
Output
Set
Cluster configuration information
- Node specific information
- Grouping information
- Shared file system setting
- Resource unit information
Initial setting of operation software
- Service auto start setting
- Activation of system monitoring function

See

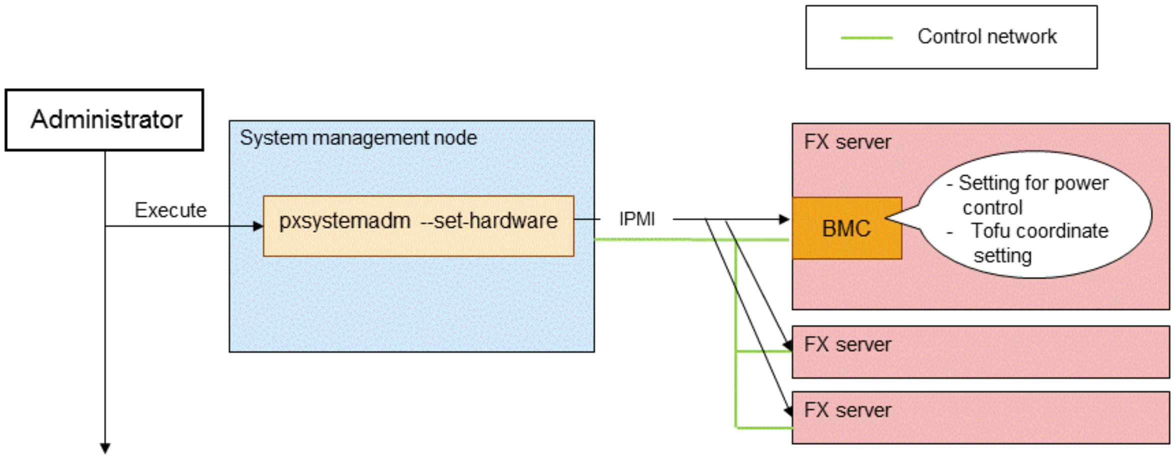
- For details on the design sheet for installation, see "3.8.3 Creating Settings of the Installation Target Node."
- For details on the procedure to register the cluster configuration information, see "3.1.2 Registering Cluster Configuration Information."

2.1.2 Overview of FX server Control Setting

The system administrator configures the FX server control setting by executing the pxsystemadm command after specifying the --set-hardware option to the command.

The pxsystemadm command configures the power control setting and the Tofu coordinates for the BMC (Baseboard Management Controller) of FX server via IPMI (Intelligent Platform Management Interface) communication through the control network.

Figure 2.3 Conceptual Diagram of FX server Control Setting



See

For details on the procedure to configure the FX server control setting, see "3.1.3 Control Setting for FX server."

2.1.3 Overview of Applying the Cluster Configuration Information

The cluster configuration, which is defined in the active system management node using the --set option of the pxsystemadm command, is applied to every node in the cluster. A process to apply the cluster configuration information differs between a target node in PRIMERGY server and a target node in FX server.

When a target node is PRIMERGY server, the pxsystemadm command instructs the target node to set the cluster configuration information using the batch operation function. The node retrieves the cluster configuration information on the system management node via HTTP communication. Then the Job Operation Software package is applied and the initial setting is made on the node.

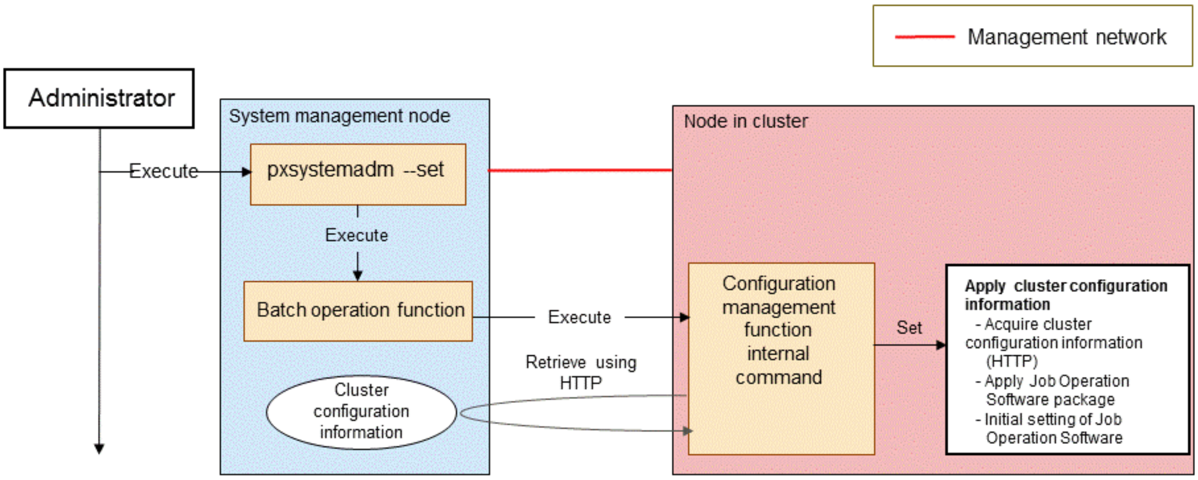
Information

When applying the Job Operation Software package, a repository set with the installation function is used. For details on the repository management using the installation function, see "2.8.5 Managing a Repository."

See

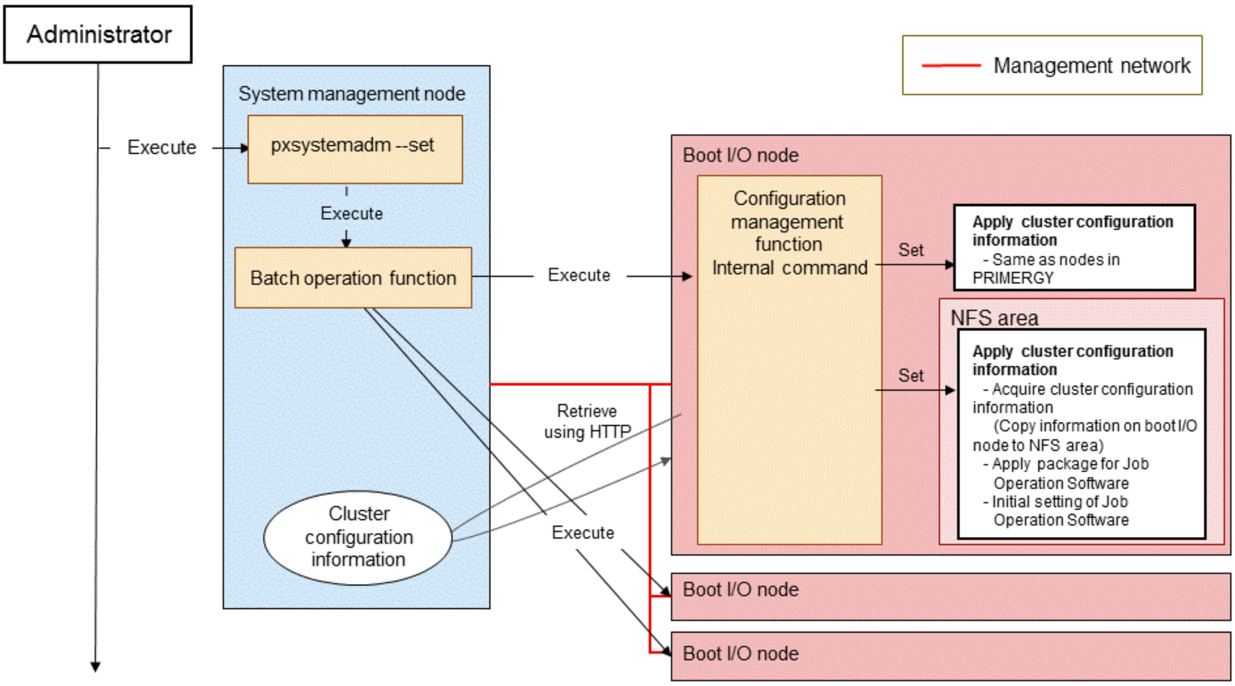
For an explanation of the batch operation function, see "2.5 Overview of the Operation Support Function."

Figure 2.4 Conceptual Diagram of Applying the Cluster Configuration Information (PRIMERGY server)


Management network
Administrator
Execute
System management node
pxsystemadm --set
Execute
Batch operation function
Cluster configuration information
Execute
Retrieve using HTTP
Node in cluster
Configuration management function internal command
Set
Apply cluster configuration information
- Acquire cluster configuration information (HTTP)
- Apply Job Operation Software package
- Initial setting of Job Operation Software

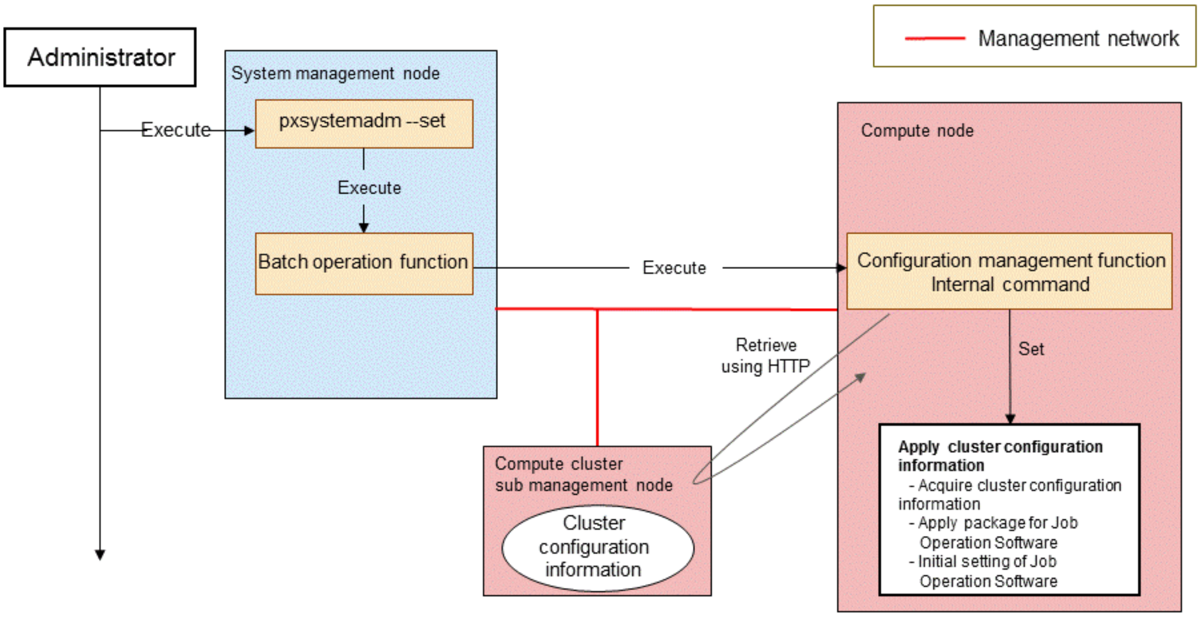
The same setting for the PRIMERGY server is configured when a target node is the boot I/O node of FX server.
If a target node is a diskless node (storage I/O node, global I/O node, or compute node), the cluster configuration information is applied to the NFS area, where the diskless node is mounted, on the boot I/O. Then the cluster configuration information can be applied to the diskless node when the boot I/O node is activated even if the diskless node is stopped.

Figure 2.5 Conceptual Diagram of Applying the Cluster Configuration Information (FX server)


Administrator
Management network
System management node
Execute
pxsystemadm --set
Execute
Batch operation function
Cluster configuration information
Execute
Retrieve using HTTP
Execute
Boot I/O node
Configuration management function Internal command
Set
Apply cluster configuration information
- Same as nodes in PRIMERGY
NFS area
Set
Apply cluster configuration information
- Acquire cluster configuration information
(Copy information on boot I/O node to NFS area)
- Apply package for Job Operation Software
- Initial setting of Job Operation Software
Boot I/O node
Boot I/O node

In the configuration with the compute cluster sub management node, the cluster configuration information is acquired from the compute cluster sub management node if the target node is a compute node in the node group.

Figure 2.6 Applying the Cluster Configuration Information Using the Compute Cluster Sub Management Node


Administrator
Execute
System management node
pxsystemadm --set
Execute
Batch operation function
Management network
Compute node
Execute
Configuration management function
Internal command
Retrieve using HTTP
Set
Compute cluster sub management node
Cluster configuration information
Apply cluster configuration information
- Acquire cluster configuration information
- Apply package for Job Operation Software
- Initial setting of Job Operation Software

See

For details on the procedure to apply the cluster configuration information, see "3.1.4 Applying the Cluster Configuration Definition."

2.2 Overview of the System Control Function

The system administrator uses the system control function to control the node power and the location LED.
The following functions are included in the system control function.

- Setting the system control function

  The system administrator sets the necessary IPMI authentication information to use the system control function, and also sets a parameter (the default simultaneous execution number) that is used when controlling the power. Once these settings are completed, the system control function can be used.
  In an environment, such as a virtual environment where a node cannot be controlled using IPMI, the system administrator can control the power by describing the power control operation in a plug-in.

- Power control

  During system installation or maintenance, the power may be controlled by various units, such as a cluster unit and a node unit. In such a case, the power needs to be controlled considering the power consumption, node configuration, and job status. The system administrator can safely control the powers of the nodes in the system using the system control function.

  - Power control by node or boot group

    The nodes can start and stop collectively by specifying nodes or a boot group. In addition, the power status of each node can be displayed. If there is an active job in a node to be stopped, the node is stopped by default after waiting until the job is completed.
    To control the power supplies of multiple nodes, the power supplies are controlled by a given number of nodes considering the drastic change in the power. The number is defined as the simultaneous execution number for controlling the power supplies. The system administrator can change this number anytime using a command parameter or plug-in.

  - Batch control of the system power

    To use this function when the entire system or cluster is required to start or stop at the same time, the power supplies of the nodes in a cluster can be controlled in an appropriate order.
    The entire system, including multi-purpose nodes, can be simultaneously controlled by performing a process before or after the batch power control using the pre/post plug-in.

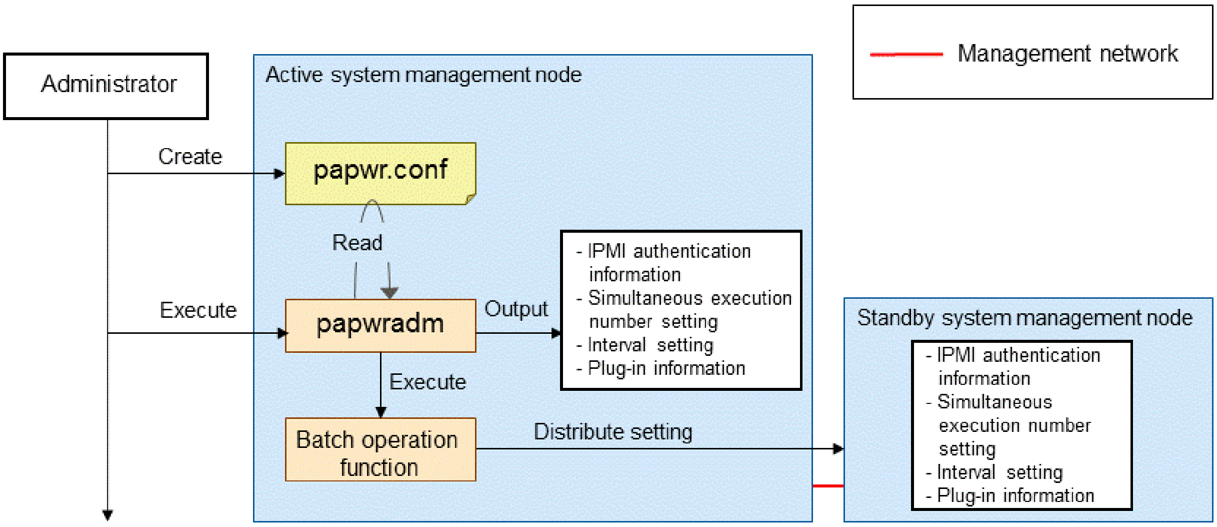
- Controlling the location LED

  A location LED on FX server lights up to notify which node is failed among many nodes to the operator (CE) when the hardware of FX server is failed.

2.2.1 Overview of the System Control Function Settings

The system administrator applies the system control function settings to the system by describing necessary settings in papwr.conf, which is the system control function configuration file, and by executing the papwradm, which is a command to apply the system control setting, on the active system management node.
The papwradm command distributes the settings to the standby system management node.

Figure 2.7 Conceptual Diagram of the System Control Function Settings


Management network
Administrator
Active system management node
Create
papwr.conf
Read
Execute
papwradm
Output
- IPMI authentication information
- Simultaneous execution number setting
- Interval setting
- Plug-in information
Execute
Batch operation function
Distribute setting
Standby system management node
- IPMI authentication information
- Simultaneous execution number setting
- Interval setting
- Plug-in information

See

- For details on the procedure to set the system control function, see "3.2.1 Configuring the System Control Function."
- For details on the batch operation function, see "2.5 Overview of the Operation Support Function."

2.2.2 Overview of the Power Control Functions for the Nodes

The system administrator can obtain the power control status and the power status from a node in the cluster using the papwrctl command. The following power control operations are available.

Table 2.1 Power Control Operation Types

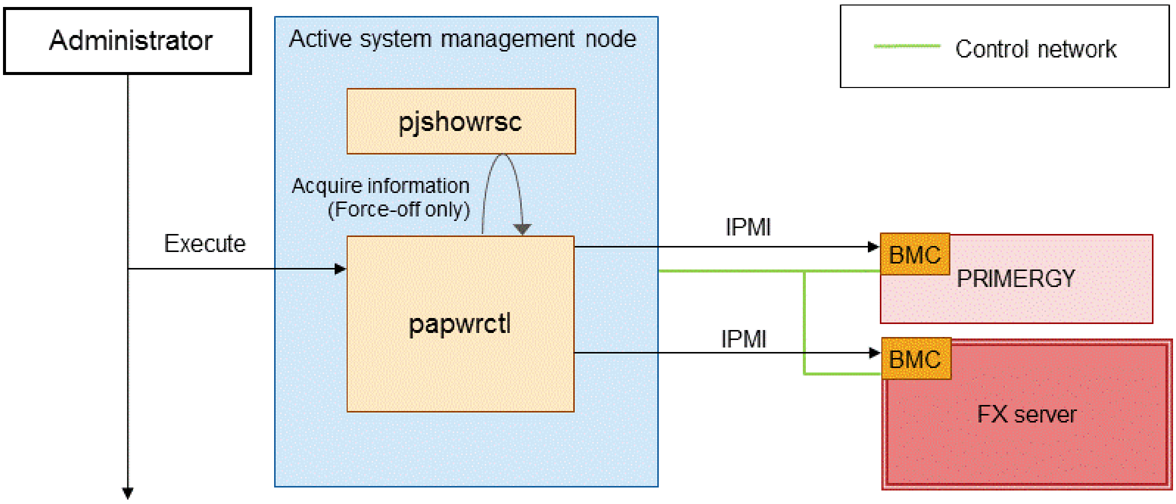
Parameter Name	Operation	Description
on	Power on	The OS is started after turning on the power of the node.
off	Power off	The node is powered off after shutting down the OS.
force-off	Forced power off	The node is immediately powered off regardless of the OS status.
cold-reboot	Reboot	The OS is started without powering off the nodes after the OS is shut down. (This operation cannot be specified in FX server.)
cycle	cycle	The OS is restarted after it is shut down forcibly. (This operation cannot be specified in FX server.)

Parameter Name	Operation	Description
reset	Reset	The OS is restarted by resetting the hardware.
init	INIT interrupt	An INIT interrupt is generated. If the dump function is set to the OS, a dump is taken by generating an INIT interrupt.
status	Status display	The power status is displayed.

A method to instruct the power control varies depending on the operation type and the target.

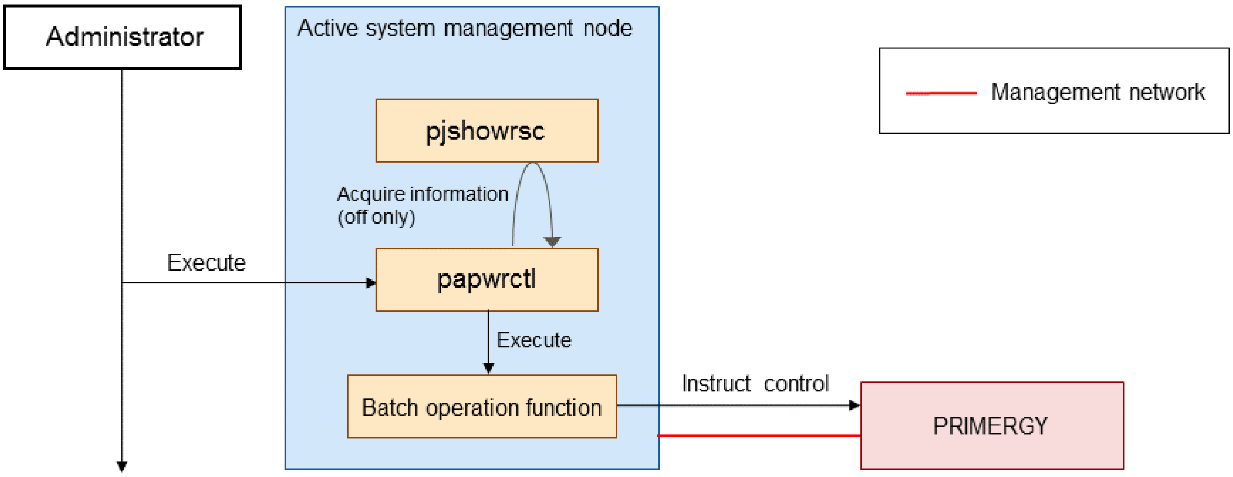
A control is instructed to the BMC via the control network using IPMI, when a control operation (on, force-off, cycle, reset, or init) is performed or the power status is acquired (status). The papwrctl command is executed to obtain job information from the pjshowrsc command in order to confirm whether a job is being processed in the node.

Figure 2.8 Conceptual Diagram of the Node Power Control Functions (On, Force-off, Cycle, Reset, Init, and Status)



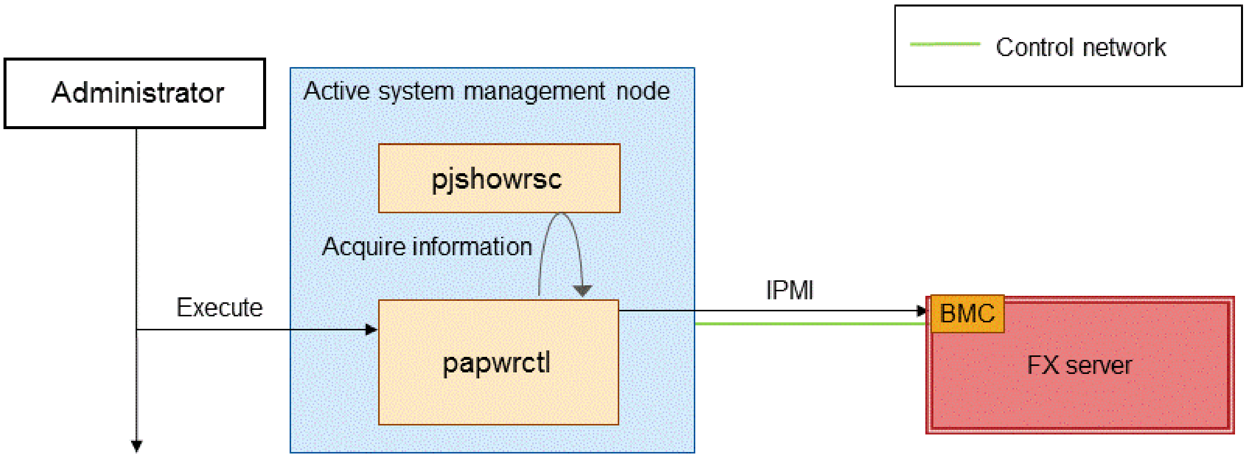
When a control operation (off or cold-reboot) is carried out on PRIMERGY server, the control instruction is made via the management network using the batch operation function (pmexe command).

Figure 2.9 Conceptual Diagram of the Node Power Control Functions for PRIMERGY server (Off and Cold-Reboot)



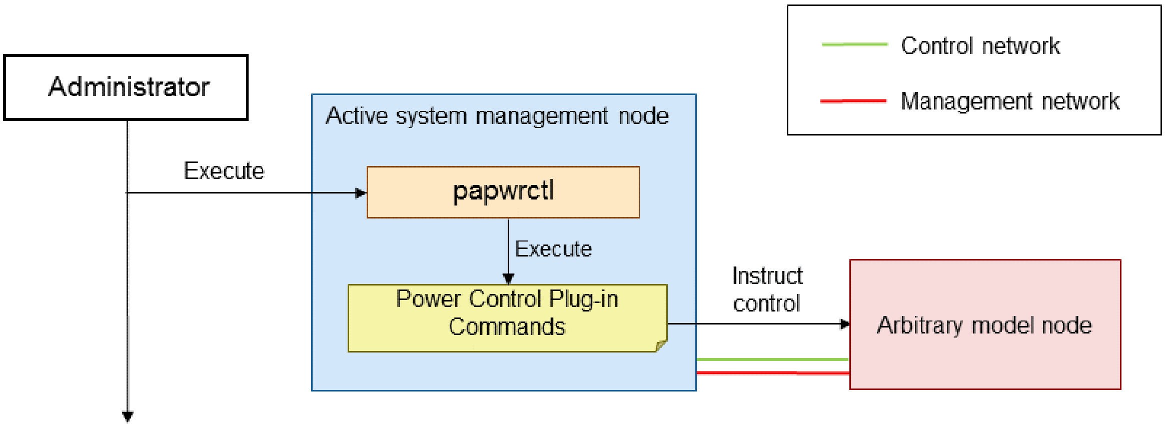
When a control operation (off) is carried out on FX server, the control instruction is made using the control network.

Figure 2.10 Conceptual Diagram of the Node Power Control Functions for FX server (Off)



A control instruction is made on any node by executing the power control plug-in command on the active system management node.

Figure 2.11 Conceptual Diagram of the Node Power Control Functions for Any Models



See

- For details on the procedure to control the node power, see "3.2.2 Power Control for Nodes."
- For details on the batch operation function, see "2.5 Overview of the Operation Support Function."
- For details on the power control plug-in command, see "3.2.1.3 Power Control Plugin."
- For details on the pjshowrsc command, see "Checking the Job Status" in the "Job Operation Software End-user's Guide."
- When the power supplies of multiple nodes are controlled, the power can be controlled by a given number of nodes in a specified range by assigning the simultaneous execution number and the necessary interval. For details on the number of nodes whose power supplies can be controlled simultaneously and the interval, see "3.2.2.1 The Number of Concurrent Executions and the Interval." You can also dynamically change the number and the interval. For details on how to dynamically change the number and the interval, see "3.2.2.2 Interval Plugin."

Note

The papwrctl command waits until the jobs are completed or the nodes are not used as job routes and executes the power control operation, if there are active jobs in the nodes that are the target of the power control or if the nodes that are the target of the power control are used as job routes. To prevent the papwrctl command from waiting, specify the --jobnowait option.

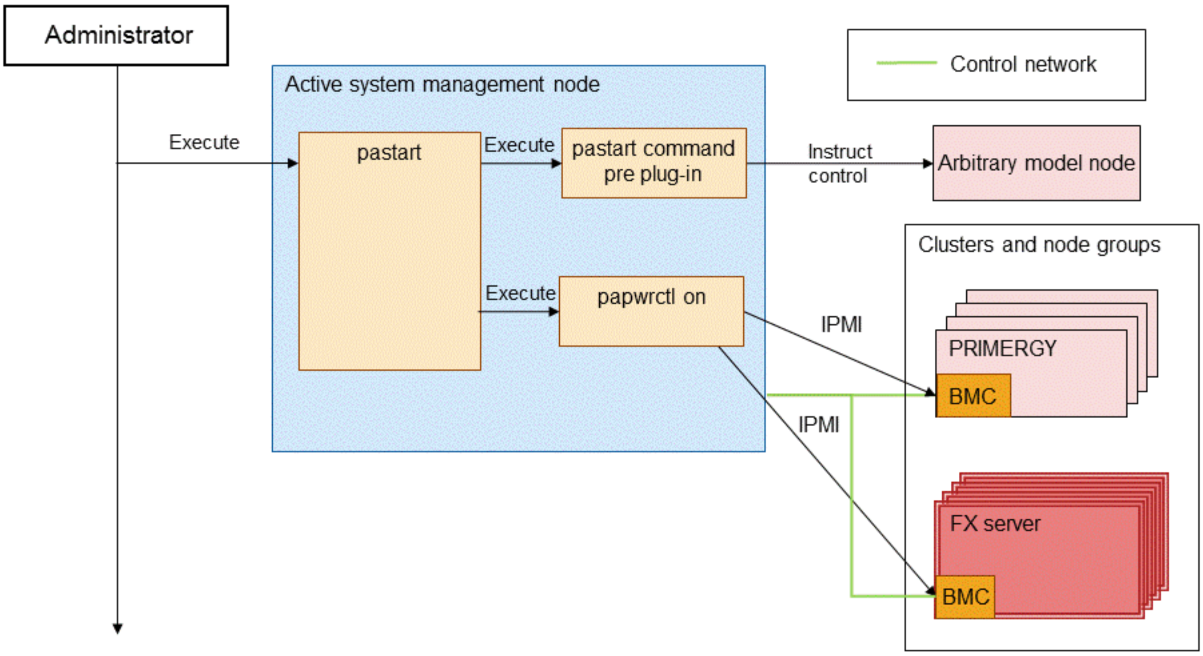
2.2.3 Overview of the Batch Power Control Function in the System

When the system is started or shut down, the nodes must be started or shut down in an appropriate order. Therefore, the system needs to be started or shut down by calling the power control function of each node in a proper order.

However, if the system administrator uses the pastart command and pastop command, the administrator can start or shut down the system without taking into account the start-up or shutdown order.

- Batch power-on function (pastart command)
  This function is used to power on the specified clusters and node groups.
  The diagram below is the conceptual diagram showing the power-on of the device before the batch power-on operation is carried out on the PRIMERGY server and FX server using the plug-in of the pastart command.
  The examples of the devices whose powers are turned on before performing the batch power-on operation are storage devices and host nodes of KVM that is a virtual environment on Linux. The system administrator can define such devices as any nodes using the configuration management function.

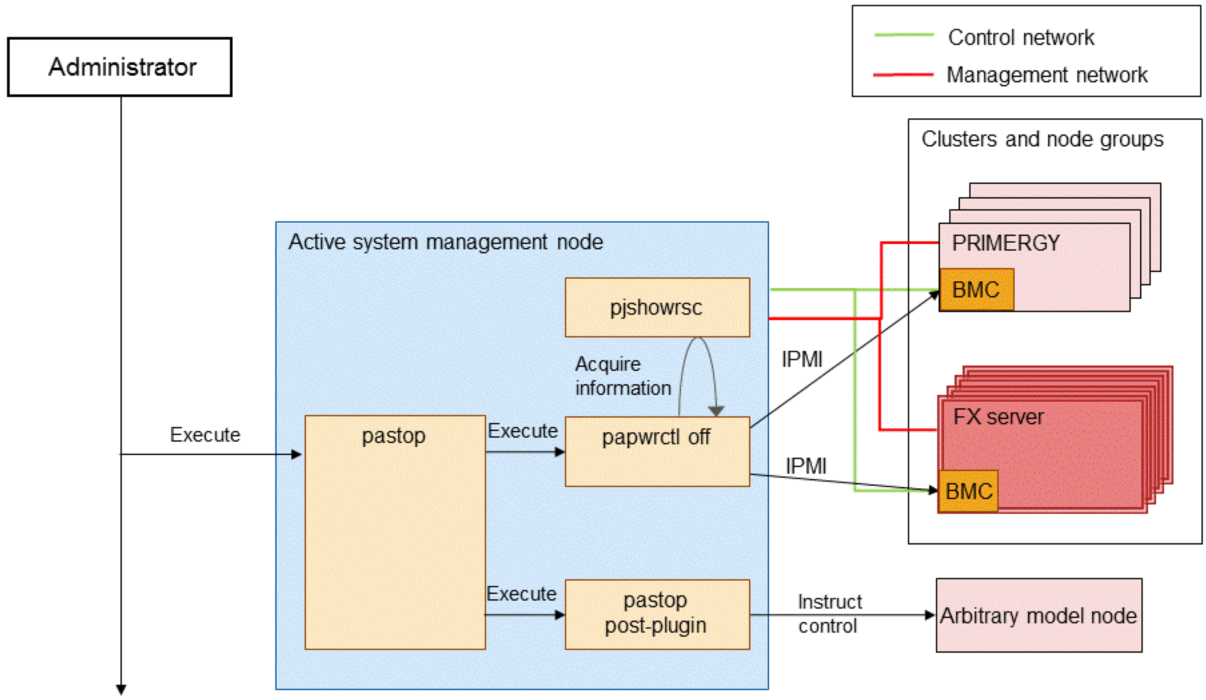
  Figure 2.12 Conceptual Diagram of the Batch Power-On Function



- Batch power-off function (pastop command)
  This function stops the power supplies of specified clusters or node groups. The system administrator can specify whether or not to wait until a job is complete before the power is turned off.
  The diagram below is the conceptual diagram showing the power-off of the device after the batch power-off operation is carried out on the PRIMERGY server and FX server using the post plug-in of the pastart command.
  The example of the devices whose power supplies are turned off after performing the batch power-off operation are storage devices and

host nodes of KVM that is a virtual environment on Linux. The system administrator can define such devices as any nodes using the configuration management function.

Figure 2.13 Conceptual Diagram of the Batch Power-Off Function



See

- For details on the procedure to batch control the system power, see "3.2.3 Collective Power Control for the System."
- For details on the operation when "off" is specified with the papwrctl command, see "2.2.2 Overview of the Power Control Functions for the Nodes."
- The pre plug-in and post plug-in can be defined before and after the batch control of the system power. For details on plug-in, see "3.2.3.2 Pre-post Plugin."
- When the power supplies of multiple nodes are controlled, the power supplies of the nodes in a specified range can be controlled gradually by setting the number of nodes whose power supplies can be controlled simultaneously and the interval. For details on the number of nodes whose power supplies can be controlled simultaneously and the interval, see "3.2.2.1 The Number of Concurrent Executions and the Interval." You can also dynamically change the number and the interval. For details on how to dynamically change the number and the interval, see "3.2.2.2 Interval Plugin."

2.2.4 Overview of Controlling the Location LED

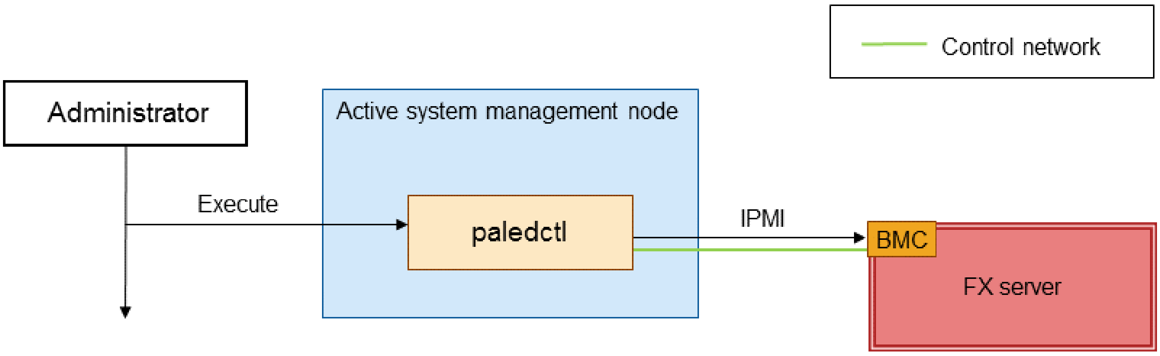
The system administrator can turn on or off the location LED of FX server using the paledctl command. The system administrator can also reference the LED status.

Table 2.2 Location LED Control Types

Parameter Name	Operation	Description
on	LED on	Turns the location LED on.
off	LED off	Turns the location LED off.
show	LED state display	Displays the location LED status.

The LED control instruction (on or off) and the LED status (show) can be acquired via the control network using IPMI.

Figure 2.14 Conceptual Diagram of Controlling the Location LED



See

For details on the procedure to control the location LED, see "3.2.4 Controlling the Location LED."

2.3 Overview of the System Monitoring Function

In a system that is configured with multiple nodes, a node where a hardware or software failure has occurred is efficiently detected and the failed hardware or software is immediately isolated from a job operation so that the failure does not affect a job operation.

The system monitoring function regularly monitors the entire system to check the hardware status or the status of the Job Operation Software service to detect an abnormal node in the system. This function also isolates the abnormal node from job operation.

The system monitoring function can be used for the following operations.

- Displaying the system information

  The system configuration information and the system operation status can be displayed.
  The system administrator can use this function to view the system status list in order to know the current system status during the installation, operation or maintenance.

- Setting the system monitoring

  The settings to monitor the system can be changed.
  The system administrator can change the setting of a timeout value for a monitoring cycle or abnormality detection based on the system operation.

- Setting the system monitoring plug-in

  The settings of a plug-in that runs when detecting an abnormality during the system monitoring or the setting to specify the default operation of the system monitoring function that is used when detecting an abnormality can be changed.
  The system administrator can add or set an operation when an abnormality occurs in the system.

- Setting the user-defined services

  Services other than the Job Operation Software service can be monitored and the monitored result can be displayed with the system operation status.
  The system administrator can detect an abnormality in an important service other than the Job Operation Software service and also monitor the nodes in the system more specifically.

- Setting an abnormality notification

  A monitoring result that is obtained by an outside mechanism other than the Job Operation Software can be notified as abnormal to update the system monitoring function to apply the result.
  By doing so, the abnormal nodes, which are detected using an external monitoring mechanism prepared by the system administrator, can be operated in the same way as the Job Operation Software, such as isolating them from the job operation.

2.3.1 Overview of Displaying the System Information

The system monitoring function can display the system "operation status" and "configuration information."
The following describes the details of the system information to be displayed.

- Displaying the system operation status

  The system operation status (job operation status, power status, service status, and Tofu route status (FX server only)) can be displayed. The system administrator can check the system status during the installation and maintenance as well as at job completion.

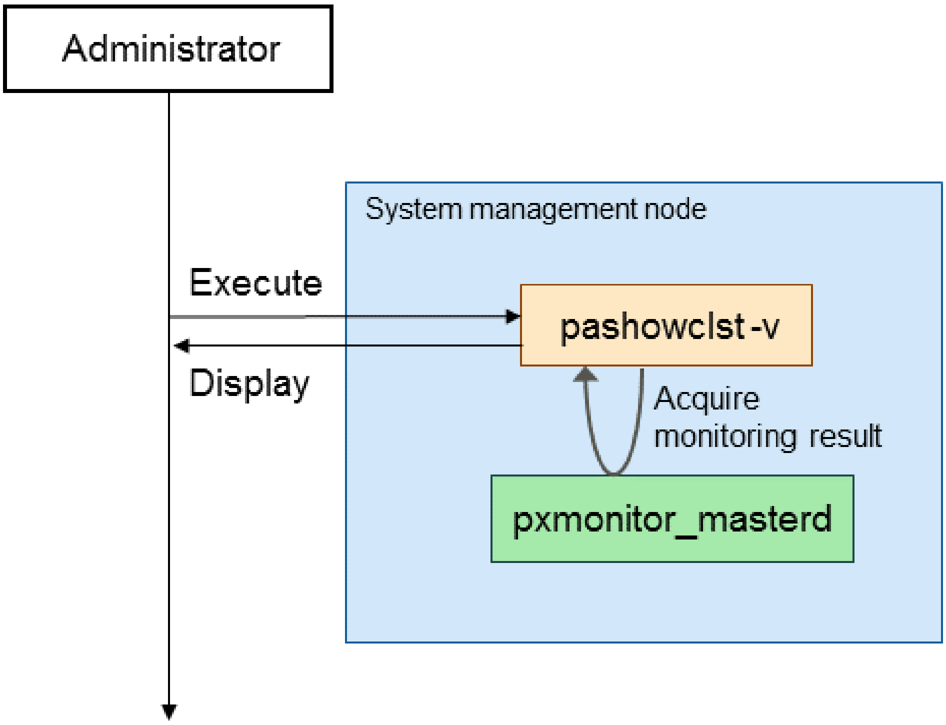
- Displaying the system configuration information

  The configuration information (grouping, node ID, IP address, Tofu coordinates, etc.), which is organized using the configuration management function during the installation, can be displayed.
  The system administrator can view the list to check whether the information set during the system design and the configuration information organized during the installation are correct.

2.3.1.1 Overview of Displaying the System Operation Status

The system administrator can check the monitoring result obtained from the pxmonitor_masterd daemon on the system management node with the pashowclst command. Detailed monitoring result can be displayed by specifying the -v option to the pashowclst command.

Figure 2.15 Conceptual Diagram of Displaying the System Operation Status



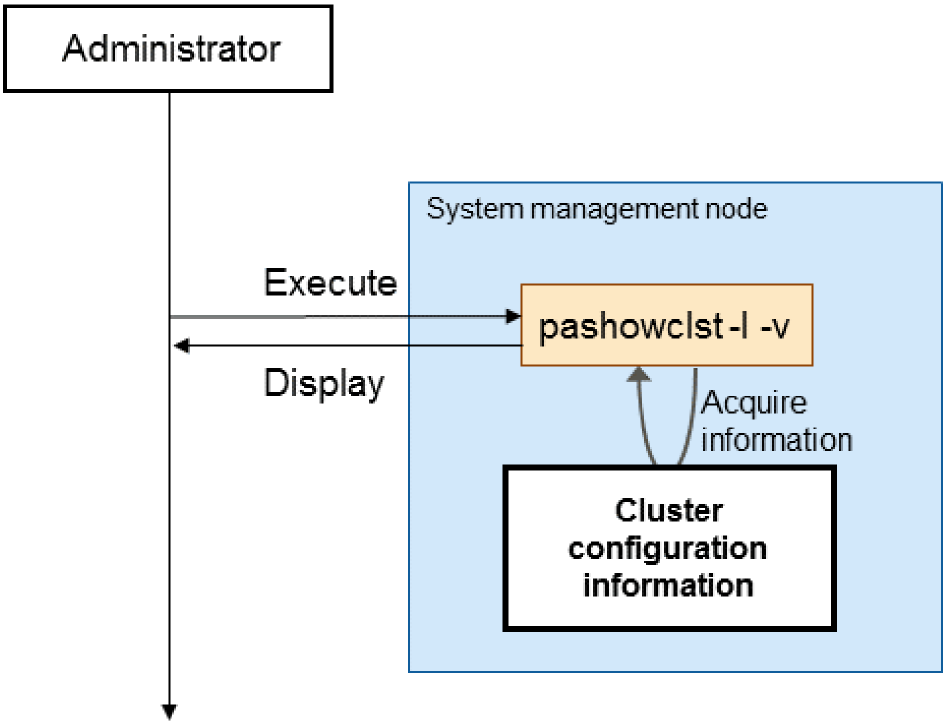
See

For details on the procedure to display the system operation status, see "3.3.2 Displaying Operation Status of the System."

2.3.1.2 Overview of Displaying the System Configuration Information

The system administrator can check the cluster configuration information organized using the configuration management function by specifying the -l option (or -v option when displaying the details) to the pashowclst command.

Figure 2.16 Conceptual Diagram of Displaying the System Configuration Information



See

- For details on the configuration management function, see "2.1 Overview of Configuration Management Function."
- For details on the procedure to display the system configuration information, see "3.3.3 Displaying System Configuration Information."

2.3.2 Overview of Setting the System Monitoring

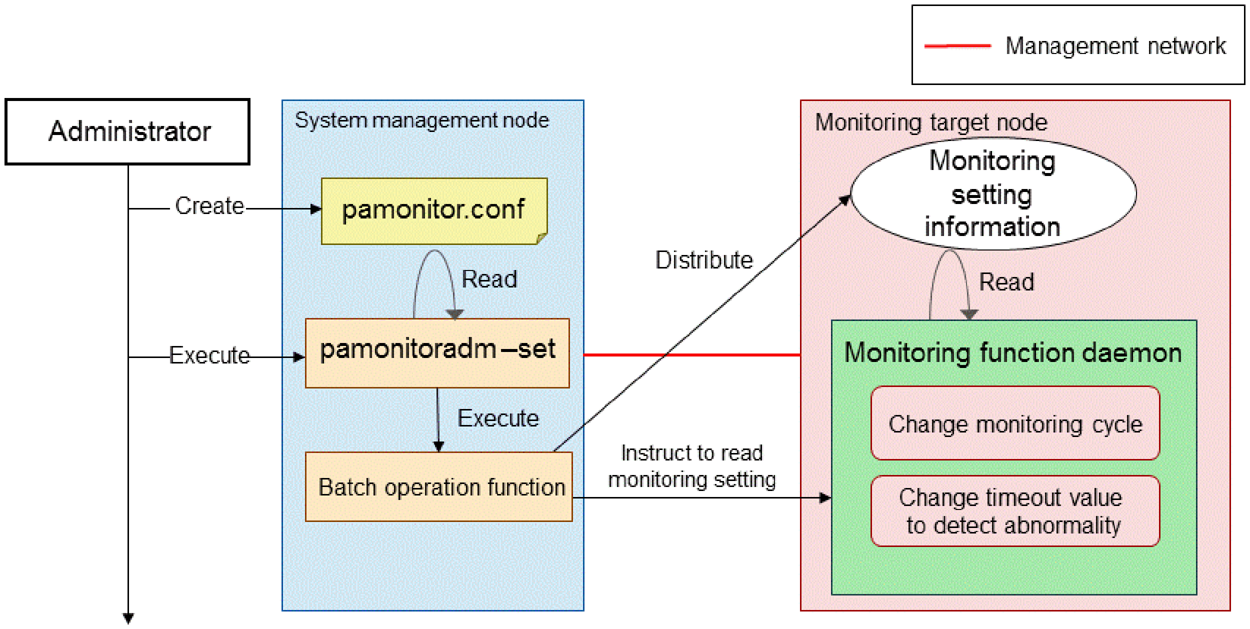
The system administrator can make the system monitoring setting by specifying the --set option to the pamonitoradm command.
The pamonitoradm command applies the changes in the system monitoring setting to all nodes in the system via the management network (compute (Tofu) network for FX server).

The pamonitoradm command uses the following file as input information.

- System monitoring configuration file, pamonitor.conf

  The system administrator describes information, such as a timeout value for regular monitoring cycle or timeout value for abnormality detection, in the pamonitor.conf file.

Figure 2.17 Conceptual Diagram of Setting the System Monitoring



See

For details on the procedure to set the system monitoring, see "3.3.4 Configuring the System Monitoring."

2.3.3 Overview of Setting the System Monitoring Plug-In

The system administrator makes the system monitoring plug-in setting by executing the pamopluginadm command after specifying the --set option to the command.

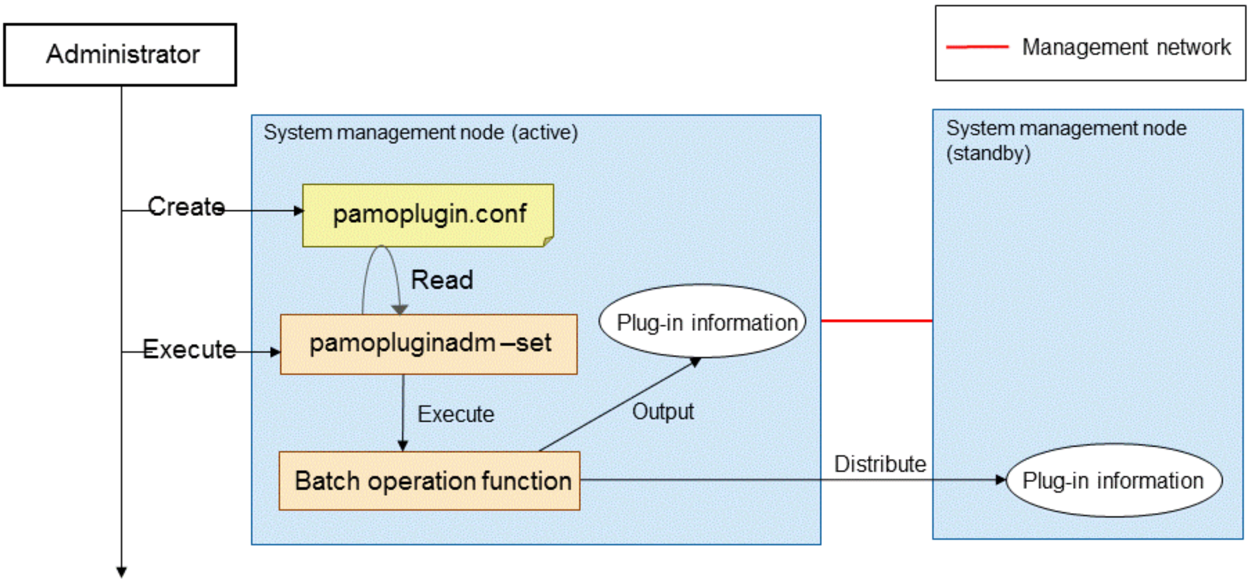
The setting is applied to the active and standby system management nodes.

The pamopluginadm command uses the following file as input information.

- System monitoring plug-in configuration file, pamoplugin.conf

  The system administrator describes information of a plug-in, which operates when detecting an abnormality, and information of valid or invalid default operation.

  Figure 2.18 Conceptual Diagram of Setting the System Monitoring Plug-In



See

For details on the procedure to set the system monitoring plug-in, see "3.3.5 Settings of the System Monitoring Plugin."

2.3.4 Overview of Setting the User-Defined Services

The system administrator sets a user-defined service by executing the paserviceadm command after specifying the --set option to the command.

The paserviceadm command applies the changes in the user-defined service to all nodes in the system via the management network (Tofu network for FX server).

The system administrator prepares the following files for the settings that are configured using the paserviceadm command and for monitoring the user-defined service after the settings.

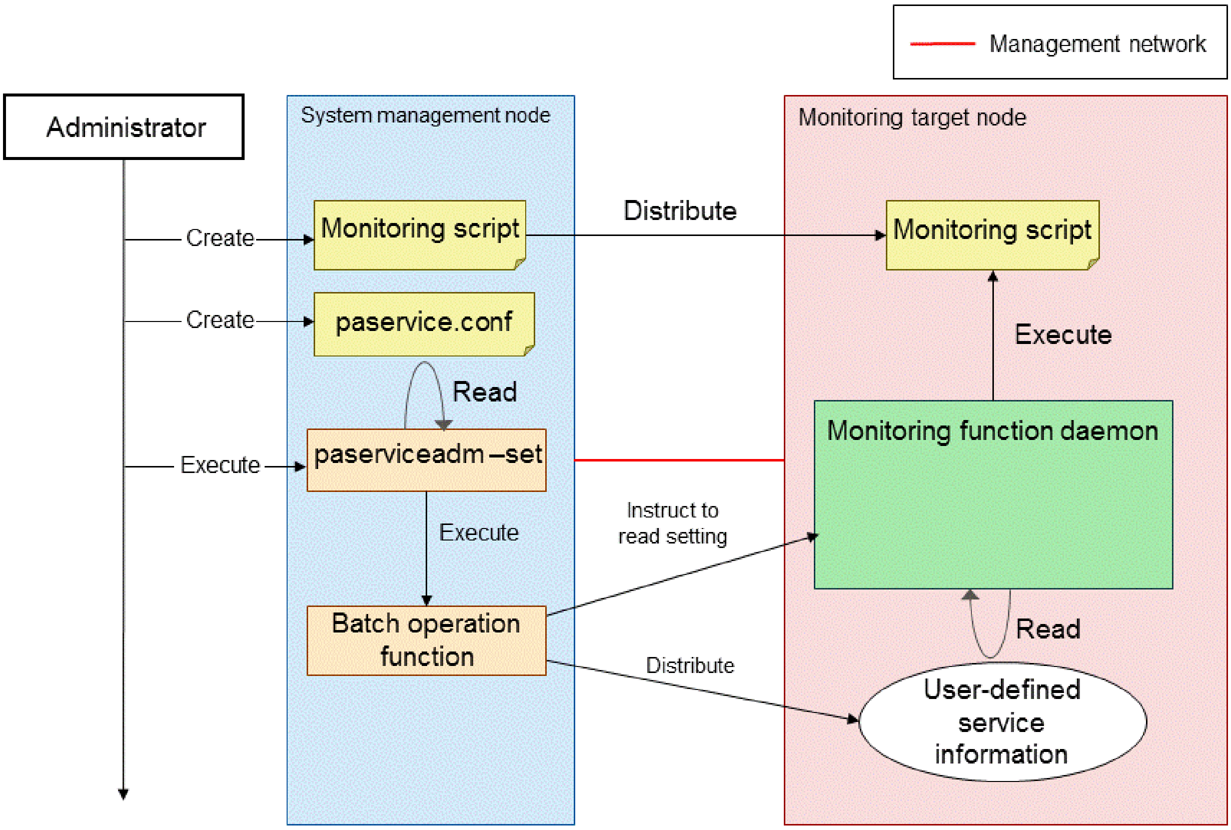
- Monitoring script

  This is a script to monitor the services that are actually running on the monitored nodes.
  The system administrator creates a monitoring script on the system management node and distributes it to the directories on the target nodes.

- User-defined service configuration file, paservice.conf

  The system administrator describes the following information in the paservice.conf file: the convenient names of the services to be newly monitored (the service names that are displayed with the pashowclst command), the nodes to be monitored, and the file path information of the monitoring scripts located on the monitoring nodes.

Figure 2.19 Overview of Setting the User-Defined Services



See

For details on the procedure to set user-defined services, see "3.3.6 Settings of the User-Defined Service."

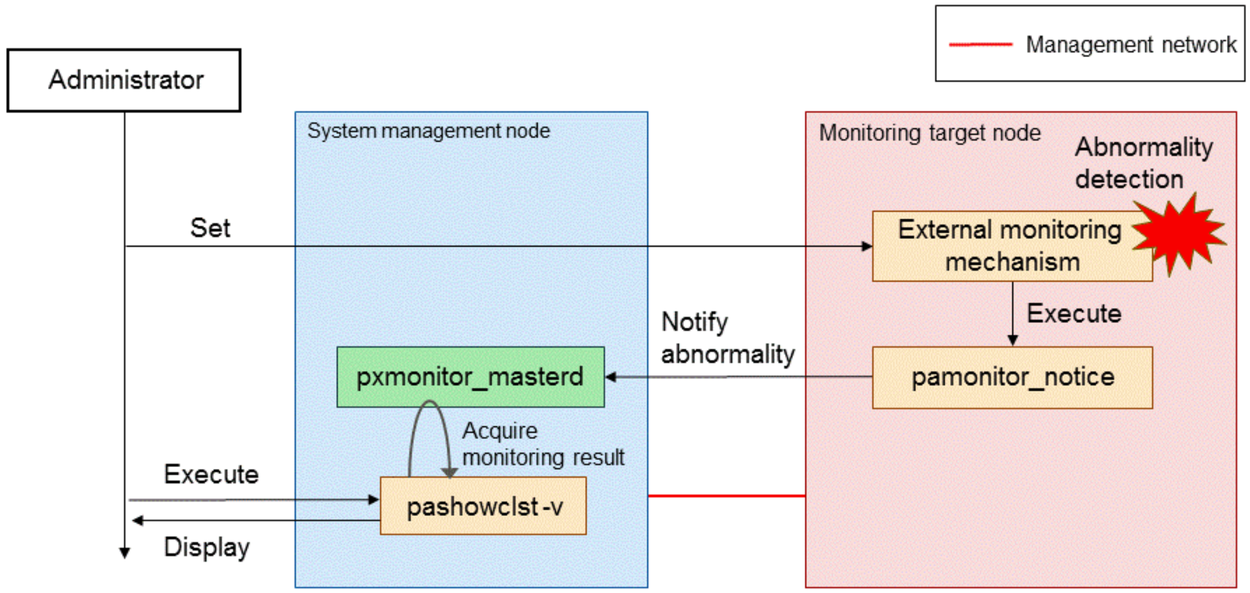
2.3.5 Overview of Setting the Abnormality Notification

If an abnormality occurs in a node, it is notified to the system monitoring function by executing the pamonitor_notice command using a script on the node or an external monitoring mechanism.

The pamonitor_notice command notifies the abnormality to the pxmonitor_masterd daemon on the system management node via the management network (Tofu network for FX server).

Detailed information can be viewed using the --detail option of the pamonitor_notice command, or by executing the pashowclst command after specifying the --detail option.

Figure 2.20 Conceptual Diagram of Setting the Abnormality Notification



See

- When executing the pamonitor_notice command, the pxmonitor_masterd daemon handles the node as an abnormal node. For details on the operation when an abnormal condition is detected in each node, see "Table 3.27 Operations for Each Node Type when Detecting Abnormality" in "3.3.1.1 Software Abnormality Monitoring."
- For details on the procedure to display the system operation status, see "3.3.2 Displaying Operation Status of the System."
- For details on the procedure to set abnormality notification, see "3.3.7 Setting of Abnormality Notifications."

2.4 Overview of the System Maintenance Function

The system administrator maintains the system using the system maintenance function to safely maintain the system after making the nodes serviceable. After the maintenance work is completed, the system administrator recovers the nodes from the serviceable state to the normal state and starts operating the system.
The system administrator performs these operations using the paclstmgr command.

- Node isolation and node installation

  "Node isolation" is to prevent a job from being assigned to a node when the node has failed or is being maintained. "Node installation" is to include the isolated node in the operation again. A job is not assigned to the isolated node.

- Switching a node

  If redundancy is provided to a node by supporting the active and standby nodes, switching between the active and standby nodes is called "failover."
  To maintain a node in redundant configuration, the system administrator can maintain the node without influencing on the system in use by switching the node from active to standby.

- Switching to and recovering from the maintenance mode

  To maintain a node, the system administrator needs to change the node to "maintenance mode."
  Although node isolation is to temporarily exclude it from the operation (a job is not assigned), in the maintenance mode, the system administrator puts the node into a safely serviceable state by ensuring that the Job Operation Software function will not operate in an unreasonable manner.
  There are the following two types of maintenance modes: a hardware maintenance mode to replace a hardware and a software maintenance mode to update software.
  The system administrator switches the node to be maintained to the appropriate maintenance mode for the maintenance work to be done, and recovers the node after the maintenance work is completed.

- Operating the Job Operation Software service

  To operate a Job Operation Software service (start, stop, or restart) after applying a package, the system maintenance function can be used. The system administrator can collectively operate the Job Operation Software services using the system maintenance function.

- Setting the automatic failover

  The automatic failover starts operating when the system monitoring function detects a service abnormality in the active node in redundant configuration.
  The system administrator can change the automatic failover setting so that unnecessary failover is not executed when maintaining the nodes in redundant configuration or when restarting the entire system.

  See

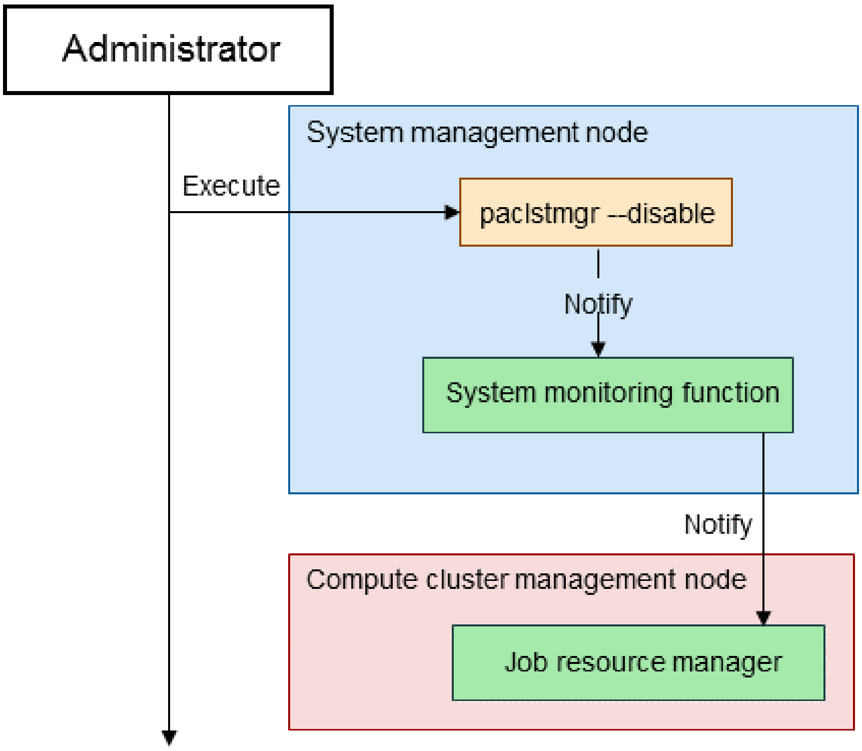
  For details on the operation when the system monitoring function detects an abnormality in a node, see "Table 3.27 Operations for Each Node Type when Detecting Abnormality" in "3.3.1.1 Software Abnormality Monitoring."

2.4.1 Overview of Isolating and Installing a Node

The system administrator executes the --disable option of the paclstmgr command on the system management node in order to isolate a node from operation.
The paclstmgr command notifies a node to be isolated to the job resource manager that is operating on the compute cluster management node through the system monitoring function. A job is not assigned to the isolated node when executing a job.

When installing the node, the node is notified in the same way as when isolating it.

Figure 2.21 Conceptual Diagram of Isolating and Installing a Node



See

For details on the isolation and installation of a node, see "3.4.1 Disconnecting and Incorporating a Node."

Information

When isolating or installing a node, the system administrator can add a necessary work before or after a node operation or service operation by creating a plug-in to which a process is described and specifying the plug-in to a paclstmgr command parameter. The system administrator can automate the process using the plug-in.

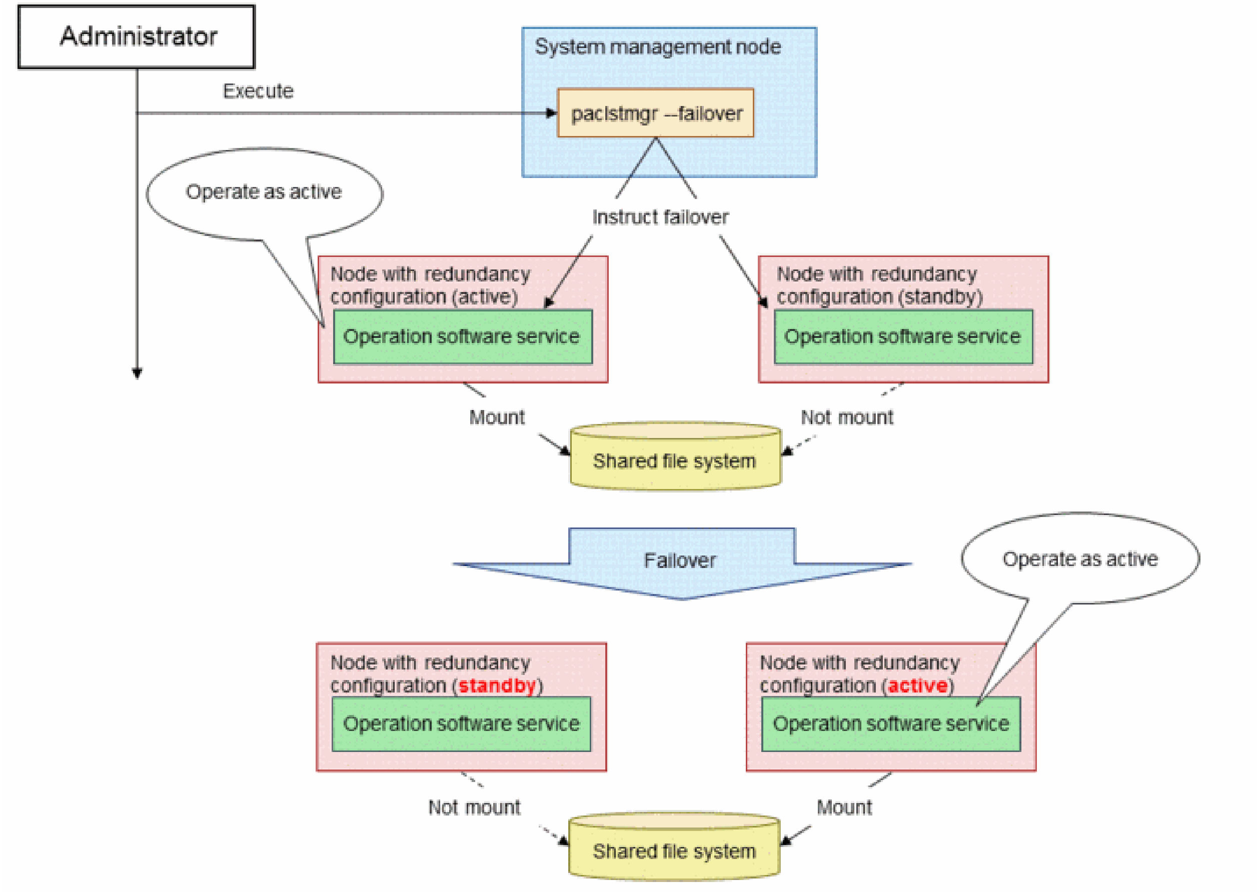
For details on the system maintenance function plug-in, see "3.4.6 System Maintenance Function Plugin."

2.4.2 Overview of Switching Nodes

The system administrator switches between the active and standby nodes by specifying the --failover option of the paclstmgr command. When a node has active/standby configuration to achieve redundancy, the active node is mounted on the shared file system and the Job Operation Software service acts as an active service. The standby node is not mounted on the shared file system and the Job Operation Software service acts as a standby service. When executing a failover, these node roles are switched.

Manually changing the switched node to the original state is called "node reset."

Figure 2.22 Conceptual Diagram of Switching Nodes


Administrator
Execute
System management node
paclstmgr --failover
Operate as active
Instruct failover
Node with redundancy configuration (active)
Operation software service
Node with redundancy configuration (standby)
Operation software service
Mount
Not mount
Shared file system
Failover
Operate as active
Node with redundancy configuration (standby)
Operation software service
Node with redundancy configuration (active)
Operation software service
Not mount
Mount
Shared file system

See

For details on the procedure to switch nodes, see "3.4.2 Switching a Node."

Information

- When switching nodes, the system administrator can add a necessary work before or after a node operation or service operation by creating a plug-in in which a process is described and specifying the plug-in to a paclstmgr command parameter. The system administrator can automate the process by using the plug-in. For details on the system maintenance function plug-in, see "3.4.6 System Maintenance Function Plugin."

- When the FEFSSR service is moved between the nodes in active/standby configuration, the paclstmgr command instructs the node to perform a failover as shown in the conceptual diagram above (Figure 2.22 Conceptual Diagram of Switching Nodes).
  For details on moving the FEFSSR service, see the "FEFS User's Guide".

2.4.3 Overview of Switching to and from the Maintenance Mode

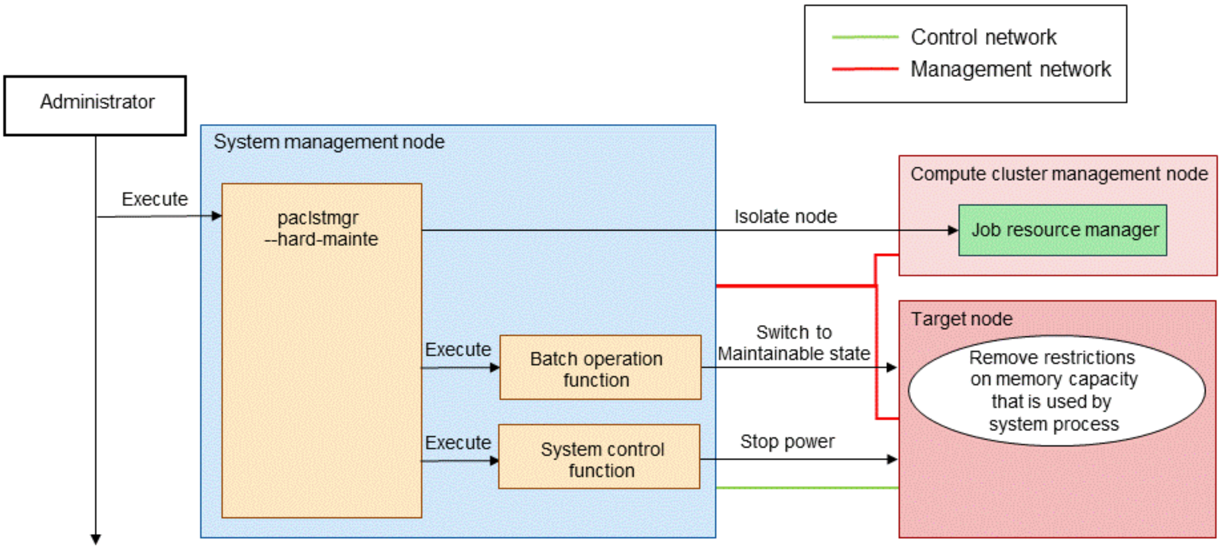
The system administrator switches a node to a maintenance mode when maintaining the system.
These are the following two types of maintenance modes: a hardware maintenance mode that needs to power off the node and a software maintenance mode that does not need to power off the node. The system administrator exits the node from the maintenance mode after completing the maintenance work.
The diagram below shows the conceptual image of the following operations.

- Switching to the hardware maintenance mode

  The system administrator switches a node to the hardware maintenance mode by specifying the --hard-mainte option to the paclstmgr command. The system administrator performs the following operation on the node to switch to the hardware maintenance mode.

  - The administrator isolates the node so that a job is not assigned to the node.
  - The memory capacity for processes other than jobs is unrestricted using the batch operation function so that the operations of the tools and commands that are executed during the maintenance are not interrupted.
  - The node is powered off using the system control function to safely replace the parts or perform other operations.

Figure 2.23 Overview of Switching to the Hardware Maintenance Mode



See

The power supplies of the nodes are disconnected based on the specifications of the system control function. Therefore, a CMU (CPU Memory Unit) and a BoB (Bunch of Blade) are powered off depending on the specified range when FX server is specified. For details on the system control function, see "2.2 Overview of the System Control Function."

When a node, such as the storage I/O node of FX server, is maintained, powering off the node may affect the job operation of the related nodes. In such a case, the related nodes, which may be affected by disconnecting the power supply of the node to be maintained, are also powered off when switching to the hardware maintenance mode. For details, see "3.4.3.2 Shifting to the Hardware Maintenance Mode."

- Switching to the software maintenance mode

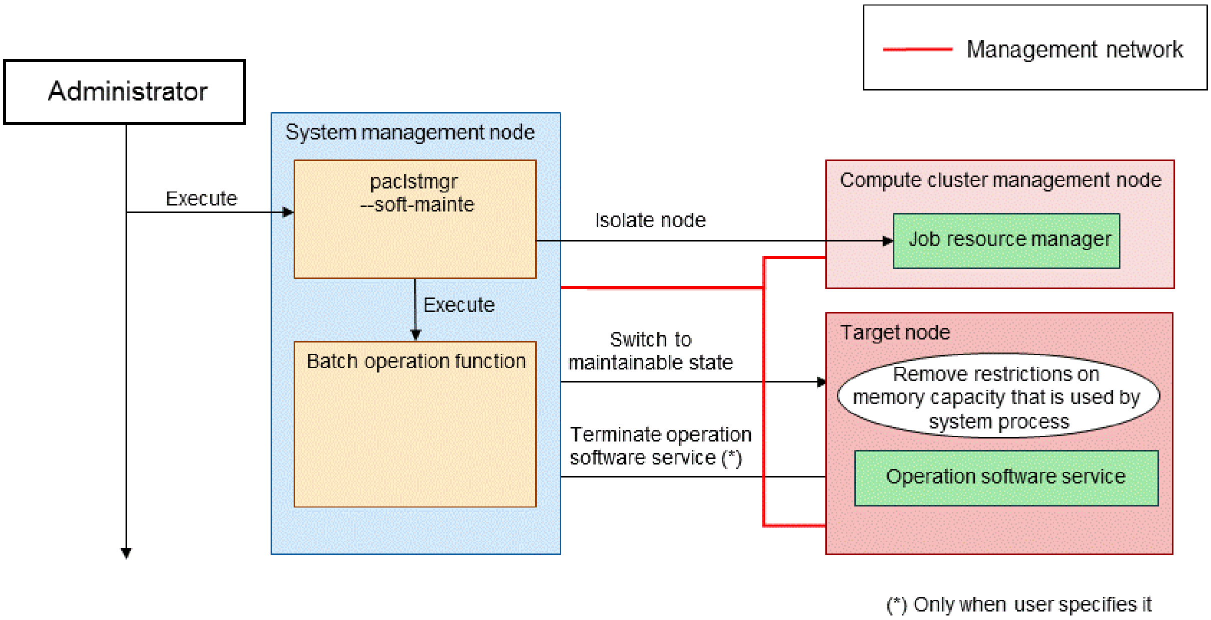
  The system administrator switches the node to the software maintenance mode by specifying the --soft-mainte option of the paclstmgr command. The system administrator performs the following operation on the node to switch to the software maintenance mode.

  The administrator isolates the node so that a job is not assigned to the node.

  - The memory capacity for processes other than jobs is unrestricted using the batch operation function so that the operations of the tools and commands that are executed during the maintenance are not interrupted.
  - The Job Operation Software service that is running on the node to be maintained is terminated using the batch operation function, so that the Job Operation Software function will not operate in an unreasonable manner.

  The system administrator selects whether or not to terminate the Job Operation Software service running on the node.

Figure 2.24 Conceptual Diagram of Switching to the Software Maintenance Mode

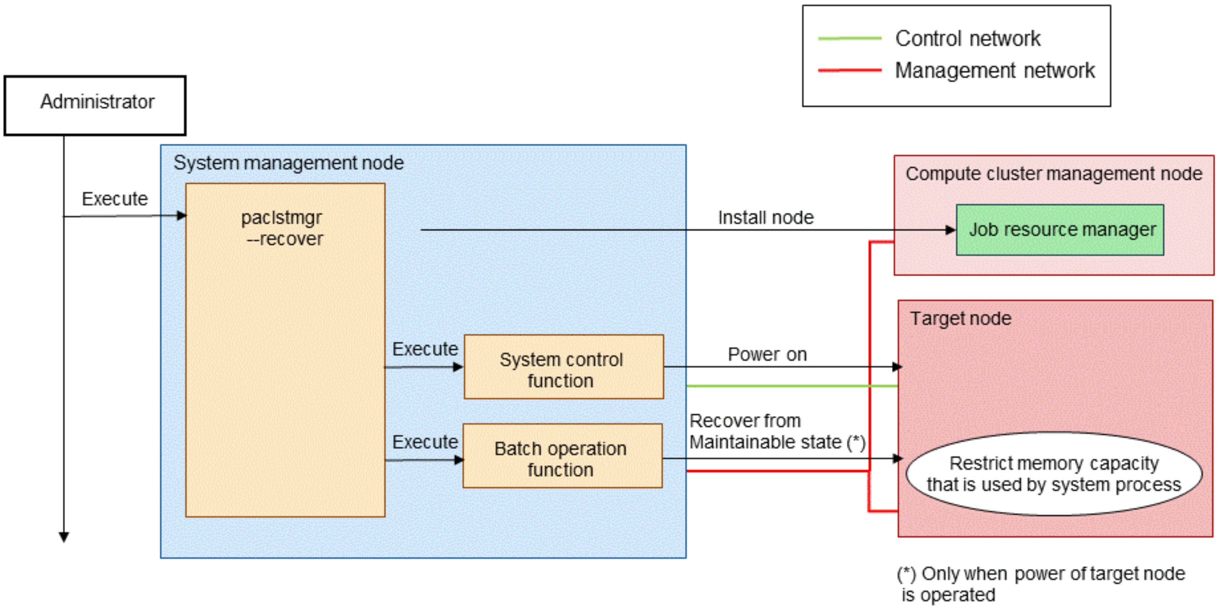


- Switching from the maintenance mode

  The system administrator switches a specified node from the maintenance mode to the ordinary operation status by specifying the --recover option to the paclstmgr command.
  The power is turned on when the node is in the hardware maintenance mode. When the node is in the software maintenance mode, the

system administrator installs the node in the operation using a desired method (e.g., node restart, service restart), because the necessity to restart the node varies depending on the maintenance work.

Figure 2.25 Conceptual Diagram of Switching from the Maintenance Mode



See

For details on the switching to and from the maintenance mode, see "3.4.3 Shifting to the Maintenance Mode and Its Recovery."

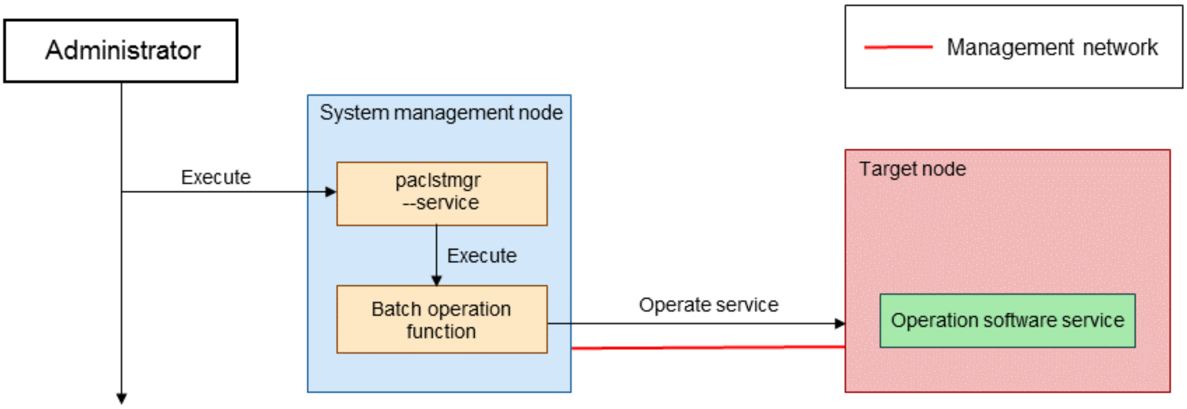
Information

When switching to and from the maintenance mode, the system administrator creates a plug-in in which a process is described and adds a necessary work before and after a node operation or service operation by specifying the plug-in to a paclstmgr command parameter. The system administrator can automate the process by using the plug-in. For details on the system maintenance function plug-in, see "3.4.6 System Maintenance Function Plugin."

2.4.4 Overview of Operating a Job Operation Software Service

The system administrator operates a Job Operation Software service by specifying the --service option to the paclstmgr command.
The paclstmgr command internally calls the batch operation function and operates the Job Operation Software service running on the node.

Figure 2.26 Conceptual Diagram of Operating the Job Operation Software Service



See

For details on the procedure to operate a Job Operation Software service, see "3.4.4 Operating the Service of the Job Operation Software."

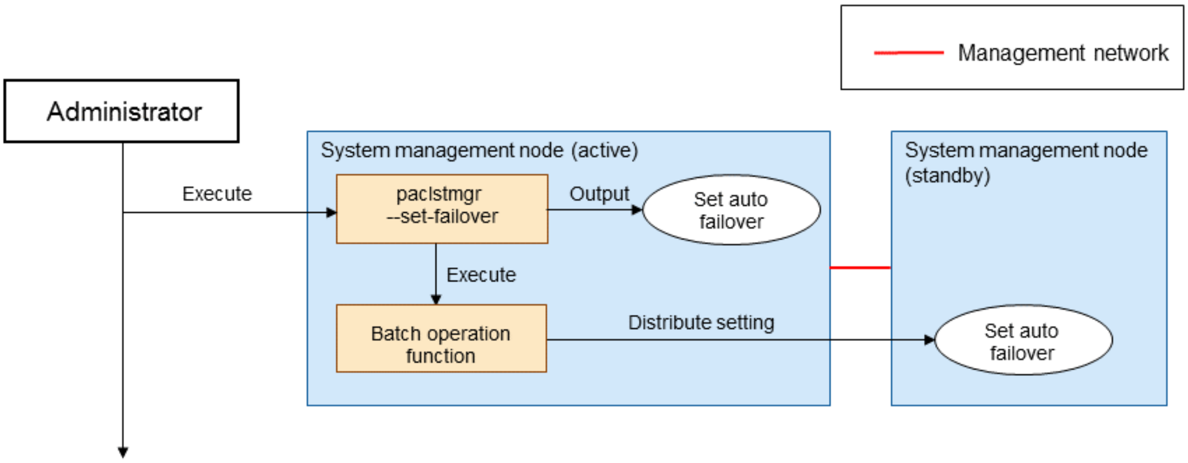
Information

When operating a Job Operation Software service, the system administrator can add a necessary work before or after a node operation or service operation by creating a plug-in in which a process is described and specifying the plug-in to a paclstmgr command parameter. The system administrator can automate the process using the plug-in. For details on the system maintenance function plug-in, see "3.4.6 System Maintenance Function Plugin."

2.4.5 Overview of Setting the Automatic Failover

The system administrator makes the automatic failover setting by specifying the --set-failover option to the paclstmgr command. The setting information of the automatic failover is output on the active system management node and synchronized with the standby system management node using the batch operation function.

Figure 2.27 Conceptual Diagram of Setting the Automatic Failover



See

For details on the procedure to set the automatic failover, see "3.4.5 Setting Automatic Failover."

2.5 Overview of the Operation Support Function

Since multiple nodes run simultaneously in the operation of a large-scale system, the same operation needs to be performed on the nodes. The system administrator can easily perform an operation (such as execution of a command and distribution of a file) on the nodes in the system using the operation support function.

The following operations can be performed using the operation support function.

- Batch command execution

  The system administrator uses the batch operation function to execute the same command on all nodes in the system or on the specified nodes at the same time. By doing so, a command can be executed from the system management node without logging in to each node.

- Batch transmission of files

  The system administrator can batch send multiple files to the nodes in the system using the batch operation function. This function is used to send the same file from the system management node to all nodes in the system or to specified nodes.

- Batch collection of files

  The system administrator can batch collect files from the nodes in the system using the batch operation function. This function is used to collect files from all nodes or specified nodes to the system management node.

- Management of dump files

  The system administrator can manage dump files in the system using the dump management function. When a dump file is created in each node, the dump file information is collected and collectively managed in the system management node. The system administrator can collect and delete the dump files on each node based on the dump file information that is collectively managed on the system management node.

- Connecting to a console on a node

  The system administrator can connect to a console on each node in the system using the operation support function (console connection function). The administrator can connect from the system management node to the consoles on all nodes in the system.

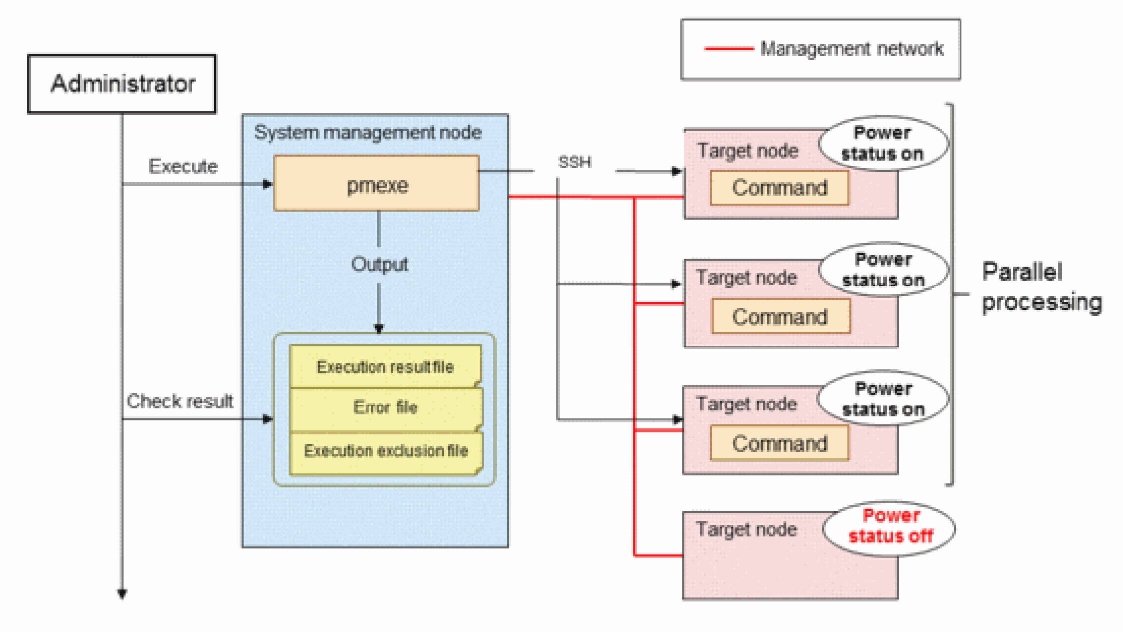
2.5.1 Overview of Batch Execution of Commands

The pmexe command executes the commands that are specified collectively to multiple nodes using the ssh connection.

When the nodes whose power supplies are disconnected are selected, such nodes are excluded before batch execution of commands, and then parallel execution of the commands is performed.

When the commands are executed at the same time, the command execution result of each node is output to an "execution result file." When there are nodes to which the commands could not be executed, the node information and the causes are output to an "error file." If there are nodes that were excluded from the command execution, the node information is output to an "execution exclusion file." The system administrator checks the batch execution result by referencing the files.

Figure 2.28 Conceptual Diagram of the Batch Operation of Multiple Commands



In a large-scale system that requires node groups due to having a lot of nodes, the pmexe command executes a command in hierarchical structure by automatically using the compute cluster sub management node and the boot I/O node as relay nodes in order to effectively perform parallel execution.

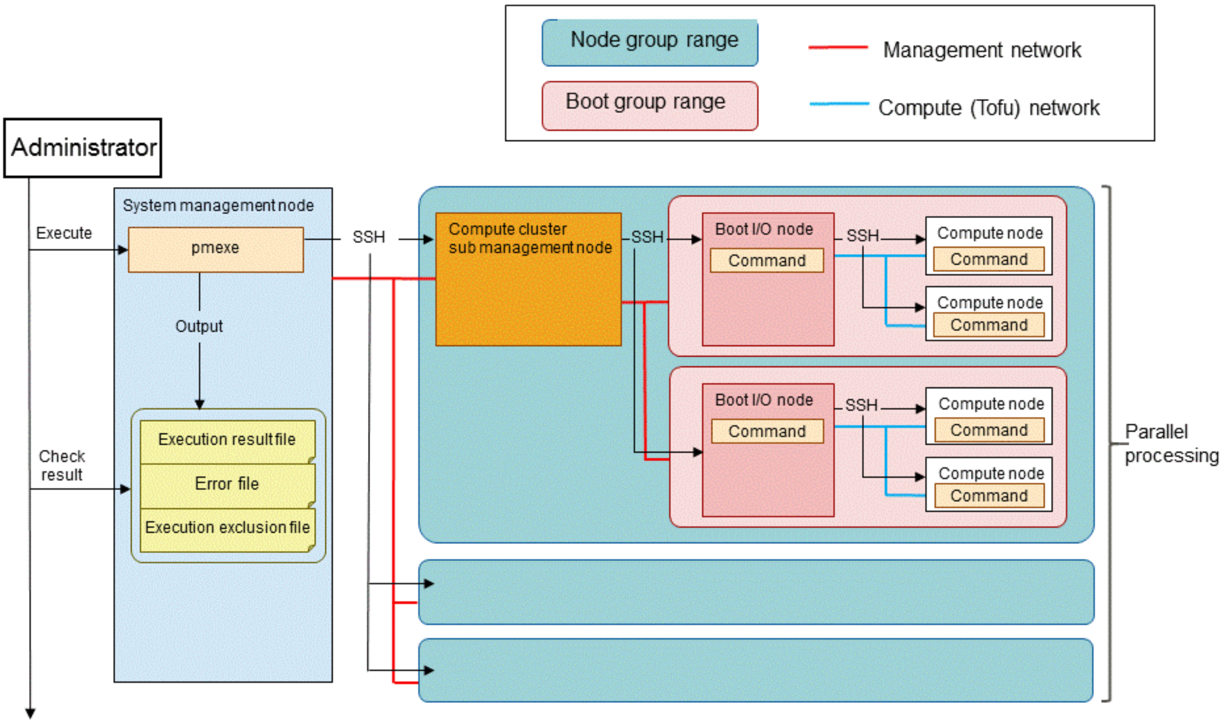
- Compute cluster sub management node

  A relay processing is performed or a command is executed to the boot I/O node in the node group.

- Boot I/O node in a node group

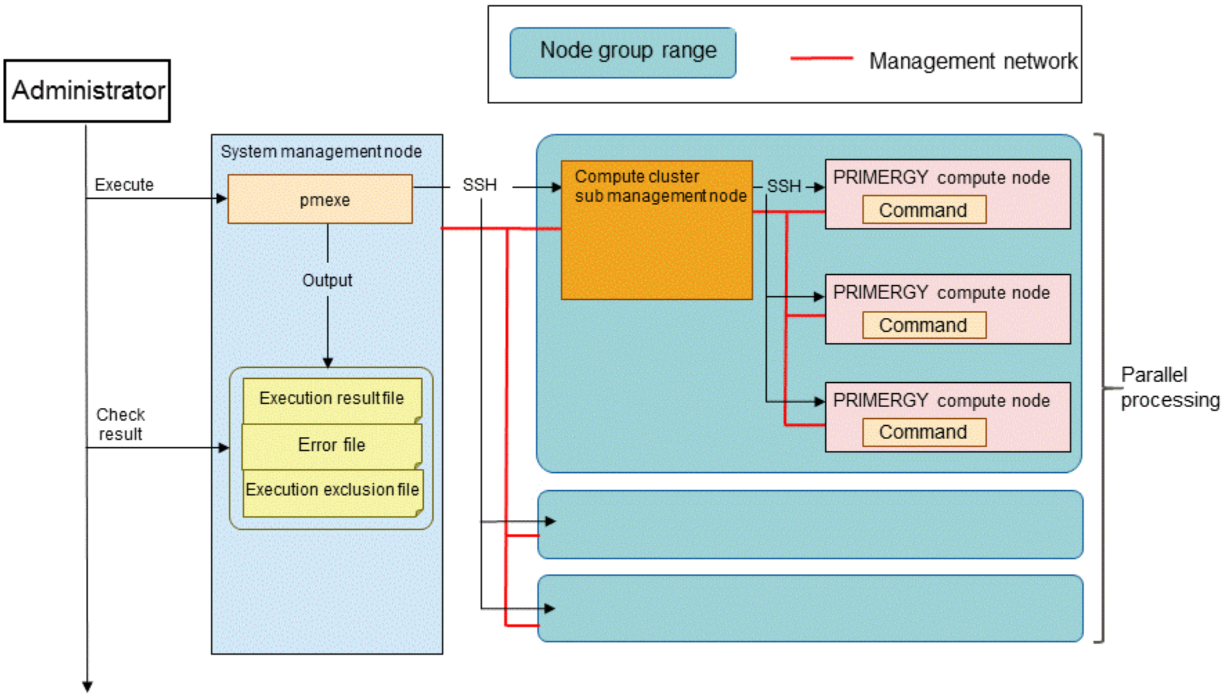
  A command is executed to the nodes in the boot node group.

Figure 2.29 Conceptual Diagram of Batch Operation of Multiple Commands (System in Large-Scale Configuration (FX server))



If a node group is configured with the PRIMERGY server, the compute cluster sub management node is used as a relay node in the same way as FX server. In this case, a command is executed from the compute cluster sub management node to the PRIMERGY server in the node group.

Figure 2.30 Conceptual Diagram of Batch Operation of Multiple Commands (System in Large-Scale Configuration (PRIMERGY server))



Information

If the compute cluster sub management node has a redundant configuration, it is used as a relay node when either the active node or standby node is being operated.

Note

- When the active and standby compute cluster sub management nodes cannot access the management network, a load to the system management node increases because a command is executed directly from the system management node to each node.
- When the active and standby compute cluster sub management nodes cannot access the management network, the time taken to execute a command increases because communication is switched to the nodes after accessing the compute cluster sub management node. If it is known in advance that the compute cluster sub management node is stopped, network access to the compute cluster sub management node can be skipped by specifying the --direct option.
- If the boot I/O node cannot be accessed via the management network, a command cannot be executed to FX server.

See

For the procedure to batch execute multiple commands, see "3.5.2 Collective Execution of Commands."

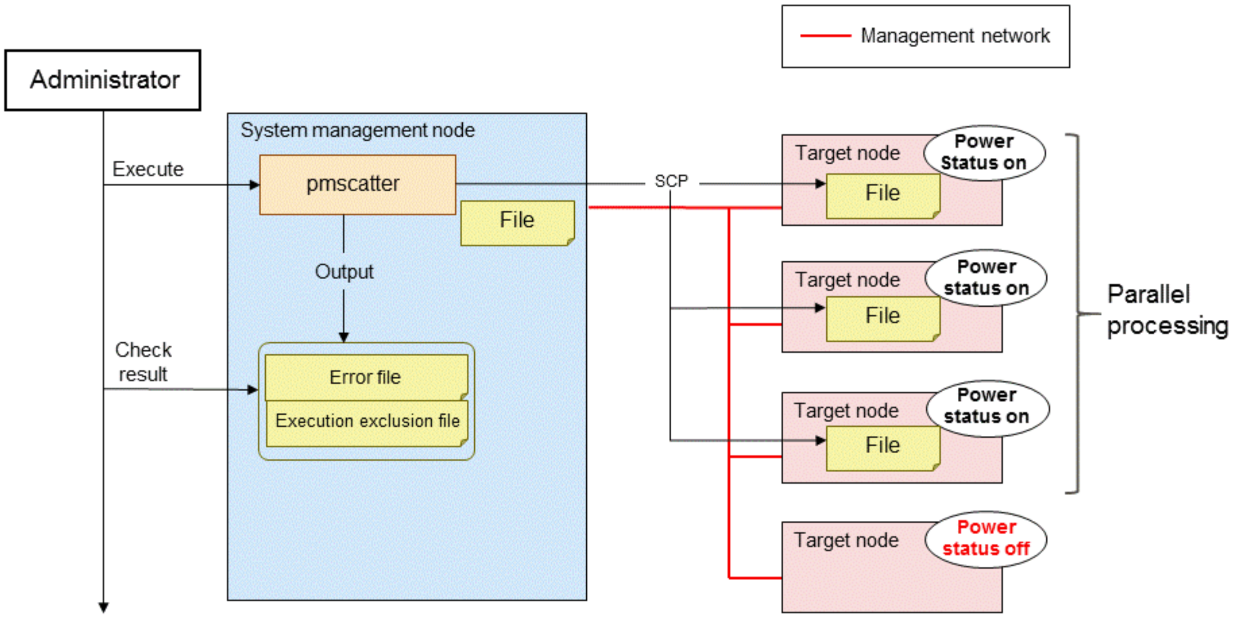
2.5.2 Overview of Batch Sending Multiple Files

The pmscatter command uses the scp command to batch send the specified files to multiple nodes from the system management node.

When the powered-off nodes are selected, such nodes are excluded before batch sending multiple files, and the parallel processing of the file transmission is performed.

If the files are failed to send, the node information and the causes are output to an "error file." If there are nodes that are excluded from the file transmission destination, the node information is output to an "execution exclusion file." The system administrator can check the batch transmission result by referencing the file.

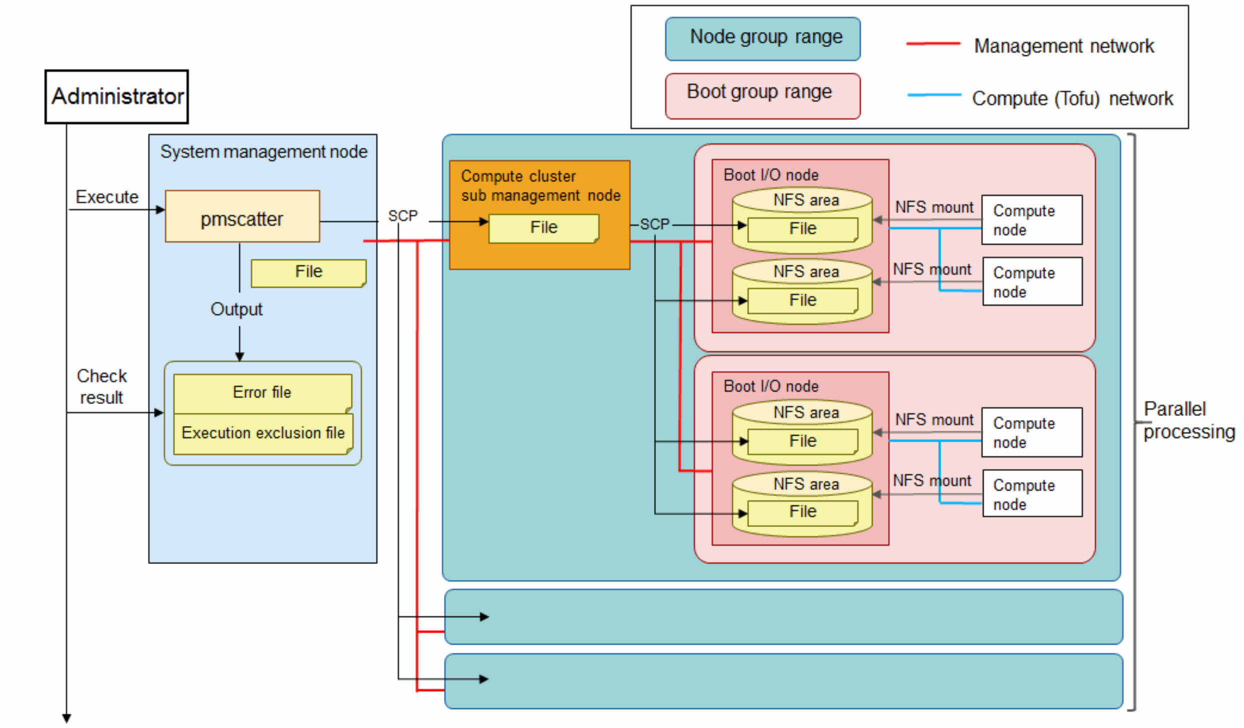
Figure 2.31 Conceptual Diagram of Batch Sending Multiple Files



In the system with node groups, the pmscatter command sends files in hierarchical structure by automatically using the compute cluster sub management node as a relay node in order to remotely and efficiently process the transmission.

The conceptual diagram below shows the procedure to send files to the boot I/O node in a node group and the compute node under the boot I/O node via the compute cluster sub management node.
The files can be sent to the nodes efficiently and parallelly using this method.

Figure 2.32 Conceptual Diagram of Batch Sending Multiple Files (System in Large-Scale Configuration (FX server))



Information

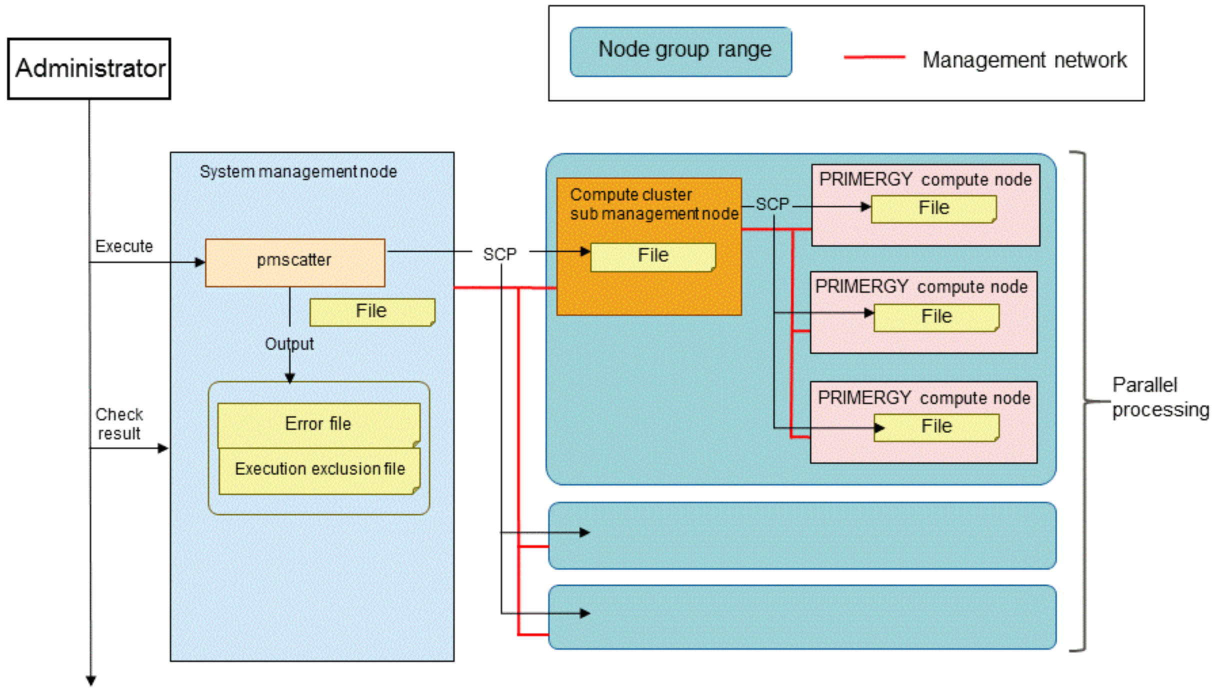
The nodes in FX server use the disk area on the boot I/O node. When sending the files to the compute node of FX server, the pmscatter command sends them to an area on the boot I/O node where the nodes are mounted.
Therefore, the files can be sent to all target nodes if a network to the boot I/O node can be accessed, even though the power of the compute node of FX server is off.

Note

- When the active and standby compute cluster sub management nodes cannot access the management network, a load to the system management node increases because files are sent directly from the system management node to each node.
- When the active and standby compute cluster sub management nodes cannot access the management network, the time taken to execute a command increases because communication is switched to the nodes after accessing the compute cluster sub management node. If it is known in advance that the compute cluster sub management node is stopped, network access to the compute cluster sub management node can be skipped by specifying the --direct option.
- If the boot I/O node cannot be accessed via the management network, files cannot be sent to FX server.

If a node group is configured with the PRIMERGY server, the compute cluster sub management node is used as a relay node in the same way as FX server. The diagram below shows the procedure to send files to the PRIMERGY server in the node group through the compute cluster sub management node.

Figure 2.33 Conceptual Diagram of Batch Sending Multiple Files (System in Large-Scale Configuration (PRIMERGY server))



See

For details on the procedure to batch send files, see "3.5.3 Delivering Files Collectively."

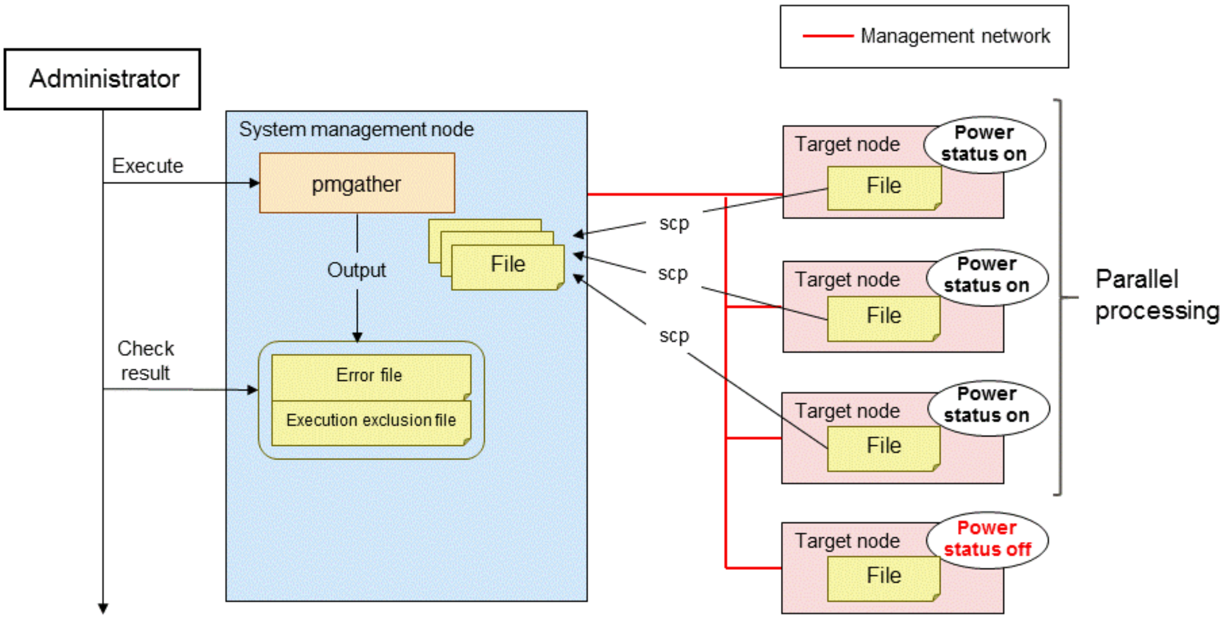
2.5.3 Overview of Batch Collecting Multiple Files

The pmgather command uses the scp command to batch collect the specified files from multiple nodes and store them in the system management node.

When the nodes whose power supplies are disconnected are selected, such nodes are excluded before batch collecting multiple files, and then the parallel execution of the file collection is performed.

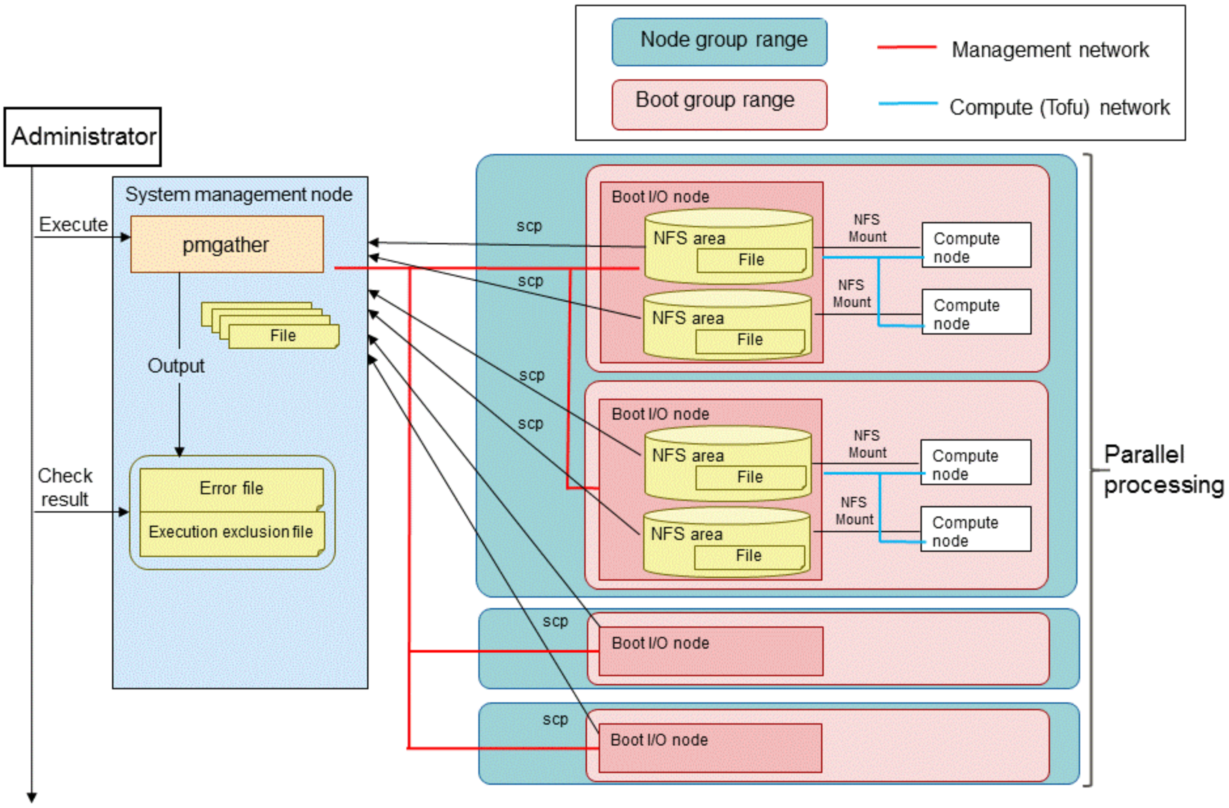
If the files fail to be collected, the node information and the causes are output to an "error file." If there are nodes that are excluded from the file collection, the node information is output to an "execution exclusion file." The system administrator can check the batch collection result by referencing the file.

Figure 2.34 Conceptual Diagram of Batch Collecting Multiple Files



Since a disk area in FX server is on the boot I/O node, files are collected from the NFS area, on which the diskless node on the boot I/O node is mounted, when collecting the files from the compute node in FX server.

Figure 2.35 Conceptual Diagram of Batch Collecting Multiple Files (FX server)



Information

In FX server, files are collected from the NFS area on which the diskless node on the boot I/O node is mounted. Therefore, the files can be collected if a network to the boot I/O node can be accessed, even though the power supplies of the nodes are disconnected.

See

For details on the procedure to batch collect multiple files, see "3.5.4 Collecting Files Collectively."

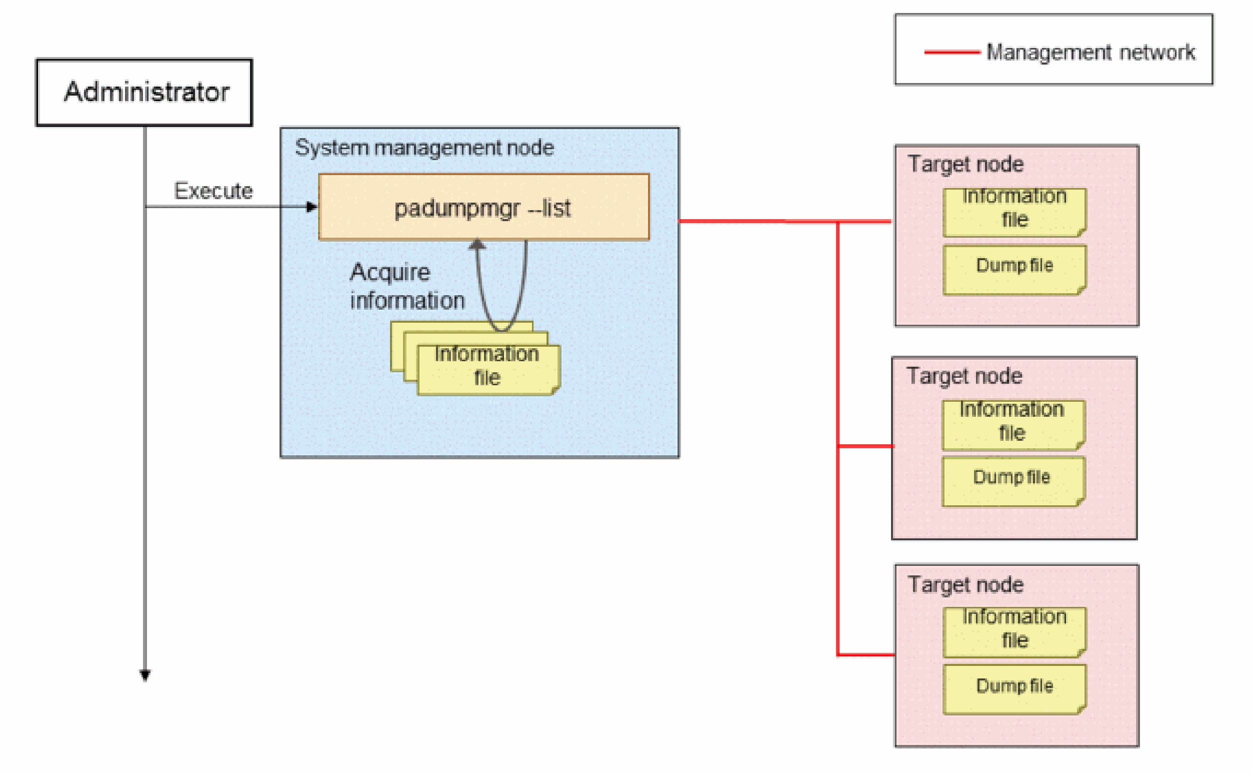
2.5.4 Overview of Managing Dump Files

The system administrator manages the dump files created on each node using the manage command, padumpmgr, for dump files.
The dump files created on the node are collected and managed on the system management node. The scp command is used to collect dump files and information files.

When dump files are created on each node, the files (information files), in which the dump file information is saved when starting the OS of the node where the dump files are created, are automatically collected on the system management node.

The system administrator can view the dump file list, which was created on each node based on the dump file information, by executing the padumpmgr command.

Figure 2.36 Conceptual Diagram of Viewing the Dump File List



The system administrator collects the dump files to be analyzed on the system management node using the padumpmgr command.

When the nodes from which the dump files are collected are the nodes in the PRIMERGY server or any node, the padumpmgr command uses the scp command to collect the dump files via the management network.

If the target is the FX server, the system administrator can acquire the dump files in the same way as the PRIMERGY server or any node.

The padumpmgr command collects the two types of dump files, "emergency dump file" and "memory dump file," using the following method.

- Emergency dump files
  They are collected from the BMC via HTTP using the control network.

- Memory dump files
  They are collected using the scp command from the boot I/O node using the management network.

Information

- There are the following two types of dump files in FX server: emergency dump files with minimum information that are used for the initial research when a problem occurs and memory dump files in which the memory details used by the OS are stored.

- In FX server, the emergency dump files are stored on the BMC, and the memory dump files are stored on the boot I/O node. Both dump files can be sent to the system management node using the padumpmgr command.

- The type of the dump files used in the PRIMERGY server is only memory dump files. The emergency dumps are not used.

Figure 2.37 Conceptual Diagram of Collecting Dump Files (PRIMERGY server)

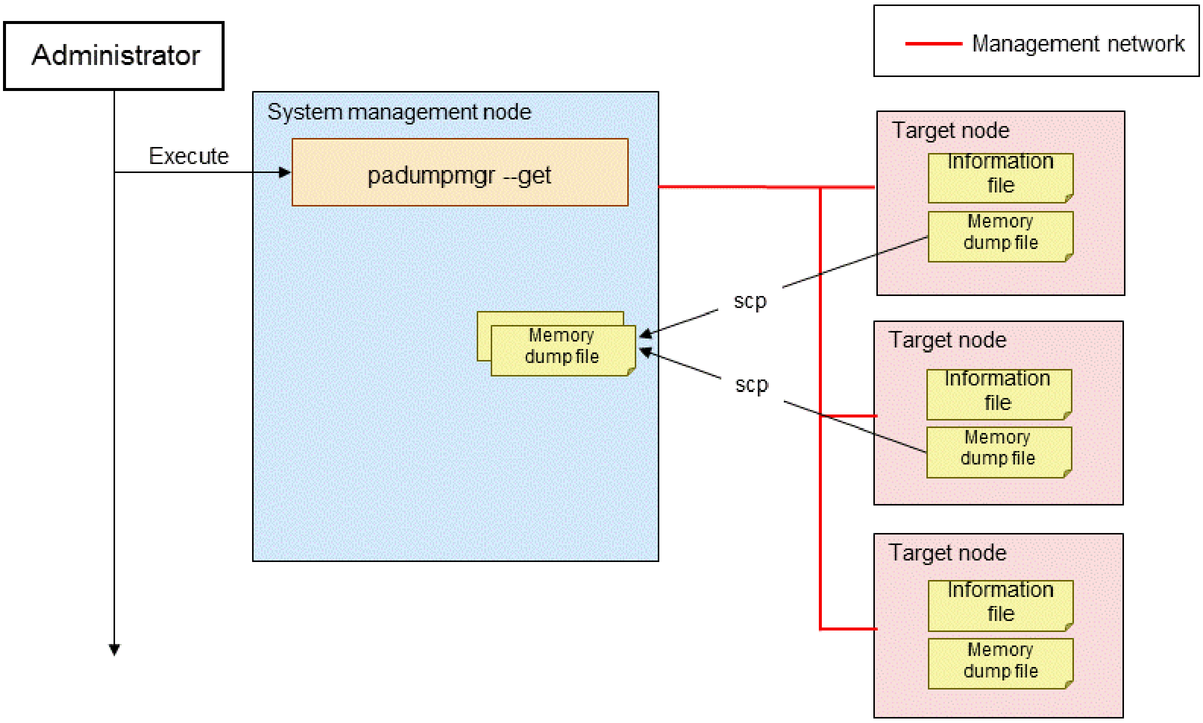
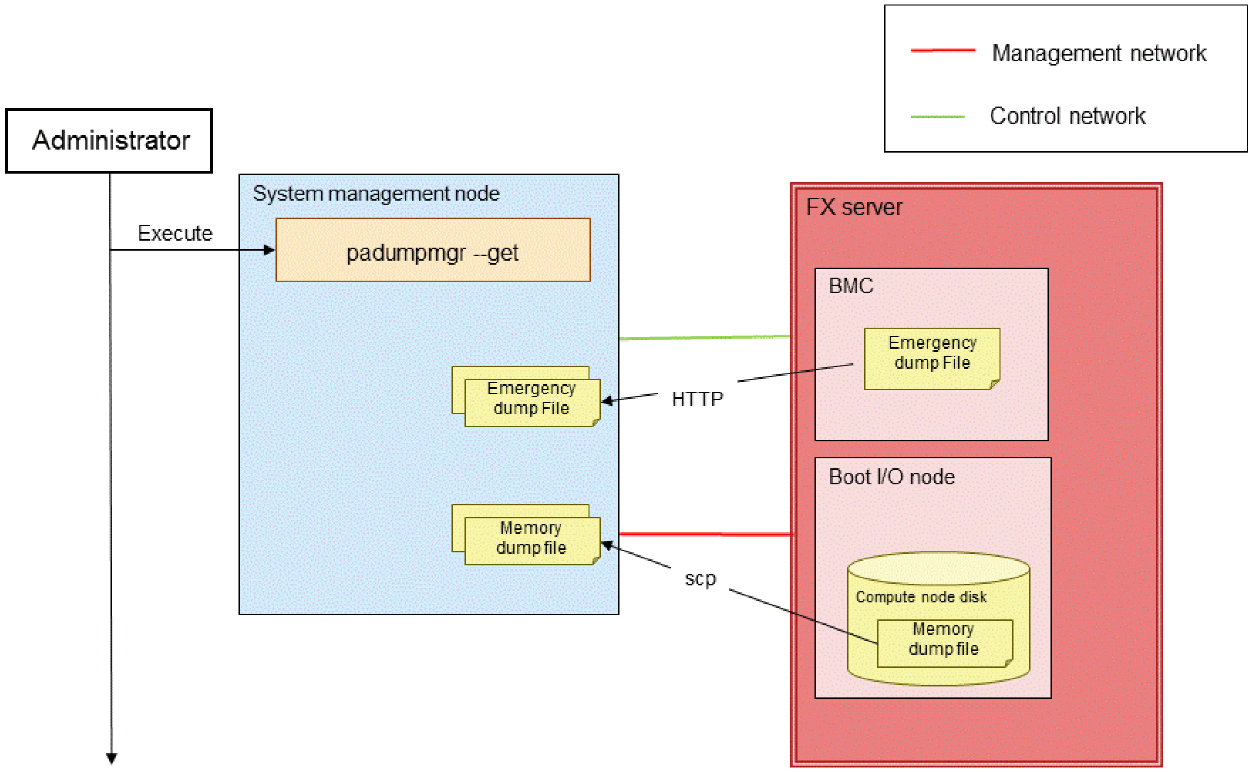


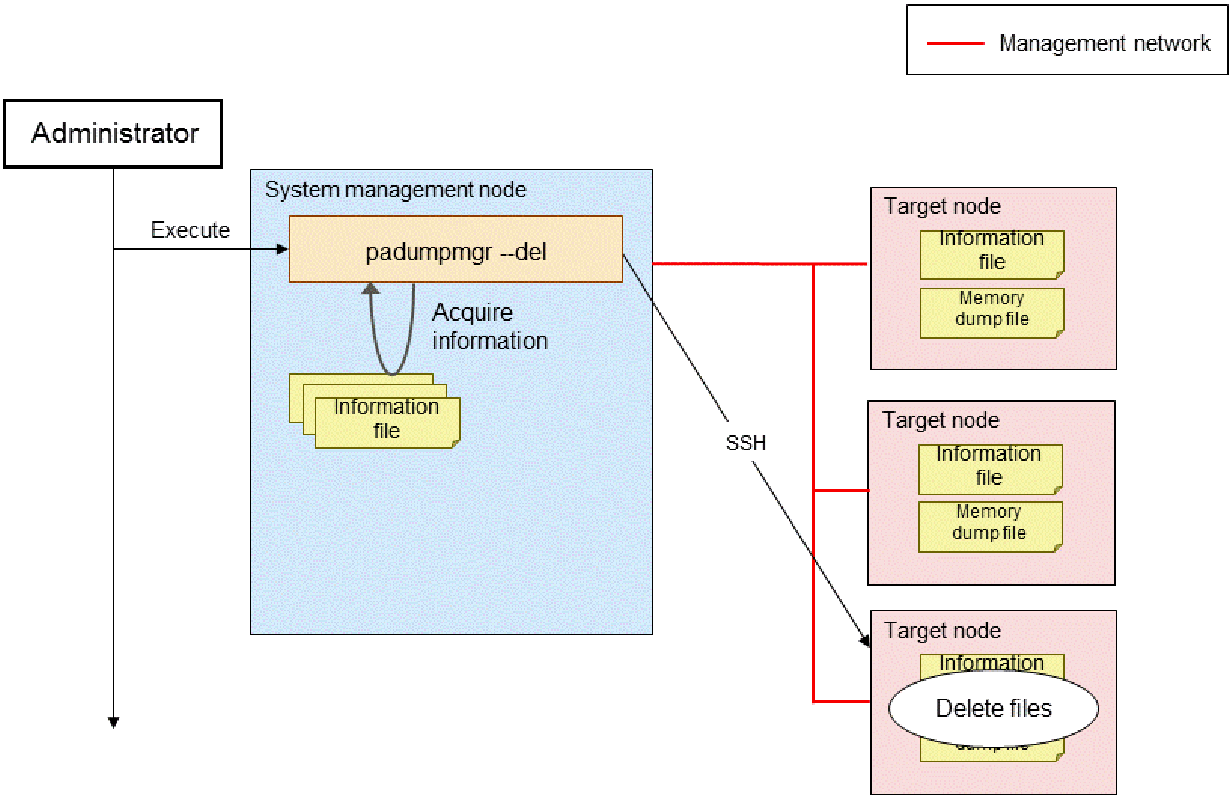
Figure 2.38 Conceptual Diagram of Collecting Dump Files (FX server)



The system administrator deletes unnecessary dump files using the padumpmgr command by referencing the dump file list managed on the system management node.

The dump files in FX server cannot be deleted using the padumpmgr command. The dump files are saved up to the third generation. The oldest file is automatically deleted each time a new dump file is created.

Figure 2.39 Conceptual Diagram of Deleting Dump Files (PRIMERGY server)

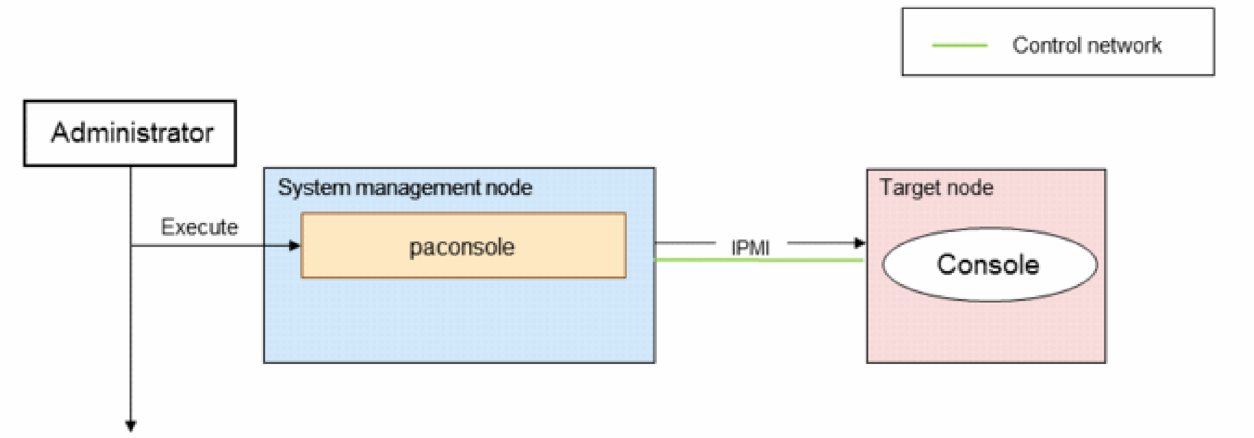


2.5.5 Overview of Connecting to a Node Console

A connection is made from the system management node to the console of each node using the console connection command, paconsole. A mechanism used for console connection varies depending on the model of the target node.

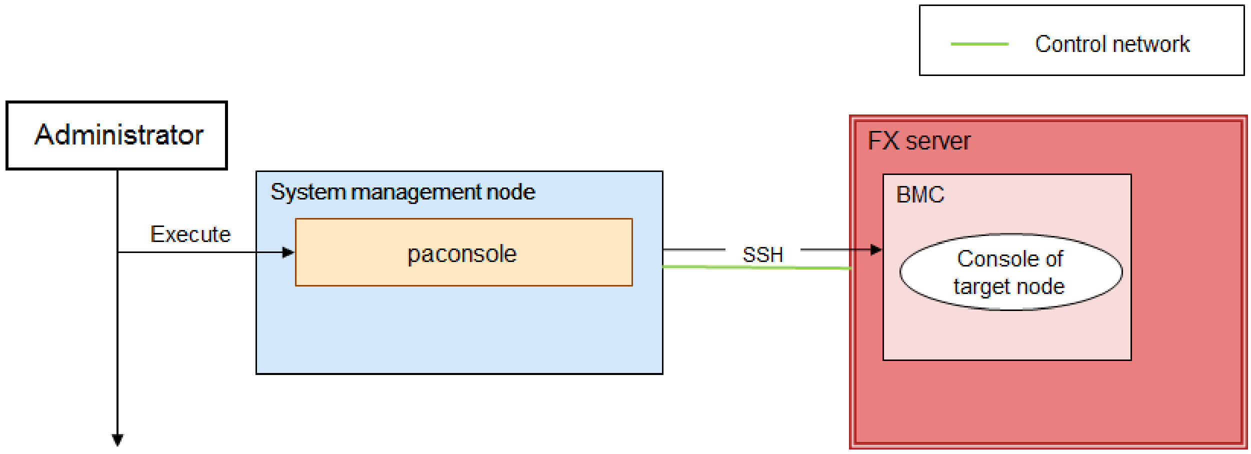
The control network is used to connect to a console in the PRIMERGY server via IPMI.

Figure 2.40 Conceptual Diagram of paconsole Command (PRIMERGY server)



A connection to a console in the FX server is made on the FX server console port using the control network via ssh.

Figure 2.41 Conceptual Diagram of paconsole Command (FX server)



2.6 Overview of Log Management Function

It is difficult to collect necessary logs or monitor the logs of each node when a problem occurs because multiple nodes run simultaneously in the operation of a large-scale system. However, the system administrator can easily collect or monitor the logs of the nodes in the system using the log management function.

The following operations can be performed using the log management function.

- Collecting research materials

  Research materials (including the logs and configuration files of the OS and the Job Operation Software) can be batch collected from each node to research the cause when a problem occurs.

- Log monitoring

  The detailed system status can be checked by monitoring the log files of each node and executing a process, such as e-mail notification, based on a specific message.

- Supporting to create a configuration file

  A configuration file for multiple nodes can be easily created because the node specific information (host name, etc.) in the configuration file can be changed based on each node.

- Regular log collection

  The system status can be often checked by regularly collecting the log messages from each node in one place.

2.6.1 Collecting Research Materials

The system administrator can batch collect necessary materials to research the Job Operation Software from multiple nodes by using the research material collection function. This function is used in the following situations.

- Collecting research materials by specifying nodes

  The system administrator can collect research materials by specifying nodes if the administrator knows the nodes to collect research materials. Groups, such as boot groups and node groups, can also be specified.

- Collecting research materials by specifying a problem

  If the system administrator does not know the nodes to collect research materials when a problem occurs, the administrator can batch collect necessary research materials from the nodes related to the problem by executing the research material collection command after specifying the nodes and the problem.

- Collecting research materials by specifying a job ID

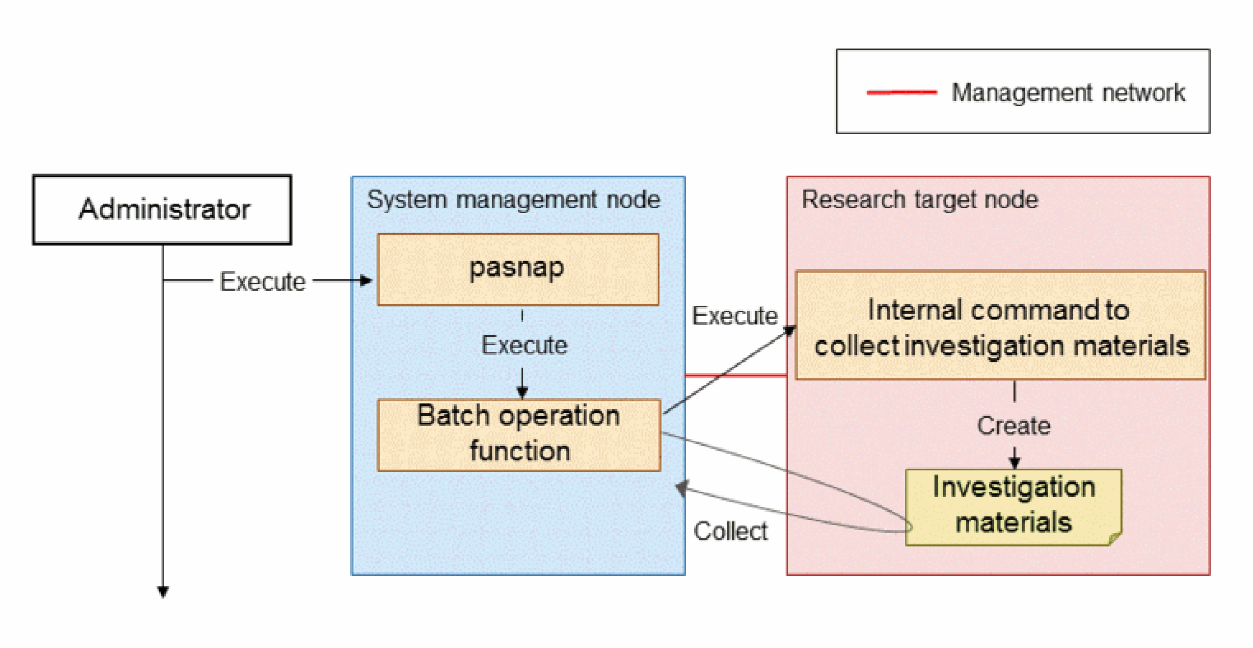
  If the administrator wants to collect research materials related to a job, the administrator can collect the research materials, which were created at the time of executing the job, from the compute node from which the job was executed with the research material collection command after specifying the job ID.

2.6.1.1 Overview of the Research Material Collection Function

- Overview of collecting research materials by specifying a node

The system administrator can create and transfer research materials by specifying the nodes to be researched and executing the pasnap command. The batch operation function is used to execute an internal command of the research material collection function and to transfer files.

Figure 2.42 Conceptual Diagram of Collecting Research Materials by Specifying a Node

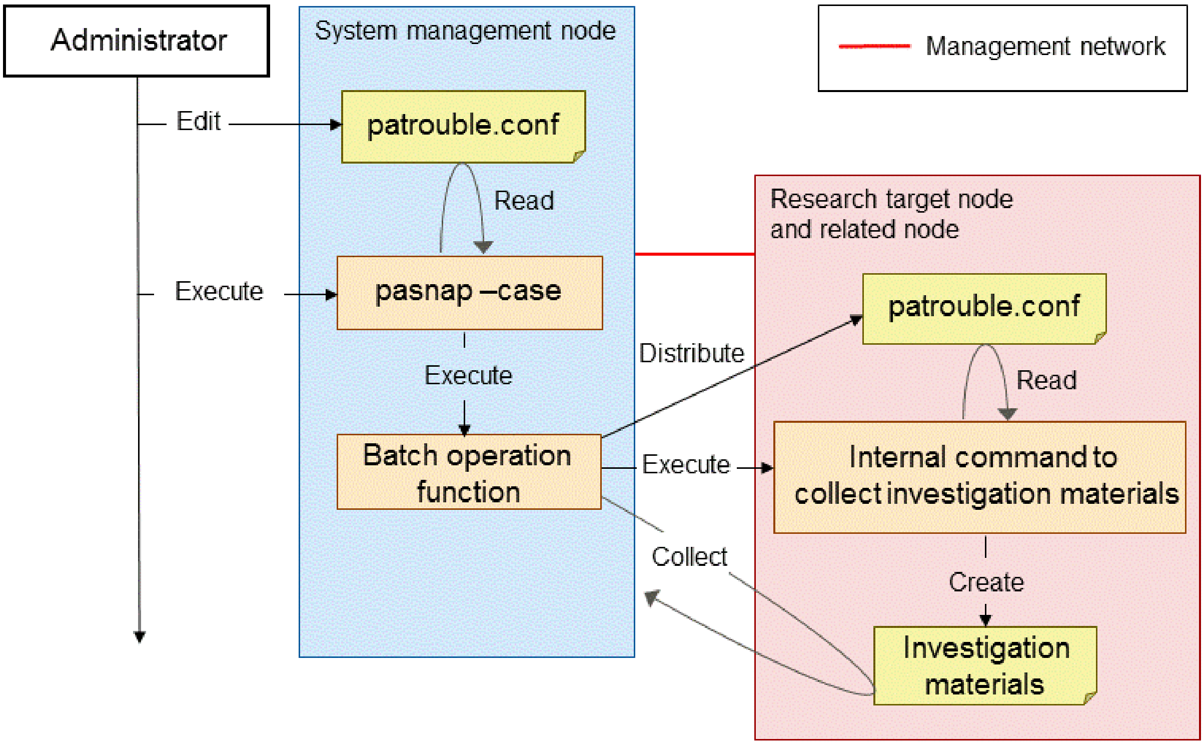


- Overview of collecting research materials by specifying a problem

The system administrator can batch collect research materials from the nodes related to a problem by specifying the problem described in the problem definition file, patrouble.conf, to the --case option of the pasnap command.

If you want to add original problems, you can define the problems by editing the problem definition file, patrouble.conf.

Figure 2.43 Conceptual Diagram of Collecting Research Materials by Specifying a Problem

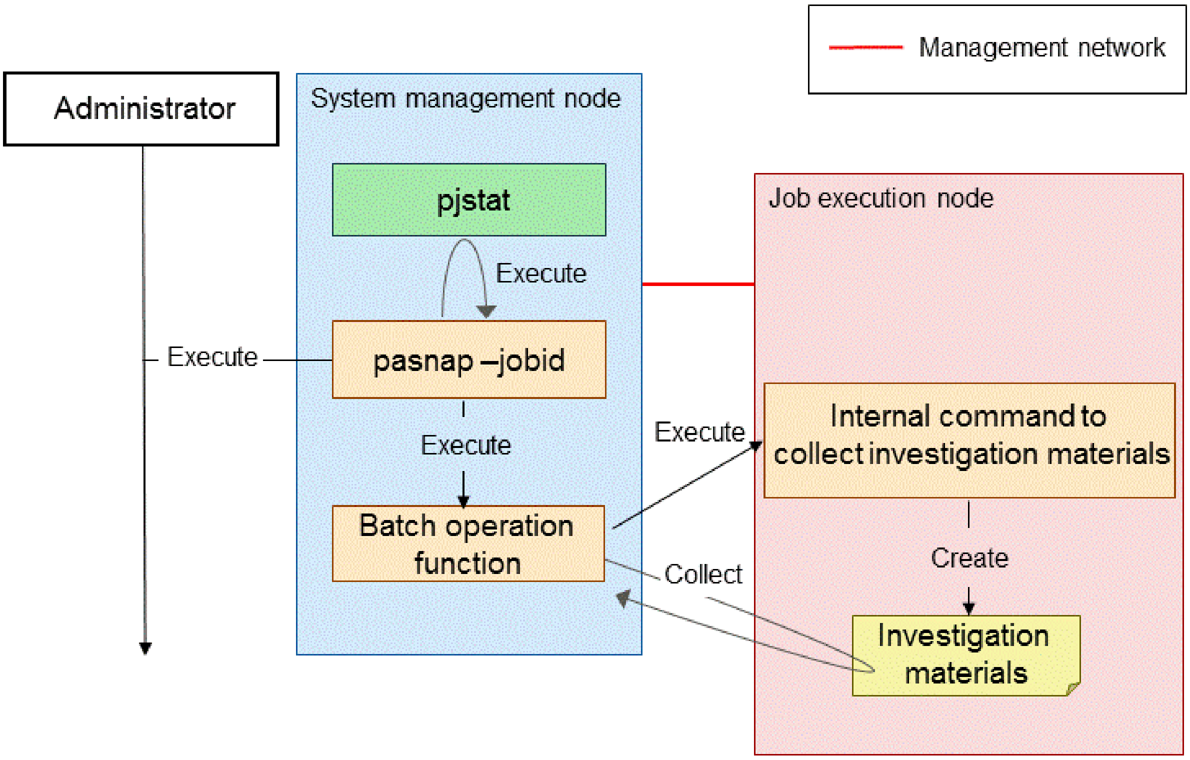


- Overview of collecting research materials by specifying a job ID

The system administrator can collect the research materials that are created during the job execution by specifying the job ID to the --jobid option of the pasnap command and executing the research material collection command.

The pjstat command is used to collect job information.

Figure 2.44 Conceptual Diagram of Collecting Research Materials by Specifying a Job ID



See

- For the explanation of the research material collection function, see "3.6.1 Investigation Data Collection."
- For the explanation of the batch operation function, see "2.5 Overview of the Operation Support Function."
- For details on the pjstat command, see "Checking the Job Status" in the "Job Operation Software End-user's Guide."

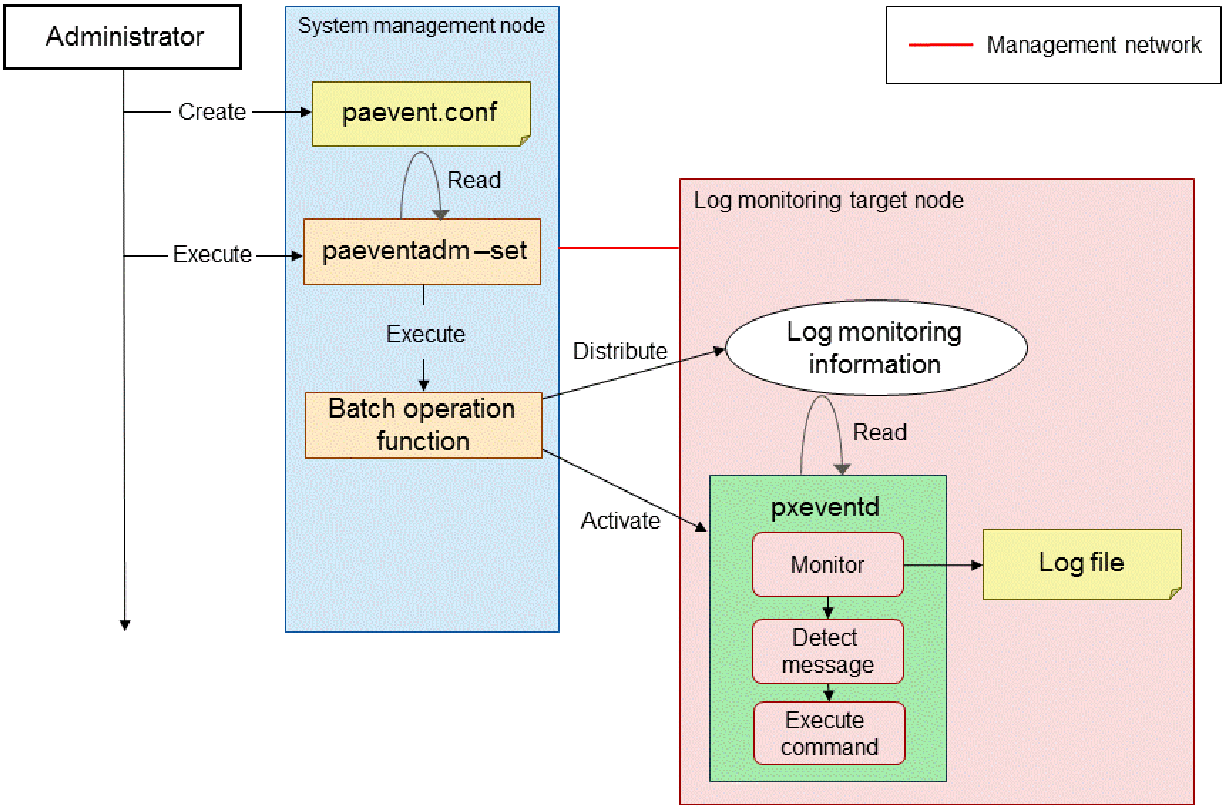
2.6.2 Log Monitoring

The system administrator can execute a specific process, such as e-mail transmission, when a specific message is detected while monitoring the logs, which are output from the nodes, using the log monitoring function.

2.6.2.1 Overview of Log Monitoring

The system administrator can make the log monitoring setting by creating a log monitoring configuration file, paevent.conf, and executing the log monitoring setting command, paeventadm.
After configuring the log monitoring setting, the log monitoring daemon, pxeventd, starts and the log files are monitored as specified. The log monitoring daemon executes a command that is set when detecting a specified log message.
The batch operation function is used to distribute the configuration files or to set a log monitoring daemon with the log monitoring setting command.

Figure 2.45 Conceptual Diagram of Log Monitoring



See

For details on log monitoring, see "3.6.2 Log Monitoring."

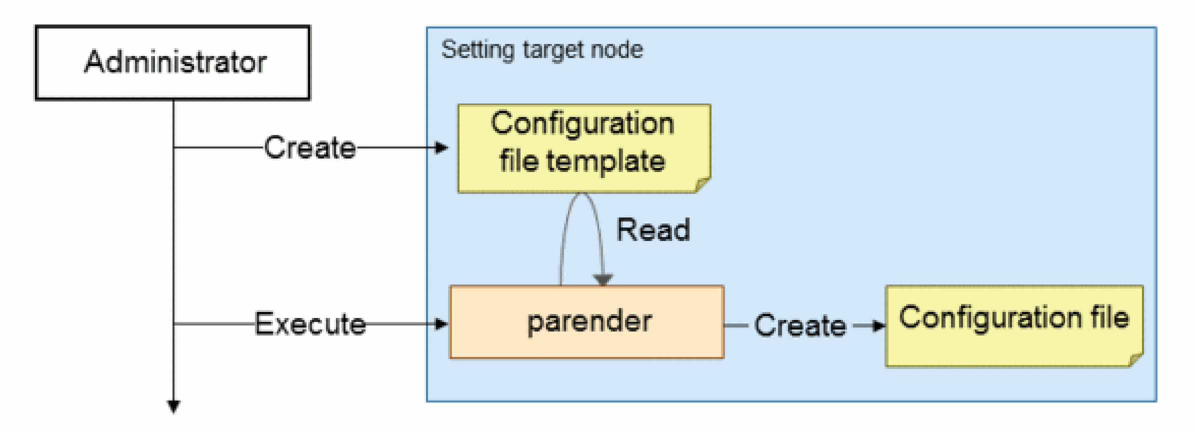
2.6.3 Supporting to Create a Configuration File

The system administrator can set the specific information (node ID, etc.) of the Job Operation Software by using this function when creating a configuration file, such as open software.

2.6.3.1 Overview of Supporting to Create a Configuration File

The system administrator can create a configuration file template. The administrator describes a specific keyword in an area, in which the configuration information such as node ID is described, in a configuration file template. The keyword in the configuration file template is changed to the configuration information of each node by specifying the configuration file template and executing the configuration file creation support command, parender.

Figure 2.46 Conceptual Diagram of Supporting to Create a Configuration File



See

For details on the support of creating a configuration file, see "3.6.3 Supporting Creation of a Configuration File."

2.6.4 Regular Log Collection

The regular log collection function can be used to monitor the log files on each node and to transfer added messages to one node. The regular log collection function uses the open source software, Filebeat (*1) and Logstash (*2).

The following describes the features of each software and the nodes to which the software is applied.

Table 2.3 Features of Filebeat and Logstash and the Nodes to Which the Software is Applied

Software	Features	Nodes to be Applied
Filebeat	- Monitoring log files and transferring messages - Less CPU and memory loads - Operable in the ARM architecture (except for FX server)	Boot I/O node Compute node (PRIMERGY server) ARM architecture node (except for FX server)
Logstash	- Monitoring log files and transferring messages - Reception of the log messages transferred from Filebeat - Decoding the log messages	Nodes other than above

(*1) Filebeat: Elastic open source software

https://www.elastic.co/jp/products/beats/filebeat

(*2) Logstash: Elastic open source software

https://www.elastic.co/jp/products/logstash

2.6.4.1 Overview of Regular Log Collection

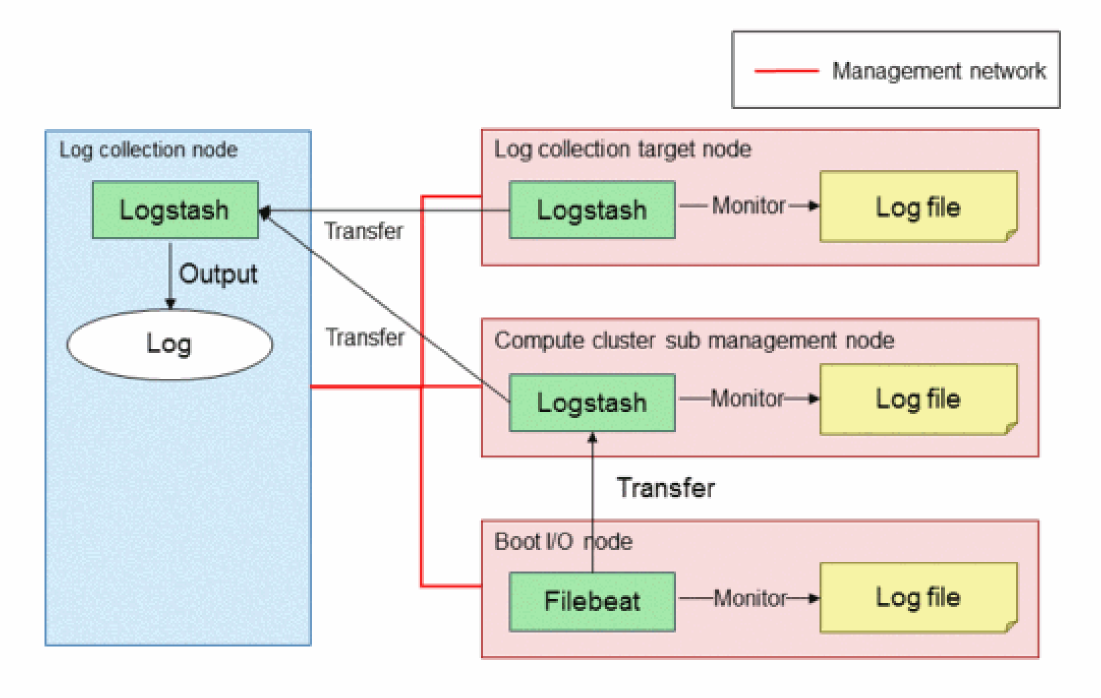
The system administrator makes the log message transmission setting for Logstash and Filebeat to the nodes from which logs are collected, configures the log message reception setting to the log collection node, and consolidates the logs of each node on the log collection node by starting each daemon.

When collecting the logs of the boot I/O node or compute node (PRIMERGY server), the logs are consolidated in the compute cluster sub management node by the node group and then consolidated to the log collection node in order to prevent access being concentrated at the log collection node.

The log collection node is a node to collect the logs of each node that is connected to the management network.

The batch operation function and the configuration file creation support function can be used to configure the Logstash and Filebeat settings.

Figure 2.47 Overview of Regular Log Collection



See

For the explanation of regular log collection, see "3.6.4 Periodical Log Collection."

2.7 Overview of the Software Environment Check Function

When installing a node or when performing a maintenance work, such as applying a package, the operations, including setting the Job Operation Software and applying a package, are carried out to multiple nodes. After the node installation or maintenance work, the system administrator needs to check whether the settings are made to multiple nodes as expected.
The system administrator can check the package application status, such as the application status of the Job Operation Software, and also check whether the setting is configured as intended using the software environment check function. The system administrator checks the status of the nodes to determine whether they can be used in the operation according to the result.

- Checking the software environment

  The following things can be checked using the software environment check function.

  - Checking a package application
    After installing a node or updating a package, the system administrator can check whether the Job Operation Software package or a package specified by the administrator is applied to the target nodes.

  - Checking the existence of configuration files
    When a setting is made to the nodes in the system, the system administrator can check that the path, permission, owner, and owned group of a file created on a node are set as intended.

  - Checking the result of comparing the file details
    When the system administrator needs to set multiple nodes in the system to the same setting, the administrator can check whether there are differences in the configuration files on multiple nodes using this function.

  - Checking the execution result of a command
    The system administrator can execute a command to a node in the system and check the result.

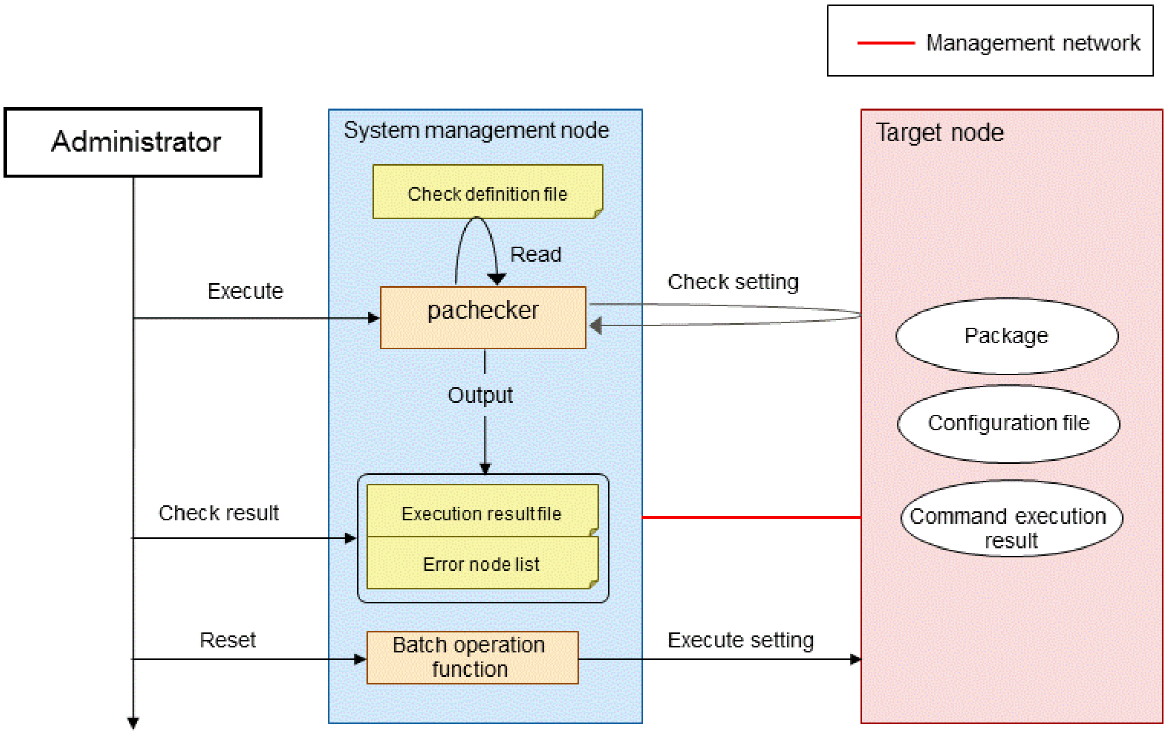
  The system administrator can check the items above by executing the pachecker command.
  The check items related to the Job Operation Software are prepared by default. However, the system administrator can check other items, such as the setting of a tool created by an OS or the administrator, by creating a check definition file.

2.7.1 Overview of the Software Environment Check Function

The conceptual diagram below shows how to use the software environment check function.

The system administrator checks the setting of a node by executing the pachecker command. The result of each check item is output to the "execution result file" and "error node list."
The system administrator can reset the setting using the batch operation function by referencing these files.

Figure 2.48 Conceptual Diagram of the Software Environment Check Function



See

For details on the software environment check function, see "3.7 Details of the Software Environment Check Function."

2.8 Overview of the Installation Function

In a large-scale cluster system that is configured with multiple nodes, the system administrator needs to efficiently install the system within a limited time.
The installation function includes the following functions for the system administrator to perform a parallel installation of an OS on multiple nodes that comprise the system.

- Installing an OS on PRIMERGY server

  The system administrator can network install an OS on PRIMERGY server by using the combined functions provided by the installation function.

- Installing an OS on FX server

  The system administrator can network install an OS on FX server by using the combined functions provided by the installation function.

- Initial setting of the installation function

  The initial setting of the server must be configured as preconditions to operate the install function.
  The system administrator configures the initial setting of the server by executing the initialization command after creating a configuration file, in which an operation environment is described, on the installer node. Then each installation function can be used.

- Repository management

  The system administrator manages a repository to be referenced when installing an OS or applying an update package.
  The system administrator can create repositories, display a list of the created repositories, update and delete repositories, and perform other operations.

- Creating an installation target node setting

  The system administrator can create information required for installing an OS on a node using a file provided in Microsoft Excel format (hereinafter called the "design sheet for installation" file).

- Creating initrd for installation

  The system administrator can create a boot image (initrd) required for OS installation.

- Creating a root file system

  The system administrator can create a root file system (rootfs) required for image-based OS installation.

- Association with ServerView Suite

  The system administrator can configure a setting to install a kernel module provided from the ServerView Suite, which is required to install an OS on PRIMERGY server, and to automatically install software included in ServerView.
  This function enables the system administrator to install an OS on PRIMERGY server without taking into account the ServerView Suite specifications.

- Installing a kernel module

  A kernel module is installed for initrd that is used for OS installation.
  The installed kernel module is used when a special kernel module is necessary.

- Registering node information
  The system administrator can register the node information required to install an OS on an installer node.
  The system administrator can create a necessary environment to network install an OS, such as adding the node information entry of the DHCP server and creating PXE configuration files and kickstart files.

- Checking the installation status

  The system administrator can check the OS installation status.
  When an error occurs during the OS installation, information to identify the cause and information to handle the error are displayed.

- Synchronizing the materials and settings for installation

  When an installer node has a redundant configuration or when a relay node exists in the system, the system administrator needs to distribute (synchronize) a repository created on the active installer node or materials and settings for installation to the standby installer node and relay node. The system administrator synchronizes the materials using the installation function.

- Checking the service operation status

  The system administrator checks whether the services necessary to operate the installation function are running.
  The system administrator can prevent a problem from happening due to a service stop during the maintenance by checking the necessary service status using this function before performing the maintenance that is carried out using the installation function.

2.8.1 Installation Function Knowledge Required as a Precondition

This section describes the basic knowledge that needs to be acquired as a precondition to use the installation function.

2.8.1.1 Basic Configuration of Installation Function

The installation function is used to network install an OS for a node using the PXE boot.
To network install an OS, the system administrator needs to set up necessary services on a node that is used as a server for the installation.

The installation function has some configuration elements for network installation. "Table 2.4 Configuration Elements of the Installation Function" shows the configuration elements.

Table 2.4 Configuration Elements of the Installation Function

Configuration Element	Meaning
Installer node	This node has functions as an install server. The example of an installer node is a system management node.
Relay node	This node supports an installer node in a large-scale environment. Since this node has the repository duplication function and TFIP server function, this node balances a load when installing an OS or when applying a fix package. The example of a relay node is the compute cluster sub management node. Nodes in a node group obtain necessary materials from a relay node during the installation.
Installation target node	This is a collective name of the nodes on which OSs are installed using the installation function. Standby installer node and relay node are included in the installation target nodes.
Repository	This is a directory in which the RPM package files that are applied to each node are collected. A package is retrieved from a repository by creating a repository per OS version and using the yum command. A master is stored in a repository on an installer node. Replication data is stored in a relay node to balance a load. A repository is also used to apply a package during the maintenance.
DHCP server	This is a server function to assign an IP address for installation to an installation target node. An installer node plays this server function. In FX server, the boot I/O node acts as a DHCP server in a BoB.
HTTP server	This server has a function that enables an installation target node to reference repositories or the installation function settings. This server operates on an installer node or relay node.
TFTP server	This server distributes the OS information (kernel image, initrd image, etc.) required for installation during the network installation to an installation target node. An installer node and a relay node act as a TFTP server. In FX server, the boot I/O node acts as a TFTP server in a BoB.
NTP server	This is a server that distributes the current time. An installer node and a relay node act as a TFTP server. In FX server, the boot I/O node acts as an NTP server in a BoB.
NFS server	The boot I/O node in FX server is an NFS server. This server is necessary to mount a root file system on the boot I/O node from the NFS diskless node.
Status management daemon (pxinstd)	This daemon is used to manage the installation status of an installation target node. The daemon operates on an installer node and a relay node.

The DHCP server cannot be changed because an installer node always serves the server function.

A relay node has the same functions as the installer node on the TFTP server, HTTP server, and NTP server.
A node belonging to a node group uses a relay node as a server for installation. A relay node itself and a node that does not belong to a node group use an installer node as a server for installation.

The table below shows whether the services, which are used with the installation function, are available by node type.

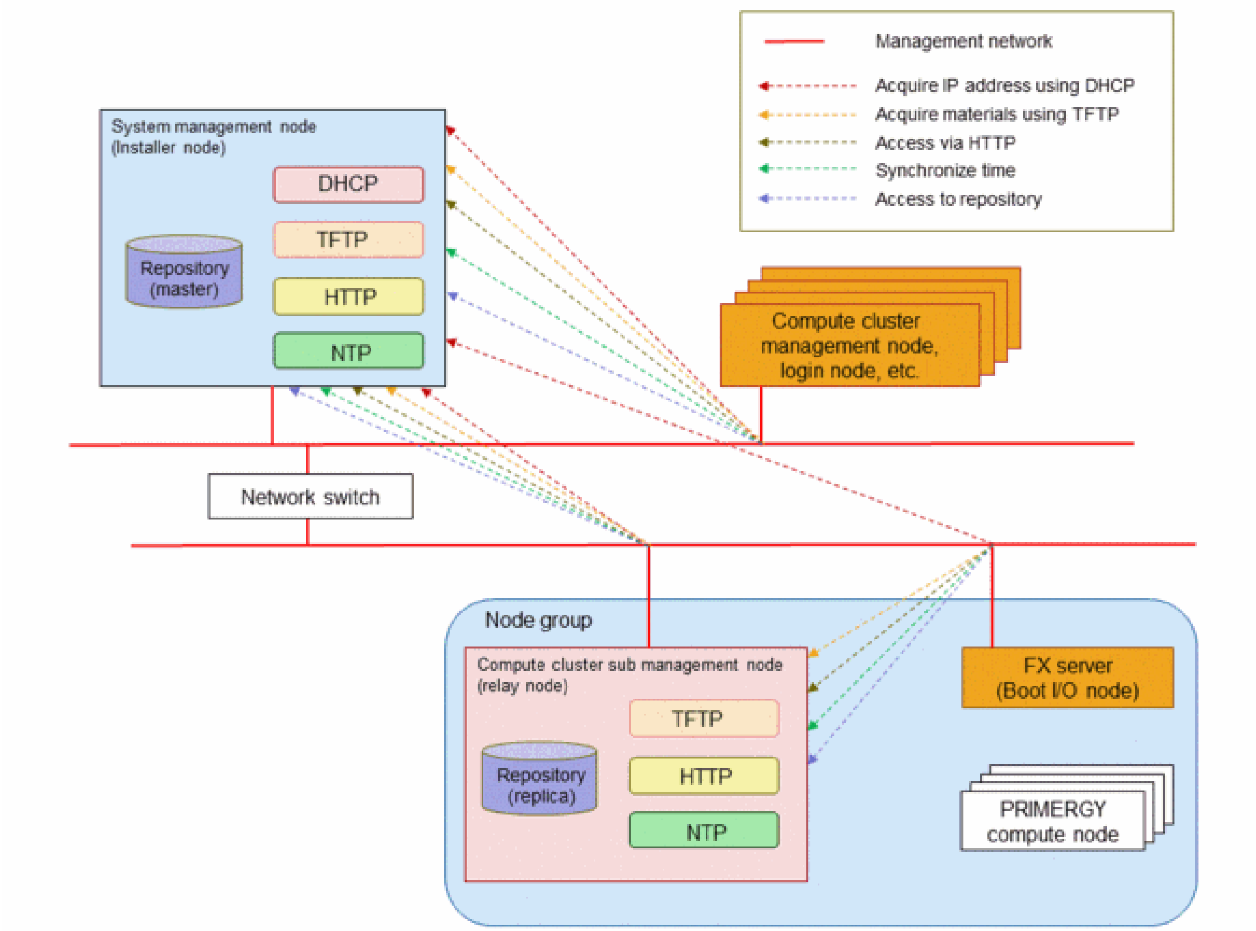
Yes: The installation function uses the service.
No: The installation function does not use the service.

Table 2.5 Installation Function Service List

Node Type	Service (Daemon Name)					
	DHCP (dhcpd)	TFTP (xinetd)	HTTP (httpd)	NTP (chronyd)	NFS (nfsd, etc.)	pxinstd (pxinstd)
Installer node (System management node)	Yes	Yes	Yes	Yes (Server + client function)	No	Yes
Relay node (Compute cluster sub management node)	No	Yes	Yes	Yes (Server + client function)	No	Yes
Installation target node (Compute cluster management node, login node, etc.)	No	No	No	Yes (Client function only)	No	-
FX server (Boot I/O node)	Yes (Inside BoB)	Yes (Inside BoB)	Yes (Inside BoB)	Yes (Inside BoB, server + client function)	Yes (Inside BoB, server)	-
FX server (Except for boot I/O node)	No	No	No	Yes (Client function only)	Yes (Client)	-

The diagram below shows the relationship among the elements that configure the installation function.

Figure 2.49 Diagram Showing the Relationship Among the Nodes Related to the Installation Function

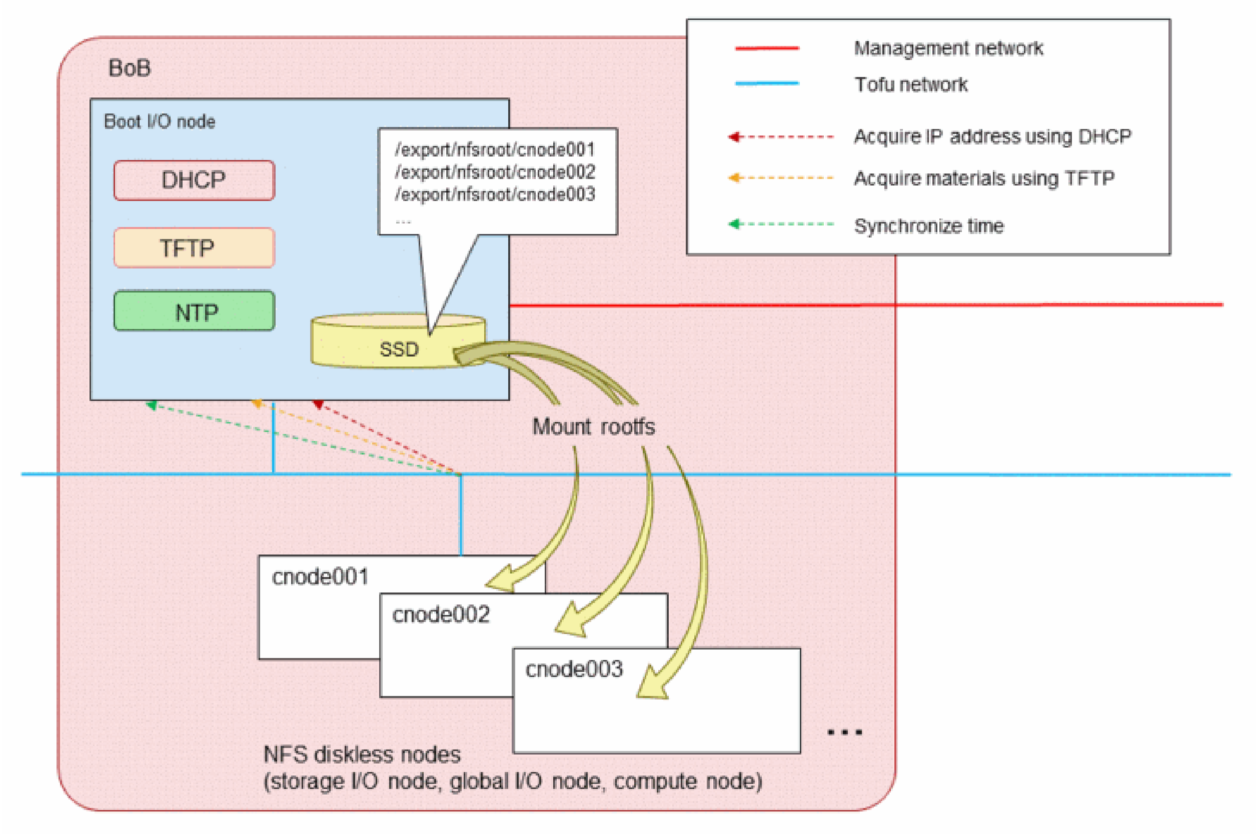


2.8.1.2 Configuration of FX server

FX server has a configuration with no disk devices, except for the boot I/O node. Therefore, the boot I/O node acts as an NFS server in a BoB, and the nodes other than the boot I/O node function as NFS diskless nodes that are mounted as a root file system on an area on the boot I/O node.

The boot I/O node in the BoB performs all server functions required for the installation function. The installation function does not access other BoBs. The figure below shows the relationship in a BoB.

Figure 2.50 Installation Function Configuration in FX server



2.8.1.3 Terminating the Services During the Operation

Some services used by the installation function can terminate after completing the installation work.
Terminating these services does not affect a job operation. However, all necessary services must be activated again when an operation (such as changing the configuration during the maintenance, reconfiguration of a node, backup/restore, and application of a fix package (*)) is performed using the installation function.

The system administrator can check the operation status of the service used by the installation function using the service status check function (see "3.8.11 Checking the Service Operation State"). The table below shows whether each service can be terminated or not during the operation.

(*) A fix package cannot be applied using the installation function. However, when a package manager (yum) is used, the repository created using the installation function, the httpd setting, and yum setting are required.

Yes: Services that can be terminated during the operation
No: Services that cannot be terminated during the operation (services required for a job operation)
-: No applicable service

Table 2.6 Termination of the Services During the Operation

Node Type/ Service (Daemon Name)	DHCP (dhcpd)	TFTP (xinetd)	HTTP (httpd)	NTP (chronyd)	NFS (nfsd, etc.)	pxinstd (pxinstd)
Installer node (System management node)	Yes	(*1)	(*1)	(*2)	-	Yes
Relay node (Compute cluster sub management node)	-	(*1)	(*1)	(*2)	-	Yes

Node Type/ Service (Daemon Name)	DHCP (dhcpd)	TFTP (xinetd)	HTTP (httpd)	NTP (chronyd)	NFS (nfsd, etc.)	pxinstd (pxinstd)
Installation target node (Compute cluster management node, login node, etc.)	-	-	-	(*2)	-	-
FX server (Boot I/O node)	No	No	-	(*2)	No	-
FX server (Except for boot I/O node)	-	-	-	(*2)	No	-

(*1) Stop these services during system operation, except for performing maintenance tasks using the installation function (Apply or remove packages, back up or restore nodes, rebuild nodes.)
(*2) The service can be terminated. However, it is not recommended to terminate the service because a time lag occurs if the service stops for a long time, which may affect a job operation.

The table below shows the influences when each service is terminated.

Table 2.7 Influences Due to Service Termination

Node Type	Service	Influences Due to Service Termination
Installer node (System management node)	DHCP	- Unable to install the OSs of all nodes - Unable to back up and restore all nodes
	TFTP	Unable to install the OSs of the nodes (the nodes that do not pass a relay node) directly under the installer node
	HTTP	- Unable to install the OSs of the nodes directly under the installer node - Unable to apply a package to the nodes directly under the installer node
	NTP	Unable to synchronize the time of the nodes directly under the installer node
	pxinstd	Unable to check the installation status during the reconfiguration, backup, and restoration of nodes
Relay node (Compute cluster sub management node)	TFTP	Unable to install the OSs of the nodes under a relay node, except for the diskless node in FX server
	HTTP	- Unable to install the OSs of the nodes under a relay node - Unable to apply a package for the nodes under a relay node
	NTP	Unable to synchronize the time of the nodes under a relay node
	pxinstd	Unable to check the installation status during the reconfiguration, backup, and restoration of the nodes under a relay node
FX server (Boot I/O node)	DHCP	Unable to start the nodes in a BoB
	TFTP	Unable to start the nodes in a BoB
	NTP	Unable to synchronize the time of the nodes in a BoB
	NFS	Unable to operate the nodes in a BoB
FX server (Except for boot I/O node)	NTP	Unable to synchronize the time of the nodes
	NFS	Unable to operate the node

2.8.1.4 Configuration for High Availability

The installation function realizes the high-availability using the node redundancy and node switching function (failover function).
Both active and standby nodes can have a function as a repository server or an install server, by introducing redundancy in the configuration of the system management node, which is an installer node, and the compute cluster sub management node, which is a relay node.

If an error occurs in the active node, a failover is triggered by the system monitoring function. By doing so, the installation function service can continue operating. The table below shows the resources and settings to which redundancy is provided for the installation function.

Table 2.8 Resources to Provide Redundancy for the Installation Function

Node Type	Resource/Setting		Method to Realize High Availability	Start Status of Service (Yes: Started, No: Not Started)	
				Active	Standby
Installer node (System management node)	Repository		Provide redundancy to node	No service	No service
	Settings and materials for OS installation	DHCP server setting	Provide redundancy to node	Yes	No
		TFTP server setting	Provide redundancy to node	Yes	Yes
		HTTP server setting	Provide redundancy to node	Yes	Yes
		NTP server setting	Provide redundancy to node	Yes	Yes
Relay node (Compute cluster sub management node)	Repository		Provide redundancy to node	No service	No service
	Settings and materials for OS installation	TFTP server setting	Provide redundancy to node	Yes	Yes
		HTTP server setting	Provide redundancy to node	Yes	Yes
		NTP server setting	Provide redundancy to node	Yes	Yes

The services, except for DHCP, are always activated on both active and standby nodes.
A client can use a service without taking into account which node is active by accessing through a virtual IP address (the representative IP address of the management network) that is activated by the system management function.

A service is not activated on the standby node because one DHCP service needs to be used in the same network segment. A service is changed when switching between the active and standby nodes.

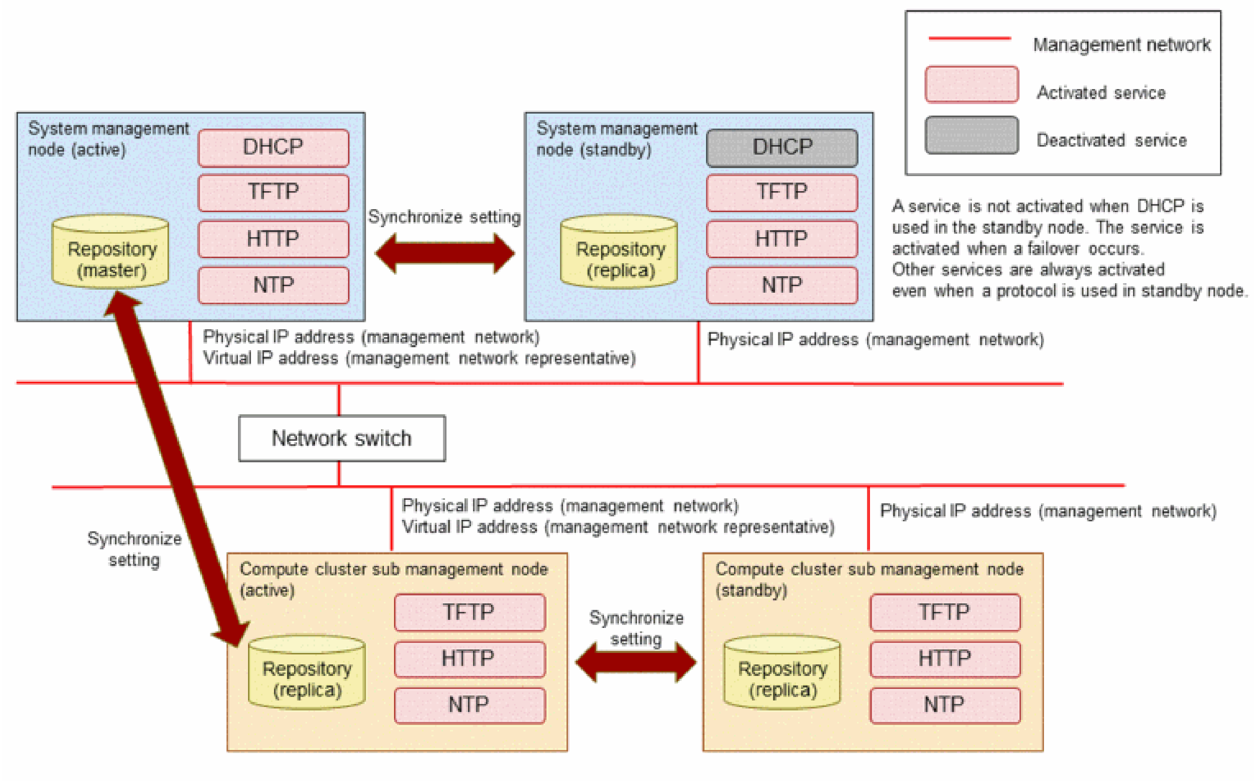
See

For details on failover, see "Table 3.27 Operations for Each Node Type when Detecting Abnormality" in "3.3.1.1 Software Abnormality Monitoring."

The system monitoring function enables a status monitoring failover for the active and standby nodes. The resources for the installation function are changed by switching the installation function when a failover occurs.

Synchronization with the standby installer node and the standby relay node must be performed using the synchronization function when the active installer node setting is changed or a repository configuration is changed (package registration and deletion, etc.), because the resources among the active and standby nodes are not automatically synchronized. The figure below shows the configuration to realize the high availability of the installation function.

Figure 2.51 Configuration for High Availability


Management network
Activated service
Deactivated service
System management node (active)
DHCP
TFTP
HTTP
NTP
Repository (master)
Synchronize setting
System management node (standby)
DHCP
TFTP
HTTP
NTP
Repository (replica)
A service is not activated when DHCP is used in the standby node. The service is activated when a failover occurs. Other services are always activated even when a protocol is used in standby node.
Physical IP address (management network)
Virtual IP address (management network representative)
Physical IP address (management network)
Network switch
Synchronize setting
Physical IP address (management network)
Virtual IP address (management network representative)
Physical IP address (management network)
Compute cluster sub management node (active)
TFTP
HTTP
NTP
Repository (replica)
Synchronize setting
Compute cluster sub management node (standby)
TFTP
HTTP
NTP
Repository (replica)

2.8.1.5 Overview of Network Installation

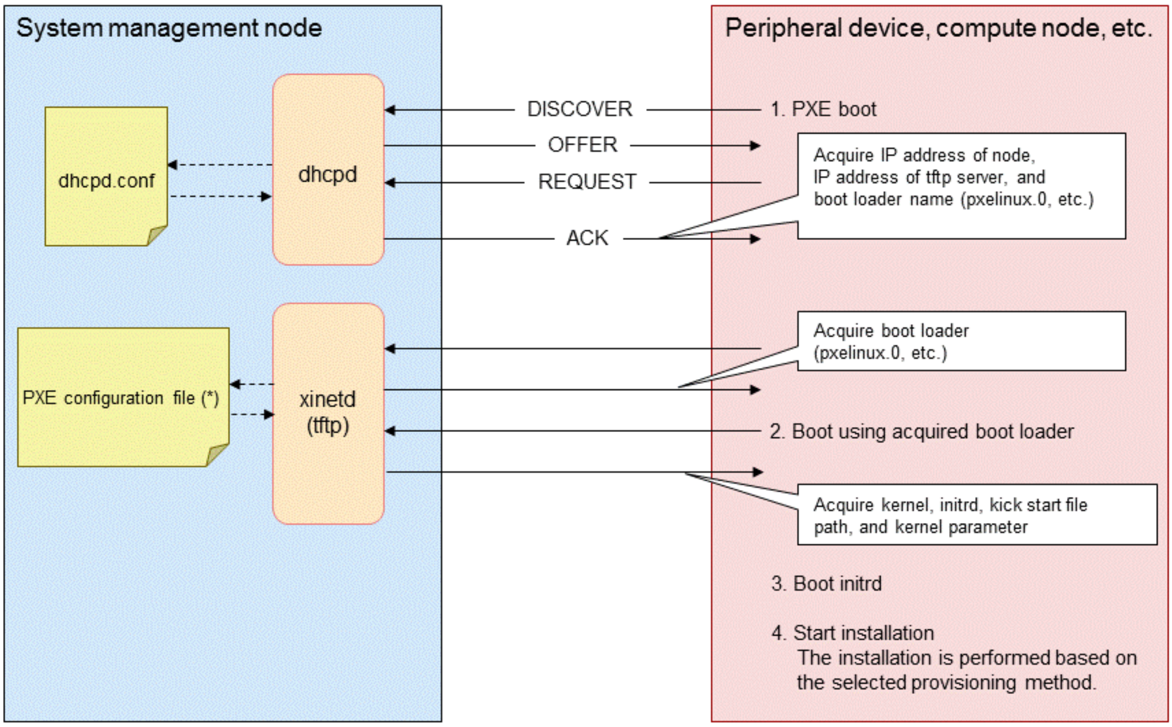
When an OS is installed using the installation function, the OS is network installed based on the PXE boot mechanism. The OS installation starts once an IP address for installation, kernel image, and initrd image have been acquired by PXE boot. The PXE boot operation is the same even any provisioning method that is supported by the installation function is selected. However, the diskless node in FX server is exception because a node does not need to start when installing an OS.

See

For details on the provisioning method supported by the installation function, see "2.8.2.2 Provisioning Method."

The figure below shows the OS installation flow using PXE boot.

Figure 2.52 Network Installation Flow


System management node
dhcpd.conf
dhcpd
PXE configuration file (*)
xinetd
(tftp)
Peripheral device, compute node, etc.
DISCOVER
OFFER
REQUEST
ACK
1. PXE boot
Acquire IP address of node,
IP address of tftp server, and
boot loader name (pxelinux.0, etc.)
Acquire boot loader
(pxelinux.0, etc.)
2. Boot using acquired boot loader
Acquire kernel, initrd, kick start file
path, and kernel parameter
3. Boot initrd
4. Start installation
The installation is performed based on
the selected provisioning method.

(*) The PXE configuration file is a configuration file in which information required for PXE boot is defined and a kernel file name, initrd file name, and other information are described.

The following is an example of the PXE configuration file.

```
# PXE configuration file for: ccm01
default Reinstall
# default localdisk
prompt 0

label localdisk
        kernel chain.c32
        append hd0 0

label Reinstall
        kernel vmlinuz-rhel-7.2-x86_64
        IPAPPEND 2
       append initrd=initrd-rhel-7.1-x86_64.img syslog='' ks=http://192.0.2.1/repos/rhel72/kickstart/
ks.192.0.2.2 ksdevice=bootif biosdevname=0 text noipv6 kssendmac selinux=0 nicdelay=60 linksleep=60
dhcptimeout=120 console=ttyS0
```

A PXE configuration file name used for the installation function is defined as eight uppercase hexadecimal digits that form an IP address.

- When an IP address assigned to DHCP is 192.0.2.1

  C0000201

- When an IP address assigned to DHCP is 192.0.2.2

  C0000202

2.8.1.6 Associating a Host Name with an IP Address

In this section, associating a host name with an IP address is called a name resolution.
When a /etc/hosts file is automatically created using the installation function, the system administrator performs name resolution by

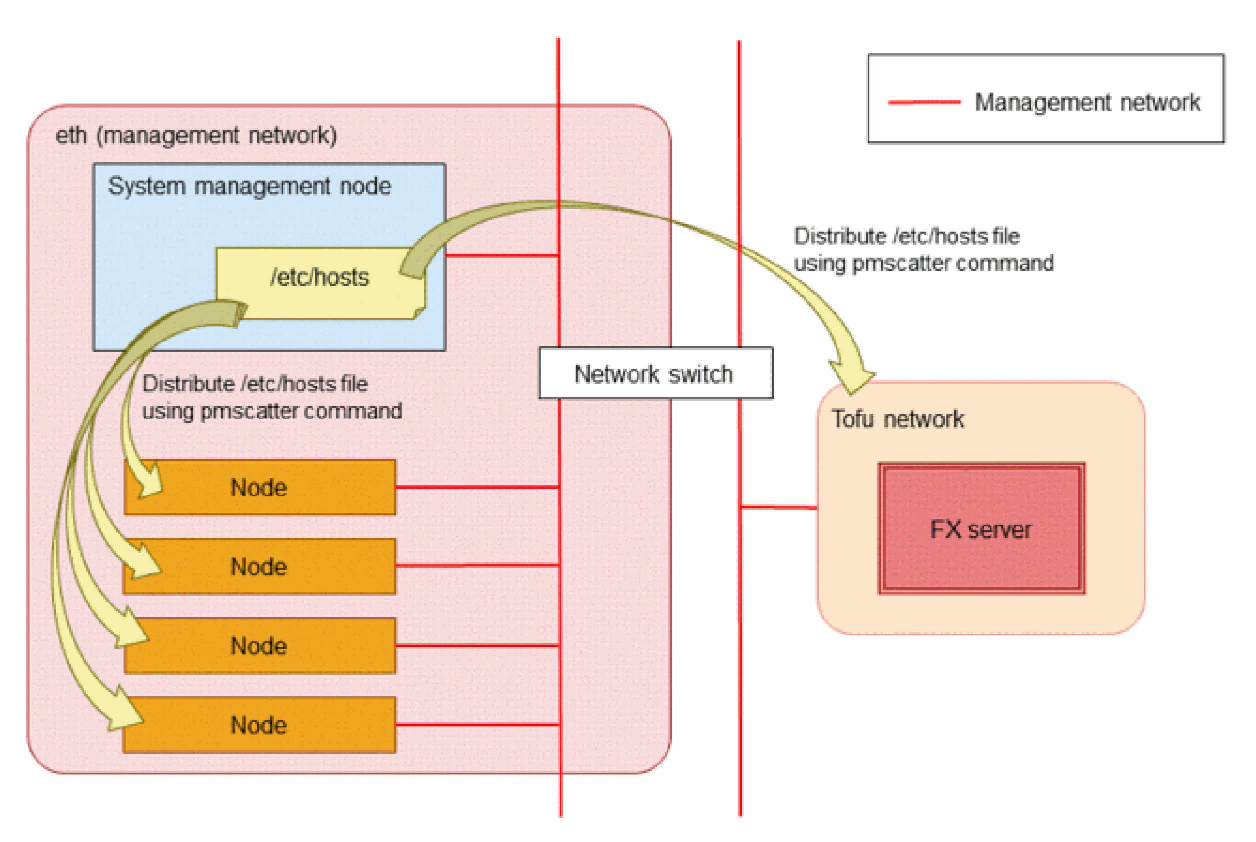
distributing the /etc/hosts file, which is automatically created on the system management node, to all nodes. The system administrator can easily distribute the file even in a large-scale environment using the batch operation function of the system management function.

See

For details on file distribution using the batch operation function, see "3.5.3 Delivering Files Collectively."

The diagram below shows how a /etc/hosts file is distributed. The system administrator can freely edit the automatically created /etc/hosts file. The system administrator needs to manually set the DNS server setting and LDAP linkage, as required.

Figure 2.53 Name Resolution Using the Installation Function



See

For details on settings for the automatic creation of a /etc/hosts file, see "2.8.4 Initial Setting of the Installation Function."

2.8.1.7 User Management of an OS

A function for user management is not provided in the installation function. User information is managed based on the procedure using functions, such as OS standard functions and the system management batch operation function.

See

For the user management procedure, see "Adding a User" in the "Job Operation Software Administrator's Guide for Maintenance."

2.8.1.8 Time Synchronization Method

Time synchronization in the system is performed using chronyd (chrony suite). The time synchronization setting is automatically configured during the node installation. A reference server is set based on the hierarchical structure of the installation function.

See

For details on the hierarchical structure of the installation function, see "2.8.1.1 Basic Configuration of Installation Function."

2.8.2 OS Installation on PRIMERGY server

A function to install an OS is provided to PRIMERGY server and the hardware of other companies (*).
An OS installation starts when the system administrator powers on a target node after configuring the necessary settings to network install an OS using this function on the installer node.

(*) The hardware of other companies can use this function if the hardware model supports PXE boot via Ethernet. If a model needs a specific kernel module to identify a network device or disk device at installation, such a model can be supported using the kernel module installation function provided by the installation function.

See

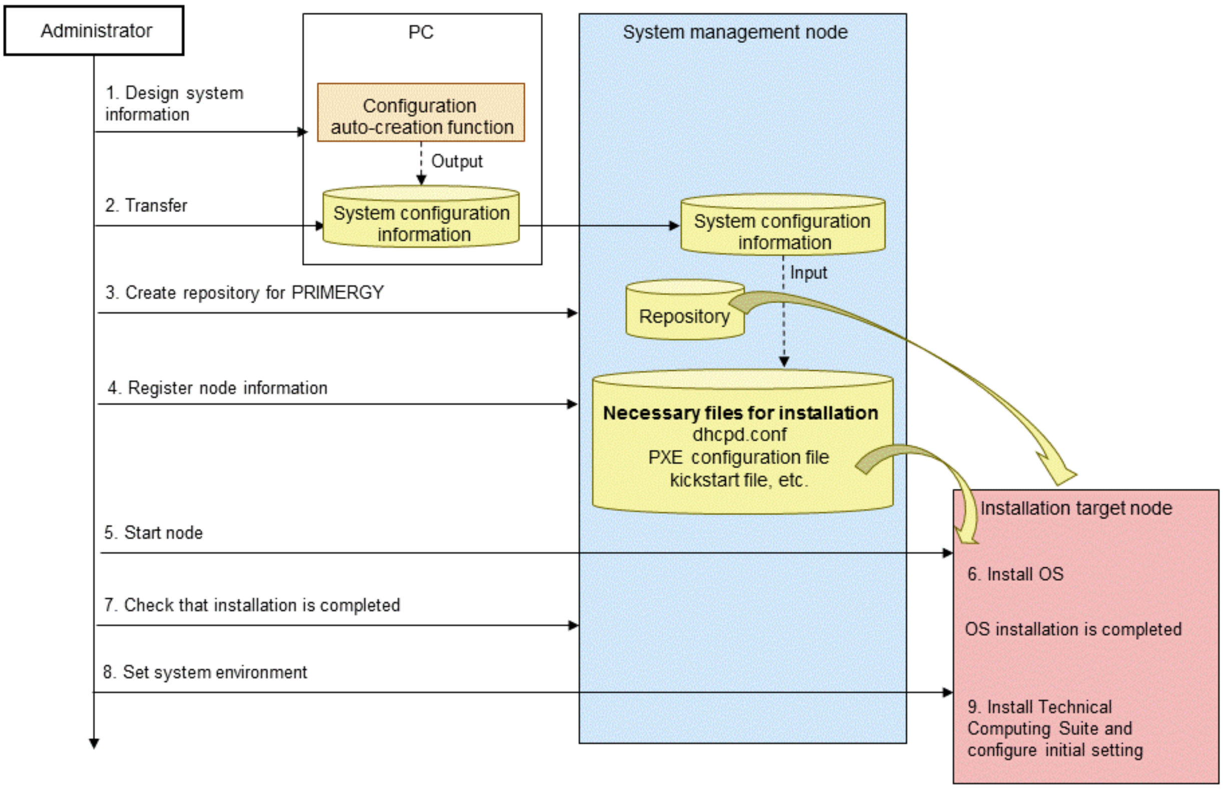
For details on the kernel module installation function, see "2.8.10 Installing a Kernel Module."

2.8.2.1 PRIMERGY server Installation Flow

The figure below shows the installation workflow using the installation function of PRIMERGY server.

A system environment can be automatically set when installing an OS.

Figure 2.54 PRIMERGY server Installation Workflow


Administrator
PC
System management node
1. Design system information
Configuration auto-creation function
Output
2. Transfer
System configuration information
System configuration information
Input
3. Create repository for PRIMERGY
Repository
4. Register node information
Necessary files for installation
dhcpd.conf
PXE configuration file
kickstart file, etc.
Installation target node
5. Start node
6. Install OS
7. Check that installation is completed
OS installation is completed
8. Set system environment
9. Install Technical Computing Suite and configure initial setting

2.8.2.2 Provisioning Method

The installation function supports multiple methods to install an OS in order to be able to select a provisioning method based on the customer's system configuration and installation scene. The table below shows the methods supported in PRIMERGY server.

Information

- The "provisioning" using the installation function means that an OS is installed on a node and the environment setting is configured.
- The provisioning method for FX server is an image-based OS installation.

Table 2.9 Supported OS Installation Method (PRIMERGY server)

OS Installation Method	Overview	Application and Selection Standards
Image-based OS installation	The system administrator installs an OS by distributing the OS image that is created using the OS image creation function. An OS can be speedily installed to distribute a root file system (rootfs) created on the installer node.	Recommended method Since an OS can be speedily installed, this is a method recommended at the initial installation.
Package-based OS installation	An OS is installed using the standard RHEL installation function (anaconda, kickstart, etc.). An OS is installed by downloading a necessary RPM package from a repository on an installer node and applying it to rootfs. This is the automated version of the manual installation using a general DVD.	This installation method is assumed to be used when a kickstart file is uniquely customized and detailed settings are configured at installation. This method is targeted at system administrators with high skills.
Restoring backup images	An OS is installed by restoring a disk image (cloning) using the backup/restore function. OS installation is performed using the disk images backed up on a pre-installed node. The settings are applied to the installation target node using the Job Operation Software settings and the disk images backed up after the OS is set.	This method is used to save time to configure the Job Operation Software or any user settings after installing an OS. This installation method is also assumed to be used to recover hardware by managing the generation of the disk images when the hardware fails or to switch the version of a fix package.

2.8.2.3 Hardware Setting of PRIMERGY server

2.8.2.3.1 IPMI Setting

In PRIMERGY server, the power can be controlled (control with the papwrctl command of the system management function, or ipmitool) by setting an IP address for IPMI on BIOS or UEFI beforehand.
The installation function supports a method to automatically set an IP address for IPMI without requiring setting the address beforehand. In this case, the power must be manually turned on during an OS installation.

An OS can be installed using one of the following two patterns.

- An IP address for IPMI is set on BIOS or UEFI in advance so that the power can be controlled remotely (using the papwrctl command, ipmitool. etc.). Then an OS is installed by remotely powering on.
- An OS is installed by manually turning on the power at the initial installation without setting an IP address for IPMI in advance. The IP address for IPMI is automatically set using the installation function when installing an OS.

The system administrator selects either method based on the number of nodes configuring the system and the system configuration.

See

An IP address for IPMI needs to be set using the configuration auto-creation function in order to automatically set the IP address. For details on how to set the setting, see "2.8.6 Creating Installation Target Node Settings."

2.8.2.3.2 Setting a Boot Order

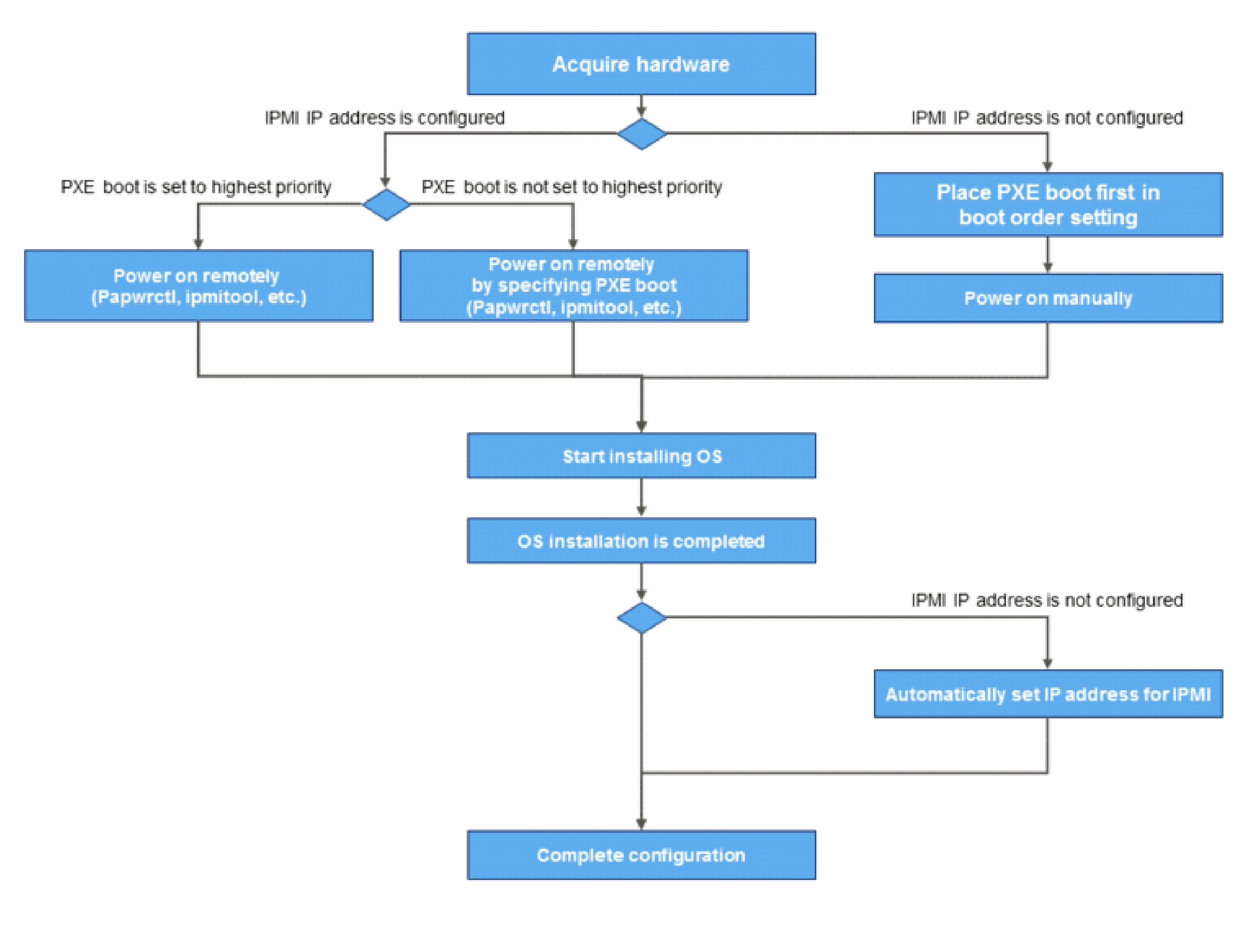
The system administrator needs to select PXE boot in the BIOS or UEFI boot order setting in order to use the PXE boot mechanism when installing an OS using the installation function.
In addition, if your hardware has multiple PXE-bootable network devices, set the BIOS boot priority of the NICs you want to use for PXE booting to be higher than the NICs you do not want to use for PXE booting.

If the papwrctl command still cannot PXE boot, set PXE boot as the highest priority in the BIOS and boot.

2.8.2.3.3 A Power-on Method That is Different According to the BIOS or UEFI Setting Status

This section describes a workflow when the setting of an IP address for IPMI and the PXE boot settings are made in advance and a workflow when these settings are not carried out beforehand.

Figure 2.55 Difference in Workflow Due to BIOS or UEFI Setting


Acquire hardware
IPMI IP address is configured
IPMI IP address is not configured
PXE boot is set to highest priority
PXE boot is not set to highest priority
Place PXE boot first in boot order setting
Power on remotely (Papwrctl, ipmitool, etc.)
Power on remotely by specifying PXE boot (Papwrctl, ipmitool, etc.)
Power on manually
Start installing OS
OS installation is completed
IPMI IP address is not configured
Automatically set IP address for IPMI
Complete configuration

2.8.2.3.4 Operation Example of PXE Boot

The following examples show how to PXE boot when the top priority is given to hard disk boot in a BIOS or UEFI boot order.

The following example shows PXE boot using ipmitool.

- Setting to forcibly PXE boot at the next start-up

```
# ipmitool -H ctrl_net -U user_name chassis bootdev pxe
Password:******                                           (*)
Set Boot Device to pxe
```

ctrl_net: IP address of the control network
user_name: User name for IPMI

(*) Enter the password for IPMI.

- Power-on

```
# ipmitool -H ctrl_net -U user_name chassis power on
Password:******                                             (*)
Chassis Power Control: Up/On
```

ctrl_net: IP address of the control network
user_name: User name for IPMI

(*) Enter the password for IPMI.

A node starts up when PXE boot is forcibly performed at the next start-up by specifying "chassis bootdev pxe." "chassis bootdev pxe" need not to be specified if PXE boot is placed first in the boot order setting on BIOS or UEFI.

The following example shows PXE boot using the papwrctl command.

```
# papwrctl -c clstname -n nodeid --boottype pxe on
[INFO] PSM 2004 papwrctl All power control was executed
```

clstname: Cluster name
nodeid: Node ID

See

For details on the papwrctl command, see "papwrctl command" in the "Job Operation Software Command Reference."

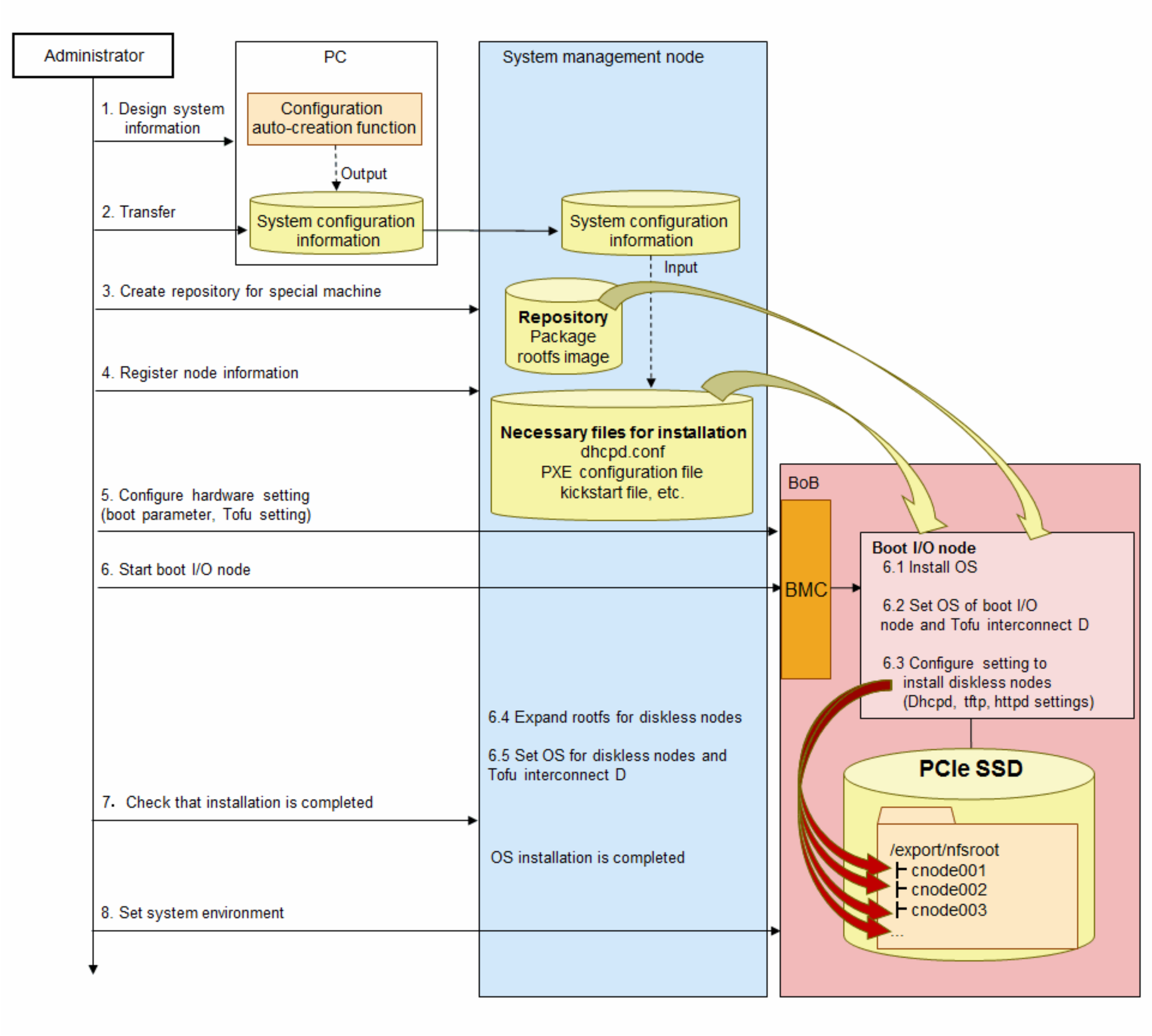
2.8.3 Installing an OS on FX server

The installation function provides a function to install an OS to FX server.
An OS installation method is different between the boot I/O node with disk devices and the diskless node (compute node, global I/O node, and storage I/O node) without disk devices in FX server.
Like PRIMERGY server, an OS is network installed using PXE boot on the boot I/O node.
An OS is installed on a diskless node by expanding rootfs in the NFS area for diskless node of the boot I/O node when installing the OS on the boot I/O node. An OS is installed by BoB unit.

2.8.3.1 Workflow to Install the OSs on FX server

The figure below shows the workflow to install OSs on FX server.
The OSs are installed by BoB unit. The OSs for all nodes in a BoB are installed when powering on the boot I/O node.

Figure 2.56 Workflow to Install the OSs on FX server

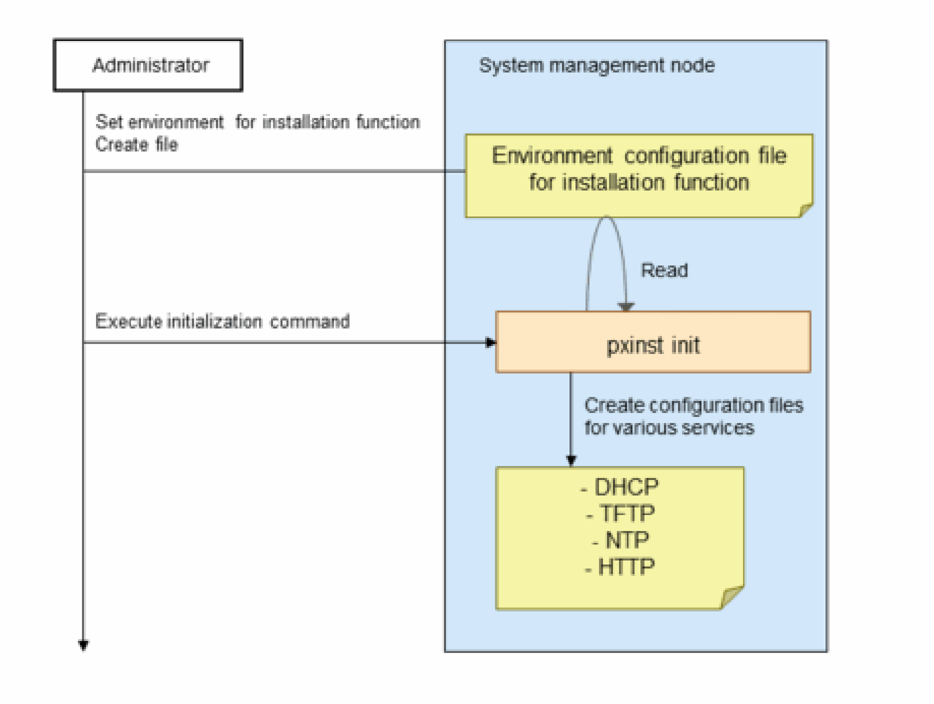

Administrator
PC
System management node
1. Design system information
Configuration auto-creation function
Output
2. Transfer
System configuration information
System configuration information
Input
3. Create repository for special machine
Repository
Package
rootfs image
4. Register node information
Necessary files for installation
dhcpd.conf
PXE configuration file
kickstart file, etc.
5. Configure hardware setting (boot parameter, Tofu setting)
BoB
BMC
6. Start boot I/O node
Boot I/O node
6.1 Install OS
6.2 Set OS of boot I/O node and Tofu interconnect D
6.3 Configure setting to install diskless nodes (Dhcpd, tftp, httpd settings)
6.4 Expand rootfs for diskless nodes
6.5 Set OS for diskless nodes and Tofu interconnect D
PCIe SSD
/export/nfsroot
├ cnode001
├ cnode002
├ cnode003
…
7. Check that installation is completed
OS installation is completed
8. Set system environment

2.8.4 Initial Setting of the Installation Function

This function is used to configure the initial server setting that is required as a precondition to operate the installation function.
The system administrator configures the initial setting by executing the initialization command after creating a configuration file, in which an operation environment is described, on the installer node.

This function is executed using the init sub-command of the pxinst command. The diagram below shows how the initial setting of the installation function is made.

Figure 2.57 Conceptual Diagram of the Initial Installation Function Setting


Administrator
System management node
Set environment for installation function
Create file
Environment configuration file
for installation function
Read
Execute initialization command
pxinst init
Create configuration files
for various services
- DHCP
- TFTP
- NTP
- HTTP

The table below provides the items set in the initial setting of the installation function.

Table 2.10 Functions for the Initial Setting of Installation Function

Setting Item	Overview
DHCP server setting	Sets the operation environment of the DHCP server for the installer node.
TFTP server setting	Sets the operation environment of the TFTP server for the installer node.
NTP server setting	Sets the operation environment of the NTP server and clients for the installer node.
HTTP server setting	Sets the operation environment of the HTTP server for the installer node.
ssh setting	Creates and registers an ssh authentication key that is used in a cluster.
Installer node setting	Configures the initial setting to operate a node, from which a command was executed, as an installer node.

See

For details on the initial setting of the installation function, see "3.8.1 Initial Settings of the Installation Function."

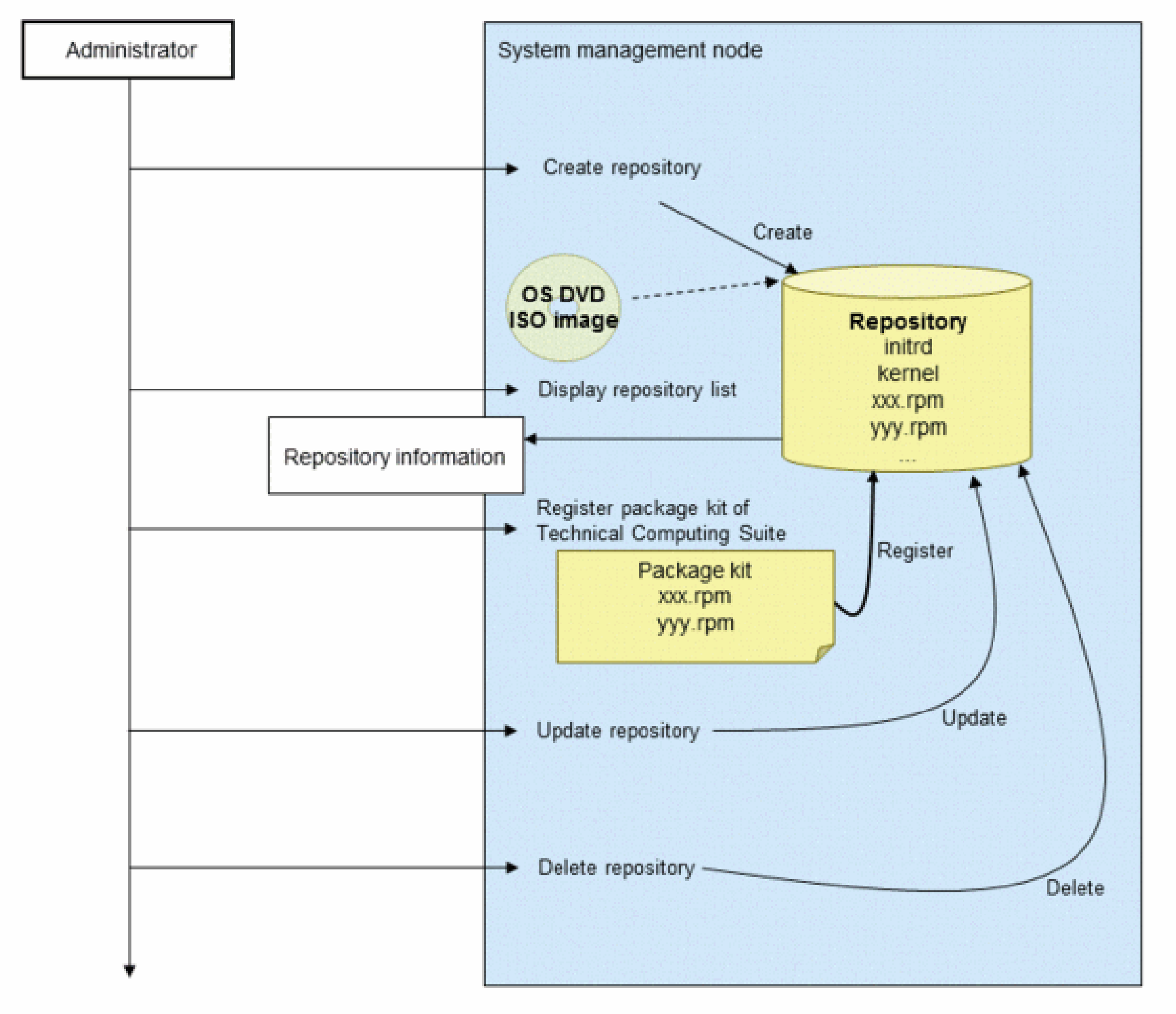
2.8.5 Managing a Repository

The repository management function is used to manage a repository to be referenced when installing an OS or applying a fix package. The system administrator can operate a package, such as installing, updating, deleting, and downgrading a package, by creating a repository from the DVD in an OS or an ISO image file. The yum command, which is an OS standard function, is used to operate a package.

The system administrator can easily operate multiple nodes using the batch operation function for the system management. The created repository can be used not only when installing an OS, but also when applying a fix package during the maintenance.

This function is executed with the repo sub-command of the pxinst command. The diagram below shows how the repository management function is used.

Figure 2.58 Conceptual Diagram of the Repository Management Function



The table below shows the list of the functions provided by the repository management function.

Table 2.11 Repository Management Function List

Function	Overview
Creating repository	A new repository is created from the OS DVD or an ISO image file.
Displaying repository list	The list of created repository information is displayed.
Registering package kit of Technical Computing Suite	The RPM package files of Technical Computing Suite are registered to a repository. A package can be applied to each node.
Updating repository	A repository is updated to the latest status after adding, updating, or deleting the package for the repository.
Deleting repository	A repository is deleted.

See

For details on the repository management function, see "3.8.2 Managing the Repository."

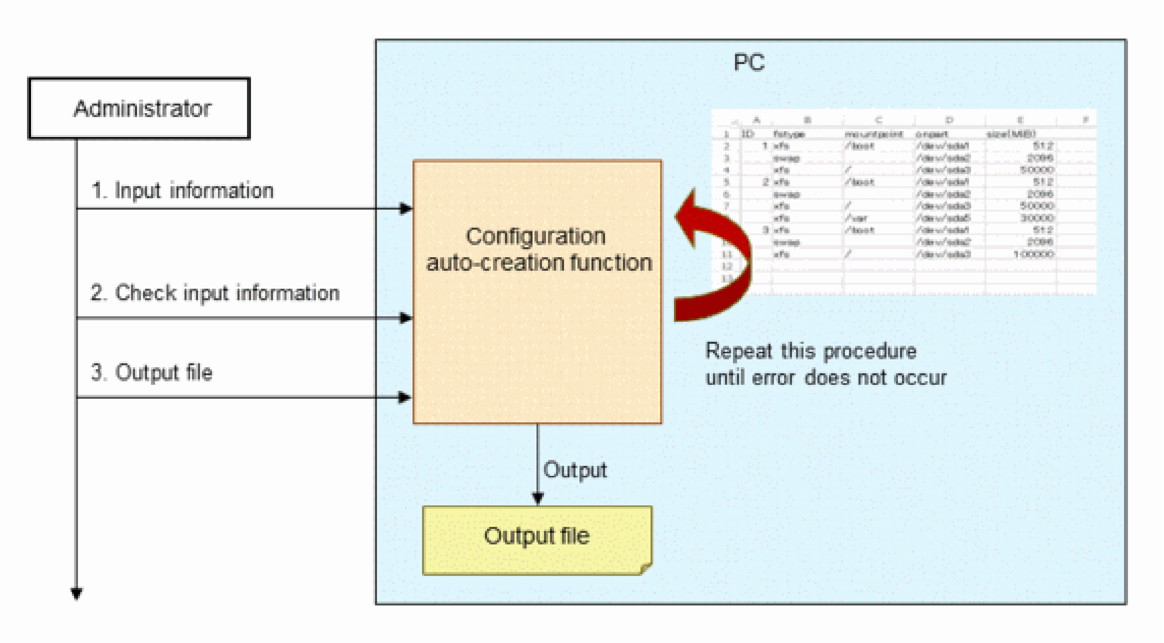
2.8.6 Creating Installation Target Node Settings

The system administrator enters the necessary information for OS installation in a design sheet for installation on a PC.
The installation function configures the settings to install the OS based on the configuration file that was output after entering the information.

A design sheet for installation is used when a new system is installed, a cluster is added, and a node is added. The design sheet is not used to delete a cluster or node.

The conceptual diagram below shows the procedure to use a design sheet for installation.

Figure 2.59 Conceptual Diagram of the Procedure to Use a Design Sheet for Installation


Administrator
1. Input information
2. Check input information
3. Output file
PC
Configuration auto-creation function
Repeat this procedure until error does not occur
Output
Output file

	A	B	C	D	E	F
1	ID	fstype	mountpoint	onpart	size(MiB)	
2	1	xfs	/boot	/dev/sda1	512	
3		swap		/dev/sda2	2096	
4		xfs	/	/dev/sda3	50000	
5	2	xfs	/boot	/dev/sda1	512	
6		swap		/dev/sda2	2096	
7		xfs	/	/dev/sda3	50000	
		xfs	/var	/dev/sda5	30000	
	3	xfs	/boot	/dev/sda1	512	
		swap		/dev/sda2	2096	
11		xfs	/	/dev/sda3	100000	
12						
13						

See

For details on the design sheet for installation, see "3.8.3 Creating Settings of the Installation Target Node."

2.8.6.1 Operation Environment

A design sheet for installation can be filled out in the following environment.

- Microsoft Windows 8.1, 10
- Microsoft Excel 2010, 2013 (32 bit, 64 bit), 2016 (32 bit, 64 bit)

2.8.7 Creating initrd for Installation

This function is used to create an initial RAM disk (initrd) that is required for OS installation. This function is executed using the initrd sub-command of the pxinst command.

See

For details on the initrd creation function, see "3.8.4 Creating initrd for Installation."

2.8.8 Creating a Root File System

This function is used to create a root file system (rootfs) that is required for image-based installation.

This function is executed using the rootfs sub-command of the pxinst command.

See

For details on the root file system creation function, see "3.8.5 Creating the Root File System."

2.8.9 Associating with ServerView Suite

This function is used to configure a setting to install a kernel module, which is provided from ServerView Suite and is required to install an OS on PRIMERGY server, and to automatically install software included in ServerView.

This function is executed using the svs sub-command of the pxinst command.

See

For details on the function to associate with ServerView Suite, see "3.8.6 Linkage With ServerView Suite."

2.8.10 Installing a Kernel Module

This function is used to install a kernel module for a boot image (initrd) used for installation.

This function is executed using the kmod sub-command of the pxinst command.

See

For details on the kernel module installation function, see "3.8.7 Incorporating a Kernel Module."

2.8.11 Registering Node Information

This function is used to set necessary node information for OS installation on the installer node. This function is used to add the node information entries of the DHCP server, create PXE configuration files and kickstart files, and so on.

This function is executed using the node sub-command of the pxinst command.

See

For details on the node information registration function, see "3.8.8 Registering Node Information."

2.8.12 Checking the Installation Status

This function is used to display the status to confirm the OS installation status.

This function is executed using the status sub-command of the pxinst command.

See

For details on the installation status confirmation function, see "3.8.9 Checking Installation States."

2.8.13 Synchronizing the Materials and Settings for Installation

When an installer node has a redundant configuration or when a relay node exists in the system, the system administrator needs to distribute (synchronize) a repository that is created on the active installer node or the materials and settings for installation to the standby installer node and relay node. The materials and settings need to be synchronized with these nodes in order to:

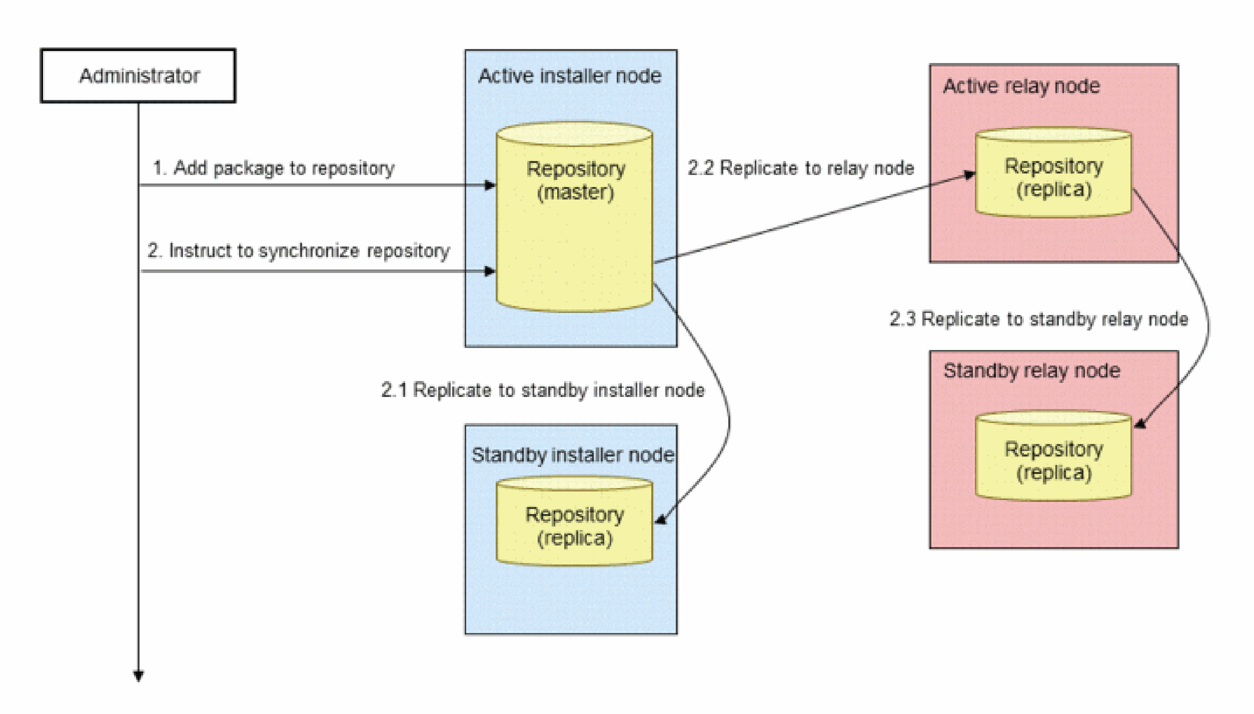
- Continue referencing a repository that is used to install or maintain a node when the active installer node fails.

- Implement a hierarchical load balance by replicating a repository to a relay node and retaining the materials for installation.

Since there are a wide variety of materials that need to be synchronized with the nodes, the system administrator provides a replication function so that the administrator can operate without taking into account the materials.
This function is executed using the sync sub-command of the pxinst command. For details on the replication function, see "3.8.10 Synchronization of the Materials and Settings for Installation." For details on how to realize the high availability of the installation function, see "2.8.1.4 Configuration for High Availability."

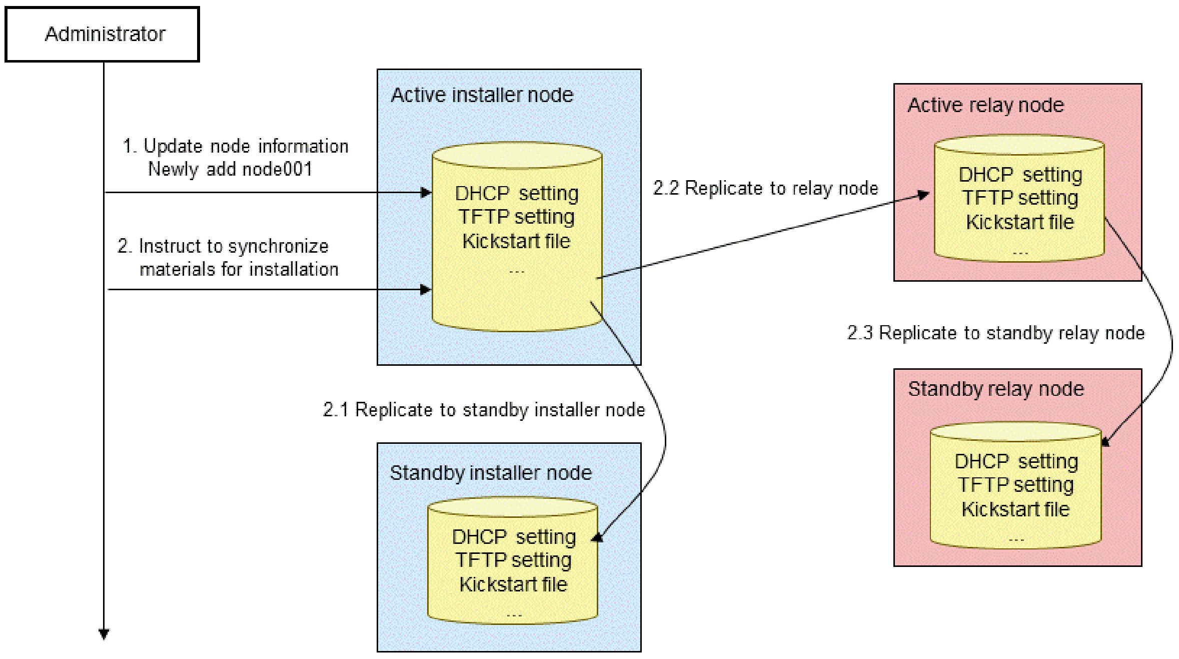
The conceptual diagram below shows how to replicate a repository.

Figure 2.60 Conceptual Diagram of Replicating a Repository


Administrator
Active installer node
Repository
(master)
1. Add package to repository
2. Instruct to synchronize repository
2.2 Replicate to relay node
Active relay node
Repository
(replica)
2.3 Replicate to standby relay node
Standby relay node
Repository
(replica)
2.1 Replicate to standby installer node
Standby installer node
Repository
(replica)

The conceptual diagram below shows how to replicate the materials for installation.

Figure 2.61 Conceptual Diagram of Replicating the Materials for Installation



If the materials could not synchronize with a target node (when a command is terminated abnormally), the setting may not be applied to the node. The system administrator needs to synchronize the materials with the node again after removing the cause based on the solution in the error message.

2.8.13.1 Overview of Using the Replication Function

If the repository and node information is registered (see "2.8.11 Registering Node Information") after installing the standby installer node or relay node, the system administrator needs to synchronize the setting with the standby installer node or relay node.

When the standby installer node or relay node is installed, the setting is automatically synchronized with the active installer. Therefore, the system administrator does not need to manually synchronize the setting with the active installer using the replication function.

The table below provides the situations when the replication function needs to be used for synchronization.

Table 2.12 Situations When Synchronization is Necessary

Situations When Synchronization is Necessary	Purpose
Creating or deleting repository	To apply the information of a created or deleted repository
Adding or deleting package to/from repository	To apply the information of an added or deleted package
Registering node information	To apply the registered node information.
Updating node information (manually updating file)	To apply the updated node information.

An operation may not be able to continue normally, if a synchronization is not carried out manually after performing an operation that requires synchronization. In this case, the system administrator needs to handle the problem by referencing the error message or "Job Operation Software Troubleshooting" based on the situation.

The table below provides the common errors that occur if a synchronization is not performed.

Table 2.13 Common Errors that Occur If a Synchronization is not Performed, and the Solution

Situations When Synchronization is Necessary	Errors That Occur When Synchronization is not Performed	Detection Method and Solution
Creating repository	Fail to install the OSs of the nodes under a relay node because the created repository information is not applied to the relay node.	Solve the problem by using the installation status confirmation function (see "3.8.9 Checking Installation States") and checking the console of the installation target node based on "Job Operation Software Troubleshooting."
Adding package to repository	Fail to apply an update package by executing the yum command to the nodes under a relay node, because the updated repository information is not applied to the relay node.	Synchronize the repository setting according to the message because the existence of the update package can be checked with the yum command.
Registering node information	Fail to install the nodes under a relay node because the registered node information is not applied to the relay node.	Solve the problem by using the installation status confirmation function (see "3.8.9 Checking Installation States") and checking the console of the installation target node based on "Job Operation Software Troubleshooting."
Updating node Information (manually updating file)	Fail to install the nodes under a relay node because the registered node information is not applied to the relay node.	Solve the problem by using the installation status confirmation function (see "3.8.9 Checking Installation States") and checking the console of the installation target node based on "Job Operation Software Troubleshooting."

See

For details on the replication function, see "3.8.10 Synchronization of the Materials and Settings for Installation."

2.8.14 Checking the Service Operation Status

The service status check function is used to check whether the services required to operate the installation function are running.
This function only checks the service operation status and does not check whether the setting of each service is correct or not. The system administrator checks the necessary service status using this function before performing the maintenance with the installation function. This function prevents a problem occurring due to the termination of a service during the maintenance. The table below shows the nodes and services supported by this function.

Yes: Services that can be checked
-: Services that do not exist

Table 2.14 Nodes and Services that can be Checked Using the Service Status Check Function

Node Type/ Service (Daemon Name)	DHCP (dhcpd)	TFTP (xinetd)	HTTP (httpd)	NTP (chronyd)	NFS (nfsd, etc.)
Installer node (System management node)	Yes	Yes	Yes	-	Yes
Relay node (Compute cluster sub management node)	-	Yes	Yes	-	Yes
FX server (Boot I/O node)	Yes	Yes	-	Yes	-

See

For details on the function to check the service status, see "3.8.11 Checking the Service Operation State."

2.9 Overview of the Backup/Restore Function

If a problem occurs due to a hard disk failure or when applying a fix package, it takes a lot of time and effort for the system administrator to start over from installing the OSs on the nodes in order to restore the nodes to the status before the problem occurs. The backup/restore function is provided to reduce such a time and effort. This function is used to restore a node from the disk images.
The system administrator can perform the following operations using the backup/restore function.

- Disk image backup
  The disk device contents of a specified node is saved as a disk image. The disk image can be saved on any node. The saved disk image is used by the restore function when configuring a node or a problem occurs.

- Restoring a disk image
  A backed up disk image is restored on the original node or another node. A node can be speedily restored to the previous status by restoring the collected disk image when a configuration is performed by replicating the node, when a hard disk fails, when a fix package is applied, or when other problems occur.

- Checking the status
  The progress of a node backup/restore process and its success/failure are displayed.

See

- For details on the backup of disk images, see "3.9.2 Backup of the Disk Image."

- For details on the restoration of disk images, see "3.9.3 Restoring a Disk Image."

- For details on the status confirmation, see "3.9.4 Checking Status."

A disk image is backed up or restored using PXE boot.
This mechanism ensures that a disk image with consistent data is backed up and a node is restored to the previous status.
The restoration method using the backup/restore function enables a node to be safely and speedily operable again.
The backup/restore function can be used when a problem occurs, as well as in various situations, such as when replicating (cloning) a node and when switching the version of a fix package.

2.9.1 Preconditions to Use the Backup/Restore Function

As the preconditions to use the backup/restore function, the server functions, which describe in the subsequent section, must be operated, and node information must be registered using the installation function. The following sections describe the details.

2.9.1.1 Necessary Server Functions

The backup/restore function can back up or restore a disk image using the following functions.

- PXE boot that uses the network installation mechanism of the installation function

- Mounting a directory via NFS to which a disk image is saved

The table below shows the functions necessary for each node.

Note

- The server function settings must be configured and the server functions must be operated using the installation function in order to execute PXE boot. For details on the network install mechanism and how to check the operation of the server functions used to network install, see "2.8 Overview of the Installation Function."

- The system administrator must configure the NFS server public setting for a directory where a disk image is saved. In addition, the NFS server must be in operation in order to mount the directory using NFS. For details on the NFS server function setting and the operation check, see the OS manuals for the NFS server.

Table 2.15 Functions Required for the Active System Management Node

Function	Description
DHCP server	Is used to assign an IP address when a target node starts using PXE boot.
TFTP server	Is used to download a kernel or initrd for backing up or restoring a target node.
HTTP server	Enables a target node to acquire the node-specific information, etc.

Table 2.16 Functions Required for a Node to Which a Disk Image is Saved

Function	Description
NFS server	Is used to mount a directory via NFS from/to which a disk image is read/write. When backing up a disk image, the directory is mounted using NFS from a target node to write the disk image. When restoring the disk image, the directory is mounted using NFS from the target node to read the disk image.

Table 2.17 Functions Required for a Node to be Backed Up or Restored

Function	Description
PXE function	This function is supported by the hardware. This function is used to PXE boot a node to transfer a file system for backing up or restoring to the node, or to start the file system.

2.9.1.2 Required Node Information

The backup/restore function uses the node information registered in advance using the installation function. The table below provides the node information required when backing up or restoring.

See

For details on the registration method, see "3.8.8 Registering Node Information" in "3.8 Details of the Installation Function."

Table 2.18 Required Node Information

Item	At Backup	At Restoration	Remarks
Host name, IP address, or node ID	Mandatory	Mandatory	The registration of the node information must be completed.
Target disks or partitions	Mandatory for PRIMERGY server	A partition must be specified when a specific partition is restored.	

2.9.2 Configuration Elements of the Backup/Restore Function

The table below provides the configuration elements of the backup/restore function.

Table 2.19 Configuration Elements of the Backup/Restore Function

Configuration	Description
Execution node of backup/restore command	This node is executed on the active system management node.
Node to which disk image is saved	A collected disk image is saved to this node. The disk image can be saved in a directory on a node whose NFS server public setting is completed.

Configuration	Description
Backup/restoration target	In PRIMERGY server, the backup/restoration target is a node to which network installation can be performed. In FX server, the backup/restoration target is a BoB.
Ethernet	The backup/restore function uses Ethernet. This function uses the management network for the nodes that are managed using the configuration management function.

2.9.3 Overview of Using the Backup/Restore Function

The backup/restore function is used when installing a new system or when changing the system configuration. The table below shows the actual situations when the backup/restore function is used.

Table 2.20 Situations When the Backup/Restore Function is Used

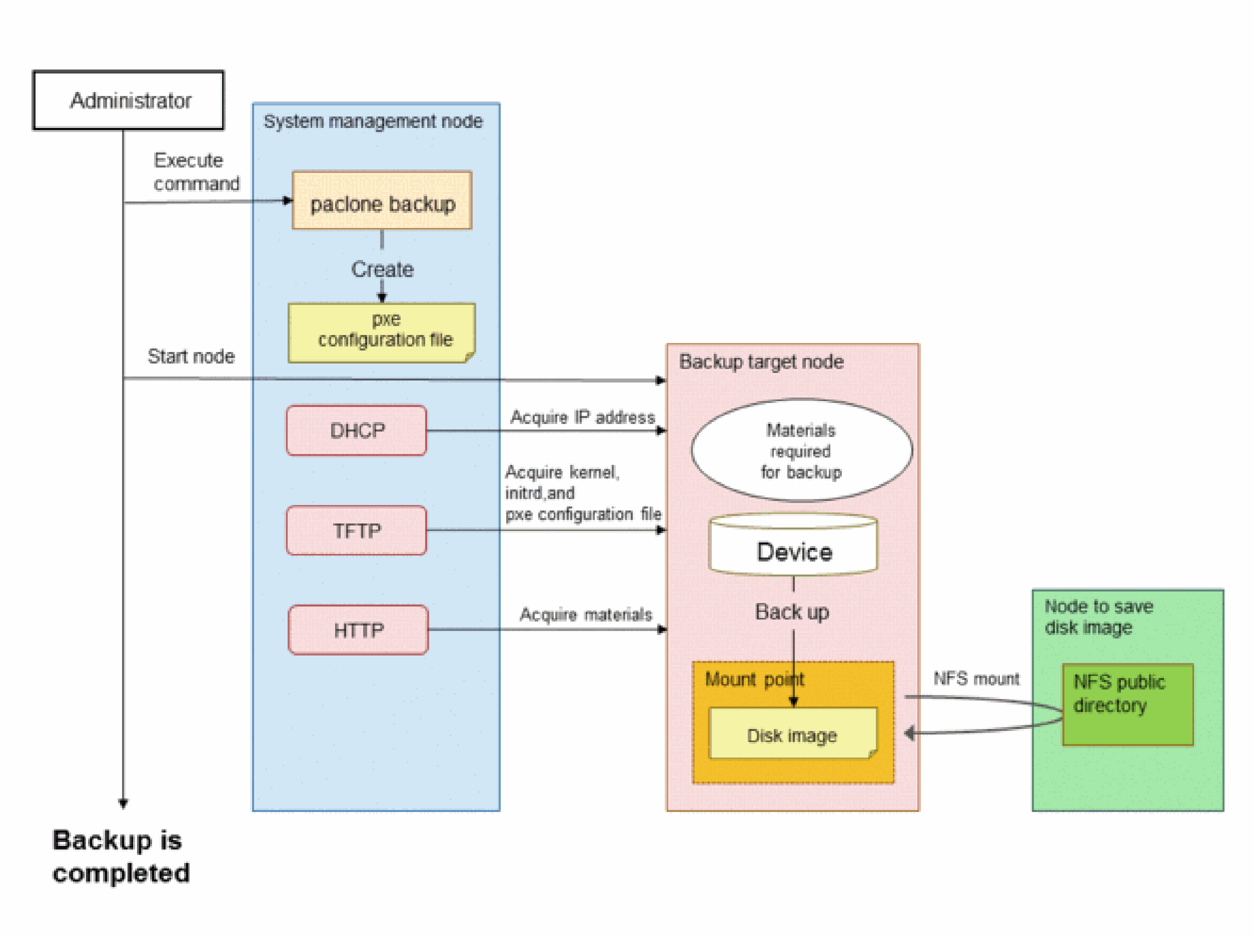
Classification	Operation	Description
Installation	Configuring cluster	A configured node is backed up, and then restored and configured on another node.
	Adding node	A collected disk image is restored on a node to be added.
Maintenance	Backing up latest node status	The latest node status is backed up when applying an OS update package or a fix package for Technical Computing Suite and when changing various settings.
	Restoring node	A node is restored using the restore function when a node needs to be restored due to hardware failure or when a problem occurs after applying a fix package.
	Changing configuration	A configuration is changed using the restore function when the configuration needs to be changed to switch the version of a fix package, and so on.

2.9.3.1 Overview of Backing up a Disk Image

The system administrator executes the backup sub-command of the paclone command to register backup information, and starts a target node to back up a disk image.
The diagram below shows the procedure to back up a disk image using the paclone command.

The diagram below is an example of saving a disk image to a node other than the system management node.

Figure 2.62 Conceptual Diagram of Backing Up a Disk Image

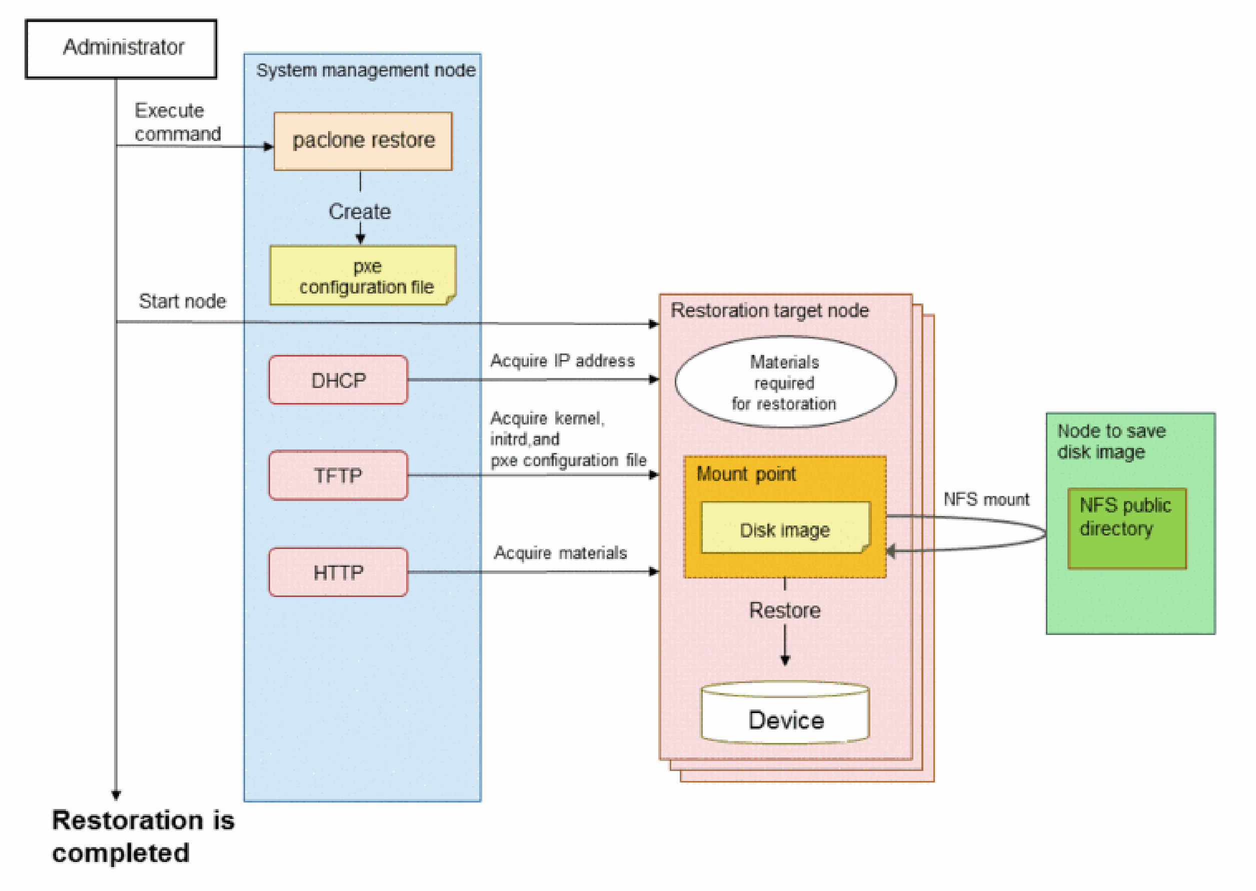


2.9.3.2 Overview of Restoring a Disk Image

The system administrator executes the restore sub-command of the paclone command to register the restoration information, and starts a target node to restore a disk image.
The diagram below shows the procedure to restore a disk image using the paclone command.

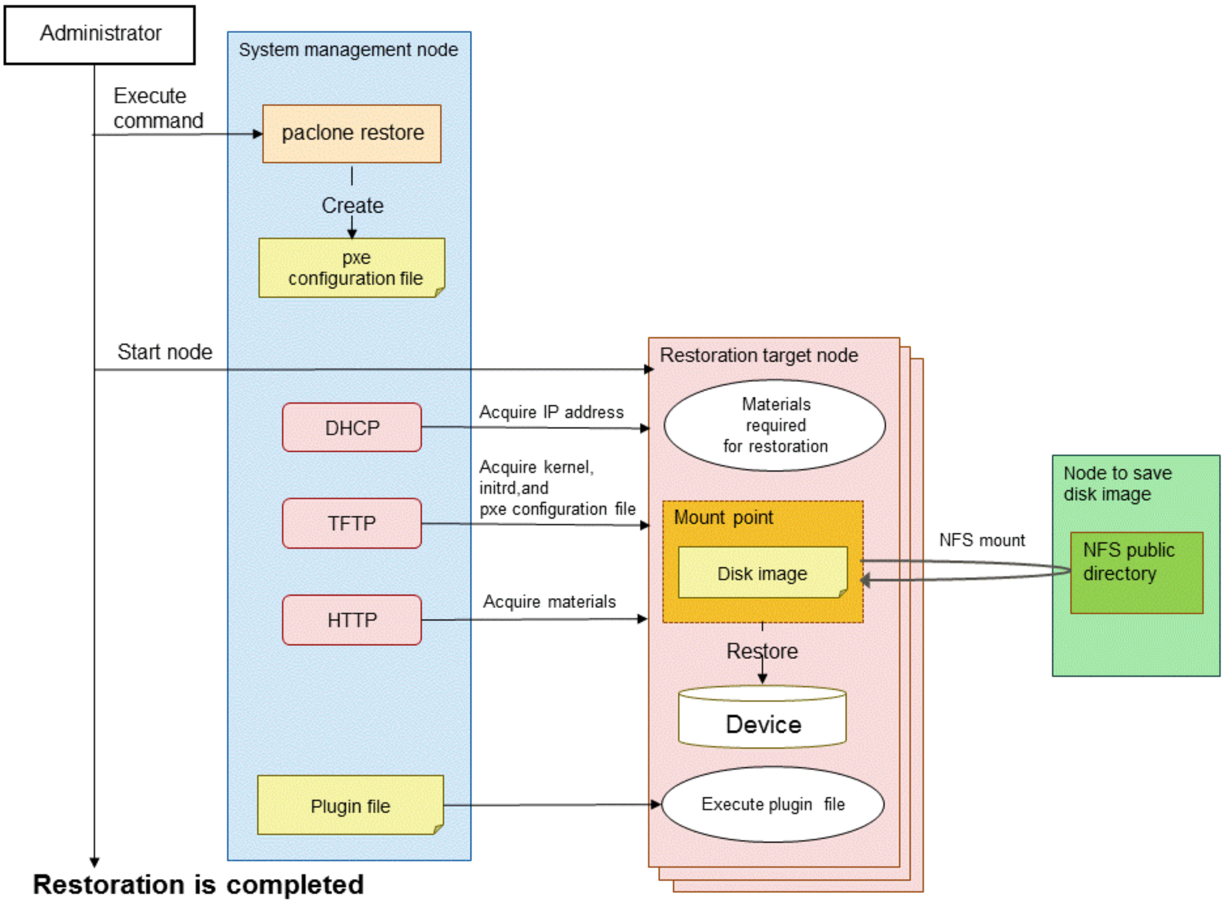
The diagram below is an example of restoring to multiple nodes a disk image that was saved on a node other than the system management node.

Figure 2.63 Conceptual Diagram of Restoring a Disk Image



When restoring a disk image, the plug-in command, in which a process is described, can be executed to a target node.
The diagram below shows the procedure to execute the plug-in command at restoration.
The diagram below is an example of restoring to multiple nodes a disk image that was saved on a node other than the system management node, to execute a specified plug-in command.

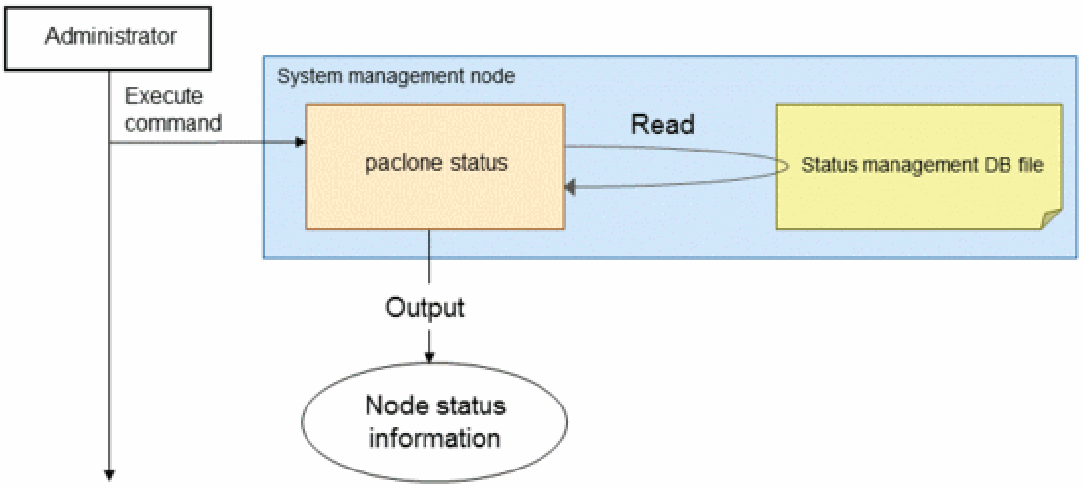
Figure 2.64 Conceptual Diagram of Executing Restoration Plug-In at Restoration



2.9.3.3 Overview of Checking the Status

The system administrator executes the status sub-command of the paclone command to display the backup/restore process status on a node. When checking the status, the status of each node is output by referencing the status management DB file.

Figure 2.65 Conceptual Diagram of Checking the Status



Chapter 3 Details of the System Management Function

This chapter describes the details on the functions that the system management function provides.

3.1 Details of the Configuration Management Function

Configure the configuration management function after deciding the cluster configuration of the system for introducing the Job Operation Software.

3.1.1 Cluster Configuration Information

When adding a cluster or node, consider the cluster configuration according to the size of the system and the applications of the node. To register the considered cluster configuration on the system, the system administrator is to describe necessary information when creating the pxsystem.conf file as described in "3.1.2 Registering Cluster Configuration Information."
This section describes information necessary for configuring the configuration management function.

See

- For the methods of thinking used when considering a cluster configuration, see "Considering Cluster Configurations" in the "Job Operation Software Setup Guide."
- This section shows examples of describing a designed cluster configuration in the pxsystem.conf file. For details on the format and items of this file, see "pxsystem.conf file" in the "Job Operation Software Command Reference."

Information

The description examples of the pxsystem.conf file described in this section may omit some of its file sections that this section does not describe.

3.1.1.1 Node-Specific Information

The system administrator is to use the configuration management function to register node-specific information such as IP address in the system. This information is intended for use by the functions of the Job Operation Software.
Node-specific information in the configuration management function is as follows.

Table 3.1 Node-Specific Information

Item	Description
ManageNetMaster	Representative IP address of the management network
ManageNet	IP address of the management network
ControlNetSrv	IP address of the control network (for the system management node only)
ControlNet	IP address of the control network
TofuNet	IP address of the Tofu network (for FX server only)
Coordinate	Tofu coordinates (for FX server only)
Model	Model name (which is PG for PRIMERGY server and FT for FX server)
PowerControlOnly	Setting that disables any functions other than the control function (*)

(*) For any devices such as disk unit and network switch, define them as the multiuse node of any node. This enables power control using the power control plugin of the system control function. When incorporating such devices to a cluster configuration, enable the setting that disables any functions other than the control function. This is to prevent the service monitoring or other functions of the Job Operation Software from running.
By doing so, only the system control function becomes effective among the functions of the Job Operation Software, disabling the system

monitoring function and the operation support function to operate the defined devices.
For details on power control using the power control plugin, see "3.2.1.3 Power Control Plugin."

Regarding node-specific information, describe sections for each node type when creating the pxsystem.conf file as described in "3.1.2 Registering Cluster Configuration Information."
The following is an example of defining a system management node in a redundant configuration. This describes the representative IP address of the management network and the SMM sections representing the system management node in the SMMGroup section.

```
Cluster {
    SMMGroup {
        ManageNetMaster = "192.0.2.1"
        SMM {
            ManageNet = "192.0.2.2"
            ControlNetSrv = "198.51.10.1"
            ControlNet = "198.51.100.1"
            Model = "PG"
        }
        SMM {
            ManageNet = "192.0.2.3"
            ControlNetSrv = "198.51.10.2"
            ControlNet = "198.51.100.2"
            Model = "PG"
        }
    }
}
```

See

The specific information of FX server is described in the "node information definition file for FX server" of the installation function, not in the pxsystem.conf file. For details on the node information definition file for FX server, see "3.8.8.6 Node Information Definition File for FX server" in "3.8 Details of the Installation Function."

3.1.1.2 Node Group Definition

When building a large-scale compute cluster, install a compute cluster sub management node, and then define its subordinate nodes as a node group. The purpose of this is for load sharing when the compute cluster management node monitors its subordinate nodes.
If defining a node group, describe the NodeGroup section when creating the pxsystem.conf file as described in "3.1.2 Registering Cluster Configuration Information."
The following is an example of defining a node group.

Node Group Definition for FX server

The following is an example of describing the NodeGroup section of the pxsystem.conf file when defining a node group for FX server. This example does not include the definition of anything other than node groups such as the system management node, as well as the description of the compute cluster sub management node (CCS section).

Example of (X,Y,Z) = (14,12,24) Configuration

In an (X,Y,Z) = (14,12,24) configuration, define 4 node groups whose number of boot groups is 216, and 9 node groups whose number of boot groups is 240 in a cluster.
Describe two or more NodeGroup sections, and then describe "NumberOfNodeGRoups" and "NumberOfBootGroups" in each of the NodeGroup sections.

```
Cluster {
    NodeGroup {
        CCS {
            Specific information of compute cluster sub management node
        }
        Model = "FT"
        NumberOfNodeGroups = "4"
        NumberOfBootGroups = "216"
    }
    NodeGroup {
        CCS {
            Specific information of compute cluster sub management node
        }
        Model = "FT"
        NumberOfNodeGroups = "9"
        NumberOfBootGroups = "240"
    }
}
```

Node Group Definition for PRIMERGY server

When defining the node group of the PRIMERGY server, describe node-specific information in the NodeGroup section.

```
Cluster {
    NodeGroup {
        CCS {
            Specific information of compute cluster sub management node
        }
        CN {
            Specific information of PRIMERGY server
        }
        CN {
            ...
    }
    NodeGroup {
        CCS {
            Specific information of compute cluster sub management node
        }
        CN {
            Specific information of PRIMERGY server
        }
        CN {
            ...
    }
}
```

3.1.1.3 Rule Allocation for IP Address

In a large-scale system, you need to register a large number of nodes as cluster configuration information.
The system administrator can collectively define the IP addresses of many nodes by describing the rules of calculating the IP address of a node in the pxsystem.conf file.
This is called the rule allocation for IP address.
Describe the IP address allocation rule in pxsystem.conf, the cluster configuration definition file, when adding a cluster or node. This makes it possible to define many nodes collectively and allocate serial IP addresses.

Information

The rule allocation for IP address provided by the configuration management function targets both the IP address of the management network and the IP address of the control network in a node of PRIMERGY server or any node.

To define the IP address allocation rule, describe the ManageNetRule section and the ControlNetRule section in the pxsystem.conf file.
There are two types of rule allocation for IP address: "Rule Allocation for Each Node Type" and "Rule Allocation for Each Node Group."

Rule Allocation for Each Node Type

You can collectively define many nodes by defining a network address, net mask, and starting IP address for each node type.
The following is an example of defining the IP address allocation rule in the case when adding 10 OSS nodes to the storage cluster "storage."

IP Address to be Allocated	Item	Value
IP address of management network	Network address	192.0.2.0
	Net mask	255.255.255.0
	Starting IP address	192.0.2.201
IP address of control network	Network address	198.51.100.0
	Net mask	255.255.255.0
	Starting IP address	198.51.100.201

```
Operation {
    Type = "add-node"
}
Cluster {
    ClusterName = "storage"
    MasterGroup {
        OSS {
            Model = "PG"
            NumberOfNodes = "10"
            ManageNetRule {
                NetworkAddress = "192.0.2.0"
                NetMask = "255.255.255.0"
                StartIPaddress = "192.0.2.201"
            }
            ControlNetRule {
                Networkaddress = "198.51.100.0"
                NetMask = "255.255.255.0"
                StartIPaddress = "198.51.100.201"
            }
        }
    }
}
```

Describing the rule as shown above adds the OSS nodes as follows.

Table 3.2 Example of Rule Allocation for IP Address

No.	Management Network	Control Network
1	192.0.2.201	198.51.100.201
2	192.0.2.202	198.51.100.202
3	192.0.2.203	198.51.100.203
4	192.0.2.204	198.51.100.204
5	192.0.2.205	198.51.100.205
6	192.0.2.206	198.51.100.206
7	192.0.2.207	198.51.100.207
8	192.0.2.208	198.51.100.208
9	192.0.2.209	198.51.100.209
10	192.0.2.210	198.51.100.210

By additionally describing both a unit as the separation of a mask and a mask to be incremented, you can also increment the specified mask value when allocating IP addresses to any number of nodes.
The following is an example of defining the IP address allocation rule in the case when adding 128 compute nodes of PRIMERGY server to the compute cluster "compute."

Table 3.3 Example of IP Address Allocation Rule

IP Address to be Allocated	Item	Value
IP address of management network	Network address	192.0.2.0
	Net mask	255.255.255.0
	Starting IP address	192.0.2.101
	Unit as separation of mask	Node
	Number of units	64 (*1)
	Mask to be incremented	255.255.255.0
IP address of control network	Network address	198.51.100.0
	Net mask	255.255.255.0
	Starting IP address	198.51.100.101
	Unit as separation of mask	Node
	Number of units	32 (*2)
	Mask to be incremented	255.255.246.0

(*1) A mask is separated for every 64 nodes.
(*2) A mask is separated for every 32 nodes.

```
Operation {
    Type = "add-node"
}
Cluster {
    ClusterName = "compute"
    MasterGroup {
        CN {
            Model = "PG"
            NumberOfNodes = "128"
            ManageNetRule {
                NetworkAddress = "192.0.2.0"
                NetMask = "255.255.255.0"
                StartIPaddress = "192.0.2.101"
                MaskUnit = "Node"
                NumberOfUnit = "64"
                IncrementMask = "255.255.255.0"
            }
            ControlNetRule {
                NetworkAddress = "198.51.100.0"
                NetMask = "255.255.255.0"
                StartIPaddress = "198.51.100.101"
                MaskUnit = "Node"
                NumberOfUnit = "32"
                IncrementMask = "255.255.246.0"
            }
        }
    }
}
```

Describing the rule as shown above adds the PRIMERGY server compute nodes as follows.

Table 3.4 Example of Rule Allocation for IP Address

No.	Management Network	Control Network
1	192.0.2.101	198.51.100.101
2	192.0.2.102	198.51.100.102
3	192.0.2.103	198.51.100.103

No.	Management Network	Control Network
4	192.0.2.104	198.51.100.104
Omitted		
31	192.0.2.131	198.51.100.131
32	192.0.2.132	198.51.100.132
33	192.0.2.133	198.51.110.101
34	192.0.2.134	198.51.110.102
Omitted		
63	192.0.2.163	198.51.110.131
64	192.0.2.164	198.51.110.132
65	192.0.3.101	198.51.120.101
66	192.0.3.102	198.51.120.102
Omitted		
95	192.0.3.131	198.51.120.131
96	192.0.3.132	198.51.120.132
97	192.0.3.133	198.51.130.101
98	192.0.3.134	198.51.130.102
Omitted		
125	192.0.3.161	198.51.130.129
126	192.0.3.162	198.51.130.130
127	192.0.3.163	198.51.130.131
128	192.0.3.164	198.51.130.132

Rule Allocation for Each Node Group

You can collectively define many nodes by defining a network address, net mask, and starting IP address in a node group.
The following is an example of defining the IP address allocation rule in the case when adding a node group to the compute cluster "compute."

IP Address to be Allocated	Item	Value
IP address of management network	Network address	192.0.2.0
	Net mask	255.255.255.0
	Starting IP address	192.0.2.201
IP address of control network	Network address	198.51.100.0
	Net mask	255.255.255.0
	Starting IP address	198.51.100.201

```
Cluster {
    ClusterName = "compute"
    NodeGroup {
        ManageNetRule {
            NetworkAddress = "192.0.2.0"
            NetMask = "255.255.255.0"
            StartIPaddress = "192.0.2.101"
        }
        ControlNetRule {
            NetworkAddress = "198.51.100.0"
            NetMask = "255.255.255.0"
            StartIPaddress = "198.51.100.101"
        }
        CCS {
            Model = "PG"
            NumberOfNodes = "2"
        }
        CN {
            Model = "PG"
            NumberOfNodes = "64"
        }
    }
}
```

Describing the rule as shown above adds a node group as follows.

Table 3.5 Example of Rule Allocation for IP Address

No.	Node Type	Management Network (Representative)	Management Network	Control Network
1	CCS	192.0.2.101 (*)	192.0.2.102	198.51.100.101
2	CCS	192.0.2.101 (*)	192.0.2.103	198.51.100.102
3	CN	-	192.0.2.104	198.51.100.103
4	CN	-	192.0.2.105	198.51.100.104
Omitted				
60	CN	-	192.0.2.161	198.51.100.160
61	CN	-	192.0.2.162	198.51.100.161
62	CN	-	192.0.2.163	198.51.100.162
63	CN	-	192.0.2.164	198.51.100.163
64	CN	-	192.0.2.165	198.51.100.164
65	CN	-	192.0.2.166	198.51.100.165
66	CN	-	192.0.2.167	198.51.100.166

(*) If the compute cluster sub management node is in a redundant configuration, the representative IP address is the IP address of the management network assigned to the first node group.

By describing an additional mask value and allocation unit, you can also increment the specified mask value when allocating IP addresses to any number of nodes.
The following is an example of defining the IP address allocation rule in the case when dividing 128 compute nodes of PRIMERGY server into 2 node groups and adding them to the compute cluster "compute."

IP Address to be Allocated	Rule	Value
IP address of management network	Network address	192.0.2.0
	Net mask	255.255.255.0
	Starting IP address	192.0.2.101
	Unit as separation of mask	Node
	Number of units	32 (*1)
	Mask to be incremented	255.255.246.0
IP address of control network	Network address	198.51.100.0
	Net mask	255.255.255.0

IP Address to be Allocated	Rule	Value
	Starting IP address	198.51.100.101
	Unit as separation of mask	NodeGroup
	Number of units	1 (*2)
	Mask to be incremented	255.255.255.0

(*1) A mask is separated for every 32 nodes (including the representative IP address).
(*2) A mask is separated for every one node group.

```
Cluster {
    ClusterName = "compute"
    NodeGroup {
        NumberOfNodeGroups = "2"
        ManageNetRule {
            NetworkAddress = "192.0.2.0"
            NetMask = "255.255.255.0"
            StartIPaddress = "192.0.2.101"
            MaskUnit = "Node"
            NumberOfUnit = "32"
            IncrementMask = "255.255.246.0"
        }
        ControlNetRule {
            NetworkAddress = "198.51.100.0"
            NetMask = "255.255.255.0"
            StartIPaddress = "198.51.100.101"
            MaskUnit = "NodeGroup"
            NumberOfUnit = "1"
            IncrementMask = "255.255.255.0"
        }
        CCS {
            Model = "PG"
            NumberOfNodes = "2"
        }
        CN {
            Model = "PG"
            NumberOfNodes = "64"
        }
    }
}
```

Describing the rule as shown above adds a node group as follows.

- First node group

  IP addresses are allocated to the nodes in the first node group as follows.

Table 3.6 Example of Rule Allocation for IP Address (First Node Group)

No.	Node Type	Management Network (Representative)	Management Network	Control Network
1	CCS	192.0.2.101	192.0.2.102	198.51.100.101
2	CCS	192.0.2.101	192.0.2.103	198.51.100.102
3	CN	-	192.0.2.104	198.51.100.103
4	CN	-	192.0.2.105	198.51.100.104
Omitted				
30	CN	-	192.0.2.131	198.51.100.130
31	CN	-	192.0.2.132	198.51.100.131

No.	Node Type	Management Network (Representative)	Management Network	Control Network
32	CN	-	192.0.12.101	198.51.100.132
33	CN	-	192.0.12.102	198.51.100.133
Omitted				
63	CN	-	192.0.12.132	198.51.100.163
64	CN	-	192.0.22.101	198.51.100.164
65	CN	-	192.0.22.102	198.51.100.165
66	CN	-	192.0.22.103	198.51.100.166

- Second node group

  IP addresses are allocated to the nodes of the second node group as follows.

Table 3.7 Example of Rule Allocation for IP Address (Second Node Group)

No.	Node Type	Management Network (Representative)	Management Network	Control Network
1	CCS	192.0.22.104	192.0.22.105	198.51.101.101
2	CCS	192.0.22.104	192.0.22.106	198.51.101.102
3	CN	-	192.0.22.107	198.51.101.103
4	CN	-	192.0.22.108	198.51.101.104
Omitted				
30	CN	-	192.0.22.134	198.51.101.130
31	CN	-	192.0.22.135	198.51.101.131
32	CN	-	192.0.32.101	198.51.101.132
33	CN	-	192.0.32.102	198.51.101.133
Omitted				
63	CN	-	192.0.32.132	198.51.101.163
64	CN	-	192.0.42.101	198.51.101.164
65	CN	-	192.0.42.102	198.51.101.165
66	CN	-	192.0.42.103	198.51.101.166

When providing the rule allocation to the IP addresses of the nodes of a node group, you can describe it in parallel with the allocation rule for each node type. If describing them in parallel, the IP address for each node type takes precedence.
The following is an example of defining the IP address allocation rule for node groups and separately defining such rule for the compute cluster sub management nodes.

```
Cluster {
ClusterName = "compute"
NodeGroup {
    NumberOfNodeGroups = "2"
    ManageNetRule {
        NetworkAddress = "192.0.2.0"
        NetMask = "255.255.255.0"
        StartIPaddress = "192.0.2.101"
        MaskUnit = "NodeGroup"
        NumberOfUnit = "1"
        IncrementMask = "255.255.255.0"
    }
    ControlNetRule {
        NetworkAddress = "198.51.100.0"
```

```
            NetMask = "255.255.255.0"
            StartIPaddress = "198.51.100.101"
            MaskUnit = "NodeGroup"
            NumberOfUnit = "1"
            IncrementMask = "255.255.255.0"
        }
        CCS {
            Model = "PG"
            NumberOfNodes = "2"
            ManageNetRule {
                NetworkAddress = "192.0.2.0"
                NetMask = "255.255.255.0"
                StartIPaddress = "192.0.2.1"
            }
            ControlNetRule {
                NetworkAddress = "198.51.100.0"
                NetMask = "255.255.255.0"
                StartIPaddress = "198.51.100.1"
            }
        }
        CN {
            Model = "PG"
            NumberOfNodes = "64"
        }
    }
}
```

Describing the rule as shown above adds a node group as follows.

- First node group

Table 3.8 Example of Rule Allocation for IP Address (First Node Group)

No.	Node Type	Management Network (Representative)	Management Network	Control Network
1	CCS	192.0.2.1	192.0.2.2	198.51.100.1
2	CCS	192.0.2.1	192.0.2.3	198.51.100.2
3	CN	-	192.0.2.101	198.51.100.101
4	CN	-	192.0.2.102	198.51.100.102
Omitted				
30	CN	-	192.0.2.128	198.51.100.128
31	CN	-	192.0.2.129	198.51.100.129
32	CN	-	192.0.2.130	198.51.100.130
33	CN	-	192.0.2.131	198.51.100.131
Omitted				
63	CN	-	192.0.2.161	198.51.100.161
64	CN	-	192.0.2.162	198.51.100.162
65	CN	-	192.0.2.163	198.51.100.163
66	CN	-	192.0.2.164	198.51.100.164

- Second node group

Table 3.9 Example of Rule Allocation for IP Address (Second Node Group)

No.	Node Type	Management Network (Representative)	Management Network	Control Network
1	CCS	192.0.2.4	192.0.2.5	198.51.100.3
2	CCS	192.0.2.4	192.0.2.6	198.51.100.4
3	CN	-	192.0.3.101	198.51.101.101
4	CN	-	192.0.3.102	198.51.101.102
Omitted				
30	CN	-	192.0.3.128	198.51.101.128
31	CN	-	192.0.3.129	198.51.101.129
32	CN	-	192.0.3.130	198.51.101.130
33	CN	-	192.0.3.131	198.51.101.131
Omitted				
63	CN	-	192.0.3.161	198.51.101.161
64	CN	-	192.0.3.162	198.51.101.162
65	CN	-	192.0.3.163	198.51.101.163
66	CN	-	192.0.3.164	198.51.101.164

3.1.1.4 Configuring the Shared File System for a Redundant Configuration

In the system management node and the compute cluster management node in a redundant configuration, there is an area (/var/opt/FJSVtcs/shared_disk) which stores management information related to logs, dump files and job operations. This area is called shared_disk.
When using shared_disk in a redundant configuration node, describe the information for mounting shared_disk with an active system node in pxsystem.conf.

The following is an example of describing the setting of shared_disk, which is used in the compute cluster management node, in pxsystem.conf.

- by-id of a device to be mounted as shared_disk: /dev/disk/by-id/scsi-00000000-part7
- Mount point: /var/opt/FJSVtcs/shared_disk
- Type of the file system: xfs
- Option specified in the mount command at the time of the mounting: "rw"

```
Cluster {
    HA {
        Node {
            NodeType = "CCM"
            Mount {
                Device = "/dev/disk/by-id/scsi-00000000-part7"
                MountPoint = "/var/opt/FJSVtcs/shared_disk"
                FSType = "xfs"
                OptList = "rw"
            }
        }
    }
}
```

3.1.2 Registering Cluster Configuration Information

To register the cluster configuration information, describe the configuration of a cluster in pxsystem.conf, the cluster configuration definition file, and then set it up using the --create option of the pxsystemadm command. The system administrator is to register the cluster

configuration information on the active system management node.
Place pxsystem.conf in the following location on the active system management node.

```
/etc/opt/FJSVtcs/pxsystem.conf
```

Information

When introducing the Job Operation Software, the pxsystem.conf file does not exist. Copy the sample file (/etc/opt/FJSVtcs/sample/pxsystem.conf), and then place it on the path described above.

3.1.2.1 Creating a Template of the Cluster Configuration Definition File

When adding a cluster or node, you can create a template of the pxsystem.conf file in order to make it easier to input this file. To create the template, use nodes.conf, the node configuration information file which the "design sheet for installation" of the installation function outputs.

1. Entering values on the design sheet for installation

   On a Windows PC, the system administrator is to enter values on the design sheet for installation.
   For details on the design sheet for installation, see "3.8.3 Creating Settings of the Installation Target Node."

2. Transferring the file

   Transfer nodes.conf, the node information definition file which the design sheet for installation outputs, to any path on the system management node.

3. Executing the pxsystemadm command

   Create the template by specifying a path having the nodes.conf file as the argument of the --convert option of the pxsystemadm command.
   The following is an example of creating a template from the nodes.conf file on the directory /var/tmp/ on the system management node.

   ```
   # pxsystemadm --convert /var/tmp/nodes.conf
   [INFO] PSM 1081 pxsystemadm The file was output.(filename)
   ```

   filename: The name of the output file

   The template of the pxsystem.conf file is output under the current directory at the time of executing the command, as pxsystem.conf_*command execution date and time*. Then, the system administrator must input insufficient item(s) of the template.

   The items requiring this input are as follows:

   - Any part that is not commented out and enclosed with {{ }}. (Commented-out items have '#' on the first column.)

   - Input commented-out items (which have '#' on the first column) as needed, according to the cluster configuration.

   If there is any other part(s) necessary to be changed, edit such part(s).

   According to the following rules, values are automatically written to the template of the pxsystem.conf file.

   Table 3.10 Rules of the Template of the pxsystem.conf File

No.	Rule of Template of pxsystem.conf File
1	In the item Model, "PG" is entered as a fixed value. To use any node, edit the template.
2	The definition of a node group is automatically described when the nodes.conf file contains the compute cluster sub management node.
3	When the number of the compute cluster sub management nodes described in the nodes.conf file is an even number, the template automatically regards the compute cluster sub management nodes as being in a redundant configuration. When the number of the compute cluster sub management nodes is an odd number, the template automatically regards the compute cluster sub management nodes as being in a non-redundant configuration. The allocation order is the order of defining the node information definition files (which means the definition order in the design sheet for installation).

No.	Rule of Template of pxsystem.conf File
4	If defining a node group, the compute nodes described in the nodes.conf file are evenly described in each node group.
5	To the management network, the template assigns the IP address specified in [Provision Network] of the design sheet for installation.
6	If specifying "bmc" in [device] of [Other Network] on the design sheet for installation, the IP address of the network specifying "bmc" is assigned to the control network.

The following shows an example of the template of pxsystem.conf to be output.

```
Operation {
    # Specify operation type.                                                     (*1)
    # [example] Type = "add-clst"
    Type = {{ope_type}}                                                           (*2)
}
Cluster {
    # Cluster name (up to 63 characters)
    # You can use single-byte alphanumeric characters (case-sensitive) and
    # hyphens. However, a hyphen cannot be used for the first character.
    # The underscore "_" cannot be used.
    ClusterName = {{cluster_name}}

...

        LN {
            ControlNet = 203.0.113.189                                            (*3)
            ManageNet = 198.51.100.6
            Model = PG
        }
...
```

(*1) This describes an item requiring additional description in a commentary form.
(*2) This outputs an item requiring additional description by enclosing it with {{ }}.
(*3) This automatically outputs an item from the contents of the nodes.conf file.

See

For the meaning of each of the section names and items, see "pxsystem.conf file" in the "Job Operation Software Command Reference."

3.1.2.2 Adding a Cluster

The configuration management function enables you to create a new system and add a cluster to the existing system.

1. Creating the cluster configuration definition file pxsystem.conf

   Create pxsystem.conf, which is the cluster configuration definition file.

   The following is an example of setting up the pxsystem.conf file.
   To add a cluster, specify "add-clst" in the item Type of the Operation section, and then describe the configuration information of the added cluster.

```
Operation {
    Type = "add-clst"                                                 # (*1)
}
Cluster {
...                                                                   # (*2)
}
```

   (*1) Specify "add-clst," which represents addition of a cluster.
   (*2) Describe information on the added cluster.

See

For the meaning of each of the section names and items, see "pxsystem.conf file" in the "Job Operation Software Command Reference."

2. Registering the contents of the pxsystem.conf file

   - When the added cluster contains FX server

     Use the --create option of the pxsystemadm command to apply the cluster configuration definition to the system management node.
     Specify the --ft option, and then specify the path of the directory that stores the node information definition file for FX server of the installation function.

     ```
     # pxsystemadm --create pxsystem.conf --ft node_ft
     Operation/Type is "add-clst".                             <- (*1)
     Do you really want to continue (y/n)? y                   <- (*2)
     [INFO] PSM 1082 pxsystemadm The setting was completed.
     ```

     (*1) This shows the type of an operation to be implemented.
     (*2) This requires your confirmation before the registration.

     ### See

     To create the node information definition file for FX server, use the design sheet for installation.
     For details on the design sheet for installation, see "3.8.3 Creating Settings of the Installation Target Node."

   - When the added cluster does not contain FX server

     Use the --create option of the pxsystemadm command to apply the cluster configuration definition to the system management node.

     ```
     # pxsystemadm --create pxsystem.conf
     Operation/Type is "add-clst".
     Do you really want to continue (y/n)? y
     [INFO] PSM 1082 pxsystemadm The setting was completed.
     ```

3. Checking the set contents

   To check the set contents, specify the -a option in the pashowclst command.

   ```
   # pashowclst -a
   ...
   ```

   Confirm that all the following information is correct:

   - The specific information on the added node is correct.

   - In the case when setting shared_disk, the displayed mount information is correct.

   At this point of time, the functions of the Job Operation Software do not work on the newly added node, displaying a hyphen on the service running status (in the SRV_STATUS field). In addition, the node running status (in the STATUS field) displays "Stopped," for the system monitoring function is not running.

   You can see these statuses by performing the procedure described in "3.1.4 Applying the Cluster Configuration Definition" for the added node.

   If no hardware exists, such as when introducing a system in stages, the power status (in the PWR_STATUS field) of a target node is "Unknown." You can see the power status of the node by connecting any prepared hardware to the control network.

See

For details on the contents displayed by the pashowclst command, see "3.3.3.3 Displaying All Information of the System."

3.1.2.3 Deleting a Cluster

Before deleting a cluster, stop job operations of the cluster.

Note

Nodes in the deletion range stop various services of the Job Operation Software, preventing the functions from running if restarting the nodes. If restarting a node that has stopped at the timing of performing this procedure, however, the services of the Job Operation Software start running. This is because this setting is not applicable to such node. For this reason, do not start a node while it is connected to a network in operation. Otherwise, this may affect a system in operation.

1. Creating the cluster configuration definition file pxsystem.conf

   Create pxsystem.conf, which is the cluster configuration definition file.
   To delete a cluster, specify "del-clst" in the item Type of the Operation section, and then describe the cluster name of the deleted cluster. The following is an example of deleting a cluster whose name is "cluster1."

   ```
   Operation {
       Type = "del-clst"                              # (*1)
   }
   Cluster {
       ClusterName = "cluster1"                       # (*2)
   }
   ```

   (*1) Specify "del-clst," which represents deletion of a cluster.
   (*2) Specify the cluster name to be deleted.

2. Registering the contents of the pxsystem.conf file

   Use the --create option of the pxsystemadm command to apply the cluster configuration definition to the system management node.

   ```
   # pxsystemadm --create pxsystem.conf
   Operation/Type is "del-clst".                                <- (*1)
   Do you really want to continue (y/n)? y                      <- (*2)
   [INFO] PSM 1082 pxsystemadm The setting was completed.
   ```

   (*1) This shows the type of an operation to be implemented.
   (*2) This requires your confirmation before the registration.

3. Checking the set contents

   To check the set contents, specify the -a option in the pashowclst command.

   ```
   # pashowclst -a
   ...
   ```

   Confirm that the cluster, which you described in the pxsystem.conf file in step 1, has been deleted.

   See

   For details on the contents displayed by the pashowclst command, see "3.3.3.3 Displaying All Information of the System."

3.1.2.4 Adding a Node

The configuration management function enables you to create a new system and add a cluster to the existing system.

1. Creating the cluster configuration definition file pxsystem.conf

   Create pxsystem.conf, which is the cluster configuration definition file.
   To add a node, specify "add-node" in the item Type of the Operation section, and then describe the name of a cluster to which the node is added and the configuration information of the added node. The following is an example of adding two node groups to a cluster (cluster name: cluster1). Each of these node groups contains 216 boot groups.

```
Operation {
    Type = "add-node"                                                 # (*1)
}
Cluster {
    ClusterName = "cluster1"
    NodeGroup {                                                       # (*2)
        NumberOfNodeGroups = "2"
        ManageNetRule {
            NetworkAddress = "192.0.2.0"
            NetMask = "255.255.0.0"
            StartIPaddress = "192.0.2.101"
            MaskUnit = "NodeGroup"
            NumberOfUnit = "1"
            IncrementMask = "255.255.255.0"
        }
        ControlNetRule {
            NetworkAddress = "198.51.100.0"
            NetMask = "255.255.0.0"
            StartIPAddres = "198.51.100.101"
            MaskUnit = "NodeGroup"
            NumberOfUnit = "1"
            IncrementMask = "255.255.255.0"
        }
        CCS {
            Model = "PG"
            NumberOfNodes = "2"
        }
        Model = "FT"
        NumberOfBootGroups = "216"
    }
}
```

   (*1) Specify "add-node," which represents addition of a node.
   (*2) Describe information on the added node. Do not describe nodes that the cluster configuration has already defined.

   See

   For the meaning of each of the section names and items, see "pxsystem.conf file" in the "Job Operation Software Command Reference."

2. Registering the contents of the pxsystem.conf file

   - When the added node contains FX server

     Use the --create option of the pxsystemadm command to apply the cluster configuration definition to the system management node.
     Specify the --ft option, and then specify the path of the directory that stores the node information definition file for FX server of the installation function.

```
# pxsystemadm --create pxsystem.conf --ft node_ft
Operation/Type is "add-node".                                <- (*1)
Do you really want to continue (y/n)? y                      <- (*2)
[INFO] PSM 1082 pxsystemadm The setting was completed.
```

     (*1) This shows the type of an operation to be implemented.
     (*2) This requires your confirmation before the registration.

See

To create the node information definition file for FX server, use the design sheet for installation.
For details on the design sheet for installation, see "3.8.3 Creating Settings of the Installation Target Node."

- When the added node does not contain FX server

  Use the --create option of the pxsystemadm command to apply the cluster configuration definition to the system management node.

```
# pxsystemadm --create pxsystem.conf
Operation/Type is "add-node".                             <- (*1)
Do you really want to continue (y/n)? y                   <- (*2)
[INFO] PSM 1082 pxsystemadm The setting was completed.
```

  (*1) This shows the type of an operation to be implemented.
  (*2) This requires your confirmation before the registration.

3. Checking the set contents

   To check the set contents, specify the -a option in the pashowclst command.

```
# pashowclst -a
...
```

   Confirm that all the following information is correct:

   - The specific information on the added node is correct.

   - In the case when setting shared_disk, the displayed mount information is correct.

   At this point of time, the functions of the Job Operation Software do not work on the newly added node, displaying a hyphen on the service running status (in the SRV_STATUS field). In addition, the node running status (in the STATUS field) displays "Stopped," for the system monitoring function is not running. You can see these statuses by performing the procedure described in "3.1.4 Applying the Cluster Configuration Definition" for the added node.

   If no hardware exists, such as when introducing a system in stages, the power status (in the PWR_STATUS field) of a target node is "Unknown." You can see the power status of the node by connecting any prepared hardware to the control network.

   See

   For details on the contents displayed by the pashowclst command, see "3.3.3.3 Displaying All Information of the System."

3.1.2.5 Deleting a Node

Delete a node from a cluster while excluding the node from the operation.

Note

Nodes in the deletion range stop various services of the Job Operation Software, preventing the functions from running if restarting the nodes. If restarting a node that has stopped at the timing of performing this procedure, however, the services of the Job Operation Software start running. This is because this setting is not applicable to such node. For this reason, do not start a node while it is connected to a network in operation. Otherwise, this may affect a system in operation.

1. Creating the cluster configuration definition file pxsystem.conf

   Create pxsystem.conf, which is the cluster configuration definition file.
   To delete a node, specify "del-node" in the item Type of the Operation section, and then describe the IDs of the node and group to be deleted.

```
Operation {
    Type = "del-node"                                                 # (*1)
}
Cluster {
    ClusterName = "cluster1"
    NodeGroup {                                                       # (*2)
        NodeGroupID = "0x03"
    }
    BootGroup {
        BootGroupID = "0x0103"
    }
    Node {
        NodeID = "0xFFFF0005"
    }
}
```

(*1) Specify "del-node," which represents deletion of a node.
(*2) Describe information on the deleted node.

2. Registering the contents of the pxsystem.conf file

   Use the --create option of the pxsystemadm command to apply the cluster configuration definition to the system management node.

```
# pxsystemadm --create pxsystem.conf
Operation/Type is "del-node".                               <- (*1)
Do you really want to continue (y/n)? y                     <- (*2)
[INFO] PSM 1082 pxsystemadm The setting was completed.
```

(*1) This shows the type of an operation to be implemented.
(*2) This requires your confirmation before the registration.

3. Checking the set contents

   To check the set contents, specify the -a option in the pashowclst command.

```
# pashowclst -a
...
```

Confirm that the node, which you described in the pxsystem.conf file in step 1, has been deleted.

See

For details on the contents displayed by the pashowclst command, see "3.3.3.3 Displaying All Information of the System."

3.1.2.6 Changing the Configuration Information

The configuration management function enables you to change the specific information of nodes (such as IP address), which is defined when adding a cluster or node. This function also allows you to change the setting for switching the file system in a redundant configuration.
Change the configuration information while excluding a target cluster from the operation.
To change the configuration information, specify "update" in the item Type of the Operation section. Then, describe the node ID to be changed and the specific information of the node to be changed.

1. Creating the cluster configuration definition file pxsystem.conf

   Create pxsystem.conf, which is the cluster configuration definition file.

   The following is an example of setting up the pxsystem.conf file.

```
Operation {
    Type = "update"                                               # (*1)
}
Cluster {
    ClusterName = "clstname"
    Node {
        NodeID = "0xFFFF0001"
        ManageNet = "192.0.2.254"                                 # (*2)
    }
}
```

(*1) Specify "update," which represents changing of the configuration information.
(*2) Describe an item to be changed. (This example shows when changing the IP address of the management network.)

See

For the meaning of each of the section names and items, see "pxsystem.conf file" in the "Job Operation Software Command Reference."

2. Registering the contents of the pxsystem.conf file

   Use the --create option of the pxsystemadm command to apply the cluster configuration definition to the system management node.

```
# pxsystemadm --create pxsystem.conf
Operation/Type is "update".                                  <- (*1)
Do you really want to continue (y/n)? y                      <- (*2)
[INFO] PSM 1082 pxsystemadm The setting was completed.
```

   (*1) This shows the type of an operation to be implemented.
   (*2) This requires your confirmation before the registration.

3. Checking the set contents

   To check the set contents, specify the -a option in the pashowclst command.

```
# pashowclst -a
...
```

   Confirm that the specific information of the node, which you described in the pxsystem.conf file in step 1, has been changed.

See

For details on the contents displayed by the pashowclst command, see "3.3.3.3 Displaying All Information of the System."

3.1.3 Control Setting for FX server

Perform the control setting for FX server after implementing the cluster configuration definition and completing the installation of FX server. To perform the control setting for FX server, execute the --set-hardware option of the pxsystemadm command when the control network for FX server is available for communication.

1. Executing the setting command

   Execute the --set-hardware option of the pxsystemadm command to perform the control setting for FX server.

```
# pxsystemadm --set-hardware -c clstname
Do you really want to continue (y/n)? y                   <- (*1)
[INFO] PSM 1082 pxsystemadm The setting was completed.
```

   clstname: Cluster name
   (*1) This requires your confirmation before the setting.

Information

If some hardware does not exist in the system, including the case of installing FX server in stages, use the scope option to execute the command only in a range where hardware exists.

The following is an example of executing the pxsystemadm command against nodes, whose boot group IDs are from 0x0101 to 0x0103, in the compute cluster cluster1.

```
# pxsystemadm --set-hardware -c cluster1 --bootgrp 0x0101-0x0103
Do you really want to continue (y/n)? y
[INFO] PSM 1082 pxsystemadm The setting was completed.
```

After FX server is installed, execute the pxsystemadm command again for the installed nodes.

2. Checking the set contents

   Execute the --check-hardware option of the pxsystemadm command to check the control setting for FX server.

```
# pxsystemadm --check-hardware -c clstname
[INFO] PSM 1083 pxsystemadm The setting check succeeded.
```

clstname: Cluster name

If there is an FX server whose control setting has not been made or whose setting has an abnormality, the error list file is output to the current directory.

```
# pxsystemadm --check-hardware -c clstname
[ERR.] PSM 1001 pxsystemadm There are nodes which has not setting of the hardware.(filename)
```

filename: The name of the error list file output to the current directory

See

For the method of coping with the problem when the error list file is output, see "Failed Reflection of Cluster Configuration Definitions" in the "Job Operation Software Troubleshooting."

3.1.4 Applying the Cluster Configuration Definition

Regarding the cluster configuration that the cluster configuration definition configures to the system management node, apply it to the system using the --set option of the pxsystemadm command.

```
# pxsystemadm --set -c clstname
Do you really want to continue (y/n)? y                           <- (*1)
[INFO] PSM 1082 pxsystemadm The setting was completed.
```

clstname: Cluster name

(*1) This requires your confirmation before the registration.

If a node fails in this application due to some reason such as stoppage of the power supply, the error list file is output to the current directory. In such a case, check the contents of the file and remove the cause. Then, execute the command again for the node that failed in this application.

```
# pxsystemadm --set -c clstname
Do you really want to continue (y/n)? y
[ERR.] PSM 0020 pxsystemadm There are nodes that failed in the setting.(filename)
```

clstname: Cluster name
filename: The name of the error list file output to the current directory

The following is an example of the error list file when SSH connection to target nodes fails. (In this example, the node IDs of these nodes are 0xFFFF0008 and 0xFFFF0009.)

```
[0xFFFF0008] ssh: connect to host 192.0.2.10 port 22: No route to host
[0xFFFF0009] ssh: connect to host 192.0.2.11 port 22: No route to host
```

Diskless nodes are to be configured on the boot I/O node. Therefore, if an abnormality occurs while configuring diskless nodes, the following node IDs are output to the above error file:

- The node IDs of the boot I/O nodes (0x01010001 and 0x01030001 in this example)
- The node IDs of the nodes to be configured (0x01010008 and 0x01030009 in this example)

The following is an example of failing to apply the package of the Job Operation Software to a target node.

```
[0x01010001][0x01010008] Packages install failed.
[0x01030001][0x01030009] Packages install failed.
```

When an error list file is output, it is necessary to reconfigure the node where its configuration failed. For the method of reconfiguring it, see "Failed Reflection of Cluster Configuration Definitions" in the "Job Operation Software Troubleshooting."

3.2 Details of the System Control Function

3.2.1 Configuring the System Control Function

It is necessary to configure the system control function in order to use it. There are the following three types of the settings to use the system control function:

- Setting the IPMI authentication information (Required)
- Setting parameters during power control
- Setting the power control plugin

See

- For details on the setting of the IPMI authentication information, see "3.2.1.1 Setting the IPMI Authentication Information."
- For details on the setting of parameters during power control, see "3.2.1.2 Setting Parameters During Power Control."
- For details on the setting of the power control plugin, see "3.2.1.3 Power Control Plugin."

3.2.1.1 Setting the IPMI Authentication Information

The system administrator is to check the IPMI authentication information set on PRIMERGY server, and then describe such authentication information for each cluster in the papwr.conf file.
The following is an example of describing the user name "admin" and the password "admin" as the IPMI authentication information in a node of the model PG (PRIMERGY server) of the cluster "compute."

```
Cluster {
    ClusterName = compute
    Model {
        Model = PG
        User = admin
        Password = admin
    }
}
```

See

- For the meanings of the items of the papwr.conf file, see "papwr.conf file" in the "Job Operation Software Command Reference."
- For details on how to set the IPMI authentication information of PRIMERGY server, see the *iRMC S2 - Integrated Remote Management Controller User's Guide*.

Note

PRIMERGY server requires you to make a setting for BIOS or UEFI or use the ipmitool command to change your user name and password, so that it can ensure security. Be sure to set the same user name and password of IPMI inside the same cluster, for the user name and password are written in the system control function configuration file for each cluster.

3.2.1.2 Setting Parameters During Power Control

Assume that you want to change the default values or set the upper and lower limit values for the number of concurrent executions and the interval at the time of executing power control. Then, describe these values for each cluster in the papwr.conf file.

The following is an example of describing the default values, the upper limit values, and the lower limit values, as shown in the table below. This example describes these values for the number of concurrent executions and the interval in the case when carrying out the power-on operation in a cluster whose name is "compute."

Table 3.11 Example of Setting Default Values, Upper Limit Values, and Lower Limit Values

Model	Operation	Default Value	Upper Limit Value	Lower Limit Value
FT (FX server)	Number of concurrent executions	24	72	1
PG (PRIMERGY server)	Number of concurrent executions	30	100	1
FT	Interval	2000	10000	10
PG	Interval	10000	100000	1000

```
Cluster {
    ClusterName = compute
    OptionLimit {
        Concurrency {
            Control = on
            Default = 24:FT,30:PG
            UpperLimit = 72:FT,100:PG
            LowerLimit = 1:FT,1:PG
        }

        Interval {
            Control = on
            Default = 2000:FT,10000:PG
            UpperLimit = 10000:FT,100000:PG
            LowerLimit = 10:FT,1000:PG
        }

    }
}
```

See

- For the meanings of the items of the papwr.conf file, see "papwr.conf file" in the "Job Operation Software Command Reference."
- For the meanings of the values of the number of concurrent executions and the interval, see "3.2.2.1 The Number of Concurrent Executions and the Interval."

3.2.1.3 Power Control Plugin

Although the Job Operation Software provides "PG (PRIMERGY server)" and "FT (FX server)" as its models, this software also enables you to define any model other than these above as "any node."
The power control operation for any node varies depending on types of target hardware. Therefore, the system administrator must define the respective power control operations for any nodes. This function is called "power control plugin," and the execution format file called by the power control plugin is called "power control plugin command."

See

For details on models, see "3.1.1.1 Node-Specific Information."

The power control plugin command must meet the following conditions:

- This file must be executable on the system management node.
- The file name must consist of up to 63 characters, including only single-byte alphanumeric characters, hyphens, underscores, and periods. However, the first character cannot be a hyphen.
- The owner and the ownership group must be root and the permission must be either 0500 or 0700.

Create the power control plugin command that meets the above conditions. Then, in the papwr.conf file, describe the settings of the power control plugin for each cluster. The following is an example of registering /root/control_plugin_vm and /root/control_plugin_et as the power control plugin commands in the cluster "compute." This example registers /root/control_plugin_vm in the node whose model name is "VM," and /root/control_plugin_et in the node whose model name is "ET."

```
Cluster {
    ClusterName = compute
    Plugin {
        Model = VM
        Path = /root/control_plugin_vm
    }
    Plugin {
        Model = ET
        Path = /root/control_plugin_et
    }
}
```

See

- For the meanings of the items of the papwr.conf file, see "papwr.conf file" in the "Job Operation Software Command Reference."
- For the methods of thinking when creating the power control plugin command, see "3.2.1.4 Creating the Power Control Plugin Command."
- For the example of creating the power control plugin command, see "3.2.1.5 Example of Creating the Power Control Plugin Command."

3.2.1.4 Creating the Power Control Plugin Command

The following table shows the information on environment variables about a node to be controlled. These environment variables are passed to the power control plugin command. The system administrator is to use the information on these environment variables to create the power control plugin command.

Table 3.12 Environment Variables Passed to the Power Control Plugin Command

Environment Variable Name	Description
PWRCTL_CLSTNAME	Sets the name of a cluster containing a node to be controlled.

Environment Variable Name	Description
PWRCTL_NODEID	Sets the node ID of a node to be controlled.
PWRCTL_NODETYPE	Sets the node type of a node to be controlled.
PWRCTL_MNGNET	Sets the management network information of a node to be controlled.
PWRCTL_CTRLNET	Sets the control network information of a node to be controlled. (*)
PWRCTL_MODEL	Sets the model information of a node to be controlled.
PWRCTL_CONTROL	Sets any of the following control operations: on, off, force-off, cold-reboot, reset, cycle, init, and status

(*) If you do not set a value, a null character is set here.

If the control operation set in the environment variable PWRCTL_CONTROL is not status, the return value of the power control plugin command determines whether or not the power control is normally performed.

Table 3.13 Relationship Between the Return Value of the Power Control Plugin Command and Its Operation (in Control Operation)

Return Value	Description
0	Normal end
Other than 0	Abnormal end

If the control operation set in the environment variable PWRCTL_CONTROL is status, the return value of the power control plugin command determines the power status. The returned return value determines the power status of a node, which appears when using the papwrctl command to specify status as an argument.

Note

If the content of the power control plugin command is incorrect, the Job Operation Software may not run properly.
Especially when setting a node in a redundant configuration as any node, the failover function does not work correctly if you do not write the following operations:
- Operation to make the force-off operation stop the power supply to a target node forcibly
- Operation to make the init operation panic a target node or forcibly stop its power supply

Table 3.14 Relationship Between the Return Value of the Power Control Plugin Command and Its Operation (in Status Reference)

Return Value	Power Status Displayed by the papwrctl Command	Meaning
0	off	Power off
1	reset	During initialization
2	post	During Self-Test/OBP
3	initialization	During OBP initialization
4	OpenBoot	In state when ok prompt of OBP appears
5	os-boot	During OS startup
6	os-running	While OS is running
7	os-shutdown	During shutdown processing of OS
8	panic	Panic
9	BootFailed	Boot failure
Other than above	Unknown	Unknown

These power statuses are also used to collect the power status of the system monitoring function. The PWR_STATUS field of the pashowclst command displays these statuses as the power status of a node.

Note

If the content of the power control plugin command in status reference does not follow the above specifications, it may be impossible to refer to the power status using the pashowclst and the papwrctl commands.

3.2.1.5 Example of Creating the Power Control Plugin Command

The following is an example of creating the power control plugin command.

- Example of the power control plugin command that uses the virsh command (version 2.0.0) to control the virtual environment of KVM

  This example creates the plugin command under the following preconditions:

  - Nodes of KVM are registered as nodes of any node.
  - When defining a node by the configuration management function, the information on the control network for nodes of KVM is described in the format of "host IP address (identification ID of a virtual machine)."
  - ssh passwordless authentication is set from the system management node to a node that serves as the host of KVM.

  To use the power control plugin command to control the virtual environment of KVM, the system administrator must describe the following three operations using the execution format file:

  1. Divide the environment variable PWRCTL_CTRLNET into the host IP address of KVM and the identification ID of a virtual machine, and then store them in a variable.
  2. According to the value of the environment variable PWRCTL_CONTROL, specify the following operation contents as the arguments of the virsh command.

Table 3.15 Relationship Between the Values of the Environment Variable PWRCTL_CONTROL and the Operations of the virsh Command

Value of PWRCTL_CONTROL	Operation of virsh Command
on	start *identify_code*
off	shutdown *identify_code*
force-off	destroy *identify_code*
cold-reboot	reboot *identify_code*
reset	destroy operation, and then start
cycle	destroy operation, and then start
init	*identify_code dumppath* --crash --memory-only --format kdump-zlib
status	list --all (*)

identify_code: Identification ID of a virtual machine
dumppath: Path on the host node to output a dump

(*) This executes virsh list --all to get the status of a virtual environment, and decides the command return value according to the output content (such as running and shutdown).

Table 3.16 Relationship Between the Outputs of virsh list --all and the Return Values

Output of virsh list --all	Return Value of Power Control Plugin Command
shut off	0
idle	0

Output of virsh list --all	Return Value of Power Control Plugin Command
pmsuspended	0
running	6
blocked	10
paused	10
shutdown	7
dying	0
crashed	0

3. Return the return value by executing the virsh command to control the virtual environment. If status is specified, analyze the output of the virsh command to get the power status, and then return an appropriate return value.

```
#!/bin/bash
IP_ADDR=`echo $PWRCTL_CTRLNET | cut -d'(' -f1`
IDENTIFY_CODE=`echo $PWRCTL_CTRLNET | cut -d'(' -f2 | tr -d ')'`
if [ "$PWRCTL_CONTROL" = "on" ]
then
    OPERATION="start"
elif [ "$PWRCTL_CONTROL" = "off" ]
then
    OPERATION="shutdown"
elif [ "$PWRCTL_CONTROL" = "force-off" ]
then
    OPERATION="destroy"
elif [ "$PWRCTL_CONTROL" = "cold-reboot" ]
then
    OPERATION="reboot"
elif [ "$PWRCTL_CONTROL" = "reset" ]
then
    ssh $IP_ADDR virsh destroy $IDENTIFY_CODE
    ssh $IP_ADDR virsh start $IDENTIFY_CODE
    exit $?
elif [ "$PWRCTL_CONTROL" = "cycle" ]
then
    ssh $IP_ADDR virsh destroy $IDENTIFY_CODE
    ssh $IP_ADDR virsh start $IDENTIFY_CODE
    exit $?
elif [ "$PWRCTL_CONTROL" = "init" ]
then
    DATE=`date '+%Y%m%d%H%M%S'`
    DUMPPATH="/root/vmdump/$DATE"
    OPERATION="$DUMPPATH --crash --memory-only --format kdump-zlib"
    ssh $IP_ADDR virsh dump $IDENTIFY_CODE $OPERATION
    exit $?
elif [ "$PWRCTL_CONTROL" = "status" ]
then
    OPERATION="list --all"
    STATUS=$(LANG=C ssh $IP_ADDR virsh $OPERATION | grep -v '^#' | awk "\$2 == \"$IDENTIFY_CODE
\" { print \$3 }")
    if [ "$STATUS" = "shut" ]
    then
        exit 0
    elif [ "$STATUS" = "idle" ]
    then
        exit 0
    elif [ "$STATUS" = "pmsuspended" ]
    then
        exit 6
    elif [ "$STATUS" = "running" ]
    then
        exit 6
    elif [ "$STATUS" = "blocked" ]
    then
        exit 10
    elif [ "$STATUS" = "paused" ]
    then
        exit 10
    elif [ "$STATUS" = "shutdown" ]
    then
        exit 7
    elif [ "$STATUS" = "dying" ]
    then
        exit 0
    elif [ "$STATUS" = "crashed" ]
    then
        exit 0
    else
        exit 10
    fi
fi
ssh $IP_ADDR virsh $OPERATION $IDENTIFY_CODE
exit $?
```

3.2.1.6 Applying the Settings

Describe the set contents in the papwr.conf file, and then apply the settings by using the --set option of the papwradm command.
In addition, you can refer to the currently set contents from the --show option of the papwradm command.
The following shows the procedure to set the power control. The system administrator is to carry out all the following operations on the active system management node.

1. Creating the papwr.conf file

   Describe any necessary information in the papwr.conf file.
   Place the papwr.conf file in the following location on the active system management node.

   ```
   /etc/opt/FJSVtcs/papwr.conf
   ```

   Information

   When introducing the Job Operation Software, the papwr.conf file does not exist. Copy the sample file (/etc/opt/FJSVtcs/sample/papwr.conf), and then place it on the path described above.

   See

   For the meanings of the section names and items described in the papwr.conf file, see "papwr.conf file" in the "Job Operation Software Command Reference."

2. Registering the contents of the papwr.conf file

   Use the --set option of the papwradm command to register the contents of the papwr.conf file in the Job Operation Software. Then, replace the character string, which is specified in the item Password of the papwr.conf file, with `*`.

   ```
   # papwradm --set
   [WARNING]
   Do you really want to continue (y/n)? y
   [INFO] PSM 0082 papwradm The processing of the configuration file was completed.
   ```

Note

This operation only registers the settings and does not confirm whether the power control is actually executable or not. This allows you to perform this operation regardless of the status of a power-controlled node or whether it physically exists.

Information

If the settings are not applied to the standby system management node due to some reason such as stoppage of the power supply, the following display is output.

```
# papwradm --set
[WARNING]
Do you really want to continue (y/n)? y
[WARN] PSM 0051 papwradm There are nodes that failed in the setting.(filepath)
[INFO] PSM 0082 papwradm The processing of the configuration file was completed.
```

filepath: File name output by the pmscatter command, which the papwradm command internally executes

In such a case, see the "Job Operation Software Troubleshooting" in order to cope with this problem. Then, recreate the papwr.conf file and execute the papwradm command again.

```
# papwradm --set
[WARNING]
Do you really want to continue (y/n)? y
[INFO] PSM 0082 papwradm The processing of the configuration file was completed.
```

3. Checking the set contents

   To confirm the set contents, use the --show option of the papwradm command to display those which are currently effective. The displayed contents have the same format as the papwr.conf file.

```
# papwradm --show
...
```

4. Transferring the file to the standby system management node

   When setting the power control plugin, use the pmscatter command to transfer the file of the power control plugin command to the standby system management node. The following is an example of distributing the power control plugin command (file name: plugin1) to the directory /usr of the system management node of the compute cluster (cluster name: compute). The power control plugin command has been created under /tmp.

```
# pmscatter -c compute --nodetype SMM -p /tmp/plugin1 /usr
```

3.2.2 Power Control for Nodes

Use the papwrctl command to perform power control for nodes or refer to their power status. The system administrator is to execute the papwrctl command on the system management node.
You can specify a target node in units of a specific node(s), various kinds of groups, etc. The following is an example of using the -n option to start and stop a single node.

Starting a Node

```
# papwrctl -c clstname -n nodeid on
[INFO] PSM 2081 papwrctl All power control was executed.
```

Stopping a Node

```
# papwrctl -c clstname -n nodeid off
[WARNING]
papwrctl is going to power-off nodes of clstname
```

```
Do you really want to continue (y/n)? y
[INFO] PSM 2081 papwrctl All power control was executed.
```

clstname: Cluster name
nodeid: Node ID

Note

If the --force option is not specified at the time of control instruction accompanied with stoppage of power supply, the papwrctl command displays a prompt to request your confirmation before the processing.

There are different types of operations that you can instruct according to the model of the target node and the control range. PRIMERGY server enables you to perform the same control operations even if you specify any range. When specifying FX server, however, the control operations are carried out in units of BoB, CMU, and node in accordance with how to specify a control target. Each of these units has different kinds of executable control operations. The details are as follows.

Yes: The control operation is executable. No: The control operation is not executable.

Table 3.17 Executable Power Control Operations

Model Name	Operation							
	on	off	force-off	cold-reboot	cycle	reset	init	status
FX server (Unit of BoB)	Yes	Yes	Yes	No	No	No	No	Yes
FX server (Unit of CMU)	Yes	Yes	Yes	No	No	No	No	Yes
FX server (Unit of node)	Yes	Yes	Yes	No	No	Yes	Yes	Yes
PRIMERGY server	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Note

- When specifying "on" in the pashowclst command, suppose that the REASON field of the display result of this command shows "Alarm," "RouterFatal," "PortRouterFatal," "PortReservedAlarm," "ReservedAlarm," or "ResetRequest" for a node. Then, this node does not start. To start such node, specify the --force-all option. If the --force-all option is specified, the power supply is forced to operate even if normal startup is not guaranteed. Be careful that a booted node may not work properly.

- Before performing the init operation, you must set kdump in a target node. For details on the setting of kdump, see the document of the OS to be used.

- When concurrently specifying multiple models as nodes to be controlled, the control instruction is made in the order of the nodes of the model PG, the nodes of any node, and the nodes of the model FT.

- If you stop a compute node by specifying the --jobnowait option of the papwrctl command while this node is executing a job, the compute node may panic.

- If the target of off, reset, or init is the boot I/O node of FX server, stop any nodes other than the boot I/O node inside the same BoB in advance.

- Performing reset or init does not incorporate the compute node of PRIMERGY server into operation, displaying "Disable" in the STATUS field and "AbnormalEnd" in the REASON field. (You can check these fields using the -v option of the pashowclst command.) To recover it, execute the paclstmgr --recover command from the system management node.
  For details on how to check the status of a node, see "3.3.2 Displaying Operation Status of the System." For details on how to recover a node, see "3.4.3.3 Recovering a Node."

- If controlled nodes contain nodes in a redundant configuration and you want to stop both nodes, disable the automatic failover. If automatic failover is not disabled, the shutdown process may cause a failover that prevents the node from being shutdown. For details on the setting of the automatic failover, see "3.4.5 Setting Automatic Failover."

- If the FX server is powered down in units of CMU, be sure to power up in units of CMU. When power is turned in units of BoB or node, the Tofu port does not link up and does not start normally.

See

- To start a range specified as a controlled object in stages, set the number of concurrent executions and the interval. For details on the number of concurrent executions and the interval, see "3.2.2.1 The Number of Concurrent Executions and the Interval."
- Specifying off or force-off as the control type stops nodes, CMUs, and BoBs inside FX server in accordance with the scope option, which selects a range of a controlled object. For details on the scope option of the papwrctl command, see "papwrctl command" in the "Job Operation Software Command Reference."

To refer to the power status of a node, execute the papwrctl command by specifying status as its argument.

Referring to the Power Status of a Node

```
# papwrctl -c clstname -n nodeid status
NODE        PWR_STATUS
nodeid      on
```

The following table lists the types of the power statuses to which you can refer by the papwrctl command.

Table 3.18 Power Statuses to Which You Can Refer by the papwrctl Command

Power Status	Meaning
off	Power off
on	Power on
reset	During initialization
post	During Self-Test/OBP
initialization	During OBP initialization
OpenBoot	In state when ok prompt of OBP appears
os-boot	During OS startup
os-running	While OS is running
os-shutdown	During shutdown processing of OS
panic	Panic
BootFailed	Boot failure
Unknown	Unknown

3.2.2.1 The Number of Concurrent Executions and the Interval

When using a command for power control operations (such as papwrctl, pastart, and pastop) to specify multiple nodes and perform power control in stages, specify the number of concurrent executions and the interval. To specify them on the command line, use the -P option for the number of concurrent executions and the --interval option for the interval.
The number of concurrent executions and the interval described here have the following meanings:

- The number of concurrent executions

  This means the number of nodes that issue control instructions at the same time. For example, suppose that you set the number of concurrent executions as 30 when providing a power control instruction to 100 nodes. Then, the control instruction is issued to 30 nodes three times and then to 10 nodes once.

- Interval

  This means a time to wait after issuing a control instruction to nodes whose number is set as the number of concurrent executions.

The following is an example of powering on 100 nodes by specifying 30 nodes as the number of concurrent executions and 1000 milliseconds (10 seconds) as the interval on the command line.

```
# papwrctl -c compute -n 0xFFFF0001-0xFFFF0064 -P 30 --interval 10000 on
```

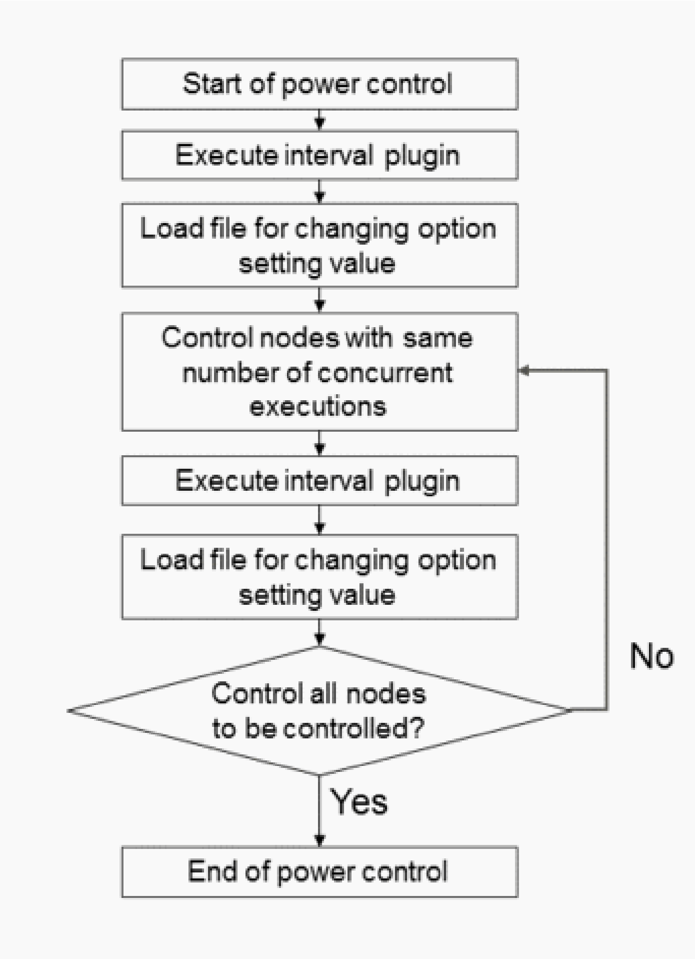
Information

- The number of concurrent executions can be a value which is at least one and equal to or less than the number of nodes inside a cluster. If the papwr.conf file sets the upper limit value and the lower limit value of the number of concurrent executions, you can set it in the range between these values.
- The interval can be a value which is equal to or more than 0 and equal to or less than 86400000 milliseconds (1 day). If the papwr.conf file sets the upper limit value and the lower limit value of the interval, you can set it in the range between these values.

3.2.2.2 Interval Plugin

The system control function can dynamically change the number of concurrent executions and the interval when it starts systems collectively. The purpose for this is to control systems without causing a rush current and exceeding the allowable value of power consumption in a facility. This function dynamically changes the number of concurrent executions and the interval right before a control instruction is provided. It executes this control in the order as shown in the following flow chart.

Figure 3.1 Execution Timing of Interval Plugin


Start of power control
Execute interval plugin
Load file for changing option setting value
Control nodes with same number of concurrent executions
Execute interval plugin
Load file for changing option setting value
Control all nodes to be controlled?
No
Yes
End of power control

The system administrator can dynamically change the number of concurrent executions and the interval. To do so, specify the --interval-plugin option and the --interval-option option at the same time from a command for power control operations (such as papwrctl, pastart, and pastop). The following shows the procedure to use this function. The system administrator is to carry out all the following procedures on the active system management node.

1. Creating the interval plugin command

   Describe the processing to be executed by the interval plugin command, and then change the permission to 0500 or 0700. The following example describes the path to create the interval plugin as /root/interval_plugin/plugin.sh.

```
# vi /root/interval_plugin/plugin.sh
```

2. Changing the permission to 0500 or 0700

   Change the permission of the interval plugin to 0500 or 0700.
   The following is an example of changing the permission of the interval plugin, which you created in step 1, to 0500.

```
# chmod 0500 /root/interval_plugin/plugin.sh
```

3. Executing the command

   The following is an execution example of specifying the --interval-plugin option and the --interval-option option in papwrctl. The following example uses the papwrctl command to supply power, but the pastart command and the pastop command are also available here.

```
# papwrctl -c clstname -n nodeid --interval-plugin /root/interval_plugin/plugin.sh \
--interval-option /root/interval_option/int_opt on
```

   clstname: Cluster name
   nodeid: Node ID

- Example of describing the interval plugin command

  Assume that there is a facility. Here, 10,000 W is the power value that may damage its equipment when exceeding this value due to some reason such as the limit of the equipment. Also, 8,000 W is the dangerous power value, and 5,000 W is the threshold as to whether the power value is still affordable. Based on this assumption, the following is an example of the interval plugin command in this facility:

  - Use the pasyspwr command to get the power information of an entire cluster. The following is a description example for a compute cluster named compute.

  - Suppose that the power value pwr obtained by the pasyspwr command exceeds the upper-limit power value pwr_alarm (10,000 W). Then, this forcibly stops the compute node inside the compute cluster named compute, causing the abnormal end of the interval plugin command.

  - Assume that the power value pwr obtained by the pasyspwr command exceeds the power value pwr_assert (8,000 W), which is at risk for excess of power. In such case, it rechecks the power consumption information after 60 seconds without performing control processing.

  - If the power value pwr obtained by the pasyspwr command exceeds the default power value pwr_default (5,000 W), it performs control under the following conditions:
    - The number of concurrent executions: 10
    - Interval: 60 seconds

  - If the power value pwr obtained by the pasyspwr command is equal to or less than the default power value pwr_default (5,000 W), it performs control under the following conditions:
    - The number of concurrent executions: 100
    - Interval: 10 seconds

```
#!/bin/bash

pwr_alarm=10000
pwr_assert=8000
pwr_default=5000

while true
do
    pwr=$(/usr/sbin/pasyspwr -c compute| awk '$1 == "TOTAL_PWR" {print $3}')
    if [ $pwr -gt $pwr_alarm ]
    then
        papwrctl -c compute -a --force --nodetype CN force-off
        exit 1
    elif [ $pwr -gt $pwr_assert ]
    then
        sleep 60
        continue
    elif [ $pwr -gt $pwr_default ]
    then
        next_conc=10
        next_int=60000
        break
    else
        next_conc=100
        next_int=10000
        break
    fi
done
file=/root/interval_option/int_opt
if [ -e $file ]
then
    rm -f $file
fi
echo "-P $next_conc --interval $next_int" > $file
chmod 0600 $file
exit 0
```

See

For details on the pasyspwr command, see "Operation With the Power Management Function" in the "Job Operation Software Administrator's Guide for Power Management."

Note

The pastart command, the pastop command, and the papwrctl command interrupt the power control processing and abnormally exit themselves when any of the following events occur:

- The interval plugin command returns with any value other than 0, the return value.
- The file for changing the option setting value contains any character string other than an option to specify the number of concurrent executions and the interval.

To continue the control processing even if any of the above states occurs, specify the --force-all option.

3.2.3 Collective Power Control for the System

To start and stop the system or a cluster, use the pastart command or the pastop command, which take account of the start/stop sequence and the existence of running jobs. The system administrator is to execute these commands on the system management node.

3.2.3.1 Starting and Stopping a Cluster

Use the pastart command to start an entire cluster and the pastop command to stop it.
Before starting a cluster, start the system management node that manages the cluster and then use this function. To stop a cluster, follow the reverse order: Stop nodes inside the cluster, and then finally stop the system management node.

Starting a Cluster

```
# pastart -c clstname
[INFO] PSM 2181 pastart All power control was executed.
```

Stopping a Cluster

```
# pastop -c clstname
[WARNING]
pastop is going to power-off all nodes of clstname
Do you really want to continue (y/n)? y
[INFO] PSM 2281 pastop All power control was executed.
```

clstname: Cluster name

The pastart command and the pastop command exit after instructing power-on/off to each node. They do not confirm whether the start/stop processing of a cluster is complete. To confirm whether the start/stop processing of a cluster is complete, specify the -w option. To check the status, use the pashowclst command. For details on how to use the pashowclst command, see "3.3.2 Displaying Operation Status of the System."

Note

- The papwrctl command also enables you to start and stop clusters and a family of nodes such as a node group. When using the papwrctl command, however, it does not take account of the start/stop sequence of nodes; it concurrently provides the start and stop instructions to a target node. For this reason, use the papwrctl command for a single node or nodes that are not dependent on each other.
  To start and stop nodes for each cluster or each node group, use the pastart command and the pastop command as described in the above example.
- The system management node is not subject to the power control provided by the pastart command and the pastop command.
- If the --force option is not specified, the pastop command displays a prompt to request your confirmation before the processing.
- If controlled nodes contain nodes in a redundant configuration and you want to stop both nodes, inhibit the failover function. If automatic failover is not inhibited, the shutdown process may cause a failover that prevents the node from being shutdown. For details on the inhibition of the failover function, see "3.4.5 Setting Automatic Failover."
- Multiuse nodes inside the compute cluster and the storage cluster are not subject to the pastart command and the pastop command. To perform power control for multiuse nodes when executing the pastart command and the pastop command, use the pre-post plugin. For details on the pre-post plugin, see "3.2.3.2 Pre-post Plugin."

- Confirming the startup completion

  Execute the pashowclst command. The cluster has successfully started if the STATUS field of all nodes displays Running. cluster1, cluster2, cluster3, fcluster, and mcluster are the names of respective clusters.

```
# pashowclst
CLUSTER  CLSTTYPE RUNNING STOPPED ERROR DISABLE
cluster1 COMPUTE  517     0       0     0
cluster2 COMPUTE  297     0       0     0
cluster3 COMPUTE  2453    0       0     0
fcluster STORAGE  4       0       0     0
mcluster MULTIUSE 4       0       0     0
```

  Note

  If a node does not start successfully, a failure may have occurred in hardware or software. Use the pashowclst command to check a node where a failure has occurred and the content of the failure. (See "3.3.2 Displaying Operation Status of the System.") Based on the check results, collect investigation data according to the "Job Operation Software Administrator's Guide for Maintenance." Then, contact a Fujitsu systems engineer (SE) or Fujitsu Support Desk with the collected data together with the display result of the pashowclst command.

- Confirming the stop completion

  To confirm that a cluster has stopped, use the pashowclst command as well. It has successfully stopped if all of the following conditions are applicable:
  - The STATUS of all nodes except the system management node (SMM) displays Stopped.
  - The PWR_STATUS field displays off.

- (For FX server only) The ARCH_STATUS field displays ICC_Stopped.
The following is a confirmation example of when the cluster name is cluster2.

```
# pashowclst -v -c cluster2
[ CLST: cluster2 ]
NODE        NODETYPE  STATUS   REASON  PWR_STATUS ARCH_STATUS    SRV_STATUS
0x00000001 SMM       Running  -       on         -              PWRD(o)
0x00000002 SMM       Running  -       on         -              PWRD(o)
0xFFFF0001 CCM       Stopped  -       off        -              -
0xFFFF0002 CCM       Stopped  -       off        -              -
0xFFFF0003 LN        Stopped  -       off        -              -
0xFF010001 BIO,CN    Stopped  -       off        ICC_Stopped    -
0xFF010002 SIO,CN    Stopped  -       off        ICC_Stopped    -
0xFF010003 GIO,CN    Stopped  -       off        ICC_Stopped    -
0xFF010004 CN        Stopped  -       off        ICC_Stopped    -
...
```

If it does not stop successfully, a failure may have occurred in hardware or software. Use the pashowclst command to check a node where a failure has occurred and the content of the failure. (See "3.3.2 Displaying Operation Status of the System.") Based on the results, collect investigation data according to the "Job Operation Software Administrator's Guide for Maintenance." Then, contact a Fujitsu systems engineer (SE) or Fujitsu Support Desk with the collected data together with the display result of the pashowclst command.

The pastart command and the pastop command start and stop each node in the following sequences.

Table 3.19 Power-On Sequence

Sequence	For Compute Cluster	For Storage Cluster
1	They start the compute cluster management node and the compute cluster sub management node, and then wait until the start processing is complete.	They start the storage cluster management node, the MGS node, and the MDS node, and then wait until the start processing is complete.
2	They provide the start instruction to the login node and the compute node (*).	They provide the start instruction to the OSS node.

(*) For FX server, it includes I/O nodes (boot I/O node, storage I/O node, and global I/O node).

Table 3.20 Power-Off Sequence

Sequence	For Compute Cluster	For Storage Cluster
1	They stop the login node and the compute node (*), and then wait until the stop processing is complete.	They stop the OSS node, and then wait until the stop processing is complete.
2	They provide the stop instruction to the compute cluster management node and the compute cluster sub management node.	They provide the stop instruction to the storage cluster management node, the MGS node, and the MDS node.

(*) For FX server, it includes I/O nodes (boot I/O node, storage I/O node, and global I/O node).

Note

- Before stopping the storage cluster, stop the compute cluster that uses the storage cluster.
- The pastart command does not start nodes for which the display result of the pashowclst command shows the "Alarm" status in the REASON field. To collectively power on clusters that contain such nodes, specify the --force-all option. If the --force-all option is specified, the power supply is forced to operate even if normal startup is not guaranteed. Be careful that a booted node may not work properly.
- The pastop command stops the compute node after waiting for the job to end. When it cannot check the state of the job, it retries this check until it is successful. To avoid waiting for the completion of the job, specify the --jobnowait option.
- If you specify the --jobnowait option of the pastop command, a compute node running a job may panic.
- The multiuse cluster does not have any control order. It collectively starts and stops multiuse nodes inside the cluster.

- You cannot specify the compute cluster or the storage cluster and the multiuse cluster concurrently.

See

To start a range specified as a controlled object in stages, set the number of concurrent executions and the interval. For details on the number of concurrent executions and the interval, see "3.2.2.1 The Number of Concurrent Executions and the Interval."

3.2.3.2 Pre-post Plugin

When starting and stopping systems collectively, there may be assumed cases in which you want to execute the pre-processing or post-processing of a command. Examples of such cases include:
- When you want to start a storage device before starting a node
- When you do not want to stop a storage device until all nodes stop, which is the opposite of the above case
For this reason, when performing a collective power-on/off, you can set plugins (pre-post plugin) which runs before and after executing a command. You can respectively set the pre-post plugins to collective start by the pastart command and collective stop by the pastop command. The system administrator is to create the plugins on the system management node.

Each of the pre-post plugins must meet the following conditions:

- Create them in a format that is executable on the system management node.
- The file name must consist of up to 63 characters, including only single-byte alphanumeric characters, hyphens, underscores, and periods. However, the first character cannot be a hyphen.
- The owner and the ownership group must be root and the permission must be either 0500 or 0700.

The following table shows the correspondence relationship between the names of the plugin scripts and the execution timings.

Table 3.21 Script List of the Pre-post Plugins

Plugin Script Name	Execution Timing
/etc/opt/FJSVtcs/scripts/pastartpre	Before collective start of pastart command
/etc/opt/FJSVtcs/scripts/pastartpost	After collective start of pastart command
/etc/opt/FJSVtcs/scripts/pastoppre	Before collective stop of pastop command
/etc/opt/FJSVtcs/scripts/pastoppost	After collective stop of pastop command

Note

- Pre-post plugins are executed only once before and after performing the power control for a cluster. A plugin script is executed only once even if you use the -c option of the pastart command and the pastop command to specify multiple clusters as the target.
- If a pre-post plugin returns abnormally (which means that the return value is not 0), the pastart command and the pastop command abnormally exit. When the plugin script of pastartpre and pastoppre returns abnormally, subsequent power controls are not performed.
- The output of pre-post plugins is output to the standard output.
- The current directory at the time of executing a pre-post plugin is /etc/opt/FJSVtcs/scripts.
- To make a multiuse node subject to collective power control, use a plugin script to define the processing.

The following describes the procedure for setting a pre-post plugin. This procedure uses pastartpre, which is a plugin script to be used before executing the pastart command, as an example. The system administrator is to carry out the following operations on the active system management node.

1. Creating a plugin script

```
# vi /etc/opt/FJSVtcs/scripts/pastartpre
```

2. Changing the permission to 0500 or 0700

```
# chmod 0500 /etc/opt/FJSVtcs/scripts/pastartpre
```

3. Delivering it to the standby system management node
   When the system management node is in a redundant configuration, use the pmscatter command to deliver the created pastartpre script to the standby system management node. If there is no standby system management node, this procedure is not necessary.

```
# pmscatter -c clstname --nodetype SMM -p /etc/opt/FJSVtcs/scripts/pastartpre /etc/opt/FJSVtcs/
scripts/pastartpre
```

   clstname: Cluster name

The following is an example of creating a pre-post plugin:

- Example of creating pastartpre, a plugin script to be used before executing the pastart command

  This example assumes the case where the system contains a node created by KVM and you want to start a node as the host of the KVM node before executing the pastart command. In this example, the node type as the host of the KVM node shall be defined as "HOST," a multiuse node, and the cluster name shall be compute.

```
# cat /etc/opt/FJSVtcs/scripts/pastartpre
#!/bin/bash
papwrctl -c compute -a --nodetype HOST -w --force on
exit $?
```

3.2.4 Controlling the Location LED

Use the paledctl command in order to turn on/off the location LED of FX server or check the LED lighting status. The system administrator is to execute the paledctl command on the system management node.

Turning on the LED

```
# paledctl -c clstname --bootgrp bootgid on
[INFO] PSM 2481 paledctl LED control was executed.
```

Turning off the LED

```
# paledctl -c clstname --bootgrp bootgid off
[INFO] PSM 2481 paledctl LED control was executed.
```

clstname: Cluster name
bootgid: Boot group ID

There are the following types of the lighting statuses of the location LED, which the paledctl command displays.

Table 3.22 Location LED Lighting Statuses to Which You Can Refer by the paledctl Command

LED Lighting Status	Meaning
OFF	LED off
ON	LED on
UNKNOWN	Unknown

- Display example of the location LED lighting status (The cluster name is "compute" and the target boot groups are 0x0101 to 0x0104.)

```
# paledctl -c compute --bootgrp 0x0101-0x0104 show
BOOTGRP STATUS
0x0101  :ON
0x0102  :ON
0x0103  :OFF
0x0104  :ON
```

Information

By specifying the boot group ID in the --bootgrp option of the paledctl command, you can control the location LED of FX server installed for each boot group (BoB).

3.3 Details of the System Monitoring Function

3.3.1 Monitoring Mechanism of the System Monitoring Function

The system monitoring function periodically monitors all nodes of the system. (This monitoring is hereinafter called as periodical monitoring).

The roles of the periodical monitoring are as follows:

- Software abnormality monitoring

  This monitors the following items on a periodical basis:
  - The alive status of the operating system (hereinafter called OS) for nodes of the entire system
  - Whether there is no abnormality in the service status of the Job Operation Software

  If it detects any abnormality in a node, the system monitoring function automatically takes a countermeasure to avoid any effect on job operations. This countermeasure includes disconnecting the node from job operations, using failover to switch the active/standby system, and collecting a dump file.

- Hardware abnormality monitoring

  This monitors the following items on a periodical basis:
  - Power status of the entire system
  - Whether there is no abnormality in hardware for nodes of FX server

  If it detects any abnormality in an FX server node, the system monitoring function automatically disconnects it from job operations in order to avoid any effect on job operations.

3.3.1.1 Software Abnormality Monitoring

The software abnormality monitoring monitors large-scale systems by using the hierarchical structure for load sharing. The "monitoring function daemons," which are placed on each node, cooperate between nodes in order to monitor each node and convey the monitoring result.

This sends information from the "monitoring function daemon" of a node in the lower layer to the "monitoring function daemon" of a node in the upper layer in stages. Finally, this collects the monitoring results of all nodes in the "monitoring function daemon" on the system management node.

The pashowclst command displays the system operation status, which means these collected monitoring results.

The hierarchical structure of nodes in the monitoring daemons has up to five layers. The monitoring daemon of each layer cooperates with the monitoring daemon in the one-level upper/lower layer between nodes, so that these daemons acquire and convey information.

The following table shows the hierarchical structure of nodes in the monitoring daemons.

Table 3.23 Hierarchical Structure of Nodes in the Monitoring Daemons

Layer	Node Type		
	Compute Cluster	Storage Cluster	Multiuse Cluster
1st layer	System management node	System management node Node serving as both system management node and storage cluster management node	System management node
2nd layer	Compute cluster management node	Storage cluster management node	Multiuse node

Layer	Node Type		
	Compute Cluster	Storage Cluster	Multiuse Cluster
3rd layer	Compute cluster sub management node Login node Multiuse node	MGS node MDS node OSS node	-
4th layer	Node serving as both compute node and boot I/O node (FX server) Compute node (PRIMERGY server)	-	-
5th layer	Node serving as both compute node and global I/O node (FX server) Node serving as both compute node and storage I/O node (FX server) Compute node (FX server)	-	-

There are three types of the monitoring daemons: pxmonitor_masterd, pxmonitor_subd, and pxmonitor_slaved. Each of them has a specific node type(s) to be arranged.

The following table shows the arrangement destinations and the roles of each of the monitoring daemons.

Table 3.24 Arrangement and Role of the Monitoring Daemons

Monitoring Daemon Name	Node Type of Arrangement Destination	Role
pxmonitor_masterd	System management node Node serving as all of system management node, compute cluster management node, and login node Node serving as both system management node and storage management node	Monitoring (*1) This monitors the OS alive status of a current node from pxmonitor_slaved of the next lower node.
		Information collection pxmonitor_subd of the next lower node sends information on the subordinate nodes. Then, this daemon collects the monitoring results of the OS alive and service status of all nodes in the system. You can use the pashowclst command to check the collected states of each node.
		Coping with the system after detecting abnormality (*1) Check the states of all collected nodes. Then, perform the automatic failover, automatic service restart, and dump collection against a node(s) having an abnormality in the OS or service.
pxmonitor_subd	Compute cluster management node (*2) Compute cluster sub management node Node serving as both compute node and boot I/O node Storage cluster management node (*2)	Monitoring (*1) This monitors the OS alive status of a current node from pxmonitor_slaved of the next lower node.
		Information collection pxmonitor_subd of the next lower node sends information on the subordinate nodes. Then, this daemon collects the monitoring results of the OS alive and service status of the subordinate nodes.
		Information transmission (*1) This sends the collected monitoring results to pxmonitor_masterd or pxmonitor_subd on the next upper layer.
pxmonitor_slaved	All nodes	Monitoring

Monitoring Daemon Name	Node Type of Arrangement Destination	Role
		This monitors the state of a service from the service or the shared file system (FEFS) of the Job Operation Software inside the local node.
		Response This responds to monitoring requests from pxmonitor_masterd or pxmonitor_subd on the next upper layer. At this time, this sends the state of a service monitored inside the local node.

(*1) A node, whose node type in the arrangement destination is the active system (including a single configuration), performs these roles.
(*2) If the node is also serving as system management node, pxmonitor_subd daemon is not arranged in the node.

As shown in "Table 3.23 Hierarchical Structure of Nodes in the Monitoring Daemons" and "Table 3.24 Arrangement and Role of the Monitoring Daemons," pxmonitor_masterd and pxmonitor_subd have different nodes to be monitored, depending on the node types of arrangement destination. They monitor nodes one level lower than nodes that they run. (For the system management node, they monitor system management nodes on the same layer.)

In addition, pxmonitor_subd sends the collection results of all nodes on a layer lower than the local node to pxmonitor_masterd or pxmonitor_subd on the next upper layer. Eventually, pxmonitor_masterd collects the results of all nodes.

The following table shows the relationship between the information collected in the monitoring daemons and the node type of arrangement destination.

Table 3.25 Relationship Between Nodes Arranging Monitoring Daemons and Collected Information

Monitoring Daemon Name	Node to Arrange Daemon	Collected Information
pxmonitor_masterd	System management node	Monitoring results of all nodes in all clusters
pxmonitor_subd	Compute cluster management node	Monitoring results of all nodes in clusters except system management node and compute cluster management node
	Compute cluster sub management node	Monitoring results of all nodes in the same node group except the local node and the pair node
	Node serving as both compute node and boot I/O node	Monitoring results of all nodes in the same boot group except the local node
	Storage cluster management node	Monitoring results of all nodes in clusters except system management node and storage cluster management node

The following describes the states that the software abnormality monitoring monitors.

- Monitoring the alive status of the OS

  On each node, the processes of the system monitoring function cooperate between nodes in order to monitor the alive status of the OS.

  Regarding the alive status of the OS, it monitors whether a network access is available to a node to be monitored. (This network access is established by a TCP/IP protocol, which is via the Tofu network for nodes except the boot I/O node of FX server and via the management network for the boot I/O node of FX server and nodes other than FX server.)

- Monitoring the service status

  To check the state of each service, it cooperates with the following functions on a node to be monitored.
  The following table lists the services to be monitored.

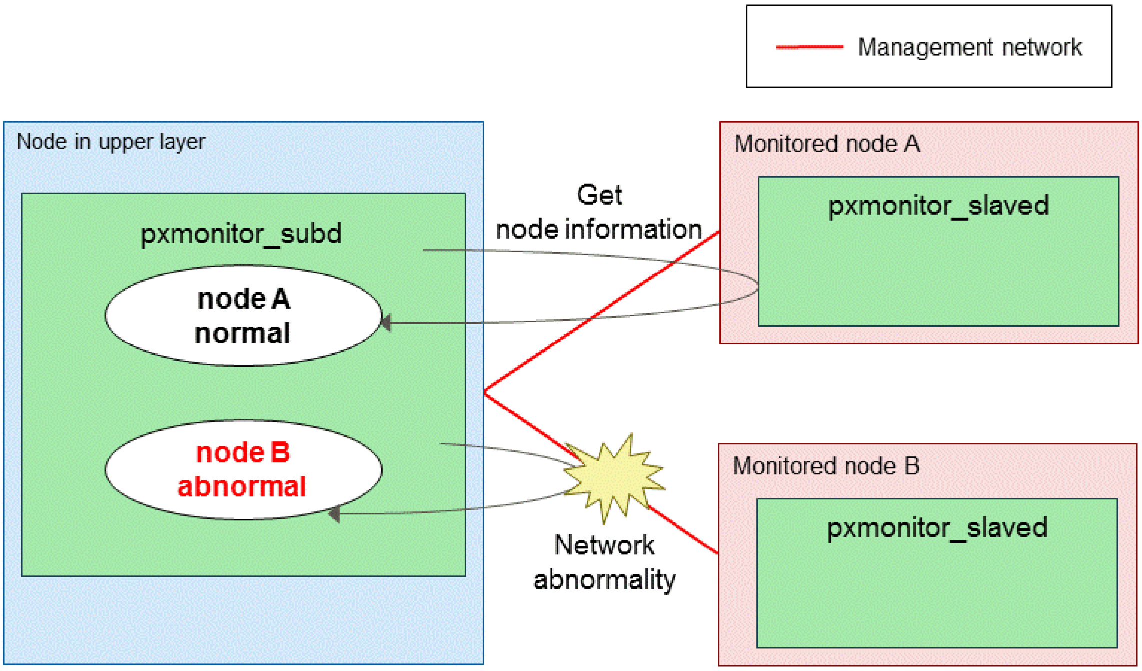
Table 3.26 Services Subject to Service Monitoring

Abbreviation	Function
PJM	Job manager function
MRD	Job resource manager (Master resource management function)

Abbreviation	Function
SRD	Job resource manager (Sub resource management function)
NRD	Job resource manager (Resource management function inside node)
PLE	Parallel execution environment
PWRD	Power management function
FEFS	File system (Client function)
FEFSSR	File system (Server/Router function)
IC	Management function of Mellanox HCA
Other than above	User-defined service

The following is an example that pxmonitor_subd monitors the OS alive status of a one-level lower node to be monitored.

Figure 3.2 Image of Monitoring the OS Alive status



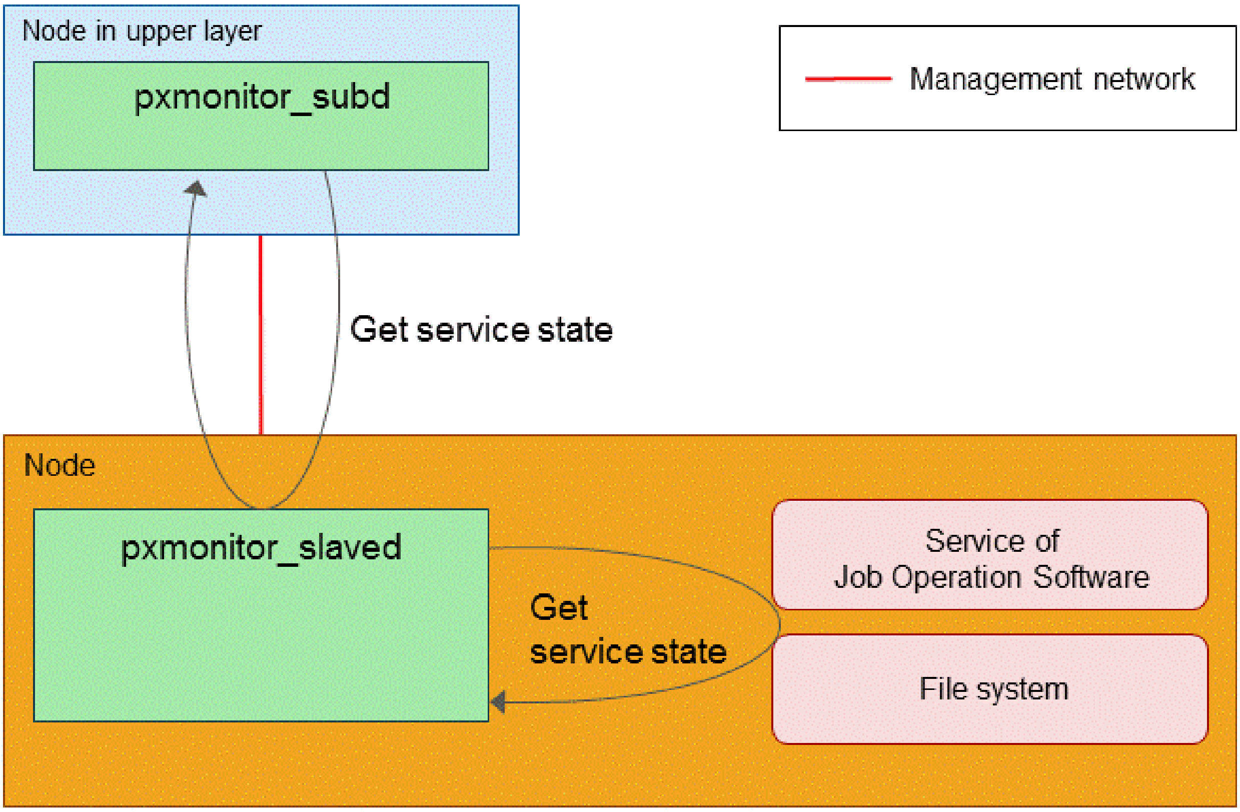
If you manage to get node information from pxmonitor_slaved, the next lower node, while monitoring the OS alive status, it is determined that this node is normal.

However, if any abnormality in the network between nodes prevents you from getting node information from one level lower of pxmonitor_slaved, it is determined that this node is abnormal.

pxmonitor_slaved monitors the service status of the Job Operation Software that runs on the local node.
pxmonitor_slaved passes the service status on the local node to pxmonitor_subd when pxmonitor_subd monitors the OS alive status.

The following is an example of pxmonitor_slaved monitoring the service on a node.

Figure 3.3 Image of Monitoring the Service Status



If it detects any abnormality of the OS alive status or any abnormality of the service status in a node, its operation after the detection of the abnormality varies, depending on the node type.
According to the node type, the node detecting any software abnormality automatically restarts a failover or service.
You can use the -v option of the pashowclst command to check the state of a node that detected an abnormality.

See

For details on displaying the -v option of the pashowclst command, see "3.3.2.2 Detailed Display of Operation Status of the System."

Table 3.27 Operations for Each Node Type when Detecting Abnormality

Node Type	Abnormality Type	Active System	Standby System
System management node	OS alive status	Failover Dump collection	Dump collection
	Service state	Failover Dump collection	Restarting service
Node serving as both system management node and storage cluster management node	OS alive status	Failover Dump collection	Dump collection
	Service state	Failover Dump collection	For PWRD service: Restarting service For IC service: None
Node serving as all of system management node, compute cluster management node, and login node	OS alive status	-	-
	Service state	Restarting service	-
Compute cluster management node	OS alive status	Failover Dump collection	Dump collection

Node Type	Abnormality Type	Active System	Standby System
	Service state	Failover Dump collection	For FEFS service: None For any service other than FEFS: Restarting service other than FEFS
Compute cluster sub management node	OS alive status	Failover Dump collection	Dump collection
	Service state	Failover Dump collection	Restarting service
Node serving as both compute node and boot I/O node	OS alive status	Disconnecting from job operations Powering off nodes subordinate to local node (for only power supply to nodes) Dump collection	-
	Service state	For SRD and PLE services: Restarting service For NRD, FEFS, and PWRD services: Disconnecting from job operations without restarting service	-
Node serving as both compute node and global I/O node	OS alive status	Disconnecting from job operations Dump collection When there is no allocation node by round-robin method, all nodes in GIO group range are disconnected from job operations (The status of FEFS service becomes "a.")	-
	Service state	Disconnecting from job operations For abnormality of FEFSSR service: When there is no allocation node by round-robin method, all nodes in GIO group range are disconnected from job operations (The status of FEFS service becomes "a.")	-
Node serving as both compute node and storage I/O node	OS alive status	Disconnecting from job operations Dump collection All nodes in SIO group range are disconnected from job operations (The status of FEFS service becomes "a.")	-
	Service state	Disconnecting from job operations For abnormality of FEFSSR service: All nodes in SIO group range are disconnected from job operations (The status of FEFS service becomes "a.")	-
Compute node	OS alive status	Disconnecting from job operations Dump collection	-
	Service state	Disconnecting from job operations	-
Login node	OS alive status	Dump collection	-
	Service state	None	-
Storage cluster management node	OS alive status	Failover Dump collection	Dump collection
	Service state	Failover Dump collection	None

Node Type	Abnormality Type	Active System	Standby System
MGS node (*1)	OS alive status	FEFS failover (*2) Dump collection	Dump collection
	Service state	FEFS failover (*2) Dump collection	Dump collection
MDS node	OS alive status	FEFS failover (*2) Dump collection	Dump collection
	Service state	FEFS failover (*2) Dump collection	Dump collection
OSS node	OS alive status	FEFS failover (*2) Dump collection	Dump collection
	Service state	FEFS failover (*2) Dump collection	Dump collection
Multiuse node	OS alive status	Dump collection	-
	Service state	None	-

(*1) This includes the node serving as both MGS and MDS nodes and the node serving as all of MGS, MDS, and OSS nodes.
(*2) For details on the failover of FEFS, see the *FEFS User's Guide*.

See

- The system monitoring plugin allows you to define the operation for when any abnormality is detected on a user-defined service. For details, see "3.3.5 Settings of the System Monitoring Plugin."
- The disconnection of a job operation occurs when detecting any abnormality in the OS alive status or the service status. This disconnection inhibits a new job submission and stops running jobs.
- See "3.4.5 Setting Automatic Failover" to define whether or not to execute failover automatically.

See

When detecting an abnormality in a node, see the "Job Operation Software Troubleshooting" in order to cope with the cause of this abnormality.

3.3.1.2 Hardware Abnormality Monitoring

The hardware status monitoring function acquires the power status of a node and monitors the state of hardware. This uses the IPMI protocol via the control network to access the BMC of PRIMERGY server nodes and the BMC of FX server from pxmonitor_masterd of the active system management node.

- FX server

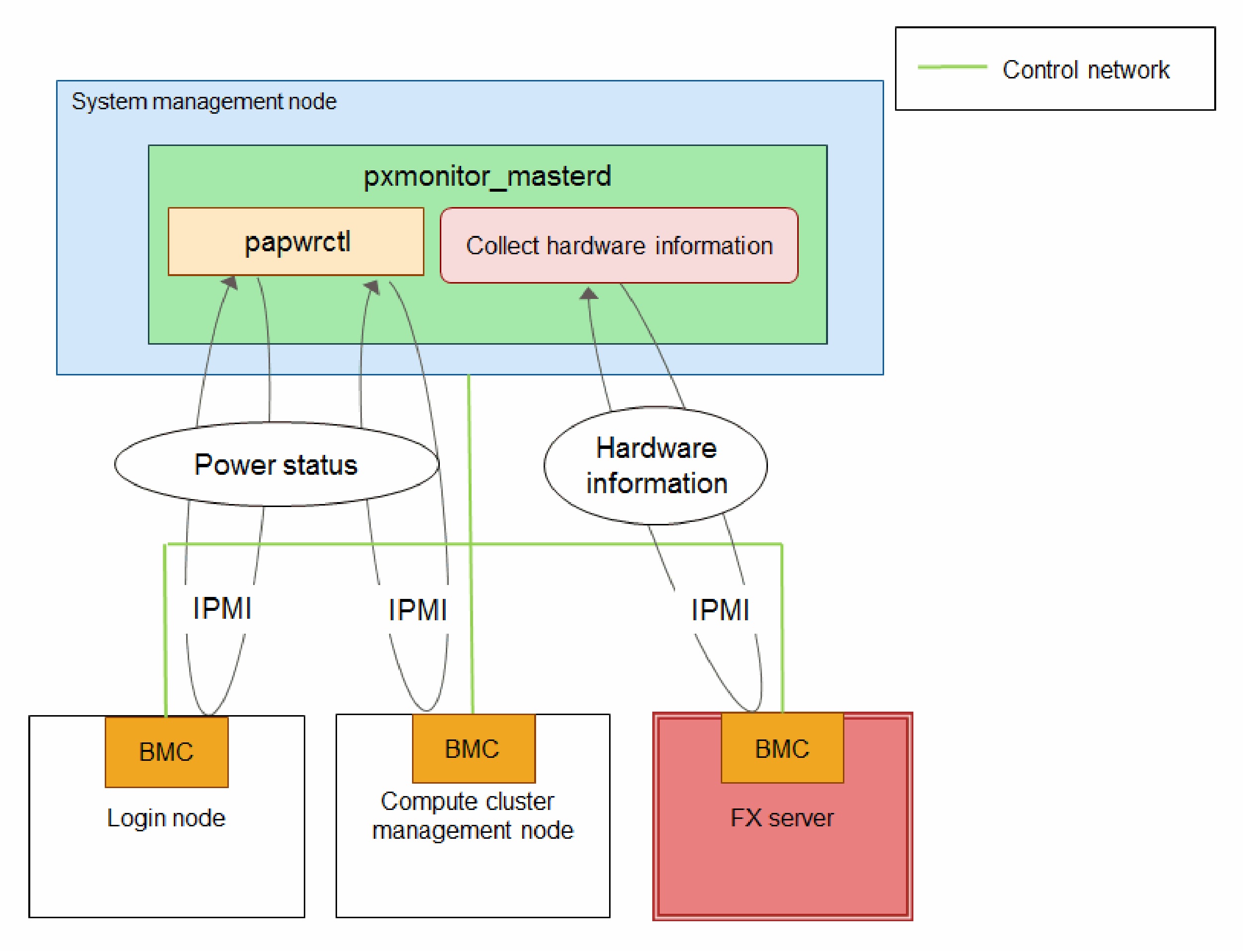
  This monitors the power status of a node and the node failure status (for CPU, memory, Tofu port, etc.).
  When detecting a failure of a node, this function determines that it is an abnormality in hardware status, and then disconnects the node from job operations.

- Nodes other than FX server

  This acquires the power status of a node. This uses the papwrctl command to get the power information of a node.

The image of the hardware abnormality monitoring is as follows.

Figure 3.4 Image of Hardware Abnormality Monitoring



After this function detects an abnormality of FX server hardware from a node, the node operates differently according to the abnormality type.

You can check the abnormality type from the REASON field, which appears when specifying the -v option of the pashowclst command.

See

For details on displaying the -v option of the pashowclst command, see "3.3.2.2 Detailed Display of Operation Status of the System."

The following table lists the operations for each of the abnormality types when detecting an abnormality.

Table 3.28 Operations for Each of the Abnormality Types when Detecting an Abnormality

Abnormality Type (REASON)	Job	Operation After Detecting Abnormality	
		Job for Corresponding Node	Job Using ICC (Channel) of Corresponding Node
ResetRequest	New job submission	No	Yes
	Running job	Yes	Yes
ReservedAlarm	New job submission	No	Yes
	Running job	Yes	Yes
Alarm	New job submission	No	Yes
	Running job	No	Yes
PortReservedAlarm	New job submission	No	No

Abnormality Type (REASON)	Job	Operation After Detecting Abnormality	
		Job for Corresponding Node	Job Using ICC (Channel) of Corresponding Node
	Running job	Yes	Yes
PortRouterFatal	New job submission	No	No
	Running job	Conditional	Conditional

Those with "Yes" on "Running job" continue the running job.
Those with "No" on "Running job" stop the running job.
Those with "Conditional" on "Running job" either stop or continue the running job, depending on the instruction for when a Tofu interconnect link is down. For an instruction not to change the communication path even though the link is down, the job stops. For an instruction to change the communication path, the bypass route is searched and the job attempts to continue.

See

- For the meanings of hardware abnormality types, see "3.3.2 Displaying Operation Status of the System."
- To incorporate a node into job operations again after the node detects an abnormality in hardware, you must perform the maintenance procedure for the system maintenance function. For details, see the "Job Operation Software Administrator's Guide for Maintenance."

3.3.2 Displaying Operation Status of the System

You can check the operation status of the system by executing the pashowclst command on the system management node.

It can display the following pieces of information:

- Summary display of the operation status of the system

  This information displays the summary of the operation status of the entire system.

- Detailed display of operation status of the system

  This information displays detailed information of the operation status of the entire system.

- Display of active/standby system status of a redundant configuration node

  This information displays the current information of the active/standby system on a redundant configuration node.

- Display of the number of nodes in maintenance work

  This information displays the number of nodes which transition to the software maintenance mode or the hardware maintenance mode due to maintenance work.

3.3.2.1 Summary Display of the Operation Status of the System

This displays the summary of the operation status of the entire system.
Execute the pashowclst command on the system management node.
Display the summary of the operation status by executing the pashowclst command without specifying an option.
The following is an example of displaying the summary of the operation status of the entire system, and the list of the meanings of the displayed items.

```
# pashowclst
CLUSTER          CLSTTYPE  RUNNING  STOPPED  ERROR    DISABLE
compute          COMPUTE   55       0        0        0
storage          STORAGE   8        0        0        0
```

CLUSTER

Cluster name

CLSTTYPE

Cluster type

RUNNING

The number of nodes whose status is Running (meaning that job operations are possible)

STOPPED

The number of nodes whose status is Stopped (meaning that they have stopped or are in start/stop processing)

ERROR

The number of nodes whose status is SoftError (meaning that an abnormality has occurred in an OS state or service)

DISABLE

The number of nodes whose status is Disable (meaning that they are going to be disconnected or have already been disconnected from job operations)

3.3.2.2 Detailed Display of Operation Status of the System

This function displays detailed information on the operation status of the system.
Detailed information of the operation status of the system is as follows:

- Node status
- Reason for transition of the status of nodes
- Power status of nodes
- Status of Tofu channel of nodes (FX server)
- Service operation status of the Job Operation Software

Display detailed information of the operation status for each node by specifying the -v option in the pashowclst command.
You can also display it by extracting its display range. For details, see "pashowclst command" in the "Job Operation Software Command Reference."
The following is an example of displaying detailed information of the operation status of the entire system, and the list of the meanings of the displayed items.

```
# pashowclst -v
[ CLST: compute ]
NODE        NODETYPE  STATUS   REASON  PWR_STATUS  ARCH_STATUS  SRV_STATUS
0x00000001  SMM       Running  -       on          -            PWRD(o)
0x00000002  SMM       Running  -       on          -            PWRD(o)
0xFFFF0001  CCM      Running  -      on        -          PJM(o),PLE(o),MRD(o),SRD(o),FEFS(o),PWRD(o)
0xFFFF0002  CCM      Running  -      on        -          PJM(o),PLE(o),MRD(o),SRD(o),FEFS(o),PWRD(o)
0xFFFF0003  LN        Running  -       on          -            PLE(o),FEFS(o)
0x01FF0001  CCS       Running  -       on          -            PLE(o),SRD(o)
0x01FF0002  CCS       Running  -       on          -            PLE(o),SRD(o)
0x01010001  BIO,CN    Running  -       os-running  ICC_Running  PLE(o),SRD(o),NRD(o),FEFS(o),PWRD(o)
0x01010002  SIO,CN   Running  -      os-running  ICC_Running  PLE(o),NRD(o),FEFSSR(o),FEFS(o),PWRD(o)
0x01010003  GIO,CN   Running  -      os-running  ICC_Running  PLE(o),NRD(o),FEFSSR(o),FEFS(o),PWRD(o)
0x01010004  CN        Running  -       os-running  ICC_Running  PLE(o),NRD(o),FEFS(o),PWRD(o)
...
[ CLST: storage ]
NODE        NODETYPE  STATUS   REASON  PWR_STATUS  ARCH_STATUS  SRV_STATUS
0x00000001  SMM       Running  -       on          -            PWRD(o)
0x00000002  SMM       Running  -       on          -            PWRD(o)
0xFFFF0001  SCM       Running  -       on          -            IC(o)
0xFFFF0002  SCM       Running  -       on          -            IC(o)
0xFFFF0003  MGS       Running  -       on          -            FEFSSR(o)
0xFFFF0004  MGS       Running  -       on          -            FEFSSR(o)
0xFFFF0005  MDS       Running  -       on          -            FEFSSR(o)
0xFFFF0006  MDS       Running  -       on          -            FEFSSR(o)
0xFFFF0007  OSS       Running  -       on          -            FEFSSR(o)
```

```
0xFFFF0008  OSS       Running  -       on          -            FEFSSR(o)
...
```

NODE

Node ID

NODETYPE

Node type

STATUS

Node status

REASON

Reason for transition of the status of nodes

PWR_STATUS

Power status of nodes

ARCH_STATUS

Status of Tofu channel of nodes

SRV_STATUS

Service operation status of the Job Operation Software

The status of nodes falls into four categories (status categories). In the summary display provided by the pashowclst command, it is grouped by status categories.

The following table shows the relationship between the statuses of nodes and status categories.

Table 3.29 The Statuses of Nodes and Classified Status Categories

Status Category	Node Status	Description
RUNNING	Running	The service of the Job Operation Software has started, so that the node is operable.
STOPPED	Stopped	The node (or the OS) has not started yet. You cannot execute a job.
	Init	The node is under start processing. You cannot execute a job.
	Stopping	The node is under stop processing. You cannot execute a job.
	Unknown	The status is unknown.
ERROR	SoftError	An abnormality in software or a service is occurring. This prevents you from executing a job.
DISABLE	PreDisable	The node is under processing of resource utilization inhibition. Running jobs wait for the completion of their execution. You cannot assign a new job.
	Disable	The node is in the unusable state for resource. You cannot execute a job.

You can use the pashowclst command to display the current status of nodes. The REASON field of the pashowclst command displays the cause why the node gets to the current status. The meanings of the displayed contents are as follows.

Table 3.30 Reason for Transition of the Node Status

Status	Displayed Contents on REASON Field		Description
	FX server	PRIMERGY server	
Stopped	-		No detailed information on the cause
Unknown	-		

<table>
<tr><th rowspan="2">Status</th><th colspan="2">Displayed Contents on REASON Field</th><th rowspan="2">Description</th></tr>
<tr><th>FX server</th><th>PRIMERGY server</th></tr>
<tr><td>Init</td><td colspan="2">-</td><td rowspan="3"></td></tr>
<tr><td>Stopping</td><td colspan="2">-</td></tr>
<tr><td rowspan="2">Running</td><td colspan="2">-</td></tr>
<tr><td colspan="2">DeadlineSchedule</td><td>Being scheduled to stop as set by the deadline schedule</td></tr>
<tr><td rowspan="2">SoftError</td><td colspan="2">NodeDown</td><td>Detecting an abnormality on the alive status monitoring of a node</td></tr>
<tr><td colspan="2">SrvDown</td><td>Detecting an abnormality of a service in association with job operations</td></tr>
<tr><td rowspan="3">PreDisable</td><td colspan="2">Manual</td><td>Resource utilization inhibition state set by user manually</td></tr>
<tr><td>ReservedAlarm</td><td>-</td><td>ReservedAlarm occurrence state (*)</td></tr>
<tr><td>PortReservedAlarm</td><td>-</td><td>PortReservedAlarm occurrence state (*)</td></tr>
<tr><td rowspan="11">Disable</td><td colspan="2">HardMaintenance</td><td>Hardware maintenance state</td></tr>
<tr><td colspan="2">SoftMaintenance</td><td>Software maintenance state</td></tr>
<tr><td colspan="2">Manual</td><td>Unusable state for resource set by user manually</td></tr>
<tr><td colspan="2">DeadlineSchedule</td><td>Unusable state for resource set by deadline schedule</td></tr>
<tr><td>-</td><td>AbnormalEnd</td><td>State in which job resource utilization is automatically inhibited because the node does not stop normally</td></tr>
<tr><td>ResetRequest</td><td>-</td><td>ResetRequest occurrence state (*)</td></tr>
<tr><td>ReservedAlarm</td><td>-</td><td>ReservedAlarm occurrence state (*)</td></tr>
<tr><td>RouterFatal</td><td>-</td><td>RouterFatal occurrence state (*)</td></tr>
<tr><td>PortRouterFatal</td><td>-</td><td>PortRouterFatal occurrence state (*)</td></tr>
<tr><td>PortReservedAlarm</td><td>-</td><td>PortReservedAlarm occurrence state (*)</td></tr>
<tr><td>Alarm</td><td>-</td><td>Alarm occurrence state (*)</td></tr>
</table>

(*) See "Table 3.37 Abnormality of FX server Hardware."

The following table shows the meanings of the power statuses of nodes that the pashowclst command displays (on the PWR_STATUS field).

Table 3.31 Power Statuses of Nodes [FX server and Any Node]

Power Status of Nodes	Description
off	Power off
reset	During initialization
initialization	During OBP initialization
os-boot	During OS startup
os-running	While OS is running
os-shutdown	During shutdown processing of OS

Power Status of Nodes	Description
panic	Panic (*1)
BootFailed	Boot failure
Unknown	Unknown (*2)

Table 3.32 Power Statuses of Nodes [PRIMERGY server]

Power Status of Nodes	Description
off	Power off
on	Power on (*3)
Unknown	Unknown (*2)

(*1) The FX server panic switches to off when memory dump collection is completed.
(*2) This applies to cases such as failure to acquire the power status.
(*3) For PRIMERGY server, the meaning of on includes during initialization, During OS startup or during shutdown processing of OS, and panic.

The following table shows the meanings of the architecture-specific statuses that the pashowclst command displays (on the ARCH_STATUS field).

Table 3.33 Statuses of Tofu Channel (ICC) of Nodes (FX server)

Architecture-Specific Status		Description
FX server	PRIMERGY server	
ICC_Running	-	ICC is available.
ICC_Stopped	-	ICC is unavailable (due to some reason such as power-off of a chassis).
ICC_Disable	-	It becomes unavailable as the channel of a new job, canceling running jobs.
ICC_PreDisable	-	It becomes unavailable as the channel of a new job.

The following table lists the abbreviations of the services that the Job Operation Software monitors.

Table 3.34 Abbreviations of the Services

Abbreviation	Function
PJM	Job manager function
MRD	Job resource manager (Master resource management function)
SRD	Job resource manager (Sub resource management function)
NRD	Job resource manager (Resource management function inside node)
PLE	Parallel execution environment
PWRD	Power management function
FEFS	File system (Client function)
FEFSSR	File system (Server/Router function)

The services monitored by the Job Operation Software vary by node. The following table lists the service names and the node types that the pashowclst command displays.

Table 3.35 Services and Node Types

Service	Node Type												
	SMM	CCM	CCS	BIO	SIO	GIO	CN	LN	SCM	MGS	MDS	OSS	Multiuse
PJM		Yes											
MRD		Yes											
SRD		Yes	Yes	Yes									
NRD				Yes	Yes	Yes	Yes						
PLE		Yes	Yes	Yes	Yes	Yes	Yes	Yes					
PWRD	Yes	Yes		Yes	Yes	Yes	Yes						
FEFS		Yes		Yes	Yes	Yes	Yes	Yes					(*)
FEFSSR					Yes	Yes				Yes	Yes	Yes	

Yes: Node type to which a service is applied
(*) This appears if the FEFS service has been installed.

Information

- For the compute node and the storage cluster management node of PRIMERGY server that mount IB, the IC service is to be monitored and displayed by the pashowclst command.
- If two or more IB-HCAs are mounted and one of them fails, this failure is treated as an abnormality of a service.

As shown below, the pashowclst command uses a symbol to display an operation status of a service.

Table 3.36 Display of Operation Status of a Service by the pashowclst Command

Display of Operation Status	Description
b	Service has not started yet (which means before start processing of the service, and during starting of the Job Operation Software).
s	During service initialization
o	During service operating
d	During stop processing of service
a	I/O error in file system (*1)
x	Service stopped or an abnormality occurred.
!	Degradation occurred. (A function is contracting.)
w	The standby system has completed starting. Failback is available.
*	Being operated in consolidation (*2)
f	During failover
?	Unknown (Example: Right after starting of the Job Operation Software)

(*1) This occurs because an abnormality in the storage I/O node or the global I/O node prevents the compute node using these nodes from using the I/O function for the file system.

(*2) Consolidation here means that one node is set as the active system node from the state of the mutual standby configuration. See "3.4.2 Switching a Node."

Information

The end status of a script created by the user determines the display of operation status of the user-defined service.
For details on the user-defined service, see "3.3.6 Settings of the User-Defined Service."

You can check the hardware abnormality status of FX server by using the pashowclst command to display the operation status of the node. The following describes the statuses displayed when a hardware abnormality occurs. The REASON field on the display of the pashowclst command displays these statuses.

Table 3.37 Abnormality of FX server Hardware

REASON	Description
ResetRequest	You need to restart the node because the memory on the node failed. The corresponding node waits for the end of a running job, preventing you from assigning a new job.
ReservedAlarm	A serious error has occurred in a node part, which causes a significant impact of performance degradation or reliability degradation. The corresponding node waits for the end of a running job, preventing you from assigning a new job.
RouterFatal	A fatal error has occurred in a Tofu port part. The corresponding node forcibly exits a running job, preventing you from assigning a new job.
PortRouterFatal	A fatal error has occurred in a Tofu port part. If the job running on the relevant node has an instruction to change the communication path when a Tofu interconnect link goes down, the Job Operation Software waits for the job to end. After the job ends, no new jobs are allocated to that node. If the job has an instruction not to change the communication path when a Tofu interconnect link goes down, the Job Operation Software forcibly terminates the job. After that, no new jobs are allocated to that node.
PortReservedAlarm	An error has occurred in a Tofu port part, which causes a significant impact of performance degradation or reliability degradation. The corresponding node waits for the end of a running job, preventing you from assigning a new job.
Alarm	A fatal error has occurred in a node part. The corresponding node forcibly exits a running job, preventing you from assigning a new job.

If any of the above hardware abnormalities occurs, perform a hardware maintenance according to the maintenance procedure.

3.3.2.3 Displaying the Update Elapsed Time of Operation Status

For nodes whose STATUS is not in the RUNNING status, you can display the elapsed time from the time when such nodes reflect the change of an operation status in the system monitoring function. To achieve this, set the -d option and the -t option in the pashowclst command.

Use this to check the time when an abnormality occurs or the elapsed time during start processing.

```
# pashowclst -v -d -t
[ CLST: compute ]
NODE       NODETYPE  STATUS     REASON       PWR_STATUS  ARCH_STATUS  SRV_STATUS
FROM_UPDATE
0xFFFF0003 LN       SoftError  NodeDown    on          -           -                          00:53:19
0x01010005 CN        Disable    RouterFatal  os-running  ICC_Disable  PLE(o),NRD(o),FEFS(o),PWRD(o)
00:03:04
```

FROM_UPDATE

The elapsed time since the last time when reflecting the change of the operation status (hour:minute:second)

Note

- After the system monitoring function receives a status change on the system management node, it takes a few seconds for the entire system to reflect this change.

The starting point of the elapsed time output to FROM_UPDATE is the time when reflecting a status update. Therefore, the elapsed time since the last update of the operation status may appear for a few seconds right after a status change occurs.

- When the active system management node is switched by failover, the update elapsed time of operation status is not carried over. Updated by status changes that occurred after a failover.

3.3.2.4 Displaying the Details of the Cause for the Transition of Operation Status

By specifying the -v option and the --detail option in the pashowclst command, you can display the details of the cause why a node status transitions.

```
# pashowclst -c clstname -v --detail
[ CLST: clstname ]

NODE        NODETYPE  STATUS     REASON    PWR_STATUS  ARCH_STATUS  SRV_STATUS              DETAIL
0x00000001  SMM       Running    -         on          -            PWRD(o)                 -
0x00000002  SMM       Running    -         on          -            PWRD(o)                 -
0xFFFF0001  CCM      SoftError  NodeDown  on          -            -                    Detect by PRM
0xFFFF0001  CCM     SoftError  NodeDown  off         -          -                   Monitoring failed
0xFFFF0003  LN        Running    -         on          -            PLE(o),FEFS(o)          -
...
```

DETAIL

Detailed information on abnormality occurrences
"-" means that no detailed information exists.

3.3.2.5 Display of Active/Standby System Status of a Redundant Configuration Node

This displays the status of the active/standby system related to nodes in a redundant configuration.
Use this to check the active/standby system when implementing the rolling update in the software maintenance.
To display the status about the active/standby system of nodes, specify the -m option in the pashowclst command.

```
# pashowclst -m
[ CLST: compute ]
NODEGRP    NODETYPE      NODE        STATUS  NODE        STATUS
SMMGRP     SMM           0x00000001  (A)     0x00000002  (S)
MASTERGRP  CCM           0xFFFF0001  (S)     0xFFFF0002  (A)
0x01       CCS           0x01FF0001  (S)     0x01FF0002  (A)
[ CLST: storage ]
NODEGRP    NODETYPE      NODE        STATUS  NODE        STATUS
SMMGRP     SMM           0x00000001  (A)     0x00000002  (S)
MASTERGRP  SCM           0xFFFF0001  (A)     0xFFFF0002  (S)
```

NODEGRP

Group to which a node belongs

SMMGRP: Control group (indicating the system management node)
MASTERGRP: Management group (indicating the compute cluster management node and the storage cluster management node)
Node group ID: Node group (indicating the compute cluster sub management node)
FILEGRP: File group (indicating the MGS node, the MDS node, and the OSS node)

NODETYPE

Node type

NODE

Node ID

STATUS

Active system (A) / Standby system (S)
For the MGS node, the MDS node, and the OSS node, a hyphen "(-)" appears here.

To check a pair node in a redundant configuration for the file server nodes (the MGS node, the MDS node, and the OSS node), specify the -m option and the --filegrp option in the pashowclst command. (The pair node becomes a pair at the time of failover.)

```
# pashowclst -m --filegrp
[ CLST: compute ]
NODEGRP    NODETYPE      NODE        STATUS  NODE        STATUS
SMMGRP     SMM           0x00000001  (S)     0x00000002  (A)
MASTERGRP  CCM           0xFFFF0001  (S)     0xFFFF0002  (A)
0x01       CCS           0x01FF0001  (S)     0x01FF0002  (A)
[ CLST: storage ]
NODEGRP    NODETYPE      NODE        STATUS  NODE        STATUS
SMMGRP     SMM           0x00000001  (A)     0x00000002  (S)
MASTERGRP  SCM           0xFFFF0001  (A)     0xFFFF0002  (S)
FILEGRP    MGS           0xFFFF0003  (-)     0xFFFF0004  (-)
FILEGRP    MGS           0xFFFF0005  (-)     0xFFFF0006  (-)
FILEGRP    MDS           0xFFFF0007  (-)     0xFFFF0008  (-)
FILEGRP    MDS           0xFFFF0009  (-)     0xFFFF000A  (-)
FILEGRP    OSS           0xFFFF000B  (-)     0xFFFF000C  (-)
FILEGRP    OSS           0xFFFF000D  (-)     0xFFFF000E  (-)
```

This does not display the status of the active/standby system for the file server nodes.

To check the status of the active/standby system (consolidation) for the file server nodes, check the status of the FEFSSR service on the SRV_STATUS field. This field appears by specifying the -v option in the pashowclst command and executing the command.

3.3.2.6 Display of the Number of Nodes in Maintenance Work

You can check the number of nodes in the maintenance mode when performing a software maintenance or a hardware maintenance.

To check the number of nodes in the hardware maintenance mode, specify the --hard-mainte option.
The following is an example of executing the pashowclst command by specifying a boot group whose boot group ID is 0x0101.

```
# pashowclst -c compute --hard-mainte --bootgrp 0x0101
[ CLST: compute ]
[ BOOTGRP: 0x0101 ]
BOOTGRP   COMPLETE  TOTAL
0x0101    16        16
```

CLST

Cluster name

BOOTGRP

Boot group ID

COMPLETE

The number of nodes whose power status becomes "off" by transitioning to the hardware maintenance mode

TOTAL

The total number of nodes in a specified range

To check the number of nodes in the software maintenance mode, specify the --soft-mainte option.
The following is an example of executing the pashowclst command by specifying a boot group whose boot group ID is 0x0101.

```
# pashowclst -c compute --soft-mainte --bootgrp 0x0101
[ CLST: compute ]
[ BOOTGRP: 0x0101 ]
BOOTGRP           SRV_RUN  SRV_STOP  TOTAL
0x0101            0        16        16
```

CLST

Cluster name

BOOTGRP

Boot group ID

SRV_RUN

The number of nodes where a service is operating

SRV_STOP

The number of nodes where all services stop

TOTAL

The total number of nodes in a specified range

3.3.3 Displaying System Configuration Information

You can check the system configuration information by executing the pashowclst command on the system management node.

It can display the following pieces of information:

- Summary display of the configuration information of the system

  This displays the summary of the configuration information of the entire system.

- Detailed display of the configuration information of the system

  This displays detailed information on the configuration information of the entire system.

- Displaying all information of the system

  This displays detailed information on the configuration information and the operation status of the entire system.

3.3.3.1 Summary Display of the System Configuration Information

Display the summary of the node configuration information of the entire system.
This enables you to check the information on the node type or the grouping of a node that configures the system.
Use the -l option of the pashowclst command to check the summary of the configuration information of the system.

When specifying the -l option, you can also specify another option to specify the display range. For details, see "pashowclst command" in the "Job Operation Software Command Reference."

```
# pashowclst -l
CLUSTER             GRP_NUM           NODE_NUM
NAME     CLSTTYPE  NODEGRP  BOOTGRP  TOTAL   SMM CCM  LN  CCS   BIO  SIO    GIO   CN  SCM  MGS  MDS
OSS
compute   COMPUTE  1        3       55      2    2   1   2     3    3      3    48   0    0    0    0
storage   STORAGE   -        -        8       2    0    0   0     0     0      0    0    2    2
2    2
```

CLUSTER

Information on a cluster to be displayed

NAME: Cluster name
CLSTTYPE: Cluster type
(COMPUTE: Compute cluster, STORAGE: Storage cluster, and MULTIUSE: Multiuse cluster)

GRP_NUM

Information on a group structure

NODEGRP: The number of node groups
BOOTGRP: The number of boot groups

Note

Node groups and boot groups do not exist in the storage cluster and the multiuse cluster. A hyphen (-) appears instead.

NODE_NUM

Information on the number of nodes

TOTAL: The total number of nodes inside a cluster
SMM to OSS: The number of nodes for each node type

When a multiuse node is defined, this appears for each type after the OSS.

Information

If a node has a configuration serving for multiple node types, the node is counted as the number of SMM to OSS nodes for each of these node types. Therefore, the total number shown in TOTAL may not match the number of SMM to OSS nodes.

3.3.3.2 Detailed Display of the Configuration Information of the System

Display the details on the node configuration information of the entire system.
To display the detailed information, specify both the -l option and the -v option at the same time in the pashowclst command.
The displayed items vary depending on node types.

```
# pashowclst -v -l
[ CLST: compute ]
NAME       CLSTTYPE  USEDBY
compute    COMPUTE   -
[ NODETYPE: SMM ]
NODE        MODEL  MNG_NET_MASTER  MNG_NET     CTRL_NET_SRV  CTRL_NET     HOSTNAME
0x00000001  PG     192.0.2.20      192.0.2.21  192.0.20.21   192.0.20.24  smm1
0x00000002  PG     192.0.2.20      192.0.2.22  192.0.20.22   192.0.20.25  smm2
[ NODETYPE: CCM ]
NODE        MODEL  MNG_NET_MASTER  MNG_NET     CTRL_NET     HOSTNAME
0xFFFF0001  PG     192.0.2.30      192.0.2.31  192.0.20.34  ccm1
0xFFFF0002  PG     192.0.2.30      192.0.2.32  192.0.20.35  ccm2
[ NODETYPE: LN ]
NODE        MODEL  MNG_NET          CTRL_NET     HOSTNAME
0xFFFF0003  PG     192.0.2.51       192.0.20.54  login1
[ NODETYPE: CCS ]
NODE        MODEL  MNG_NET_MASTER  MNG_NET     CTRL_NET      HOSTNAME
0x01FF0001  PG     192.0.2.70      192.0.2.71  192.0.20.114  ccs1
0x01FF0002  PG     192.0.2.70      192.0.2.72  192.0.20.115  ccs2
[ NODETYPE: BIO ]
NODE        MODEL  MNG_NET       CTRL_NET            COMPUTE_NET   COORDINATE   HOSTNAME
0x01010001  FT     192.0.2.73    192.0.20.1(4)       198.51.100.5  0,0,2,0,0,0  bio01
[ NODETYPE: SIO ]
NODE        MODEL  MNG_NET       CTRL_NET            COMPUTE_NET   COORDINATE   HOSTNAME
0x01010002  FT     -             192.0.20.1(10)      198.51.100.11 0,0,1,1,0,0  sio01
[ NODETYPE: GIO ]
NODE        MODEL  MNG_NET       CTRL_NET            COMPUTE_NET   COORDINATE   HOSTNAME
0x01010003  FT     -             192.0.20.1(2)       198.51.100.3  0,0,1,0,0,0  gio01
[ NODETYPE: CN ]
NODE        MODEL  MNG_NET       CTRL_NET            COMPUTE_NET   COORDINATE   HOSTNAME
0x01010001  FT     192.0.2.73    192.0.20.1(4)       198.51.100.5  0,0,2,0,0,0  bio01
0x01010002  FT     -             192.0.20.1(10)      198.51.100.11 0,0,1,1,0,0  sio01
0x01010003  FT     -             192.0.20.1(2)       198.51.100.3  0,0,1,0,0,0  gio01
0x01010004  FT     -             192.0.20.1(0)       198.51.100.1  0,0,0,0,0,0  01-cn0
0x01010005  FT     -             192.0.20.1(6)       198.51.100.7  0,0,3,0,0,0  01-cn4
...
[ CLST: storage ]
```

```
NAME       CLSTTYPE  USEDBY
storage    STORAGE   compute
[ NODETYPE: SMM ]
NODE        MODEL  MNG_NET_MASTER  MNG_NET     CTRL_NET_SRV  CTRL_NET     HOSTNAME
0x00000001  PG     192.0.2.20      192.0.2.21  192.0.20.21   192.0.20.24  smm1
0x00000002  PG     192.0.2.20      192.0.2.22  192.0.20.22   192.0.20.25  smm2
[ NODETYPE: SCM ]
NODE        MODEL  MNG_NET_MASTER  MNG_NET     CTRL_NET     HOSTNAME
0xFFFF0001  PG     192.0.2.59      192.0.2.60  192.0.20.60  scm1
0xFFFF0002  PG     192.0.2.59      192.0.2.61  192.0.20.61  scm2
[ NODETYPE: MGS ]
NODE        MODEL  MNG_NET     CTRL_NET     FILEGRP_ID  HOSTNAME
0xFFFF0003  PG     192.0.2.62  192.0.20.62  1           mgs1
0xFFFF0004  PG     192.0.2.63  192.0.20.63  1           mgs2
[ NODETYPE: MDS ]
NODE        MODEL  MNG_NET     CTRL_NET     FILEGRP_ID  HOSTNAME
0xFFFF0005  PG     192.0.2.64  192.0.20.64  2           mds1
0xFFFF0006  PG     192.0.2.65  192.0.20.65  2           mds2
[ NODETYPE: OSS ]
NODE        MODEL  MNG_NET     CTRL_NET     FILEGRP_ID  HOSTNAME
0xFFFF0007  PG     192.0.2.66  192.0.20.66  3           oss1
0xFFFF0008  PG     192.0.2.67  192.0.20.67  3           oss2
```

NAME

Cluster name

CLSTTYPE

Cluster type

USEDBY

This displays the name of a compute cluster using a storage cluster if the cluster type is the storage cluster.

NODETYPE

Node type

NODE

Node ID

MODEL

Model name

MNG_NET_MASTER

Representative IP address of the management network

MNG_NET

IP address of the management network

CTRL_NET_SRV

Server IP address of the control network (IP address to access the BMC of each node)

CTRL_NET

IP address of the control network
Model name "FT": IP address of BMC (Node number)
Model name "PG": IP address of BMC

HOSTNAME

Host name

COMPUTE_NET

Tofu communication IP address for calculation

COORDINATE

Tofu coordinates (X,Y,Z,a,b,c)

FILEGRP_ID

File group ID

3.3.3.3 Displaying All Information of the System

Display the detailed information on the configurations and operations of all nodes inside the system, and the mount configuration information on the shared disk inside the system. To display such information, specify the -a option in the pashowclst command to execute it.

```
# pashowclst -a
[ HA ]
CLUSTER          NODETYPE      NODE
compute          SMM           0x00000001,0x00000002
    DEVICE            /dev/disk/by-id/xxxxxxxxxxxxxxxxxxxxxx
    MOUNT_POINT       /var/opt/FJSVtcs/shared_disk
    FS_TYPE           xfs
    OPT_LIST          rw
CLUSTER          NODETYPE       NODE
compute          CCM            0xFFFF0001,0xFFFF0002
    DEVICE            /dev/disk/by-id/xxxxxxxxxxxxxxxxxxxxxx
    MOUNT_POINT       /var/opt/FJSVtcs/shared_disk
    FS_TYPE           ext4
    OPT_LIST          rw
[ CLST: compute ]
NAME      CLSTTYPE  USEDBY
compute   COMPUTE   -
[ NODETYPE: SMM ]
NODE        STATUS      REASON          PWR_STATUS      ARCH_STATUS      SRV_STATUS
            MODEL       MNG_NET_MASTER  MNG_NET         CTRL_NET_SRV     CTRL_NET     HOSTNAME
            RELAY_NODE  RELAY_MNG_NET   RELAY_HOSTNAME  FROM_UPDATE      DETAIL
0x00000001  Running     -               on              -                PWRD(o)
            PG          192.0.2.20      192.0.2.21      192.0.20.21      192.2.20.24       smm1
            -           -               -               12:11:34         -
0x00000002  Running     -               on              -                PWRD(o)
            PG          192.0.2.20      192.0.2.22      192.0.20.22      192.2.20.25       smm2
            -           -                -              12:11:34         -
[ NODETYPE: CCM ]
NODE        STATUS      REASON          PWR_STATUS       ARCH_STATUS  SRV_STATUS
            MODEL       MNG_NET_MASTER  MNG_NET          CTRL_NET     HOSTNAME
            RELAY_NODE  RELAY_MNG_NET   RELAY_HOSTNAME   FROM_UPDATE  DETAIL
0xFFFF0001  Running     -               on               -
PJM(o),PLE(o),MRD(o),SRD(o),FEFS(o),PWRD(o)
            PG          192.0.2.30      192.0.2.31       192.0.20.34  ccm1
            -           -               -                12:11:34     -
0xFFFF0002  Running     -               on               -
PJM(o),PLE(o),MRD(o),SRD(o),FEFS(o),PWRD(o)
            PG          192.0.2.30      192.0.2.32       192.0.20.35  ccm2
            -           -               -                12:11:34     -
[ NODETYPE: LN ]
NODE        STATUS      REASON         PWR_STATUS       ARCH_STATUS  SRV_STATUS
            MODEL       MNG_NET        CTRL_NET         HOSTNAME
            RELAY_NODE  RELAY_MNG_NET  RELAY_HOSTNAME   FROM_UPDATE  DETAIL
0xFFFF0003  Running     -              on               -            PLE(o),FEFS(o)
            PG          192.0.2.51     192.0.20.54      login1
            -           -              -                12:11:34     -
[ NODETYPE: CCS ]
NODE        STATUS      REASON          PWR_STATUS        ARCH_STATUS   SRV_STATUS
            MODEL       MNG_NET_MASTER  MNG_NET           CTRL_NET      HOSTNAME
            RELAY_NODE  RELAY_MNG_NET   RELAY_HOSTNAME    FROM_UPDATE   DETAIL
```

```
0x01FF0001  Running     -               on                -             PLE(o),SRD(o)
            PG          192.0.2.70      192.0.2.71        192.0.20.114  ccs1
            -           -               -                 12:11:34      -
0x01FF0002  Running     -               on                -             PLE(o),SRD(o)
            PG          192.0.2.70      192.0.2.72        192.0.20.115  ccs2
            -           -               -                 12:11:34      -
[ NODETYPE: BIO ]
NODE        STATUS      REASON         PWR_STATUS       ARCH_STATUS      SRV_STATUS
            MODEL       MNG_NET        CTRL_NET         COMPUTE_NET      COORDINATE     HOSTNAME
            RELAY_NODE  RELAY_MNG_NET  RELAY_HOSTNAME   FROM_UPDATE      DETAIL
0x01010001  Running     -              os-running       ICC_Running
PLE(o),SRD(o),NRD(o),FEFS(o),PWRD(o)
            FT          192.0.2.73     192.0.20.1(4)    198.51.100.5     0,0,2,0,0,0    bio01
            -           -              -                12:11:34         -
[ NODETYPE: SIO ]
NODE        STATUS      REASON         PWR_STATUS       ARCH_STATUS      SRV_STATUS
            MODEL       MNG_NET        CTRL_NET         COMPUTE_NET      COORDINATE     HOSTNAME
            RELAY_NODE  RELAY_MNG_NET  RELAY_HOSTNAME   FROM_UPDATE      DETAIL
0x01010002  Running     -              os-running       ICC_Running
PLE(o),NRD(o),FEFSSR(o),FEFS(o),PWRD(o)
            FT          -              192.0.20.1(10)   198.51.100.11    0,0,1,1,0,0    sio01
            0x01010001  192.0.2.73     bio01            12:11:34         -
[ NODETYPE: GIO ]
NODE        STATUS      REASON         PWR_STATUS       ARCH_STATUS      SRV_STATUS
            MODEL       MNG_NET        CTRL_NET         COMPUTE_NET      COORDINATE     HOSTNAME
            RELAY_NODE  RELAY_MNG_NET  RELAY_HOSTNAME   FROM_UPDATE      DETAIL
0x01010003  Running     -              os-running       ICC_Running
PLE(o),NRD(o),FEFSSR(o),FEFS(o),PWRD(o)
            FT          -              192.0.20.1(2)    198.51.100.3     0,0,1,0,0,0    gio01
            0x01010001  192.0.2.73     bio01            12:11:34         -
[ NODETYPE: CN ]
NODE        STATUS      REASON         PWR_STATUS       ARCH_STATUS      SRV_STATUS
            MODEL       MNG_NET        CTRL_NET         COMPUTE_NET      COORDINATE     HOSTNAME
            RELAY_NODE  RELAY_MNG_NET  RELAY_HOSTNAME   FROM_UPDATE      DETAIL
0x01010004  Running     -             os-running       ICC_Running      PLE(o),NRD(o),FEFS(o),PWRD(o)
            FT          -              192.0.20.1(0)    198.51.100.5     0,0,0,0,0,0     01-cn0
            0x01010001  192.0.2.73     bio01            12:11:34         -
...
[ NODETYPE: DISK ]
NODE        STATUS      REASON         PWR_STATUS      ARCH_STATUS  SRV_STATUS
            MODEL       MNG_NET        CTRL_NET        FILEGRP_ID   HOSTNAME
            RELAY_NODE  RELAY_MNG_NET  RELAY_HOSTNAME  PWRCTL_ONLY  FROM_UPDATE      DETAIL
0xFFFF0005  Running     -              on              -            -
            ST          192.0.2.64     192.0.20.64     2            disk1
            -           -              -               yes          12:11:34         -
[ CLST: storage ]
NAME     CLSTTYPE  USEDBY
storage    STORAGE   -
[ NODETYPE: SMM ]
NODE        STATUS      REASON          PWR_STATUS      ARCH_STATUS      SRV_STATUS
            MODEL       MNG_NET_MASTER  MNG_NET         CTRL_NET_SRV     CTRL_NET          HOSTNAME
            RELAY_NODE  RELAY_MNG_NET   RELAY_HOSTNAME  FROM_UPDATE      DETAIL
0x00000001  Running     -               on              -                PWRD(o)
            PG          192.0.2.20      192.0.2.21      192.0.20.21      192.2.20.24       smm1
            -           -               -               12:11:34         -
0x00000002  Running     -               on              -                PWRD(o)
            PG          192.0.2.20      192.0.2.22      192.0.20.22      192.2.20.25       smm2
            -           -               -               12:11:34         -
[ NODETYPE: SCM ]
NODE        STATUS      REASON          PWR_STATUS      ARCH_STATUS      SRV_STATUS
            MODEL       MNG_NET_MASTER  MNG_NET         CTRL_NET         HOSTNAME
            RELAY_NODE  RELAY_MNG_NET   RELAY_HOSTNAME  FROM_UPDATE      DETAIL
0xFFFF0001  Running     -               on              -                IC(o)
```

```
            PG          192.0.2.59      192.0.2.60      192.0.20.60      scm1
            -           -               -               12:11:34         -
0xFFFF0002  Running     -               on              -                IC(o)
            PG          192.0.2.59      192.0.2.61      192.0.20.61      scm2
            -           -                     -         12:11:34         -
[ NODETYPE: MGS ]
NODE        STATUS      REASON          PWR_STATUS      ARCH_STATUS      SRV_STATUS
MODEL       MNG_NET     CTRL_NET        FILEGRP_ID      HOSTNAME
            RELAY_NODE  RELAY_MNG_NET   RELAY_HOSTNAME  FROM_UPDATE      DETAIL
0xFFFF0003  Running     -               on              -                FEFSSR(o)
            PG          192.0.2.62      192.0.20.62     1                mgs1
            -           -               -               12:11:34         -
0xFFFF0004  Running     -               on              -                FEFSSR(o)
            PG          192.0.2.63      192.0.20.63     1                mgs2
            -           -               -               12:11:34         -
[ NODETYPE: MDS ]
NODE        STATUS      REASON          PWR_STATUS      ARCH_STATUS      SRV_STATUS
MODEL       MNG_NET     CTRL_NET        FILEGRP_ID      HOSTNAME
            RELAY_NODE  RELAY_MNG_NET   RELAY_HOSTNAME  FROM_UPDATE      DETAIL
0xFFFF0005  Running     -               on              -                FEFSSR(o)
            PG          192.0.2.64      192.0.20.64     2                mds1
            -           -               -               12:11:34         -
0xFFFF0006  Running     -               on              -                FEFSSR(o)
            PG          192.0.2.65      192.0.20.65     2                mds2
            -           -               -               12:11:34         -
[ NODETYPE: OSS ]
NODE        STATUS      REASON          PWR_STATUS      ARCH_STATUS      SRV_STATUS
MODEL       MNG_NET     CTRL_NET        FILEGRP_ID      HOSTNAME
            RELAY_NODE  RELAY_MNG_NET   RELAY_HOSTNAME  FROM_UPDATE      DETAIL
0xFFFF0007  Running     -               on              -                FEFSSR(o)
            PG          192.0.2.66      192.0.20.66     3                oss1
            -           -               -               12:11:34         -
0xFFFF0008  Running     -               on              -                FEFSSR(o)
            PG          192.0.2.67      192.0.20.67     3                oss2
            -           -               -               12:11:34         -
...
```

HA

Mount information on the shared disk (shared_disk)

CLUSTER

Cluster name

NODETYPE

Node type

NODE

Node ID

DEVICE

Name of a device to be mounted
(by-id or UUID and the volume label)

MOUNT_POINT

Mount point of the file system

FS_TYPE

Type of the file system

OPT_LIST

Option specified in the mount command

CLST

System configuration information and operation status

NODETYPE

Node type

NODE

Node ID

STATUS

Node status

REASON

Reason why a node status transitions to the current status

PWR_STATUS

Power status of the node

ARCH_STATUS

Status of Tofu channel

SRV_STATUS

Service name and its operation status

MODEL

Model name

MNG_NET_MASTER

Representative IP address of the management network

MNG_NET

IP address of the management network

CTRL_NET_SRV

Server IP address for the control network

CTRL_NET

IP address of the control network
Model "FT": IP address of BMC (Node number)
Model "PG": IP address of BMC

HOSTNAME

Host name

RELAY_NODE

Node ID of a node serving as the relay to log in to a display object

RELAY_MNG_NET

IP address on the management network for RELAY_NODE

RELAY_HOSTNAME

Host name for RELAY_NODE

PWRCTL_ONLY

Setting whether or not to perform only power control (for the multiuse node only)

COMPUTE_NET

IP address of the network for Tofu communication

COORDINATE

Information on Tofu coordinates (X,Y,Z,a,b,c)

FILEGRP_ID

File group ID

FROM_UPDATE

Elapsed time since last operation status change (hour:minute:second)

DETAIL

Detailed information on abnormality occurrences
"-" means that no detailed information exists.

3.3.4 Configuring the System Monitoring

This section describes the following two procedures for configuring the system monitoring:

- Setting for the system monitoring
- Monitoring setting for the IC service (Mellanox HCA)

3.3.4.1 Setting for the System Monitoring

Describe the setting content in pamonitor.conf, the system monitoring configuration file, and then apply the setting by using the --set option of the pamonitoradm command. In addition, you can refer to the currently set contents from the --show option of the pamonitoradm command.
The following shows the procedure to set the system monitoring. Perform all the following operations on the active system management node.

1. Creating the system monitoring configuration file

   Create pamonitor.conf, the system monitoring configuration file, in order to monitor the system.

   Place pamonitor.conf in the following location on the active system management node.

   ```
   /etc/opt/FJSVtcs/pamonitor.conf
   ```

   Information

   When introducing the Job Operation Software, the pamonitor.conf file does not exist. Copy the sample file (/etc/opt/FJSVtcs/sample/pamonitor.conf), and then place it on the path described above.

   The following is an example of describing the system monitoring configuration file when changing the following times to 600 seconds:
   - The interval for monitoring the OS alive status of a cluster named compute
   - The timeout time until detecting an abnormality on the alive status monitoring for the OS

   For details on the system monitoring configuration file, see "pamonitor.conf file" in the "Job Operation Software Command Reference."

   ```
   Cluster {
       ClusterName = compute
       Monitor {
           NodeInterval = 600
           NodeTimeout = 600
       }
   }
   ```

2. Executing the system monitoring setting command

   Apply the settings by using the --set option of the pamonitoradm command.

```
# pamonitoradm --set
[WARNING]
Do you really want to continue (y/n)y
[INFO] PSM 0082 pamonitoradm The processing of the configuration file was completed.
```

Information

If a node fails in this application due to some reason such as stoppage of the power supply, the node ID of this node is output to the error list file (pamonitoradm_*clstname*_failed_*date*) on the home directory.

```
# pamonitoradm --set
[WARNING]
Do you really want to continue (y/n)y
[WARN] PSM 0051 pamonitoradm There are nodes that failed in the setting.(path/
pamonitoradm_clstname_failed_date)
```

path: Path of the directory to which the file is output
clstname: Cluster name
date: Date and time when executing the command (year, month, day, hour, minute, and second)

The following is an example of the error list file.

```
0xFFFF0008
0xFFFF0009
```

By specifying the error list file in the -f option of the pamonitoradm command, you can reexecute it to a node to which you failed to apply the settings.

```
# pamonitoradm -c clstname -f pamonitoradm_clstname_failed_date --set
Do you really want to continue (y/n)? y
[INFO] PSM 0082 The processing of the configuration file was completed.
```

3. Checking the set contents

   Check the contents that you set.
   The following is an example displayed when making the settings according to the description example of step 1.

```
# pamonitoradm --show
Cluster {
    ClusterName = "compute"
    Monitor {
        NodeInterval = "600"
        NodeTimeout = "600"
        NodeRetry = "10" # default
        ServiceInterval = "60" # default
        ServiceInterval_CN = "60" # default
        ServiceTimeout = "60" # default
        MasterLogLevel = "1" # default
        SubLogLevel = "1" # default
        SlaveLogLevel = "1" # default
    }
    IC {
        DeviceName = "mlx"
        PortNumber = "1,2"
        PeakRate = "100 Gb/sec,56 Gb/sec,40 Gb/sec"
    }
}
```

   For items that are not explicitly described in the pamonitor.conf file, the default settings of the system monitoring function appear. For the default settings, the character string "# default" appears after the setting values.

3.3.4.2 Monitoring Setting for the IC Service (Mellanox HCA)

By changing the monitoring setting for the IC service, you can change the conditions for determining that the IC service is abnormal. When monitoring the IC service, the system monitoring function runs under the condition of abnormality detection, which determines whether the performance value of the Mellanox HCA falls below the standard value.

By default, it determines as follows.

Table 3.38 Standard Value of the Performance Value of the IC Service

Performance Value	Status	Display of pashowclst Command
200 Gb/sec (2X NDR)	Normal	IC(o)
200 Gb/sec (4X HDR)		
100 Gb/sec (2X HDR)		
100 Gb/sec (4X EDR)		
56 Gb/sec (4X FDR)		
40 Gb/sec (4X FDR10)		
40 Gb/sec (4X QDR)		
Other than above	Degraded	IC(!)

If you want to monitor only a specific port, such as a Dual Port Mellanox HCA, you can change the DeviceName setting.
To change the IC service monitoring setting, describe the setting content in pamonitor.conf, the system monitoring configuration file. Then, apply the setting by using the --set option of the pamonitoradm command. In addition, you can refer to the currently set contents from the --show option of the pamonitoradm command.

The following shows the procedure for the monitoring setting of the IC service. Perform all the following operations on the active system management node.

1. Creating the system monitoring configuration file

   Create pamonitor.conf, the system monitoring configuration file, in order to monitor the system.
   Place pamonitor.conf in the following location on the active system management node.

   ```
   /etc/opt/FJSVtcs/pamonitor.conf
   ```

   The following is an example of describing the system monitoring configuration file when monitoring the IC service.

   - Cluster name: compute.
   - Dtermining that any performance value other than EDR is abnormal
   - Monitoring only port 0.

   ```
   Cluster {
       ClusterName = compute
       IC {
           PeakRate = "100 Gb/sec "
           DeviceName = "mlx5_0"
       }
   }
   ```

   For details on the system monitoring configuration file, see "pamonitor.conf file" in the "Job Operation Software Command Reference."

2. Executing the system monitoring setting command

   Apply the settings by using the --set option of the pamonitoradm command.

   ```
   # pamonitoradm --set
   [WARNING]
   ```

```
Do you really want to continue (y/n)y
[INFO] PSM 0082 pamonitoradm The processing of the configuration file was completed.
```

3. Checking the set contents

   Check the contents that you set.
   The following is an example displayed when making the settings according to the description example of step 1.

```
# pamonitoradm --show
Cluster {
    ClusterName = "compute"
    IC {
        PeakRate = "100 Gb/sec"
        DeviceName = "mlx5_0"
        PortNumber = "1,2" # default
    }
    Monitor {
        NodeInterval = "60"
        NodeTimeout = "300"
        NodeRetry = "10"
        ServiceInterval = "60"
        ServiceInterval_CN = "60"
        ServiceTimeout = "60"
        MasterLogLevel = "1"
        SubLogLevel = "1"
        SlaveLogLevel = "1"
    }
}
```

4. Applying the setting of the IC section

   If changing the IC section, restart the monitoring function process of a node that defines the IC service as a monitoring target, applying the settings of the abnormality detection condition of the IC.

   To restart the monitoring function process, use the --soft-mainte option and the --recover option of the paclstmgr command.

   - For the compute node (PRIMERGY server)

     The following is an example of the procedure to apply the setting when the name of the compute cluster is compute and the target node is the compute node.

     a. Use the paclstmgr command to disconnect the compute node from job operations.

```
# paclstmgr -c compute --nodetype CN --model PG -a --disable
```

     b. Use the pashowclst command to confirm that the compute node has been disconnected from job operations.
        Confirm that the STATUS field displays "Disable" and the REASON field displays "Manual."

```
# pashowclst -v -c compute --nodetype CN
[ CLST: compute ]
[ NODETYPE: CN ]
NODE        NODETYPE  STATUS   REASON  PWR_STATUS  ARCH_STATUS  SRV_STATUS
0x01FF0003  CN        Disable  Manual  on          -            (*)
PLE(o),NRD(o),FEFS(o),PWRD(o),IC(o)
0x01FF0004  CN        Disable  Manual  on          -            (*)
PLE(o),NRD(o),FEFS(o),PWRD(o),IC(o)
```

        A new line starts just after the (*) part because of the space limitation of this document, but only one line is actually displayed.

     c. After the job of the compute node completes its execution, use the paclstmgr command to shift the node to the software maintenance mode.

```
# paclstmgr -c compute --nodetype CN --model PG -a --soft-mainte --no-service-stop
Do you really want to continue (y/n)? y
<< It changes to maintenance mode. >>
```

d. Use the pashowclst command to confirm that the node has transitioned to the software maintenance mode.
Confirm that the REASON field displays "SoftMaintenance."

```
# pashowclst -v -c compute --nodetype CN
[ CLST: compute ]
[ NODETYPE: CN ]
NODE        NODETYPE  STATUS   REASON           PWR_STATUS  ARCH_STATUS  SRV_STATUS
0x01FF0003  CN        Disable  SoftMaintenance  on          -            (*)
 PLE(o),NRD(o),FEFS(o),PWRD(o),IC(o)
0x01FF0004  CN        Disable  SoftMaintenance  on          -            (*)
PLE(o),NRD(o),FEFS(o),PWRD(o),IC(o)
```

A new line starts just after the (*) part because of the space limitation of this document, but only one line is actually displayed.

e. After it transitions to the software maintenance mode, use the paclstmgr command to restart the service of the Job Operation Software, and then incorporate it into job operations.

```
# paclstmgr -c cluster1 --nodetype CN --model PG -a --recover --service-restart
<< It restarts service. >>
```

f. Confirm that the compute node has been incorporated into job operations.
Confirm that the STATUS field displays "Running."

```
# pashowclst -v -c compute --nodetype CN
[ CLST: compute ]
[ NODETYPE: CN ]
NODE        NODETYPE  STATUS   REASON  PWR_STATUS  ARCH_STATUS  SRV_STATUS
0x01FF0003  CN        Running  -       on          -            (*)
PLE(o),NRD(o),FEFS(o),PWRD(o),IC(o)
0x01FF0004  CN        Running  -       on          -            (*)
PLE(o),NRD(o),FEFS(o),PWRD(o),IC(o)
```

A new line starts just after the (*) part because of the space limitation of this document, but only one line is actually displayed.

- For the storage cluster management node

The following is an example of the procedure to apply the setting when the name of the storage cluster is storage and the node ID of the target node is 0xFFFF0001.

a. Switch the active/standby system of the node.

When the storage cluster management node is in a redundant configuration, apply the setting from the standby system node first.
If it is necessary to switch the node, execute the following.

```
# paclstmgr -c storage --failover 0xFFFF0001
[INFO] PSM 4084 paclstmgr Failover process starts now. 0xFFFF0001 is going to failover to
0xFFFF0002.
....
[INFO] PSM 4085 paclstmgr Failover process finished normally. 0xFFFF0001 was failovered
to 0xFFFF0002.
```

b. Use the pashowclst command to confirm that the node switching has been successful.
Confirm that the STATUS field of the target node displays "(S)."

```
# pashowclst -c storage --nodetype SCM -m
[ CLST: storage ]
[ NODETYPE: SCM ]
NODEGRP    NODETYPE      NODE        STATUS  NODE        STATUS
MASTERGRP  SCM           0xFFFF0001  (S)     0xFFFF0002  (A)
```

c. Use the paclstmgr command to shift the node to the software maintenance mode.

```
# paclstmgr -c storage -n 0xFFFF0001 --soft-mainte --no-service-stop
Do you really want to continue (y/n)? y
<< It changes to maintenance mode. >>
```

d. Use the pashowclst command to confirm that the node has transitioned to the software maintenance mode.
Confirm that the REASON field displays "SoftMaintenance."

```
# pashowclst -v -c storage --nodetype SCM
[ CLST: storage ]
[ NODETYPE: SCM ]
NODE        NODETYPE  STATUS   REASON           PWR_STATUS  ARCH_STATUS  SRV_STATUS
0xFFFF0001  SCM       Disable  SoftMaintenance  on          -            IC(o)
0xFFFF0002  SCM       Running  -                on          -            IC(o)
```

e. After it transitions to the software maintenance mode, use the paclstmgr command to restart the service of the Job Operation Software, and then incorporate it into job operations.

```
# paclstmgr -c storage -n 0xFFFF0001 --recover --service-restart
<< It restarts service. >>
```

f. Use the pashowclst command to confirm that the node has been incorporated into job operations.
Confirm that the STATUS field displays "Running."

```
# pashowclst -v -c storage --nodetype SCM
[ CLST: storage ]
[ NODETYPE: SCM ]
NODE        NODETYPE  STATUS   REASON           PWR_STATUS  ARCH_STATUS  SRV_STATUS
0xFFFF0001  SCM       Running  -                on          -            IC(o)
0xFFFF0002  SCM       Running  -                on          -            IC(o)
```

When you switched one node, perform the node switching again to restore the status of the active/standby system. Then, apply the settings on the pair node.

3.3.5 Settings of the System Monitoring Plugin

This section describes the following procedures for setting the system monitoring plugin:

- Setting the abnormality plugin
- Setting the automatic dump collection

3.3.5.1 Configuring the System Monitoring Plugin

Describe the setting content in pamoplugin.conf, the system monitoring plugin configuration file, and then apply the setting by using the --set option of the pamopluginadm command. In addition, you can refer to the currently set contents from the --show option of the pamopluginadm command.
The following shows the procedure to set the system monitoring plugin. Perform all the following operations on the active system management node.

1. Creating the system monitoring plugin configuration file

   Create pamoplugin.conf, the system monitoring plugin configuration file, in order to set the system monitoring plugin.
   Place pamoplugin.conf in the following location on the active system management node.

```
/etc/opt/FJSVtcs/pamoplugin.conf
```

Information

When introducing the Job Operation Software, the pamoplugin.conf file does not exist. Copy the sample file (/etc/opt/FJSVtcs/sample/pamoplugin.conf), and then place it on the path described above.

The following is an example of describing the system monitoring plugin configuration file when executing "/work/test_CN.sh > /dev/null". This is executed if the monitoring of the OS alive status detects an abnormality in the compute node (except for nodes serving multiple roles) of a cluster named compute.
For details on the system monitoring plugin configuration file, see "pamoplugin.conf file" in the "Job Operation Software Command Reference."

```
Cluster {
    ClusterName = compute
    PluginCmd {
        NodeType = CN
        ExcludeType = BIO,GIO,SIO
        ServiceName = OS
        Status = x
        Cmd = "/work/test_CN.sh > /dev/null"
    }
}
```

2. Executing the system monitoring plugin setting command

   Apply the settings by using the --set option of the pamonitoradm command.

```
# pamopluginadm --set
[WARNING]
Do you really want to continue (y/n)y
[INFO] PSM 0082 pamopluginadm The processing of the configuration file was completed.
```

Information

If a node fails in this application due to some reason such as stoppage of the power supply, the node ID of this node is output to the error list file (pamopluginadm_*clstname*_failed_*date*) on the home directory.

```
# pamopluginadm --set
[WARNING]
Do you really want to continue (y/n)y
[WARN] PSM 0051 pamopluginadm There are nodes that failed in the setting.(path/
pamopluginadm_clstname_failed_date)
```

path: Path of the directory to which the file is output
clstname: Cluster name
date: Date and time when executing the command (year, month, day, hour, minute, and second)

The following is an example of the error list file.

```
0x00000002
```

By specifying the error list file in the -f option of the pamopluginadm command, you can reexecute it to a node to which you failed to apply the settings.

```
# pamopluginadm -c clstname -f pamopluginadm_clstname_failed_date --set
Do you really want to continue (y/n)? y
[INFO] PSM 0082 The processing of the configuration file was completed.
```

3. Checking the set contents

   Check the contents that you set.
   The following is an example displayed when making the settings according to the description example of step 1.

```
# pamopluginadm --show
Cluster {
    ClusterName = "compute"
    PluginCmd {
        NodeType = "CN"
        ExcludeType = "BIO,GIO,SIO"
        ServiceName = "OS"
        Status = "x"
        Cmd = "/work/test_CN.sh > /dev/null"
    }
    Concurrency = "1024" # default
    AutoNodeOff = "on" # default
    Dump {
        AutoDump = "on" # default
        AutoForceOff = "off" # default
        WaitTime = "600" # default
        NodeType = "SMM,CCM,CCS,BIO,GIO,SIO,CN,LN,SCM,OSS,MDS,MGS" # default
    }
}
```

For items that are not explicitly described in pamoplugin.conf, the default settings of the system monitoring function appear. For the default settings, the character string "# default" appears after the setting values.

3.3.5.2 Configuring the Automatic Dump Collection

The system monitoring function automatically attempts to collect a dump when detecting an abnormality in the OS alive status.
You can enable or disable the setting for this automatic dump collection.

Describe the setting content in pamoplugin.conf, the system monitoring plugin configuration file, and then apply the setting by using the --set option of the pamopluginadm command. In addition, you can refer to the currently set contents from the --show option of the pamopluginadm command.

The following shows the procedure to set the system monitoring plugin. Perform all the following operations on the active system management node.

1. Creating the system monitoring plugin configuration file

   Create pamoplugin.conf, the system monitoring plugin configuration file, in order to set the system monitoring plugin.
   Place pamoplugin.conf in the following location on the active system management node.

   ```
   /etc/opt/FJSVtcs/pamoplugin.conf
   ```

   The following is an example of describing the system monitoring plugin configuration file when disabling the automatic dump collection for all nodes of a cluster named compute.

   For details on the system monitoring plugin configuration file, see "pamoplugin.conf file" in the "Job Operation Software Command Reference."

   ```
   Cluster {
       ClusterName = compute
       Dump {
           AutoDump = off
           AutoForceOff = off
       }
   }
   ```

2. Executing the system monitoring plugin setting command

   Apply the settings by using the --set option of the pamonitoradm command.

   ```
   # pamopluginadm --set
   [WARNING]
   Do you really want to continue (y/n)y
   [INFO] PSM 0082 pamopluginadm The processing of the configuration file was completed.
   ```

3. Checking the set contents

   Check the contents that you set.
   The following is an example displayed when making the settings according to the description example of step 1.

   ```
   # pamopluginadm --show
   Cluster {
   ```

```
    ClusterName = "compute"
    Dump {
        AutoDump = "off"
        AutoForceOff = "off"
        WaitTime = "600" # default
        NodeType = "SMM,CCM,CCS,BIO,GIO,SIO,CN,LN,SCM,OSS,MDS,MGS" # default
    }
    Concurrency = "1024" # default
    AutoNodeOff = "on" # default
}
```

3.3.6 Settings of the User-Defined Service

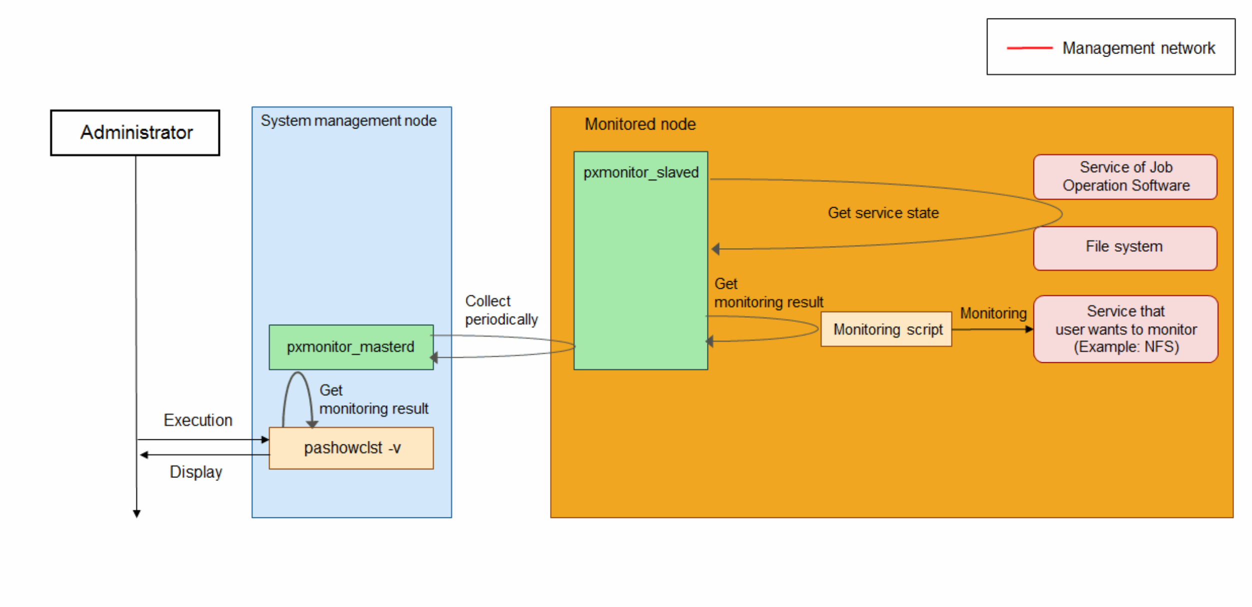
The following sections describe the mechanism of the user-defined service and its setting procedure respectively.

3.3.6.1 The Mechanism of the User-Defined Service

To monitor the user-defined service, place the "monitoring script" created by the user in a target node, and then register it in the system management function.
The system monitoring function collects the execution result of this monitoring script and uses the pashowclst command to display the status of a service defined by a user.

The following diagram shows an example of the operation when monitoring a service defined by a user.

Figure 3.5 The Mechanism of Monitoring the User-Defined Service



Regarding monitoring of the user-defined service, its time interval and timeout are subject to the value set in the system monitoring function. If the monitoring script times out, the service status becomes "x," exiting the process of the monitoring script.

3.3.6.2 Configuring the User-Defined Service

The system administrator is to prepare the monitoring script for monitoring a service to be monitored, and then distribute it to a target node in advance.
Finally, describe the setting content in paservice.conf, the user-defined service configuration file, and then apply the setting by using the --set option of the paserviceadm command. In addition, you can refer to the currently set contents from the --show option of the paserviceadm command.

The following shows the procedure for setting the user-defined service. Perform all the following operations on the active system management node.

1. Creating the monitoring script

   Create a script to monitor any service. The end status of the script determines what the display is like when using the pashowclst command to refer to the state of the service.

   The following table shows the correspondence between the end status of the script and the state display of the service.

Table 3.39 The End Status of the Script and the State Display of the Service

End Status	State
0	o
1	x
3	!
4	s
5	d
Monitoring script does not exist.	n
Other than above	?

Note

- Do not use 2 for the end status of the monitoring script. This is because 2 is the value that the system monitoring function internally uses. If the end status of the script becomes 2, the corresponding user-defined service does not appear on the pashowclst command.
- You cannot pass an argument to the monitoring script.
- The system management function does not manage the standard output and the standard error output provided when executing the monitoring script. (They are also not output to the log of the system management function.) If necessary, define a process to leave the log in the monitoring script.

Also, the setting of the system monitoring plugin allows you to define the operation when detecting any abnormality on the user-defined service.

See

For details on the procedure for setting the system monitoring plugin, see "3.3.5 Settings of the System Monitoring Plugin."

2. Distributing the monitoring script

   Distribute the created script to those nodes that are to be monitored. Run this operation as the root user.

   The following is an example of creating nfsclient_check.sh, the script to monitor the NFS service, and distributing it to a target node.

   a. Giving execution authority to the monitoring script

      Give execution authority to the monitoring script.

```
# chmod 700 nfsclient_check.sh
```

   b. Distributing the monitoring script

      Place the created monitoring script in any directory of the target node.

```
# pmscatter -c clstname -n nodeid -p nfsclient_check.sh /usr/local/bin/nfsclient_check.sh
```

clstname: Cluster name
nodeid: Target node (The system administrator determines its specification range.)

3. Creating paservice.conf, the user-defined service configuration file

   Create paservice.conf, the system monitoring configuration file, in order to perform the system monitoring.
   Place paservice.conf in the following path on the active system management node.

```
/etc/opt/FJSVtcs/paservice.conf
```

Information

When introducing the Job Operation Software, the paservice.conf file does not exist. Copy the sample file (/etc/opt/FJSVtcs/sample/paservice.conf), and then place it on the path described above.

The following is an example of describing the user-defined service configuration file when monitoring the NFS service of a login node in a cluster named compute. To monitor the NFS service, this example uses the monitoring script which was created and distributed in steps 1 and 2.

For details on the user-defined service configuration file, see "paservice.conf file" in the "Job Operation Software Command Reference."

```
Cluster {
    ClusterName = compute
    Service {
        Name = NFS
        NodeType = LN
        ChkPath = /usr/local/bin/nfsclient_check.sh
    }
}
```

4. Executing the user-defined service setting command

   Make this setting by using the --set option of the paserviceadm command.

```
# paserviceadm --set
[WARNING]
Do you really want to continue (y/n)y
[INFO] PSM 0082 paserviceadm The processing of the configuration file was completed.
```

Information

If a node fails in this application due to some reason such as stoppage of the power supply, the node ID of this node is output to the error list file (paserviceadm_*clstname*_failed_*date*) on the home directory.

```
# paserviceadm --set
[WARNING]
Do you really want to continue (y/n)y
[WARN] PSM 0051 paserviceadm There are nodes that failed in the setting.(path/
paserviceadm_clstname_failed_date)
```

path: Path of the directory to which the file is output
clstname: Cluster name
date: Date and time when executing the command (year, month, day, hour, minute, and second)

The following is an example of the error list file.

```
0xFFFF0008
0xFFFF0009
```

By specifying the error list file in the -f option of the paserviceadm command, you can reexecute it to a node to which you failed to apply the settings.

```
# paserviceadm -c clstname -f pamonitoradm_clstname_failed_date --set
Do you really want to continue (y/n)? y
[INFO] PSM 0082 paserviceadm The processing of the configuration file was completed.
```

5. Checking the set contents

   Check the contents that you set.

   The following is an example displayed when making the settings according to the description example of step 3.

```
# paserviceadm --show
Cluster {
    ClusterName = "compute"
    Service {
        Name = "NFS"
        NodeType = "LN"
        ChkPath = "/usr/local/bin/nfsclient_check.sh"
    }
}
```

6. Displaying the service status

   You can use the pashowclst command to check the state of the service obtained by the monitoring script.
   The following is an example of checking the contents that you set in steps 1 to 5. The SRV_STATUS field displays the user-defined service "NFS."

```
# pashowclst -v --nodetype LN
[ CLST: compute ]
[ NODETYPE: LN ]
NODE        NODETYPE  STATUS   REASON  PWR_STATUS  ARCH_STATUS  SRV_STATUS
0xFFFF0003  LN        Running  -       on          -            PLE(o),FEFS(o),NFS(o)
```

3.3.7 Setting of Abnormality Notifications

The following shows the procedure for setting abnormality notifications.

3.3.7.1 Setting Abnormality Notifications

Set abnormality notifications in either of the following two ways:
- The system administrator performs this setting when detecting an abnormality.
- Before a job operation, you set the pamonitor_notice command to make it run as the operation for an external monitoring mechanism when detecting an abnormality.

The following is an example of executing the pamonitor_notice command when the 0xFFFF0005 node has an OS abnormality. This node belongs to a cluster named compute.

For details on the settings of abnormality notifications, see "pamonitor_notice command" in the "Job Operation Software Command Reference."

```
# pamonitor_notice -c compute -n 0xFFFF0005 --detail "Insufficient memory"
```

To check the abnormality and the detailed information set in the pamonitor_notice command, specify the -v option and the --detail option in the pashowclst command. (The node, to which the abnormality is set, displays "SoftError" on the STATUS field and "NodeDown" on the REASON field.)

The following is an example of using the pashowclst command to display each of normal nodes and a node to which the abnormality is set in the pamonitor_notice command.

```
# pashowclst -c compute --nodetype CN -v --detail
[ CLST: compute ]
[ NODETYPE: CN ]
NODE       NODETYPE  STATUS     REASON   PWR_STATUS  ARCH_STATUS  SRV_STATUS                    DETAIL
0xFFFF0004  CN        Running    -         on          -            PLE(o),NRD(o),FEFS(o),PWRD(o)   -
0xFFFF0005  CN        SoftError  NodeDown  on          -            -
Insufficient memory
0xFFFF0006  CN        Running    -         on          -            PLE(o),NRD(o),FEFS(o),PWRD(o)   -
0xFFFF0007  CN        Running    -         on          -            PLE(o),NRD(o),FEFS(o),PWRD(o)   -
...
```

Note

- In the -c and -n options, specify a node which is in the same cluster as the command execution node.
- A failover or automatic dump collection runs for a node of which an OS abnormality occurrence is notified. If necessary, inhibit the failover function or the automatic dump collection function.
- If executing the pamonitor_notice command when the DETAIL field of the pashowclst command has already set detailed information, the detailed information is not updated.
- The detailed information is cleared in either of the following cases:
  - At the timing when a corresponding node is incorporated into job operations again (meaning when the STATUS field of the pashowclst command transitions to Running)
  - At the timing when restarting the system management node on the active system and the standby system

3.4 Details of the System Maintenance Function

3.4.1 Disconnecting and Incorporating a Node

To disconnect or incorporate a node, the system administrator is to use the paclstmgr command on the active system management node.

- Disconnecting a node

  The following is an example of disconnecting the compute node (Node ID: 0xFFFF0010) inside the cluster cluster1 from operation.

```
# paclstmgr --disable -c cluster1 -n 0xFFFF0010
```

Use the pashowclst command to check the disconnected node.
Confirm that the STATUS field of the target node displays "Disable" and its REASON field displays "Manual."

```
# pashowclst -c cluster1 -n 0xFFFF0010
[ CLST: cluster1 ]
[ NODE: 0xFFFF0010 ]
NODE        NODETYPE     STATUS       REASON          PWR_STATUS    ARCH_STATUS   SRV_STATUS
0xFFFF0010  CN           Disable      Manual          on            -
PLE(o),NRD(o),FEFS(o),PWRD(o)
```

To perform a hardware maintenance for FX server, disconnect a target node from the Tofu interconnect D mounted on the target node. To do so, specify the -ic option in the paclstmgr command. The following is an example of disconnecting the boot group (Boot group ID: 0x0101) inside the cluster cluster1 from the mounted Tofu interconnect D.

```
# paclstmgr --disable --ic -c cluster1 --bootgrp 0x0101
```

Use the pashowclst command to check the disconnected node.
Confirm that the STATUS field of the target node displays "Disable," its REASON field displays "Manual," and its ARCH_STATUS field displays "ICC_Disable."

```
# pashowclst -v -c cluster1 --bootgrp 0x0101
[ CLST: cluster1 ]
[ BOOTGRP: 0x0101 ]
NODE        NODETYPE   STATUS     REASON     PWR_STATUS    ARCH_STATUS   SRV_STATUS
```

```
0x01010001  BIO,CN     Disable    Manual     os-running    ICC_Disable
PLE(o),SRD(o),NRD(o),FEFS(o),PWRD(o)
0x01010002  SIO,CN     Disable    Manual     os-running    ICC_Disable
PLE(o),NRD(o),FEFSSR(o),FEFS(o),PWRD(o)
0x01010003  GIO,CN     Disable    Manual     os-running    ICC_Disable
PLE(o),NRD(o),FEFSSR(o),FEFS(o),PWRD(o)
0x01010004  CN       Disable   Manual    os-running   ICC_Disable  PLE(o),NRD(o),FEFS(o),PWRD(o)
0x01010005  CN       Disable   Manual    os-running   ICC_Disable  PLE(o),NRD(o),FEFS(o),PWRD(o)
0x01010006  CN       Disable   Manual    os-running   ICC_Disable  PLE(o),NRD(o),FEFS(o),PWRD(o)
0x01010007  CN       Disable   Manual    os-running   ICC_Disable  PLE(o),NRD(o),FEFS(o),PWRD(o)
0x01010008  CN       Disable   Manual    os-running   ICC_Disable  PLE(o),NRD(o),FEFS(o),PWRD(o)
0x01010009  CN       Disable   Manual    os-running   ICC_Disable  PLE(o),NRD(o),FEFS(o),PWRD(o)
0x0101000A  CN       Disable   Manual    os-running   ICC_Disable  PLE(o),NRD(o),FEFS(o),PWRD(o)
0x0101000B  CN       Disable   Manual    os-running   ICC_Disable  PLE(o),NRD(o),FEFS(o),PWRD(o)
0x0101000C  CN       Disable   Manual    os-running   ICC_Disable  PLE(o),NRD(o),FEFS(o),PWRD(o)
0x0101000D  CN       Disable   Manual    os-running   ICC_Disable  PLE(o),NRD(o),FEFS(o),PWRD(o)
0x0101000E  CN       Disable   Manual    os-running   ICC_Disable  PLE(o),NRD(o),FEFS(o),PWRD(o)
0x0101000F  CN       Disable   Manual    os-running   ICC_Disable  PLE(o),NRD(o),FEFS(o),PWRD(o)
0x01010010  CN       Disable   Manual    os-running   ICC_Disable  PLE(o),NRD(o),FEFS(o),PWRD(o)
```

- Incorporating a node

  The following is an example of incorporating the compute node (Node ID: 0xFFFF0010) inside the cluster cluster1 into operation.

```
# paclstmgr --enable -c cluster1 -n 0xFFFF0010
```

  Use the pashowclst command to check the incorporated node.
  Confirm that the STATUS field of the target node displays "Running" and its REASON field displays "-."

```
# pashowclst -c cluster1 -n 0xFFFF0010
[ CLST: cluster1 ]
[ NODE: 0xFFFF0010 ]
NODE        NODETYPE     STATUS       REASON          PWR_STATUS    ARCH_STATUS   SRV_STATUS
0xFFFF0010  CN           Running      -               on            -
PLE(o),NRD(o),FEFS(o),PWRD(o)
```

3.4.2 Switching a Node

To execute failover manually, the system administrator is to carry out the following operations using the --failover option of the paclstmgr command from the active system management node.
When switching a node, you must keep the following points in mind:

- Representative IP address

  Redundant nodes among the active and standby system nodes contain a common IP address (representative IP address) in the management network. Such redundant nodes always start on the active system node.
  The system management function provides a function to automatically switch the representative IP address of the active and standby system nodes at the time of node switching. On a node that becomes the standby system node at the time of node switching, the representative IP address stops; it starts on a node that becomes the active system node.

- Shared file system (shared_disk)

  The Job Operation Software stores logs and dump files in the directory /var/opt/FJSVtcs/shared_disk on the system management node. This software also stores the management information about job operations in the directory /var/opt/FJSVtcs/shared_disk on the compute cluster management node. This area is called the shared file system (shared_disk).

  To set the system management node or the compute cluster management node as a node in a redundant configuration, you must enable only the respective active system nodes to access the management information described above. Therefore, place the information on a sharable file system, making only the active system nodes mount and use the information. At the time of node switching, you must switch the mount from the old active system node to a new active system node.
  The system management function provides a function to automatically switch the mount of the shared file system at the time of node switching. By setting any necessary mount information for each node in a redundant configuration beforehand, this function automatically mounts it when such node becomes an active system node.

See

To configure the shared file system, use the configuration management function. For details on the configuration of the shared file system, see "3.1.1.4 Configuring the Shared File System for a Redundant Configuration."

- "Mutual standby configuration" and "consolidation"

  A configuration that operates the FEFS functions as the active system in both the redundant configuration nodes is called "mutual standby configuration."
  From the state of the mutual standby configuration, you can set one node as the active system node and the other node as the standby system node. This is called "consolidation."

  The following table shows whether a node becomes the standby/active system configuration or the mutual standby configuration when setting the node in a redundant configuration.
  The node types shown with a hyphen "-" are in a non-redundant configuration.

Table 3.40 Redundant Configuration of a Node

Node Type	Type of Redundant Configuration
System management node	Active/Standby system configuration
Compute cluster management node	
Compute cluster sub management node	
Login node	-
Boot I/O node	-
Storage I/O node	-
Global I/O node	- (*)
Compute node	-
Storage cluster management node	Active/Standby system configuration
MGS node	For details on redundancy (server configuration) of these nodes, see the *FEFS User's Guide*.
MDS node	
OSS node	
Multiuse node	-

(*) The FEFSSR service of the global I/O node uses the round-robin method. If one node inside the GIO group fails, this method enables you to use the function of the other global I/O node.
For details on the round-robin method, see the *FEFS User's Guide*.

For a node in the mutual standby configuration, the system administrator can manually perform consolidation and failback for the FEFS service. This enables him/her to maintain the node without stopping operations in the entire system.

- Failover

```
# paclstmgr -c clstname --failover nodeid1
Do you really want to continue (y/n) ? y
[INFO] PSM 4084 paclstmgr Failover process starts now. nodeid1 is going to failover to nodeid2.
....
[INFO] PSM 4085 paclstmgr Failover process finished normally. nodeid1 was failovered to nodeid2.
```

nodeid1: The node ID of the active system node (The node ID of a node that becomes the standby system node by executing the paclstmgr command)
nodeid2: The node ID of the standby system node (The node ID of a node that becomes the active system node by executing the paclstmgr command)

- Failback for a consolidated FEFS service

```
# paclstmgr -c clstname --failback nodeid1
Do you really want to continue (y/n) ? y
[INFO] PSM 4084 paclstmgr Failover process starts now. nodeid1 is going to failover to nodeid2.
....
[INFO] PSM 4085 paclstmgr Failover process finished normally. nodeid1 was failovered to nodeid2.
```

nodeid1: The node ID of the active system node
nodeid2: The node ID of the standby system node

Information

For the MGS node and the MDS node, you can execute the following command from the system management node to each node in order to check the active/standby system from the output results.

The following is an example of a node, with node ID 0xFFFF0003, as the active system node in a pair node (Node ID: 0xFFFF0003 and 0xFFFF0004) in a redundant configuration.

```
# pmexe --stdout -c clstname -n 0xFFFF0003,0xFFFF0004 "mount -t lustre"
[cmdline]
pmexe --stdout -c clstname -n 0xFFFF0003,0xFFFF0004 mount -t lustre
[cluster]
clstname
<<<<< ResultInformation >>>>>
[0xFFFF0003] /dev/sdc5 on /mnt/FJSVfefs/fefs/mdt type lustre (defaults,retry=6)
[0xFFFF0004]
```

clstname: Cluster name

For details on how to check nodes to be paired, see "3.3.2.5 Display of Active/Standby System Status of a Redundant Configuration Node."

Note

- Do not execute manual failover for the standby system MGS node and the standby system MDS node. Suppose that you wrongly execute failover for the standby system MGS node or the standby system MDS node. Then, the FEFSSR service of the active active system node shows "o" and the FEFSSR service of the standby system node shows "w" by using the pashowclst command to check the operation status of the service. The procedure for recovering such case is as follows.

  Restart the FEFSSR service of the node whose FEFSSR service is "w" (which is 0xFFFF0004 in this example). Then, confirm that the FEFSSR service has turned into "o." Carry out the following operation on the system management node.

  ```
  # pmexe -c clstname -n 0xFFFF0004 --stdout "systemctl stop FJSVfefs"
  # pmexe -c clstname -n 0xFFFF0004 --stdout "systemctl start FJSVfefs"
  ```

  clstname: Cluster name

- If you have logged in to the system management node using the representative IP address, do not perform failover for the system management node. Such failover fails because you cannot switch the representative IP address.

3.4.3 Shifting to the Maintenance Mode and Its Recovery

At the time of a software maintenance and a hardware maintenance, shift the node to the maintenance mode in order to allow you to perform the maintenance work safely.

After the maintenance work is complete, recover the node in the maintenance mode to resume operation.

3.4.3.1 Shifting to the Software Maintenance Mode

When applying a package or changing the configuration of a node, disconnect a target node from job operations. Then, use the --soft-mainte option of the paclstmgr command to shift the node to the software maintenance mode.

Also, depending on the package to be applied, you must restart the service of the Job Operation Software that runs on the node. Therefore, the system administrator is to use the --service-stop or --no-service-stop option of the paclstmgr command to specify whether to stop the service when shifting the node to the software maintenance mode.

See

Regarding the necessity of a restart after applying the package of the Job Operation Software, see "Applying Packages Through Rolling Updates" in the "Job Operation Software Administrator's Guide for Maintenance."

The following is an example of shifting a specific node to the software maintenance mode.
The system administrator is to carry out the following work on the active system management node.

- When not stopping the service of the Job Operation Software

```
# paclstmgr -c clstname -n nodeid --soft-mainte --no-service-stop
Do you really want to continue (y/n)? y                             <- (*)
```

clstname: Cluster name
nodeid: Node ID
(*) This requires your confirmation when executing this command.

- When stopping the service of the Job Operation Software

```
# paclstmgr -c clstname -n nodeid --soft-mainte --service-stop
Do you really want to continue (y/n)? y                             <- (*)
```

clstname: Cluster name
nodeid: Node ID
(*) This requires your confirmation when executing this command.

Note

- When shifting to the software maintenance mode, the FEFS service and the FEFSSR service do not stop. The system administrator is to stop these services by using the fefs_sync command.
  For details on the fefs_sync command, see the *FEFS User's Guide*.

- When shifting to the software maintenance mode, the user-defined service does not stop. At the time of a software maintenance, the system administrator is to stop monitoring the user-defined service accordingly.

- The FX server internally frees the target node's job memory during the transition to software maintenance mode. At the time of a software maintenance,be sure to specify the --soft-mainte option for paclstmgr and confirm that you are in software maintenance mode.
  If you restart with the papwrctl command or the like in the software maintenance mode state, the service starts and the job memory is acquired. If you do not release the software maintenance mode state and want to apply a patch to the kernel and then reboot, and then continue another patch application, execute the paclstmgr command again with the --soft-mainte option after the reboot to free up job memory.

3.4.3.2 Shifting to the Hardware Maintenance Mode

When hardware fails, disconnect the target node from job operations, and then use the --hard-mainte option of the paclstmgr command to shift the node to the hardware maintenance mode.

A node in the hardware maintenance mode inhibits you from using it as a means of job execution or a job channel.

The following is an example of shifting a specific node to the hardware maintenance mode.
The system administrator is to carry out the following work on the active system management node.

```
# paclstmgr -c clstname -n nodeid --hard-mainte
Do you really want to continue (y/n)? y                             <- (*)
```

clstname: Cluster name
nodeid: Node ID
(*) This requires your confirmation when executing this command.

To perform a hardware maintenance for a node of FX server, it is necessary to stop the power supply to the node to be maintained. For this reason, if a node is not to be maintained, the node may be disconnected from operation or the power may not to be supplied to the node, according to influences on job operations.

The following table shows the influence range when performing a hardware maintenance.
When shifting a node shown in "Node to be Maintained" to the hardware maintenance mode, stop the power supply to nodes shown in "Power-Off Range." Also, before shifting a node shown in "Node to be Maintained" to the hardware maintenance mode, you must disconnect nodes shown in "Range Affecting Job Operations" from job operations.

Table 3.41 Influence Range of the Hardware Maintenance Mode

Node to be Maintained	Power-Off Range	Range Affecting Job Operations
Node serving as both compute node and boot I/O node	Boot group	Boot group (*)
Node serving as both compute node and storage I/O node	SIO group	SIO group (*)
Node serving as both compute node and global I/O node	CMU	CMU (*)
Compute node	CMU	CMU (*)

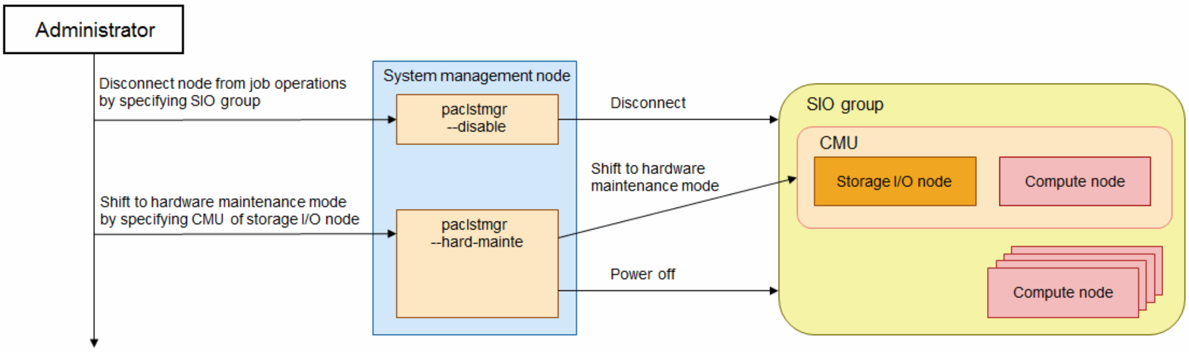
(*) If an active global I/O node does not exist in the GIO group due to stoppage of something shown in the "Power-Off Range," the corresponding one shown in "Range Affecting Job Operations" becomes the GIO group.

See

For details on disconnection of a node, see "3.4.1 Disconnecting and Incorporating a Node."

The following is an example of the influence range when replacing hardware because the storage I/O node fails.
In the above case, a target node belongs to the SIO group, which applies to the "Power-Off Range" and "Range Affecting Job Operations." Therefore, the system administrator is to disconnect the applicable SIO group from job operations, and then shift the CMU mounting the target node to the hardware maintenance mode.

Figure 3.6 Power-Off Range and Range Affecting Job Operations


Administrator
Disconnect node from job operations by specifying SIO group
Shift to hardware maintenance mode by specifying CMU of storage I/O node
System management node
paclstmgr --disable
paclstmgr --hard-mainte
Disconnect
Shift to hardware maintenance mode
Power off
SIO group
CMU
Storage I/O node
Compute node
Compute node

See

Power-off of a node complies with the specifications of the system control function. This makes it possible for CMU or BoB to be powered off according to the specification range when specifying a node of FX server. For details, see "3.2 Details of the System Control Function."

3.4.3.3 Recovering a Node

After completing the maintenance work for a node, recover it from the software maintenance mode and the hardware maintenance mode, and then incorporate it into job operations.
The system administrator is to carry out the following work on the active system management node.

```
# paclstmgr -c clstname -n nodeid --recover
```

clstname: Cluster name
nodeid: Node ID

Note

Regarding nodes of FX server, this operation restricts their amount of memory available to system processes.

To recover a node in the software maintenance mode after applying the modification package, specify an additional option according to the necessity of restarting the applied modification package. This enables you to select a restart method for the node.
For the necessity of restarting for each modification package, see "Applying Packages Through Rolling Updates " in the "Job Operation Software Administrator's Guide for Maintenance."

Table 3.42 Operations for Each Additional Option of the --recover Option

Additional Option	Operation When Specifying It
--no-restart	When specifying a node in the software maintenance mode, incorporate the node into operation without restarting the service of the Job Operation Software. An error occurs if specifying a node in the hardware maintenance mode.
--service-restart	When specifying a node in the software maintenance mode, restart the service of the Job Operation Software, and then incorporate the node into operation. An error occurs if specifying a node in the hardware maintenance mode. If the specified node range includes a node that is not in software maintenance mode, the node is excluded.
--cold-reboot	When specifying a node in the software maintenance mode, execute cold reboot, and then incorporate the node into operation. An error occurs if specifying a node in the hardware maintenance mode. If the specified node range includes a node that is not in software maintenance mode, power down of the node is executed. To start the powered down node, use the papwrctl command.
No specification	When specifying a node in the software maintenance mode, execute cold reboot, and then incorporate the node into operation. When specifying a node in the hardware maintenance mode, power it on, and then incorporate it into operation.

The following is an example of using the cold reboot to recover a node (whose node type is CN and node ID is 0xFFFF0010) in the software maintenance mode.
This is applicable to nodes whose REASON field shows SoftMaintenance when checking it with the pashowclst command.

```
# pashowclst -c cluster1 -n 0xFFFF0010
[ CLST: cluster1 ]
[ NODE: 0xFFFF0010 ]
NODE        NODETYPE  STATUS   REASON            PWR_STATUS   ARCH_STATUS  SRV_STATUS
0xFFFF0010  CN        Disable  SoftMaintenance   on           -            -
```

Specify the --cold-reboot option in the --recover option of the paclstmgr command.

```
# paclstmgr -c cluster1 -n 0xFFFF0010 --recover --cold-reboot
```

Use the pashowclst command to confirm that the STATUS field shows Running.

```
# pashowclst -c cluster1 -n 0xFFFF0010
[ CLST: cluster1 ]
[ NODE: 0xFFFF0010 ]
NODE        NODETYPE  STATUS   REASON       PWR_STATUS   ARCH_STATUS  SRV_STATUS
0xFFFF0010  CN        Running  -            on           -            PLE(o),NRD(o),FEFS(o),PWRD(o)
```

The node recovery is applicable to nodes whose REASON field shows SoftMaintenance, HardMaintenance, and AbnormalEnd when checking their status with the pashowclst command. The following table shows whether a node is recoverable or not for each REASON.

Table 3.43 Whether the --recover Option Can Recover a Node or Not

REASON	Whether Recoverable or Not
-	
HardMaintenance	Recoverable
SoftMaintenance	Recoverable
Manual	
DeadlineSchedule	
AbnormalEnd	Recoverable
ResetRequest	
ReservedAlarm	
RouterFatal	
PortRouterFatal	
PortReservedAlarm	
Alarm	

Recoverable: The --recover option can recover a node.

Note

- If a target of the --recover option contains multiple recoverable statuses, the command results in an error. Specify either of SoftMaintenance, HardMaintenance, or AbnormalEnd in the specification range of the --recover option. You can also specify the --force-all option to recover a node with multiple reasons shown in REASON.

- To execute the --recover option for a node whose REASON is AbnormalEnd, incorporate it into job operations without performing any power supply operation and service operation.

- If using the --recover option to release the maintenance mode from a node, this also releases its inhibitions of automatic failover and automatic dump collection.

- Do not specify the --no-restart sub option for a node in which the software maintenance mode stops the service of the Job Operation Software. Otherwise, the maintenance mode is released while the service is stopping, preventing job operations from resuming on the target node.

- If the FX server is performed hardware maintenance in units of CMU, be sure to recovery in units of CMU. When you perform a recovery in units of BoB or node, the Tofu port does not link up and does not start normally.

3.4.4 Operating the Service of the Job Operation Software

To operate the service of the Job Operation Software during a software maintenance, the system administrator is to use the --service option of the paclstmgr command on the active system management node.

- Stopping the service of the Job Operation Software (by specifying a node ID)

```
# paclstmgr -c clstname -n nodeid --service stop
```

clstname: Cluster name
nodeid: Node ID

- Starting the service of the Job Operation Software (by specifying a node ID)

```
# paclstmgr -c clstname -n nodeid --service start
```

clstname: Cluster name
nodeid: Node ID

- Restarting the service of the Job Operation Software (by specifying a node ID)

```
# paclstmgr -c clstname -n nodeid --service restart
```

clstname: Cluster name
nodeid: Node ID

Note

- If you stop or restart the service of the job operation software, the target node is in the stopped state. Therefore, stop and restart the service of the job operation software running on the compute node while the target node is disconnected from the job operation.
- If you stop or restart the service of the job operation software running on a boot I/O node, the compute node for that boot group is marked down. If you want to stop or restart services on a boot I/O node, disconnect the boot group from job operation, and then perform service operations on a boot group basis.
- If you restart the service in units of cluster or units of node group, stop the service and ensure that pashowclst command reflects the service down state before starting the service.

3.4.5 Setting Automatic Failover

To set the automatic failover, the system administrator is to use the --set-failover option of the paclstmgr command on the active system management node.

Information

By default, the automatic failover is enabled on all redundant configuration nodes.

- Disabling the automatic failover (by specifying a node)

```
# paclstmgr -c clstname -n 0x01FF0001 --set-failover disable
```

- Enabling the automatic failover (by specifying a node)

```
# paclstmgr -c clstname -n 0x01FF0001 --set-failover enable
```

- Checking the setting of the automatic failover (by specifying a node)

```
# paclstmgr -c clstname -n 0x00000001 --set-failover list
[ CLST: clstname ]
NODE         NODETYPE      FAILOVER
```

```
0x00000001  SMM           disable
0x00000002  SMM           disable
```

CLST

Cluster name

NODE

Node ID

NODETYPE

Node type

FAILOVER

Setting of the automatic failover (enable: The setting is enabled. disable: The setting is disabled.)

Note

- If a node has "disable" under "FAILOVER," this does not disable the failover provided by the --failover option and the failback provided by the --failback option.
- The automatic failover is inhibited for nodes in the software maintenance mode and the hardware maintenance mode. Therefore, "FAILOVER" shows "disable" when checking the status with the --set-failover option.
- When specifying the scope option in the --set-failover option and specifying only one side of a redundant configuration, any pair nodes in a redundant configuration also become the target of this setting. (The one side of a redundant configuration means either the active system or the standby system.)

```
# paclstmgr -c clstname -n 0xFFFF0001 --set-failover list
[ CLST: clstname ]
NODE        NODETYPE      FAILOVER
0xFFFF0001  CCM           disable
0xFFFF0002  CCM           disable
```

3.4.6 System Maintenance Function Plugin

The system maintenance function has a plugin function, which describes specific processing. You can execute this plugin before and after using the paclstmgr command to perform a node operation and a service operation.
This function enables you to automate necessary work, making your maintenance work more efficient.

The following is an example of executing the plugin located in any directory when shifting to the software maintenance mode.

- Executing the plugin before shifting to the software maintenance mode

```
# paclstmgr -c clstname -n nodeid --soft-mainte --service-stop --plugin-pre dirpath
```

clstname: Cluster name
nodeid: Node ID
dirpath: The path of a directory that contains the plugin

- Executing the plugin after shifting to the software maintenance mode

```
# paclstmgr -c clstname -n nodeid --soft-mainte --service-stop --plugin-post dirpath
```

clstname: Cluster name
nodeid: Node ID
dirpath: The path of a directory that contains the plugin

The plugin executed by the system maintenance function has the following rules:

- The plugin command that you can specify
    - The content of the plugin command is arbitrary; the result is not guaranteed.
    - Define the file name of the plugin command with up to 63 characters, including single-byte alphanumeric characters (case-sensitive), hyphens ("-"), underscores ("_"), and periods ("."). However, the first character cannot be a hyphen.
    - The plugin command must use the execution format, and the permission must be either 0500 or 0700.
- Execution of the plugin command
    - The plugin command is executed on the active system management node.
    - The plugin command is executed with root privileges.
    - The execution directory of the plugin command is a directory specified in the --plugin-pre option and the --plugin-post option.
    - The plugin is executed in order from the first file name, which exists in a directory specified as the argument. (This order means the output order of ls -l when the environment variable LANG=C is set.)
    - The paclstmgr command continues processing after waiting for the return of the plugin command.
- Execution result of the plugin
    - If the end status of the plugin command is not 0, it is determined to be an abnormal end, exiting the paclstmgr command abnormally.
    - The output of the plugin command is output to the standard output.

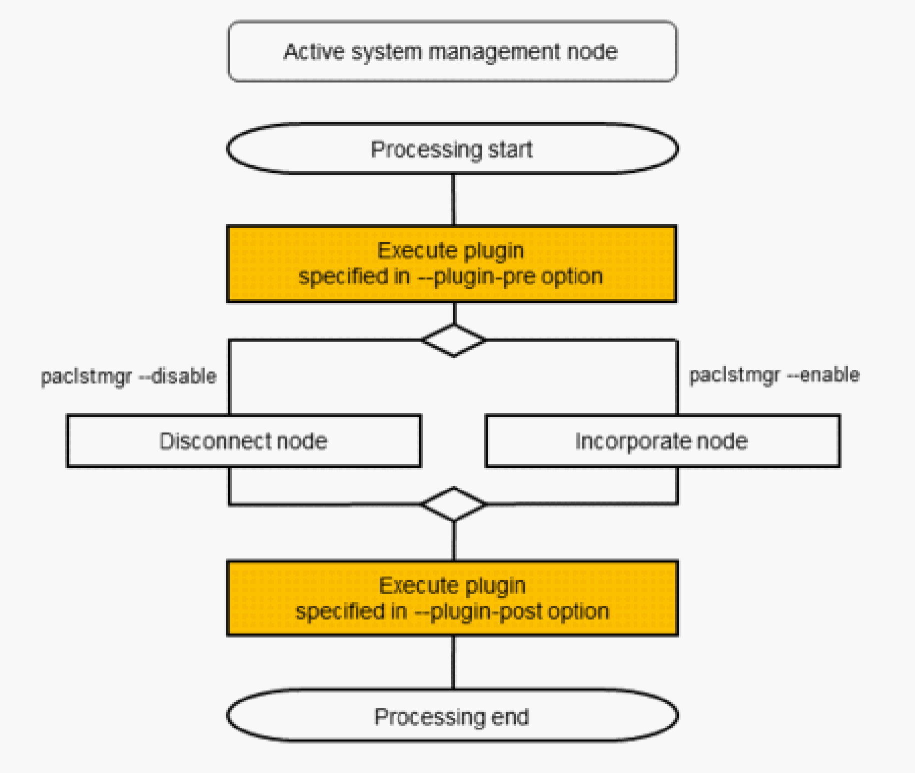
3.4.6.1 Operation Timing of the System Maintenance Function Plugin

This section describes the timing when the system maintenance function plugin runs in processing at the time of executing the paclstmgr command.
The timing of the plugin operation varies depending on operations to be performed.

When Disconnecting/Incorporating a Node

When disconnecting or incorporating a node, the system maintenance function plugin runs at the timing shown in orange as follows.

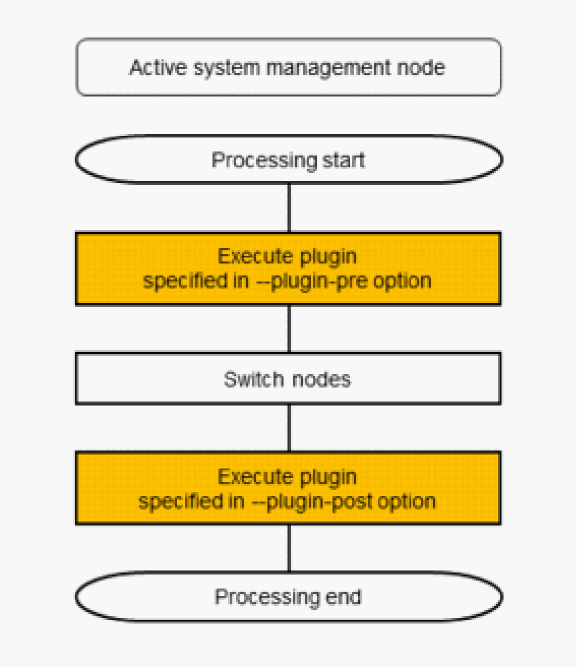
Figure 3.7 Operation Timing of the System Maintenance Function Plugin (Disconnecting/Incorporating a Node)


Active system management node
Processing start
Execute plugin specified in --plugin-pre option
paclstmgr --disable
paclstmgr --enable
Disconnect node
Incorporate node
Execute plugin specified in --plugin-post option
Processing end

When Switching a Node

When switching a node, the system maintenance function plugin runs at the timing shown in orange as follows.

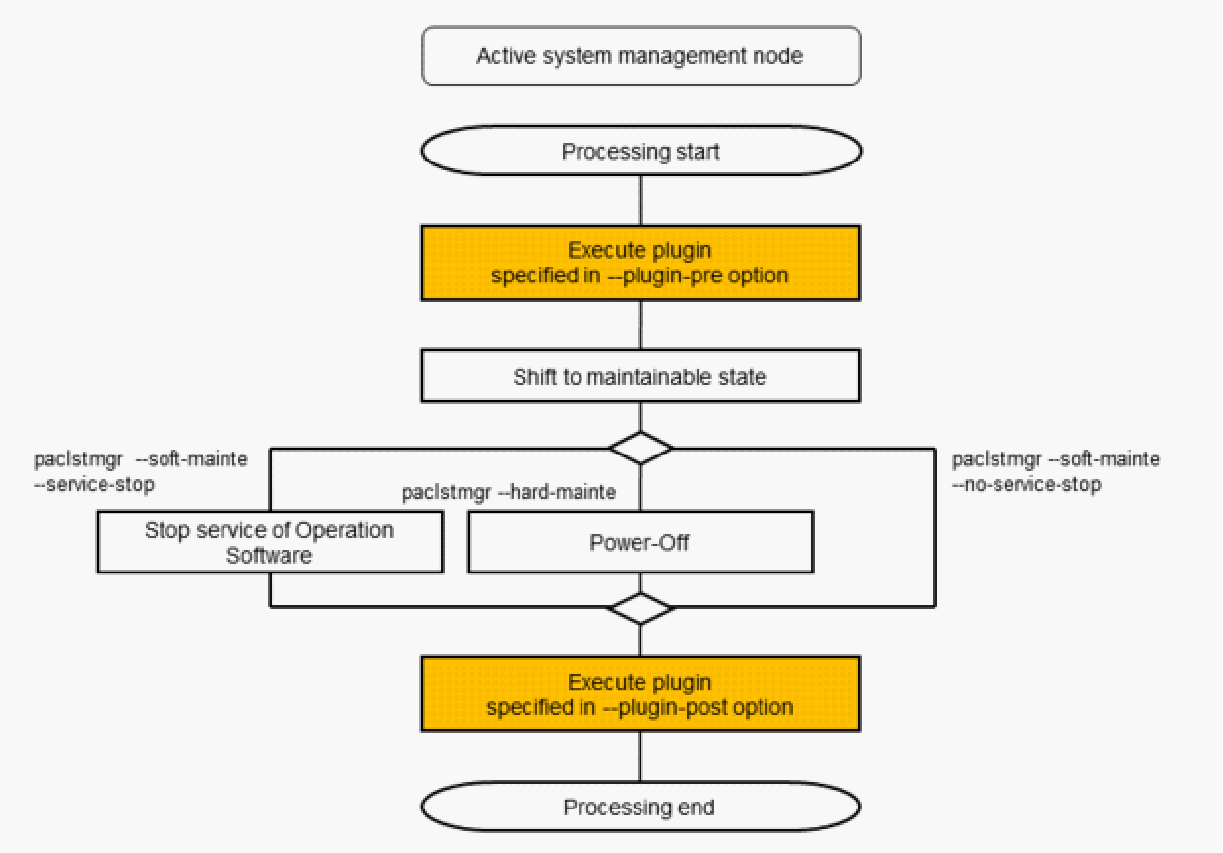
Figure 3.8 Operation Timing of the System Maintenance Function Plugin (Switching a Node)


Active system management node
Processing start
Execute plugin
specified in --plugin-pre option
Switch nodes
Execute plugin
specified in --plugin-post option
Processing end

When Shifting to the Maintenance Mode and Its Recovery

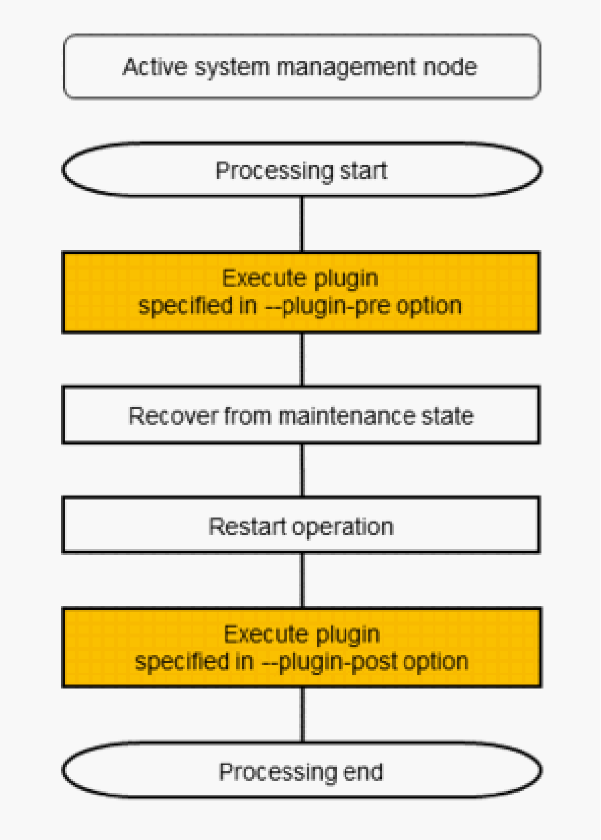
When shifting to the maintenance mode, the system maintenance function plugin runs at the timing shown in orange as follows.

Figure 3.9 Operation Timing of the System Maintenance Function Plugin (Shifting to the Maintenance Mode)



When recovering from the maintenance mode, the system maintenance function plugin runs at the timing shown in orange as follows.

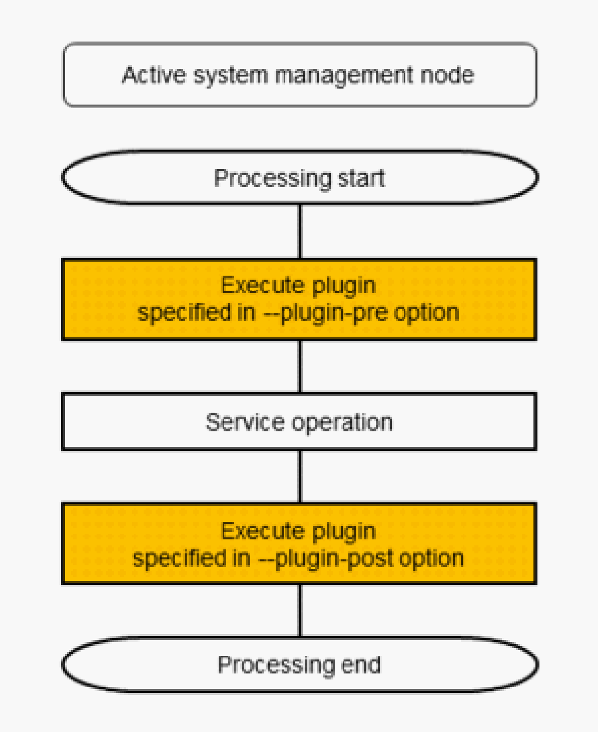
Figure 3.10 Operation Timing of the System Maintenance Function Plugin (Recovering from the Maintenance Mode)


Active system management node
Processing start
Execute plugin specified in --plugin-pre option
Recover from maintenance state
Restart operation
Execute plugin specified in --plugin-post option
Processing end

When Operating the Service of the Job Operation Software

In a service operation of the Job Operation Software, the system maintenance function plugin runs at the timing shown in orange as follows.

Figure 3.11 Operation Timing of the System Maintenance Function Plugin (Service Operation of the Job Operation Software)


Active system management node
Processing start
Execute plugin specified in --plugin-pre option
Service operation
Execute plugin specified in --plugin-post option
Processing end

3.4.6.2 Environment Variables

For information specified in an option of the paclstmgr command, an environment variable is used to pass this information to the plugin command.
By using the plugin command to refer to an environment variable, you can sort the types of operations and the processes based on target node types.

Table 3.44 Environment Variables Passed to the System Maintenance Function Plugin

Environment Variable Name	Description
CLSTMGR_CLSTNAME	Sets a cluster name specified in the -c option. Example: cluster1
CLSTMGR_SCOPE	Sets a specified scope option and argument. Example: -n 0x01010001 -n 0x01010001 --siogrp --nodegrp 0x01,0x02 --bootgrp 0x0101-0x0104 --cmu 0x01010001 --coord 0,0 --rscunit unit1 -a -f /tmp/filelist
CLSTMGR_NODETYPE	Sets a node type specified in the --nodetype option. Example: SIO
CLSTMGR_EXCLUDETYPE	Sets a node type specified in the --excludetype option. Example: BIO
CLSTMGR_MODEL	Sets a model name specified in the --model option.

Environment Variable Name	Description
	Example: PG
CLSTMGR_OPERATION	Sets a specified operation type. This sets one of the following types: "--enable" "--disable" "--hard-mainte" "--soft-mainte" "--recover" "--service stop" "--service start" "--service restart"
CLSTMGR_SOFTMAINTE_SUB	Sets a service operation type when specifying the --soft-mainte option. This sets one of the following types only when specifying the --soft-mainte option: "--no-service-stop" "--service-stop"
CLSTMGR_RECOVER_SUB	Sets a restart operation type when specifying the --recover option. This sets one of the following types only when specifying the --recover option: "--no-restart" "--service-restart" "--cold-reboot"
CLSTMGR_OPT_IC	Sets "--ic" when specifying the --ic option. When it is not set, this variable does not set a value.
CLSTMGR_OPT_K	Sets "-k" when specifying the -k option. When it is not set, this variable does not set a value.

3.5 Details of the Operation Support Function

3.5.1 About the Operation Support Function

When using the operation support function, note the following points regarding the setting of each node in the system:

- The operation support function uses SSH (the ssh command and the scp command). Therefore, the ssh command and the scp command must be executable to all nodes in a cluster.
- The operation support function uses public key authentication to authenticate each node and executes a command from the root account. It is therefore necessary to set "PubkeyAuthentication=yes" in SSH (sshd) on all nodes in order to make public key authentication available. Also, do not set "PermitRootLogin=no" because a command is executed from the root account.

Information

Key registration for public key authentication (meaning registration of a public key in each node) is already made when using the installation function to install the system management function. It is therefore unnecessary for the user to be conscious of such registration.

3.5.2 Collective Execution of Commands

To execute commands collectively, use the pmexe command of the batch operation function.

3.5.2.1 Collective Execution of Commands for Each Node

As shown below, execute the pmexe command to collectively execute commands on each node.

```
# pmexe -c clstname "command"
```

clstname: Cluster name
command: Command line to be executed on a target node

The pmexe command executes commands in parallel for all nodes in a specified range.

For a target node whose power status is not on (or os-running), the pmexe command automatically excludes the node from the target nodes for command execution, preventing any commands from running.

If all nodes are the execution target, a confirmation message is output when executing the pmexe command.
The following is an example of executing the pmexe command by specifying all nodes of a cluster called compute.

```
# pmexe -c compute "hostname"
[INFO] PSM 5081 pmexe All nodes are targets. Cluster name is compute.
Do you really want to continue (y/n)?
```

3.5.2.2 Parallel Execution of Commands

The batch operation function executes commands and transfers files in parallel for all nodes in a specified range.

The system administrator can use the -P option to reduce the number of parallel executions. The purpose of this is to reduce the load on the system management node and the target nodes for command execution that perform the batch operation function.

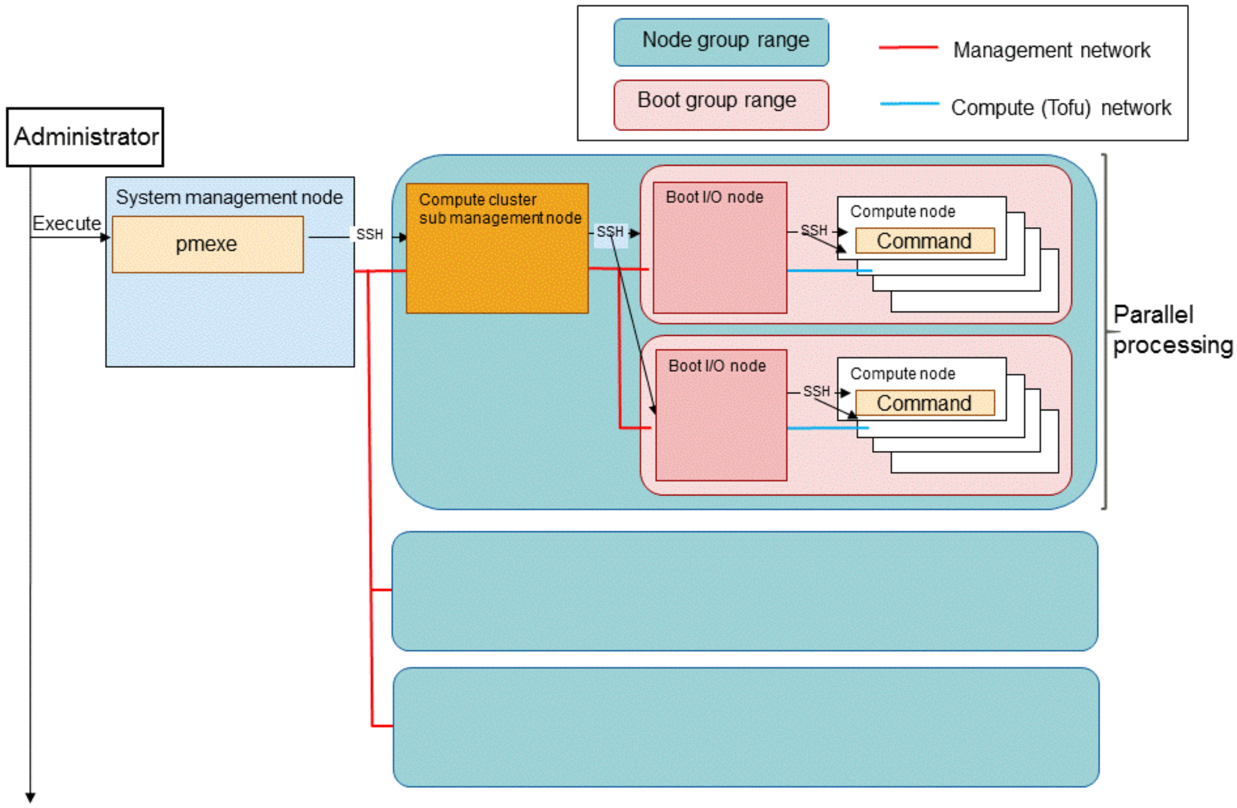
The following is an example of performing processing in parallel for each one node group.

```
# pmexe -c clstname -P 1:nodegrp command
```

clstname: Cluster name
command: Command line to be executed on a target node

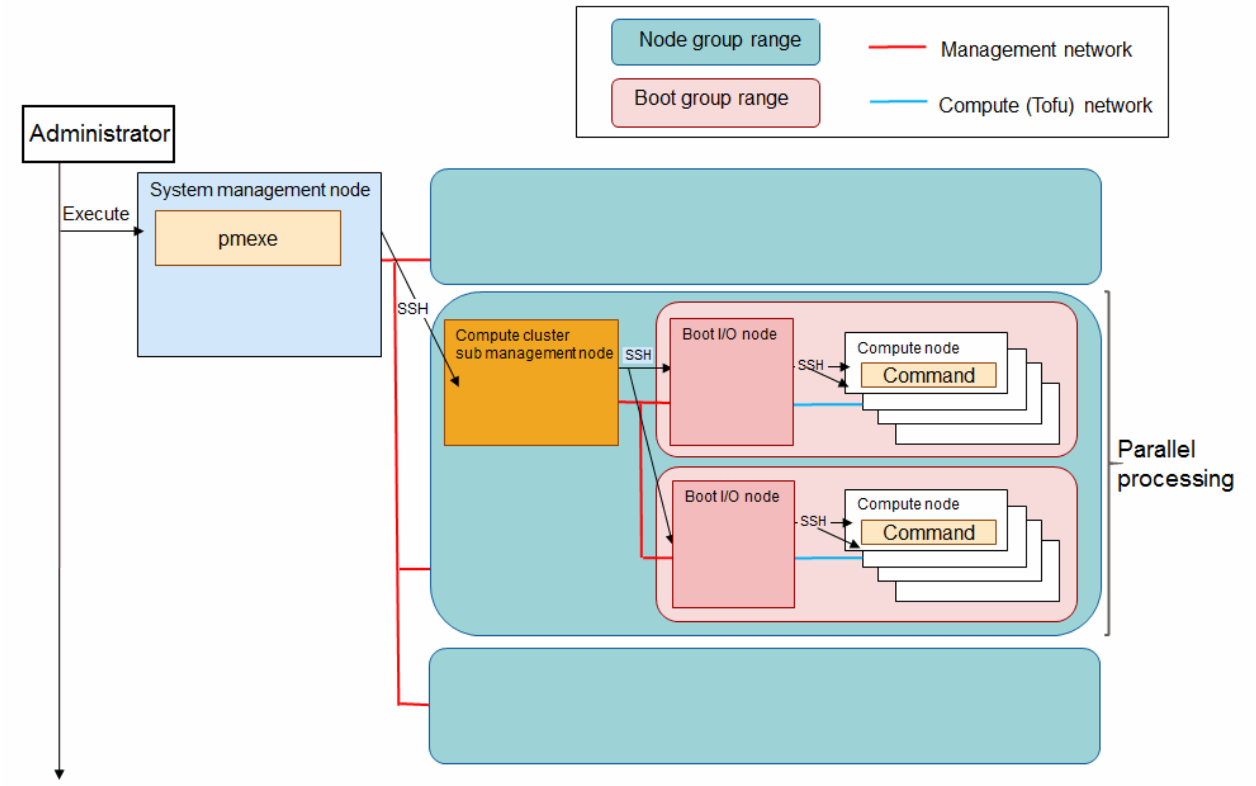
The following shows the image of execution of commands to all nodes under the first node group. These nodes are included in target nodes in a specified range.

Figure 3.12 Usage Image of Specifying the Number of Parallel Executions of the pmexe Command (Processing to the First Node Group)



The next image executes commands to nodes under the next node group after completing the execution of commands under the first node group.

Figure 3.13 Usage Image of Specifying the Number of Parallel Executions of the pmexe Command (Processing to the Next Node Group)



3.5.2.3 Outputting the Execution Result of Collective Execution

The pmexe command outputs the execution result to a file or the standard output, so that the user can discriminate the result of collectively-executed commands.

Table 3.45 Execution Result File of the pmexe Command

Output File Path	Details of Output File
pmexe_result_*date.pid*	Execution result when a command specified on a target node is executable (regardless of the success and failure of the specified command)
pmexe_failed_*date.pid*	Result when network access is unavailable to a target node, or result when a specified command itself is not executable because the command does not exist on a target node
pmexe_excluded_*date.pid*	Result of a node that is excluded from the execution target, due to such reasons as that the power supply status of the target node is not on and that PowerControlOnly=yes has been set

date: Date and time when executing the pmexe command (Example: 10:27 on July 6th, 2017 -> 201707061027)
pid: Process ID when executing the pmexe command

The execution result file of the pmexe command is output to the home directory of the root account on the executing system management node.

```
# pmexe -c compute -n 0xFFFF0001-0xFFFF0005 hostname
[INFO] PSM 5082 pmexe The execution result file was output.(/root/pmexe_result_20160401.8740)
[WARN] PSM 5051 pmexe The execution failed file was output.(/root/pmexe_failed_20160401.8740)
[INFO] PSM 5083 pmexe The execution exclude file was output.(/root/pmexe_excluded_20160401.8740)
```

By specifying the -d option and the --filename option, you can also arbitrarily change the output destination path and the output file name for the execution result file of the pmexe command.

The following is an example of changing the output destination path and the output file name for the execution result file by specifying the -d option and the --filename option in the pmexe command.

```
# pmexe -c compute -n 0xFFFF0001-0xFFFF0005 -d /var/tmp/ --filename test hostname
[INFO] PSM 5082 pmexe The execution result file was output.(/var/tmp/test_result)
[WARN] PSM 5051 pmexe The execution failed file was output.(/var/tmp/test_failed)
[INFO] PSM 5083 pmexe The execution exclude file was output.(/var/tmp/test_excluded)
```

The execution result is output to the execution result file of the pmexe command in the following formats:

- Successful execution result

```
# cat /root/pmexe_result_20160401.8740
[cmdline]
pmexe -c compute -n 0xFFFF0001-0xFFFF0005 hostname
[cluster]
compute
[0xFFFF0001] node01
[0xFFFF0002] node02
[0xFFFF0003] node03
```

- Failed execution result

```
# cat /root/pmexe_failed_20160401.8740
[cmdline]
pmexe -c compute -n 0xFFFF0001-0xFFFF0005 hostname
[cluster]
compute
[0xFFFF0004] ssh: connect to host target node IP address port 22: Connection refused
```

- Excluded execution result

```
# cat /root/pmexe_excluded_20160401.8740
[cmdline]
pmexe -c compute -n 0xFFFF0001-0xFFFF0005 hostname
[cluster]
compute
[0xFFFF0005] This node is excluded. (REASON: PWR_STATUS is not os-running)
```

By specifying the --stdout option, you can also display the result on the standard output without outputting it to the file.

```
# pmexe -c compute -n 0xFFFF0001-0xFFFF0005 --stdout hostname
[cmdline]
pmexe -c compute -n 0xFFFF0001-0xFFFF0005 hostname
[cluster]
compute
<<<<< ResultInformation >>>>>                                                         # (*1)
[0xFFFF0001] node01
[0xFFFF0002] node02
[0xFFFF0003] node03
<<<<< FailedInformation >>>>>                                                         # (*2)
[0xFFFF0004] ssh: connect to host target node IP address port 22: Connection refused
<<<<< ExcludeInformation >>>>>                                                        # (*3)
[0xFFFF0005] This node is excluded. (REASON: PWR_STATUS is not os-running.)
```

(*1) Header showing the normal result
(*2) Header showing the abnormal result
(*3) Header showing the excluded result

Note

- The error of the pmexe command itself (such as option error) is not output to the file described above, but to the standard error output.
- Collective execution using the pmexe command removes lines which include any of the following character strings; it does not output such lines.

Table 3.46 Character Strings to be Removed

Character String to be Removed	Remarks
Connection to *ipaddr* closed.	*ipaddr*: IP address of the management network on node to be executed
's password:	When the character string is at the end of a line
Warning: Permanently added	

- Collective execution using the pmexe command outputs lines including any of the following character strings to the error file.

Table 3.47 Character Strings Output to the Error File

Character String Output to Error File	Remarks
Command timed out	
Permission denied, please try again.	
Permission denied (*A character string showing an authentication method*).	When the character string is at the end of a line
resolution	When the character string is at the end of a line
bash: *Command name*: command not found	When the character string is at the end of a line, and when the beginning of a line is either bash, /bin/bash, or /usr/bin/bash
bash: *Command name*: No such file or directory	When the character string is at the end of a line, and when the beginning of a line is either bash, /bin/bash, or /usr/bin/bash
ssh connect to:	
command-line line	When the character string is at the beginning of a line
/usr/bin/chroot:	When the character string is at the beginning of a line

3.5.2.4 Collective Execution of Commands on FX server

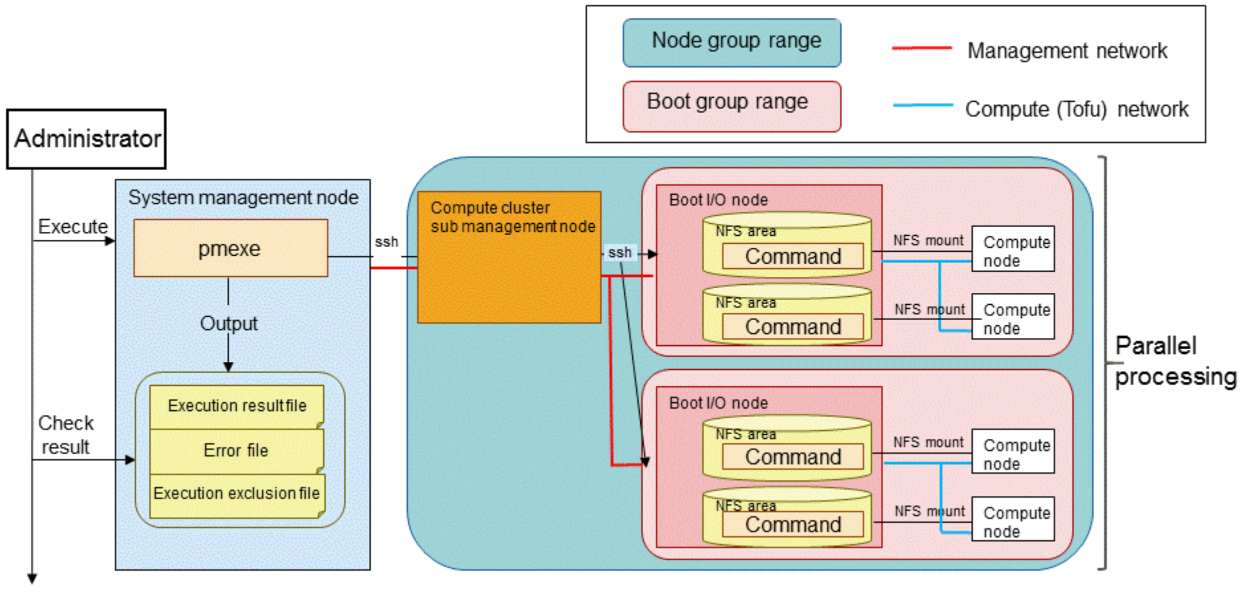
The compute node of FX server NFS-mounts the disk space on the boot I/O node. Therefore, the data may be corrupted by editing the same NFS area from both the boot I/O node and the compute node in such a case of applying a package. To avoid this symptom, the system administrator is to specify the --chroot option in the pmexe command. This allows him/her to execute a command on the boot I/O node by chrooting into the NFS area where the compute node mounts.

The following is an example of executing the hostname command to the NFS area where the compute node (Node ID: 0xFF010010) of FX server mounts. A cluster called compute contains this compute node.

```
# pmexe --chroot -c compute -n 0xFF010010 hostname
[INFO] PSM 5082 pmexe The execution result file was output. (/root/pmexe_result_20160401.8740)
```

This enables you to prevent data corruptions caused by editing the same NFS area from both the boot I/O node and the compute node in such a case of applying a package.

Figure 3.14 Usage Image of Executing a Command by Chrooting Into the NFS Area on the Boot I/O Node



3.5.3 Delivering Files Collectively

To deliver files collectively, use the pmscatter command of the batch operation function.

3.5.3.1 Delivering Files Collectively to Each Node

As shown below, execute the pmscatter command to collectively deliver files to each node.

```
# pmscatter -c clstname srcfile dstfile
```

clstname: Cluster name
srcfile: Transfer source file on the system management node
dstfile: Transfer destination file on a target node

The pmscatter command delivers files in parallel to all nodes in a specified range.
For a target node whose power status is not on (or os-running), this command automatically excludes it from the delivery target nodes, preventing files from being delivered.

If all nodes are the execution target, a confirmation message is output.
The following is an example of executing the pmscatter command by specifying all nodes of a cluster called compute.

```
# pmscatter -c compute srcfile dstfile
[INFO] PSM 5081 pmscatter All nodes are targets. Cluster name is compute.
Do you really want to continue (y/n)? y
```

3.5.3.2 Parallel Execution in Collective Delivery

As with the pmexe command, the pmscatter command also allows you to reduce the number of parallel executions.

For details on how to specify it, see "3.5.2.2 Parallel Execution of Commands" below and "pmscatter command" in the "Job Operation Software Command Reference."

3.5.3.3 Outputting the Execution Result of Collective Delivery

The pmscatter command outputs the execution result to a file, so that the user can discriminate the results of a collective delivery.

Table 3.48 Execution Result File of the pmscatter Command

Output File Path	Details of Output File
pmscatter_failed_*date.pid*	Result when network access is unavailable to a target node, or result when the file delivery itself is not executable because the specified directory does not exist on a target node
pmscatter_excluded_*date.pid*	Result of a node that is excluded from the execution target, due to such reasons as that the power supply status of the target node is not on and that PowerControlOnly=yes has been set

date: Time when executing the pmscatter command (10:27 on July 6th, 2017 -> 201707061027)
pid: Process ID when executing the pmscatter command

The execution result file of the pmscatter command is output to the home directory of the root account on the executing system management node.

```
# pmscatter -c compute -n 0xFFFF0001-0xFFFF0005 srcfile dstdir
[WARN] PSM 5051 pmscatter The execution failed file was output. (/root/pmscatter_failed_20160401.8740)
[INFO] PSM 5083 pmscatter The execution exclude file was output. (/root/
pmscatter_excluded_20160401.8740)
```

By specifying the -d option or the --filename option, you can also arbitrarily change the output destination path and the output file name for the execution result file of the pmscatter command.

The following is an example of changing the output destination path and the output file name for the execution result file by specifying the -d option and the --filename option in the pmexe command.

```
# pmscatter -c compute -n 0xFFFF0001-0xFFFF0005 -d /var/tmp/ --filename test srcfile dstdir
[WARN] PSM 5051 pmscatter The execution failed file was output. (/var/tmp/test_failed)
[INFO] PSM 5083 pmscatter The execution exclude file was output. (/var/tmp/test_excluded)
```

The execution result is output to the execution result file of the pmscatter command in the following formats.
Note that nothing is output for nodes which succeed in delivering the file.

- Failed execution result

```
# cat /root/pmscatter_failed_20160401.8740
[cmdline]
pmscatter -c compute -n 0xFFFF0001-0xFFFF0005 srcfile dstdir
[cluster]
compute
[0xFFFF0004] ssh: connect to host target node IP address port 22: Connection refused
```

- Excluded execution result

```
# cat /root/pmscatter_excluded_20160401.8740
[cmdline]
pmscatter -c compute -n 0xFFFF0001-0xFFFF0005 srcfile dstdir
[cluster]
compute
[0xFFFF0005] This node is excluded. (REASON: PWR_STATUS is not os-running)
```

3.5.4 Collecting Files Collectively

To collect files collectively, use the pmgather command of the batch operation function.

3.5.4.1 Collecting Files Collectively From Each Node

As shown below, execute the pmgather command to collectively collect files from each node.

```
# pmgather -c clstname srcfile dstdir
```

clstname: Cluster name
srcfile: Transfer source file on a target node
dstdir: Transfer destination directory to be placed on the system management node

The pmgather command collects files in parallel from all nodes in the range.
For a target node whose power status is not on (or os-running), this command automatically excludes it from the collection target nodes, preventing files from being collected.

If all nodes are the execution target, a confirmation message is output.
The following is an example of executing the pmgather command by specifying all nodes of a cluster called compute.

```
# pmgather -c compute srcfile dstdir
[INFO] PSM 5081 pmgather All nodes are targets. Cluster name is compute.
Do you really want to continue (y/n)? y
```

3.5.4.2 Parallel Execution in Collective Collection

As with the pmexe command, the pmgather command also allows you to reduce the number of parallel executions.

For details on how to specify it, see "3.5.2.2 Parallel Execution of Commands" below and "pmgather command" in the "Job Operation Software Command Reference."

3.5.4.3 Outputting the Execution Result of Collective Collection

The pmgather command outputs the execution results to a file, so that the user can discriminate the results of a collective collection.

Table 3.49 Execution Result File of the pmgather Command

Output File Path	Details of Output File
pmgather_failed_*date.pid*	Result when network access is unavailable to a target node, or result when the file collection itself is not executable because the specified file does not exist on a target node
pmgather_excluded_*date.pid*	Result of a node that is excluded from the execution target, due to such reasons as that the power supply status of the target node is not on and that PowerControlOnly=yes has been set

date: Time when executing the pmgather command (For example, "10:27 on July 6th, 2017" is displayed as "201707061027.")
pid: Process ID when executing the pmgather command

This is the same as the pmscatter command, except for the file name and the command name output into the file. For the output example, see "3.5.3.3 Outputting the Execution Result of Collective Delivery."

Note

The pmgather command uses the scp command to collect the specified files from multiple execution nodes to a system management node, but the scp command behaves differently for path expansion depending on the OS version of the system management node.
Note that the behavior of the pmgather command varies depending on the behavior of the scp command.
For example:

[The system management node is using RHEL7]
File collection succeeds even if double quotes are escaped when specifying the file names to be collected by the pmgather command. (Example: \"/etc/hosts\")

[The system management node is using RHEL8.1 or later]
If the file name collected by the pmgather command is specified by escaping double quotes, "protocol error: filename does not match request" will occur and file collection will fail. (Example: \"/etc/hosts\")

The following is an example of executing the pmgather command by specifying 0xFFFF0001 node of a cluster called compute.

```
# /usr/sbin/pmgather -c compute -n 0xFFFF0001 \"/etc/hosts\" ./
[WARN] PSM 5051 pmgather The execution failed file was output.(/root/
pmgather_failed_202010211436.80706)
# cat /root/pmgather_failed_202010211436.80706
[cmdline]
/usr/sbin/pmgather -c compute -n 0xFFFF0001 "/etc/hosts" ./
[cluster]
compute
[0xFFFF0001] protocol error: filename does not match request
```

If "protocol error: filename does not match request" occurs, the file names collected by the pmgather command are not properly specified, so review the file name specifications.
In the above example, specify the file name without escaping double quotes.

3.5.5 Managing Dump Files

To manage dump files that are collected such as when a node panics, you can use the padumpmgr command from the system management node. Execute the padumpmgr command to manage dump files collected on each node.

The padumpmgr command allows the system administrator to perform unitary management of these dump files.

- Listing dump files (padumpmgr --list)

  You can display the number and collection time of dump files for each node.

- Updating the information on dump files (padumpmgr --refresh)

  You can update the dump file list to the latest state accordingly.

- Transferring a dump file (padumpmgr --get)

  You can transfer the dump file of a specified node to the system management node.

- Deleting a dump file (padumpmgr --del)

  You can delete an unnecessary dump file.

3.5.5.1 Listing Dump Files

To display the list of dump files existing on each node, use the --list option of the padumpmgr command.
The following is an example of checking dump files that exist in a cluster called cluster.

[Displaying the total number of dump files]

The following example executes the padumpmgr command by specifying a node group whose node group ID is 0x01.

```
# padumpmgr --list -c cluster --nodegrp 0x01
[ CLST: cluster ]
[ NODEGRP: 0x01 ]
NODEGRP          NODE_SUM
0x01             10
```

[Displaying the detailed information of dump files (PRIMERGY server and any node)]

The following example executes the padumpmgr command by specifying a node group whose node group ID is 0x01.

```
# padumpmgr --list -c cluster --nodegrp 0x01 -v
[ CLST: cluster ]
[ NODEGRP: 0x01 ]
NODE       DUMPID   DATE             SIZE(M)
0x01FF0001 1        2017-08-10-10:00 30626M
```

```
0x01FF0002 2        2017-08-18-11:00 37638M
...
0x01FF0009 10       2017-08-19-12:00 37638M
```

[Displaying the detailed information of dump files (FX server)]

The following example executes the padumpmgr command by specifying a boot group whose boot group ID is 0x0101.

```
# padumpmgr --list -c cluster --bootgrp 0x0101 -v
[ CLST: cluster ]
[ BOOTGRP: 0x0101 ]
NODE       DUMPID   DATE             SIZE(M)
0x01010003 10000001 2017-08-25-13:43 -
...
0x01010010 10000008 2017-08-25-16:08 -
```

CLUSTER(CLST)

Cluster name

NODE_SUM

The total number of dumps held in a specified range

NODE

Node ID of a node from which dump files are collected

DUMPID

Dump ID (Serial number of dump files)
This is a unique value in all clusters that the system management node manages.

DATE

Date and time when a dump file is collected

SIZE(M)

This displays the size of a dump file in MiB.
For nodes of FX server, the hyphen (-) appears here.

Note

- For PRIMERGY server, only memory dumps (vmcore) are listed, not including incomplete memory dumps (vmcore-incomplete).
- For FX server, you are viewing emergency dump information on the BMC. If a memory dump is deleted, such as by rebuilding an FX server, it appears in the list but cannot be transferred.

3.5.5.2 Transferring Dump Files

To transfer dump files existing on each node to the system management node, use the --get option of the padumpmgr command.
The following is an example of transferring all the dump files in the boot group 0x0201 to a location under the directory /work/dump. This boot group exists in a cluster called cluster.

```
# padumpmgr --get -c cluster -d /work/dump/ --bootgrp 0x0201
[WARNING]
The following dumps are forwarded.
BOOTGRP          NODE_SUM
0x0201           8
Do you really want to continue(y/n)? :       <- Prompt requesting your confirmation
Dump forwarding...
```

When transferring dump files, the load on the network may increase. For this reason, the padumpmgr command requests your confirmation before the processing starts.
As shown below, the storage location of transferred dump files has a directory configuration corresponding to a node configuration.

[Storage location of transferred dump files]

```
/Any directory                <- Directory specified by the -d option
    /Cluster name
        /Node group (0x02)
            /Boot group (0x01)
                /Node ID          <- Transferred dump files are stored under the node ID directory.
                                "_nnn" is added to the end of the name of transferred dump files. (nnn
is a dump ID.)
```

Note

- You cannot transfer dump files on the boot I/O node of FX server under the following condition.
  When another BoB is adjacent to a target boot I/O node and the boot I/O node of this BoB stops, you cannot transfer dump files. This is because FX server stores the dump files of a boot I/O node in the boot I/O node of another adjacent BoB. The another adjacent BoB mentioned here means FX server in which boot group IDs lie next to each other. As shown below, three boot group IDs compose a cyclic relationship.

  Table 3.50 Relationship Between Boot I/O Nodes and Nodes Storing Dumps

Target Boot I/O Node	Boot I/O Node of Adjacent BoB
0x01010001	0x01020001
0x01020001	0x01030001
0x01030001	0x01010001
0x01040001	0x01050001
0x01050001	0x01060001
0x01060001	0x01040001

  In such a case, the following message is output when transferring dump files (meaning when executing the --get option of the padumpmgr command).

  ```
  [WARN] PSM 5451 padumpmgr It failed in forwarding the dump. (Details)
  ```

- In FX server, after transferring dump files, they still remain on a node where an abnormality occurs. You cannot delete dump files of FX server even though you specify not only the --del option of the padumpmgr command, but also the --erase option when transferring dump files. (They remain for up to three generations. After that, the OS automatically deletes them every time a new dump file is generated.)

- When transferring dump files on nodes of PRIMERGY server and any node, files collected in the same directory on Sysvol (such as fjsnap, summary, and vmlinux) are also transferred. These files are then stored in the same location as the dump files. Deleting the dump files also deletes these files.

3.5.5.3 Deleting Dump Files

To delete the dump files from a node, use the --del option of the padumpmgr command.

The following is an example of deleting the dump file from one node (Node ID: 0x01FF0003) of a cluster called cluster.

```
# padumpmgr --del -c cluster -n 0x01FF0003
[WARNING]
The following dumps are deleted.
NODE          DUMPID     DATE                 SIZE(M)
0x01FF0003    19         2018-01-16-13:49     131431M
Do you really want to continue (y/n)? y
Dump deleting....
```

To prevent you from deleting the dump file by mistake, the padumpmgr command requests your confirmation before the processing starts.

Note

- You cannot delete dump files on the boot I/O node of FX server. (They remain for up to three generations. After that, the OS automatically deletes them every time a new dump file is generated.)
- The padumpmgr command cannot remove an incomplete memory dump of the PRIMERGY server (vmcore-incomplete). If you want to delete it, please delete it according to the OS documentation.

3.5.5.4 Updating the Information on Dump Files

Normally, the dump file list is automatically updated when a new dump file is generated. If any cause such as network abnormality prevents communications between the system management node and other nodes for a certain period, however, use the --refresh option of the padumpmgr command to update the information manually.

The following is an example of refreshing the dump file list in a cluster called cluster.

```
# padumpmgr --refresh -c cluster
[WARNING]
The dumplist of cluster is updated.
Do you really want to continue(y/n)? y
Dump refreshing...
```

Updating the dump file list requires all target nodes to communicate with each other, which may increase the network load. For this reason, the padumpmgr command requests your confirmation before the processing starts.

Note

For FX server, you cannot use the --refresh option to update it. The system monitoring function automatically updates dump files of FX server.

3.5.6 Connecting to the Console of a Node

The system management function is a system consisting of a large number of nodes. This has the console connection function to establish connections to consoles, supporting the operation work of the system administrator.

The console connection function enables the system administrator to perform operations by making a console connection to each node. Use the paconsole command to connect to the console of a node.

Note

- The paconsole command uses ipmitool in order to connect the consoles of the PRIMERGY server. It is therefore necessary to make a setting for BIOS or UEFI, so that ipmitool becomes available.
- The console connection function does not support nodes of any node.
- You can make only one console connection to one node at a time.

3.5.6.1 Connecting to the Console of a Node

Use the paconsole command to connect to the console of a node.

The following is an example of connecting to the console of a node (Node ID: 0x01FF0003) in a cluster called cluster.

```
# paconsole -c cluster -n 0x01FF0003
<I/O display after console connection>
...
#.      <- Exiting console
[INFO] PSM 5381 paconsole Connection to the console closed.
```

To exit the connected console, press the [Enter] key, and then enter "#". (which is a sharp sign followed by a period).

3.6 Details of the Log Management Function

3.6.1 Investigation Data Collection

This section describes the following three procedures of the investigation data collection function:

- Collection of investigation data
- Collection of investigation data by specifying a trouble event
- Collection of investigation data by specifying a job ID

3.6.1.1 Collection of Investigation Data

1. Collecting investigation data

   The system administrator can collect investigation data by executing pasnap, the investigation data collection command, on the active system management node or the multiuse node. The system administrator can also narrow down collected investigation data on anywhere other than components. For details, see "pasnap command" in the "Job Operation Software Command Reference."
   The following is an example of collecting investigation data of the component PSM against a node which has a cluster named compute and the boot group 0x0101.

   ```
   # pasnap -c compute --bootgrp 0x0101 --component PSM
   Do you really want to continue (y/n)? y
   <<<< Execute pmexe >>>>
   [INFO] PSM 5082 pmexe The execution result file was output.(./snap/pasnap_exec_snap_result)
   [INFO] PSM 5082 pmexe The execution result file was output.(./snap/pasnap_exec_snap_bio_result)
   <<<< Execute pmgather >>>>
   [INFO] PSM 5082 pmexe The execution result file was output.(./snap/pasnap_exec_rmsnap_result)
   [INFO] PSM 5082 pmexe The execution result file was output.(./snap/pasnap_exec_rmsnap_bio_result)
   ```

   The pasnap command outputs the file of the batch operation function that is internally executed.
   For details on the files output by the pasnap command, see "3.6.1.4 Files Output by the Investigation Data Collection Command."

2. Checking investigation data

   Regarding investigation data that the investigation data collection command pasnap collects, palsnap.tgz, the compressed investigation data file is created for each node. This file is placed under the directory where executes the command, as shown below.

   ```
   snap/Cluster name/Node group ID/Boot group ID/Node ID/palsnap.tgz
   ```

   By expanding palsnap.tgz, the investigation data file, you will find that the directory for each component stores the investigation data.

3.6.1.2 Collection of Investigation Data by Specifying a Trouble Event

The system administrator can collect investigation data of a node and component related to a trouble event. To do so, specify the trouble event and execute pasnap, the investigation data collection command, on the active system management node or the multiuse node.

The following is an example of collecting investigation data by specifying the trouble event CN-NODE-DOWN for a node which has a cluster named compute and the node ID 0x01010003.

```
# pasnap -c compute -n 0x01010003 --case CN-NODE-DOWN
Do you really want to continue (y/n)? y
<<<< Execute pmexe >>>>
[INFO] PSM 5082 pmexe The execution result file was output.(./snap/pasnap_exec_snap_result)
[INFO] PSM 5082 pmexe The execution result file was output.(./snap/pasnap_exec_snap_bio_result)
<<<< Execute pmgather >>>>
[INFO] PSM 5082 pmexe The execution result file was output.(./snap/pasnap_exec_rmsnap_result)
[INFO] PSM 5082 pmexe The execution result file was output.(./snap/pasnap_exec_rmsnap_bio_result)
```

To check the trouble definition name to be specified in the --case option, refer to the trouble definition file on the following path.
You can also add the definition of any trouble event by editing the trouble definition file manually.

```
/etc/opt/FJSVtcs/patrouble.conf
```

The following is an example of describing the trouble definition file. This example collectively collects the investigation data of the following relevant nodes when specifying the compute node in the pasnap command:

- Target compute node
- Boot I/O node in the same boot group
- Compute cluster sub management node in the same node group
- Compute cluster management node in the same cluster
- System management node

For details on the trouble definition file, see "patrouble.conf file" in the "Job Operation Software Command Reference."

```
Trouble {
    TroubleName = CN-NODE-DOWN       # (*1)
    NodeTypeName = CN                # (*2)
    RelatedNode {                    # (*3)
        NodeTypeName = CN            # (*4)
        Condition = target_node      # (*5)
        Component = ALL              # (*6)
    }
    RelatedNode {
        NodeTypeName = BIO
        Condition = same_bootgrp
        Component = ALL
    }
    RelatedNode {
        NodeTypeName = CCS
        Condition = same_nodegrp
        Component = ALL
    }
    RelatedNode {
        NodeTypeName = CCM
        Condition = same_cluster
        Component = ALL
    }
    RelatedNode {
        NodeTypeName = SMM
        Condition = same_cluster
        Component = ALL
    }
}
```

(*1) Trouble definition name
(*2) Node type name of a node where a trouble occurs
(*3) Relevant node definition section
(*4) Node type name of a relevant node
(*5) Name of the relationship between the node where a trouble occurs and the relevant node
(*6) Name of the component on the relevant node that gets investigation data

Note

When collecting investigation data by specifying a trouble event, be sure to place patrouble.conf in /etc/opt/FJSVtcs/patrouble.conf. Do not change the file name, and do not delete the file.

3.6.1.3 Collection of Investigation Data by Specifying a Job ID

The system administrator can collect investigation data of the time period for executing a job. To do so, specify the job ID and execute pasnap, the investigation data collection command, on the system management node.

The following is an example of collecting investigation data by specifying a job whose job ID is *jobid*, which is executed in a cluster named compute.

```
# pasnap -c compute --jobid jobid
Do you really want to continue (y/n)? y
<<<< Execute pmexe >>>>
[INFO] PSM 5082 pmexe The execution result file was output.(./snap/pasnap_exec_snap_result)
<<<< Execute pmgather >>>>
[INFO] PSM 5082 pmexe The execution result file was output.(./snap/pasnap_exec_rmsnap_result)
```

jobid: Job ID

Information

The pjstat command allows you to refer to the information of a job. For details on the pjstat command, see "pjstat command" in the "Job Operation Software Command Reference."

3.6.1.4 Files Output by the Investigation Data Collection Command

pasnap, the investigation data collection command, uses the batch operation function inside this command.
The batch operation function outputs files as the result files of the pasnap command under the snap directory, which is created in the current directory.

```
# pasnap -c compute --bootgrp 0x0101 --component PSM
Do you really want to continue (y/n)? y
<<<< Execute pmexe >>>>
[INFO] PSM 5082 pmexe The execution result file was output.(./snap/pasnap_exec_snap_result)
[INFO] PSM 5082 pmexe The execution result file was output.(./snap/pasnap_exec_snap_bio_result)
<<<< Execute pmgather >>>>
[INFO] PSM 5082 pmexe The execution result file was output.(./snap/pasnap_exec_rmsnap_result)
[INFO] PSM 5082 pmexe The execution result file was output.(./snap/pasnap_exec_rmsnap_bio_result)
```

Information

When specifying the -d option in the pasnap command, the snap directory is created in a directory specified as an argument, and then the files are output under this directory.

The pasnap command outputs files named as shown below, according to the operations to be performed and the objects of the operations.

Table 3.51 Files That the pasnap Command Outputs

Collected Data	File Name
Execution result file	pasnap_*Operation name*_(*Object name*_)result
Error file	pasnap_*Operation name*_(*Object name*_)failed
Execution exclusion file	pasnap_*Operation name*_(*Object name*_)excluded

The following contents are set to the *Operation name.*

Table 3.52 Operation Names of the pasnap Command

Operation Name	Operation Content
exec_snap	Creating investigation data on each node
gather	Collecting investigation data on each node
exec_rmsnap	Deleting investigation data on each node

Operation Name	Operation Content
scatter	Distributing the definition file on each node

The following contents are set to the *Object name*.

Table 3.53 Object Names of the pasnap Command

Object Name	Description
bio	For information on processing of an internal command running on the boot I/O node, the *Object name* becomes bio.
nodelist	For information on distribution of an internal file to the boot I/O node, the *Object name* becomes nodelist.
troublefile	For information on distribution of an internal file when specifying the --case option, the *Object name* becomes troublefile.

Among the output files, check the content of a file whose name has "_result" at the end.

- If an error message is output to the file whose name has "_result" at the end, it means that the processing of the pasnap command for a corresponding node fails. See "Job Operation Software Troubleshooting" in order to take some measure.

  ```
  [0xFFFF0010]                          (*1)
  [0xFFFF0011] messsage                 (*2)
  ```

  message: Error message
  (*1) The processing to the node (Node ID: 0xFFFF0010) is successful. It is unnecessary to take some measure.
  (*2) The ssh communication to the node (Node ID: 0xFFFF0011) fails. It is necessary to take some measure.

- If a file is output and its name has "_failed" or "_excluded" at the end, it means that the processing of the pasnap command fails. See "Job Operation Software Troubleshooting" in order to take some measure.

3.6.2 Log Monitoring

This section describes the following two procedures of the log monitoring function:

- Configuring Log Monitoring

- Configuring Mellanox HCA monitoring

3.6.2.1 Configuring Log Monitoring

The system administrator can predefine a log file to be monitored, character strings to be searched, and operations at the time of detecting a message.

Describe the setting content in paevent.conf, the log monitoring configuration file, and then apply the setting by using the --set option of the paeventadm command. In addition, you can refer to the currently set contents from the --show option of the paeventadm command.
The following shows the procedure to set the log monitoring. Perform all the following operations on the active system management node.

1. Creating the log monitoring configuration file

   Create paevent.conf, the log monitoring configuration file, in order to perform the log monitoring.
   Place paevent.conf in the following location on the active system management node.

   ```
   /etc/opt/FJSVtcs/paevent.conf
   ```

   Information

   When introducing the Job Operation Software, the paevent.conf file does not exist. Copy the sample file (/etc/opt/FJSVtcs/sample/paevent.conf), and then place it on the path described above.

The following is an example of describing the log monitoring configuration file.
For details on the log monitoring configuration file, see "paevent.conf file" in the "Job Operation Software Command Reference."

```
Cluster {                                                         # (*1)
    ClusterName = cluster1                                        # (*2)
    NodeType {                                                    # (*3)
        NodeTypeName = SMM,BIO,CN                                 # (*4)
        Event {                                                   # (*5)
            EventName = sample1                                   # (*6)
            FileName = "/var/log/messages"                        # (*7)
            ChkString = "kerne.*segfaul"                          # (*8)
            IgnoreString = "error 1"                              # (*9)
            IgnoreString = "error 2"
            IgnoreString = "error 3"
            ExecCmd  = "echo $PAEVENT_STR1 > /tmp/result1.txt"    # (*10)
        }
    }
}
```

(*1) Section defining the settings for each cluster
(*2) Cluster name
(*3) Section defining the settings for each node type
(*4) Node type name
(*5) Section defining the log monitoring operation
(*6) Log monitoring definition name
(*7) Absolute path of a log file to be monitored
(*8) Character string of the monitoring target
(*9) Character string excluded from the monitoring target
(*10) Any command to be executed when detecting a message

2. Executing the log monitoring setting command

   Apply the settings by using the --set option of the paeventadm command.

```
# paeventadm --set
[WARNING]
Do you really want to continue (y/n) y
[INFO] PSM 0082 paeventadm The processing of the configuration file was completed.
```

Information

If a node fails in this application due to some reason such as stoppage of the power supply, the node ID of this node is output to the error list file (paeventadm_*clstname*_failed_list) on the home directory.

```
# paeventadm --set
[WARNING]
Do you really want to continue (y/n)? y
[ERR.] PSM 6305 paeventadm Failed to send files. (/root/paeventadm_clstname_failed_list)
```

clstname: Cluster name

The following is an example of the error list file.

```
0xFFFF0008
0xFFFF0009
```

By specifying the error list file in the -f option of the paeventadm command, you can reexecute it to a node to which you failed to apply the settings.

```
# paeventadm -c clstname -f paeventadm_clstname_failed_list --set
[WARNING]
Do you really want to continue (y/n)? y
[INFO] PSM 0082 paeventadm The processing of the configuration file was completed.
```

clstname: Cluster name

3.6.2.2 Configuring Mellanox HCA Monitoring

You can detect a Catastrophic Error of Mellanox HCA by monitoring a message which is output to /var/log/messages. To use Mellanox HCA on the compute node of PRIMERGY server, perform the following settings.

1. Creating the log monitoring configuration file

   Create paevent.conf, the log monitoring configuration file, in order to perform the log monitoring.
   Place paevent.conf in the following location on the active system management node.

   ```
   /etc/opt/FJSVtcs/paevent.conf
   ```

   The following is an example of describing the log monitoring configuration file.

   ```
   Cluster {                                                                # (*1)
      ClusterName = cluster1                                                # (*2)
      NodeType {                                                            # (*3)
        NodeTypeName = CN                                                   # (*4)
        Event {                                                             # (*5)
          EventName = sample1                                               # (*6)
          ChkString = "kernel: mlx4_core .+: Internal error detected:"      # (*7)
          ExecCmd  = "/bin/touch /var/opt/FJSVtcs/psm/monitor/ib_ca_error" # (*8)
        }
      }
   }
   ```

   (*1) Section defining the settings for each cluster
   (*2) Cluster name
   (*3) Section defining the settings for each node type
   (*4) Node type name
   (*5) Section defining the log monitoring operation
   (*6) Log monitoring definition name
   (*7) Character string of the monitoring target
   (*8) Command to be executed when detecting a message

   Note

   The error message of Mellanox HCA may vary depending on the types of the products. If necessary, change the character string of the monitoring target used for log monitoring, according to the Mellanox product that you use.

2. Executing the log monitoring setting command

   Apply the settings by using the --set option of the paeventadm command.

   ```
   # paeventadm --set
   [WARNING]
   Do you really want to continue (y/n)y
   [INFO] PSM 0082 paeventadm The processing of the configuration file was completed.
   ```

3.6.3 Supporting Creation of a Configuration File

The following section describes the procedure to create a configuration file. This procedure uses the function for supporting creation of a configuration file.

3.6.3.1 Creating a Configuration File

Create a configuration file from its template. To do so, execute the function for supporting creation of a configuration file by specifying a file to be converted.

1. Creating a template file

   Create sample.conf, the file which describes keywords to be converted. As shown below, surround each keyword by double curly brackets "{{ }}". For the available keywords, see "parender command" in the "Job Operation Software Command Reference."

```
clstname: "{{ PX_MYNODE.CLSTNAME }}"
nodeid: "{{ PX_MYNODE.NODEID }}"
nodetype: "{{ PX_MYNODE.NODETYPE }}"
ipaddr: "{{ PX_MYNODE.MNG_NET }}"
bobid: "{{ PX_MYNODE.CNODE_NUM }}"
```

2. Converting the template file

   Execute parender, the command for supporting creation of a configuration file, in order to convert sample.conf. The original file overwrites the converted sample.conf.

```
# parender sample.conf
```

3. Confirming the file

   Confirm that the output file is converted as expected.
   The following is an example of converting sample.conf on a node with:
   - Cluster name: compute
   - Node ID: 0x01010003
   - Node type: CN
   - Management network address: 192.0.2.103
   - Node number inside BoB: 10

```
clstname: "compute"
nodeid: "0x01010003"
nodetype: "CN"
ipaddr: "192.0.2.103"
bobid: "010"
```

3.6.4 Periodical Log Collection

The following section describes the procedure for setting log collection using Logstash and Filebeat.

3.6.4.1 Setting Periodical Log Collection

1. Applying Logstash and Filebeat

   a. Getting the packages

      Logstash and Filebeat packages for log collection are available under Elastic License and Apache License Version 2.0.

      It is your responsibility to review and select a license and download and use either.

      - Elastic License

        https://www.elastic.co/downloads/beats/filebeat
        https://www.elastic.co/downloads/logstash

      - Apache License Version 2.0

        https://www.elastic.co/downloads/beats/filebeat-oss
        https://www.elastic.co/downloads/logstash-oss

   b. Adding the packages

      To the repository, add the packages of Logstash and Filebeat that you use for log collection.
      For details on how to add the packages, see "3.8.2.6 Adding and Deleting a Package."

      If there are the standby system management node and the compute cluster sub management node, it is required to synchronize the repository.
      For details on synchronization of the repository, see "3.8.10.2 Synchronization Procedure."

c. Installing the packages

Install the packages of Logstash and Filebeat that you use for log collection.
Install Filebeat on log collection target nodes, which are the boot I/O node, the compute node (PRIMERGY server), and the ARM architecture node. Install Logstash on any other log collection target nodes as well as log collection nodes.
Logstash requires the java package. This example is as follows.

The following is an example of installing Filebeat on the boot I/O node existing in the compute cluster (whose cluster name is compute).

```
# pmexe -c compute --nodetype BIO "yum clean all; yum install -y filebeat"
[INFO] PSM 5082 pmexe The execution result file was output.(/root/
pmexe_result_20160401.8740)
```

The following is an example of installing Logstash and Java on the system management node, the compute cluster management node, the login node, and the compute cluster sub management node. These nodes exist in the compute cluster (whose cluster name is compute).

```
# pmexe -c compute --nodetype SMM,CCS,CCM,LN "yum clean all; yum install -y logstash java"
[INFO] PSM 5082 pmexe The execution result file was output.(/root/
pmexe_result_20160401.8740)
```

2. Creating the configuration file template

The sample file directory (/etc/opt/FJSVtcs/sample/) contains the samples of the following configuration files regarding periodical log collection. Use any of them in accordance with a target node.

Table 3.54 Sample File List

File Name	Target Node	Description
filebeat_BIO.yml	Boot I/O node	Configuration file to transfer the logs of all nodes inside BoB to the relay node
filebeat_CN.yml	Compute node (PRIMERGY server)	Configuration file to transfer the log of the compute node to the relay node
filebeat_arm.yml	Node of the ARM architecture	Configuration file to transfer the log of the node of the ARM architecture to the relay node
logstash_relay.conf	Compute cluster sub management node Compute cluster management node	Configuration file to transfer the log from Filebeat to the relay node and transfer the log of the local node to the log collection node
logstash_receive.conf	Log collection node	Configuration file to collect the logs from each node
logstash_relay_receive.conf	Node serving as all of system management node, compute cluster management node, and login node	Configuration file to collect the logs from each node
logstash_send.conf	Nodes other than above	Configuration file to transfer the log of the local node to the log collection node

Some system configurations require you to edit a sample file. The following table shows the edit items.

Table 3.55 Edited Parts of Sample Files

Configuration File	Edited Part	Description
filebeat_BIO.yml filebeat_CN.yml	Destination relay (Logstash) node	Changing the relay node requires a modification.
filebeat_arm.yml	Destination relay (Logstash) node (output.logstash.hosts)	Changing the relay node requires a modification.

Configuration File	Edited Part	Description
logstash_relay.conf logstash_send.conf	Destination collection node	Changing the destination requires a modification.

3. Editing the Filebeat configuration file template (Changing the destination relay node)

   By default, it describes logs so as to transmit them under the following conditions:

   - For nodes that belong to a node group, it transmits logs by taking the compute cluster sub management node as the relay node.
   - For other nodes, it transmits logs by taking the compute cluster management node as the relay node.

```
#----------------------------- Logstash output --------------------------------
output.logstash:
  # The Logstash hosts
{% if "CCS" in PX_SAMENGRP %}
{% if PX_SAMENGRP.CCS|length > 1 %}
  hosts: ["{{ PX_SAMENGRP.CCS[0].MNG_NET }}:5044", "{{ PX_SAMENGRP.CCS[1].MNG_NET }}:5044"]
{% elif PX_SAMENGRP.CCS|length > 0 %}
  hosts: ["{{ PX_SAMENGRP.CCS[0].MNG_NET }}:5044"]
{% endif %}
{% elif PX_SAMENGRP.CCM|length > 1 %}
  hosts: ["{{ PX_SAMENGRP.CCM[0].MNG_NET }}:5044", "{{ PX_SAMENGRP.CCM[1].MNG_NET }}:5044"]
{% elif PX_SAMENGRP.CCM|length > 0 %}
  hosts: ["{{ PX_SAMENGRP.CCM[0].MNG_NET }}:5044"]
{% endif %}
...
```

   To transmit logs to another node (whose IP address is 192.168.0.200), describe it as follows.

```
#----------------------------- Logstash output --------------------------------
output.logstash:
  # The Logstash hosts
  hosts: ["192.168.0.200:5044"]
```

   For other settings, see the document of Filebeat.

4. Editing the Logstash configuration file template (Changing the destination relay node)

   By default, it describes the setting for collecting logs in the system management node.

```
output {
  tcp {
    host => ["{{ PX_SAMECLST.SMM[0].MNG_NET }}"]
    port => 15044
    codec => json_lines
  }
}
```

   To make a reception setting of Logstash for another node (whose IP address is 192.168.0.200) and transmit logs to this node, describe it as follows.

```
output {
  tcp {
    host => ["192.168.0.200"]
    port => 15044
    codec => json_lines
  }
}
```

   For other settings, see the document of Logstash.

5. Distributing the configuration file template

   The system administrator is to distribute the configuration file template by using pmscatter, the collective file delivery command, on the system management node. This example is as follows.

- An example of distributing the filebeat_BIO.yml file to the boot I/O node existing in the compute cluster (whose cluster name is compute)

```
# pmscatter -c compute --nodetype BIO /etc/opt/FJSVtcs/sample/filebeat_BIO.yml /etc/filebeat/
filebeat.yml
```

- An example of distributing the logstash_relay.conf file to the compute cluster management node and compute cluster sub management node existing in the compute cluster (whose cluster name is compute)

```
# pmscatter -c compute --nodetype CCM,CCS \
/etc/opt/FJSVtcs/sample/logstash_relay.conf /etc/logstash/conf.d/logstash.conf
```

- An example of distributing the logstash_send.conf file to the login node existing in the compute cluster (whose cluster name is compute)

```
# pmscatter -c compute --nodetype LN \
/etc/opt/FJSVtcs/sample/logstash_send.conf /etc/logstash/conf.d/logstash.conf
```

- An example of distributing the logstash_receive.conf file to the log collection node

```
# pmscatter -c compute -n nodeid  \
/etc/opt/FJSVtcs/sample/logstash_receive.conf /etc/logstash/conf.d/logstash.conf
```

nodeid: Node ID of the log collection node

6. Creating the configuration file from the configuration file template

From the system management node, use pmexe, the collective execution command to execute parender, the command for supporting creation of a configuration file on each node. This expands the keyword part in the configuration file, making it an official configuration file. This example is as follows.

- An example of expanding the filebeat.yml file to the boot I/O node existing in the compute cluster (whose cluster name is compute)

```
# pmexe -c compute --nodetype BIO "parender /etc/filebeat/filebeat.yml"
[INFO] PSM 5082 pmexe The execution result file was output.(/root/pmexe_result_20160401.8740)
```

- An example of expanding the logstash.conf file to the compute cluster management node, the compute cluster sub management node, and the login node existing in the compute cluster (whose cluster name is compute)

```
# pmexe -c compute --nodetype CCM,CCS,LN "parender /etc/logstash/conf.d/logstash.conf"
[INFO] PSM 5082 pmexe The execution result file was output.(/root/pmexe_result_20160401.8740)
```

- An example of expanding the logstash.conf file on the log collection node

```
# pmexe -c compute -n nodeid "parender -c compute /etc/logstash/conf.d/logstash.conf"
[INFO] PSM 5082 pmexe The execution result file was output.(/root/pmexe_result_20160401.8740)
```

7. Setting for starting the logstash service

To start the logstash service, edit the configuration file, and then execute the command. First, edit the configuration file (/etc/logstash/startup.options) in order to start a daemon with root privileges. By default, the execution user group of the daemon is logstash as shown below.

```
    # user and group id to be invoked as
LS_USER=logstash
LS_GROUP=logstash
```

In this situation, you cannot collect logs such as /var/log/messages, which can only be read with root priviledges. It is therefore recommended to rewrite it into root as follows.

```
    # user and group id to be invoked as
LS_USER=root
LS_GROUP=root
```

To edit the configuration file for multiple nodes, the system administrator is to distribute the configuration file by using pmscatter, the collective file delivery command, on the system management node. This example is as follows.

- An example of distributing the startup.options file to the compute cluster management node, the compute cluster sub management node, and the login node existing in the compute cluster (whose cluster name is compute). This example is for a file edited on the system management node.

```
# pmscatter -c compute --nodetype CCM,CCS,LN /etc/logstash/startup.options /etc/logstash/
startup.options
```

Then, use pmexe, the collective execution command to perform the service starting setting (/usr/share/logstash/bin/system-install) from the system management node. This example is as follows.

- An example of starting the logstash service on the system management node

```
# /usr/share/logstash/bin/system-install
Successfully created system startup script for Logstash
```

- An example of making the logstash service setting to the compute cluster management node, the compute cluster sub management node, and the login node existing in the compute cluster (whose cluster name is compute)

```
# pmexe -c compute --nodetype CCM,CCS,LN "/usr/share/logstash/bin/system-install"
[INFO] PSM 5082 pmexe The execution result file was output.(/root/pmexe_result_20160401.8740)
```

8. Starting the service

From the system management node, use pmexe, the collective execution command to start the log collection service and automatic service startup settings of each node. This example is as follows.

- An example of starting the logstash service on the system management node

```
# systemctl start logstash
# systemctl enable logstash
```

- An example of starting the logstash service and automatic service startup settings for the compute cluster management node, the compute cluster sub management node, and the login node existing in the compute cluster (whose cluster name is compute)

```
# pmexe -c compute --nodetype CCM,CCS,LN "systemctl start logstash"
[INFO] PSM 5082 pmexe The execution result file was output.(/root/
pmexe_result_201905071740.35274)
# pmexe -c compute --nodetype CCM,CCS,LN "systemctl enable logstash"
[INFO] PSM 5082 pmexe The execution result file was output.(/root/
pmexe_result_201905071740.35241)
```

- An example of starting the filebeat service for the boot I/O node existing in the compute cluster (whose cluster name is compute)

```
# pmexe -c compute --nodetype BIO "systemctl start filebeat"
[INFO] PSM 5082 pmexe The execution result file was output.(/root/
pmexe_result_201905071741.35495)
# pmexe -c compute --nodetype BIO "systemctl enable filebeat"
[INFO] PSM 5082 pmexe The execution result file was output.(/root/
pmexe_result_201905071741.35475)
```

9. Checking collected logs

By default, the following directory of the system management node stores collected logs.

```
/var/log/logstash/YYYYMMDD/clstname/nodegid/bootgid/format
```

YYYYMMDD: Date of the log
clstname: Cluster name
nodegid: Node group ID
bootgid: Boot group ID
format: Log format name (either of syslog, FJSVtcs, or FJSVfefs)

The following is an example of the storage file path of the log format FJSVtcs, which has the node ID 0x01FF0001 on April 2nd, 2018.

```
/var/log/logstash/20180402/compute/0x01/0xFF/FJSVtcs
```

Also, a collected log file is described in the following format.

```
nodeid filename message
```

nodeid: Node ID
filename: Log file name
message: Log message

The following is an example of collecting a log message that is output to the log file pxinstd.log with the node ID 0x01FF0001.

```
0x01FF0001 pxinstd.log 21:19:25 2018-04-06 15:42:22:048804 [INFO] [INST] 9999 - 3093 starting
pxinstd
```

3.7 Details of the Software Environment Check Function

3.7.1 Check Items of the Software Environment Check Function

The pachecker command enables you to check the following items:

- Checking the application state of the package of the Job Operation Software
- Confirming the setting reflection of the Job Operation Software

Note

The pachecker command references rpmdb internally. While the pachecker command is running, do not run any commands that manipulate rpmdb, such as the rpm or dnf commands.

Information

Suppose that a system uses the Technical Computing Suite File System and the Technical Computing Suite Development Studio. Then, when checking the application state of the package of the Job Operation Software, check the application of the package for them as well.

Thus, if a system uses the Technical Computing Suite File System and the Technical Computing Suite Development Studio, checking the application state of the package of the Job Operation Software includes checking the application state of the package of:
- Technical Computing Suite File System, and
- Technical Computing Suite Development Studio

When checking the application state of the package of the Job Operation Software, check whether the package included in the package kit is applied to each node type.

When confirming the setting reflection of the Job Operation Software, confirm the following items.

Table 3.56 Confirmation Contents of Setting Reflection of the Job Operation Software

Function Name	Object Command	Confirmation Contents of Setting Reflection
System management function	pxsystemadm	Confirming that the node configuration information of each node is correct
System management function	papwradm	Confirming that each node reflects the system control function setting

Function Name	Object Command	Confirmation Contents of Setting Reflection
System management function	pamonitoradm	Confirming that each node reflects the system monitoring setting
System management function	paserviceadm	Confirming that each node reflects the user-defined service setting
System management function	pamopluginadm	Confirming that each node reflects the monitoring plugin setting
System management function	paeventadm	Confirming that each node reflects the log monitoring condition setting
Job management function	papjmadm	Confirming that each node reflects the job operation setting of an entire cluster
Job management function	pmpjmadm	Confirming that each node reflects the job operation setting for each resource unit
Job management function	papjmstatsadm	Confirming that each node reflects the statistical information setting
Power management function	papwrmgradm	Confirming that each node reflects the power management function setting

Other than those above, you can describe any item to be checked in the definition file (hereinafter called as the check definition file), place it in a designated location, and then execute the pachecker command. This allows you to check the state of any package and settings.

You can check the following types:

- Checking the application state of the package
- Checking whether the file exists
- Checking the comparison result of the contents of the file
- Checking the execution result of any command

3.7.2 Checking the Software Environment

This section describes the procedure for checking the software environment.

1. Creating the check definition file

   When the system administrator sets any check items, he/she must create the check definition file.
   Describe the check definition file in the yaml format.

   See

   For details on the format and items of the check definition file, see "pachecker command" in the "Job Operation Software Command Reference."

   In checking the software environment, you can perform, for example, the following checks:

   - Confirming that the package python is installed on the node of the model FT
   - Confirming that the file /etc/yum.repos.d/bio1.repo exists in the boot I/O node
   - Confirming that the file /etc/modprobe.d/libmlx4.conf, in which the permission is 644 and the owner's UID is 0 (root), exists in the compute cluster management node and the login node
   - Confirming that the file /etc/hosts with all node types is the same as the node ID 0x00000001
   - Confirming that the file /etc/sysctl.conf on the boot I/O node is the same as the node ID 0x01010001
   - Confirming that the return value is 0 when executing the command line "/usr/sbin/ip a |grep ib0 |grep UP" on the global I/O node

- Confirming that there is no difference in the output when executing the command line "cat /proc/cpuinfo |grep MHz" on the compute node

The following is an example of creating the check definition file to check the above items.

```
Package:                                              (*1)
-                                                     (*2)
   Package: python                                    (*3)
   Model: FT                                          (*4)
   Action: "Install python package"                   (*5)

FileExist:                                            (*6)
-
   Name: bio1.repo                                    (*7)
   Path: /etc/yum.repos.d/bio1.repo                   (*8)
   NodeType: BIO                                      (*9)

-
   Name: libmlx4.conf
   Path: /etc/modprobe.d/libmlx4.conf
   NodeType: CCM,LN
   Permission: 644                                    (*10)
   User: 0                                            (*11)

FileMatch:                                            (*12)
-
   Name: hosts
   SrcNodeID: 0x00000001                              (*13)
   SrcPath: /etc/hosts                                (*14)
   DstPath: /etc/hosts                                (*15)
   NodeType: ALL
   Action: "pmscatter /etc/hosts"

-
   Name: sysctl.conf
   SrcNodeID: 0x01010001
   SrcPath: /etc/sysctl.conf
   DstPath: /etc/sysctl.conf
   NodeType: BIO

Command:                                              (*16)
-
   Name: ibup
   Type: ReturnCode                                   (*17)
   ExecCmd: "/usr/sbin/ip a |grep ib0 |grep UP"       (*18)
   Expected: 0                                        (*19)
   NodeType: GIO

-
   Name: cpuMHz
   Type: Diff
   ExecCmd: "cat /proc/cpuinfo |grep MHz"
   NodeType: CN
```

(*1) The section to check the difference in versions of the package.
(*2) Write a hyphen at the beginning of each check item.
(*3) The name of the package to be checked.
(*4) The model to be checked.
(*5) The message output together with the execution result when the execution result determines that it is FAILED.
(*6) The section to check whether the file exists.
(*7) The name of a check item.
(*8) The absolute path of an object file.
(*9) The node type to be checked. (When specifying "ALL," all node types are to be checked.)

(*10) The permission of an object file.
(*11) UID of the owner of an object file.
(*12) The section to check the difference in the comparison result of the contents of the file.
(*13) The node ID at the comparison source.
(*14) The absolute path of the file at the comparison source.
(*15) The absolute path of the file at the comparison destination.
(*16) The section to check the execution result of any command.
(*17) The check method for any command (in which ReturnCode means a return value and Diff means a difference).
(*18) The command to be executed, or the script created by the administrator.
(*19) The expected value of the return value of the command to be executed.

2. Placing the check definition file

   To enable the check definition file created in step 1, place it on /etc/opt/FJSVtcs/plugin/psm/check/ or arbitrary directory of the active system management node.
   The following is an example of copying the check definition file check1, which is created under the directory /tmp, on /etc/opt/FJSVtcs/plugin/psm/check/.

```
# cp /tmp/check1 /etc/opt/FJSVtcs/plugin/psm/check/
```

   Note

   If you want to use check definition files that are placed in arbitrary directory, you need to specify the directory using the --check option.

3. Checking the software environment

   Execute the pachecker command to check the setting.
   The following is an example of checking all nodes in a cluster called cluster1.

```
# /usr/sbin/pachecker -c cluster1
[INFO] PSM 4181 pachecker Result of the check items are OK.
[INFO] PSM 4182 pachecker The execution result file was output.
/root/pachecker_pkg_cluster1_result_20180102001122
/root/pachecker_conf_cluster1_result_20180102001122
/root/pachecker_plugin_check1_cluster1_result_20180102001122
```

4. Checking the check results

   Check the content of the output file to determine whether you can incorporate the node to be checked into operation.
   For details on the content of the files output by the pachecker command, see "3.7.3 Checking the Check Result."

3.7.3 Checking the Check Result

3.7.3.1 Output File

The pachecker command outputs the execution result file and the error node list.
The execution result file outputs the execution result for each check type when executing the pachecker command. The system administrator is to refer to this file in order to check the execution result.

The following describes the format and the output unit of the execution result file.

The pachecker command outputs the execution result file in units of "checking the package application state of the Job Operation Software," "confirming the setting reflection of the Job Operation Software," and "any check items defined by the system administrator." The output paths of the execution result file are as follows:

1. Check result of the package application state of the Job Operation Software

```
dirname/pachecker_pkg_clstname_result_YYYYMMDDhhmmss
```

dirname: Directory specified in the -d option
clstname: Cluster name
YYYYMMDDhhmmss: Command execution date and time

2. Confirmation result of the setting reflection of the Job Operation Software

   ```
   dirname/pachecker_conf_clstname_result_YYYYMMDDhhmmss
   ```

   dirname: Directory specified in the -d option
   clstname: Cluster name
   YYYYMMDDhhmmss: Command execution date and time

3. Check result of any check items defined by the system administrator

   ```
   dirname/pachecker_plugin_checkfilename_clstname_result_YYYYMMDDhhmmss
   ```

   dirname: Directory specified in the -d option
   clstname: Cluster name
   YYYYMMDDhhmmss: Command execution date and time

To the execution result file, "Check result," "Message," "Target node," and "Execution result" are output for each "Check item." The check items described by the system administrator are output after classifying them for each "Check method."

The following describes the output contents and the items of the execution result file.

```
[ CLST: clstname ]
[ Check type ]
[ Check method ]

**************************************************
TARGET: Check item  RESULT: Check result        ACTION: Message
==================================================
nodeid-nodeid
==================================================
Execution result
==================================================
...

**************************************************
TARGET: Check item  RESULT: Check result        ACTION: Message
==================================================
nodeid-nodeid
==================================================
Execution result
```

Table 3.57 Output Contents of the Execution Result File

Item	Description
Check type	- JOB OPERATION SOFTWARE PACKAGE (Package of the Job Operation Software) - JOB OPERATION SOFTWARE SETTING (Setting of the Job Operation Software) - PLUGIN (Any check items defined by the system administrator)
Check method	Check method for PLUGIN (Any check items defined by the system administrator) - Package (Checking the application state of the package) - FileExist (Checking whether the file exists) - FileMatch (Checking the comparison result of the contents of the file) - Command (Checking the execution result of any command)
Check item	The package name and the setting name of the Job Operation Software, and the names of check items described on the check definition file
Check Result	- SAME (meaning that no difference exists)

Item	Description
	- DIFF (meaning that a difference exists) - OK (meaning that it matches a specified expected value) - FAILED (meaning that it does not match a specified expected value)
Execution result	Execution result of checking for check items
Message	Method of coping with a check item whose execution result shows FAILED

Information

The error node list is a file that outputs the node ID for which the execution result of the pachecker command shows FAILED. The format of the error node list makes it possible to directly specify the list in the -f option of the batch operation command. The system administrator can use this list to perform efficient resetting by executing the pmexe command or the like.

The error node list is output for each check item.
The output paths of the error node list for each check item are as follows:

- Check result of the package application state of the Job Operation Software

```
dirname/pachecker_pkg_clstname_error_item_YYYYMMDDhhmmss
```

dirname: Directory specified in the -d option
clstname: Cluster name
item: Name of a check item
YYYYMMDDhhmmss: Command execution date and time

- Confirmation result of the setting reflection of the Job Operation Software

```
dirname/pachecker_conf_clstname_error_item_YYYYMMDDhhmmss
```

dirname: Directory specified in the -d option
clstname: Cluster name
item: Name of a check item
YYYYMMDDhhmmss: Command execution date and time

- Check result of any check items defined by the system administrator

```
dirname/pachecker_plugin_checkfilename_clstname_error_item_YYYYMMDDhhmmss
```

dirname: Directory specified in the -d option
checkfilename: Check definition file name
clstname: Cluster name
item: Name of a check item
YYYYMMDDhhmmss: Command execution date and time

To the error node list, a "node ID" and "reason for FAILED" are output for each "Check item."

The following describes the output contents and the items of the error node list.

```
# Check item
nodeid  # reason
nodeid  # reason
```

Table 3.58 Output Contents of the Error Node List

Item	Description
nodeid	Node ID of a node that becomes FAILED
reason	Reason for FAILED

3.7.3.2 Checking the Package Application State of the Job Operation Software

Check the package application state of the Job Operation Software by confirming that the difference in versions of the package of the Job Operation Software indicates an intended version in the target range.

The target range does not always contain one version, for a maintenance by rolling update may mix different versions. "SAME" is output to the execution result file when versions are the same in the target range, and "DIFF" is output to the file when versions are different in such range. Therefore, the system administrator must check this information to determine whether the package is correctly applied.

If the check results of the package application state of the Job Operation Software becomes "FAILED," the message of the coping method is output as "ACTION." The system administrator must refer to this coping method to perform the resetting.

The following table shows the output information and the meanings of the execution result file.

Table 3.59 Output Information of the Execution Result File (Checking the Package Application State of the Job Operation Software)

Check Result	Output Content of Execution Result	Meaning
SAME	Package name, version, release, and architecture	Check results are the same in specified target range.
DIFF	Package name, version, release, and architecture	Check results are different in specified target range.
FAILED	(NotInstalled)	Not installed
FAILED	(CannotCheck)	Checking is unavailable due to some reason such as the impossibility of connection.

The following is an output example of the execution result file.

```
[ CLST: cluster1 ]

[ JOB OPERATION SOFTWARE PACKAGE ]
**************************************************
TARGET:FJSVpxpsm.x86_64   RESULT:SAME    ACTION:-    # (*1)
==================================================
0x00000001,0x01FF0001,0xFFFF0001
==================================================
FJSVpxpsm-2.0.1-02.x86_64

**************************************************
TARGET:FJSVpxpsm.aarch64  RESULT:DIFF    ACTION:-    # (*2)
==================================================
0x01010001-0x01010004,0x01010006-0x01010010
==================================================
FJSVpxpsm-2.0.1-02.aarch64
==================================================
0x01010005
==================================================
FJSVpxpsm-2.0.1-01.aarch64
```

(*1) This displays the version of the FJSVpxpsm.x86_64 package, which matches the node of the model PG in cluster1.
(*2) This displays the version of the FJSVpxpsm.aarch64 package and the version of the node of the model FT in cluster1, which do not match each other.

3.7.3.3 Confirming the Setting Reflection of the Job Operation Software

To confirm the setting reflection of the Job Operation Software, confirm whether the target node reflects the setting information of the Job Operation Software. "OK" is output if the node reflects the setting. If not, "FAILED" is output.

If the check results of the setting reflection of the Job Operation Software shows "FAILED," the message of the coping method is output as "ACTION." The system administrator must refer to this coping method to perform the resetting.

The following table shows the output information and the meanings of the execution result file.

Table 3.60 Output Information of the Execution Result File (Confirming the Setting Reflection of the Job Operation Software)

Check Result	Output Content of Execution Result	Meaning
OK	SAME	The node reflects the setting correctly.
FAILED	DIFF	The node does not reflect the setting correctly.
FAILED	(CannotCheck)	Checking is unavailable due to some reason such as the impossibility of connection.

The following is an output example of the execution result file.

```
[ CLST: cluster1 ]

[ JOB OPERATION SOFTWARE SETTING ]
**************************************************
TARGET:pamonitoradm    RESULT:FAILED   ACTION:Execute pamonitoradm command with --set option    # (*1)
==================================================
0x00000001-0x00000002,0xFFFF0001-0xFFFF0002,0x01010001
==================================================
SAME
==================================================
0x01010002-0x01010004
==================================================
DIFF
==================================================
0x01010005
==================================================
(CannotCheck)
```

(*1) Checking is unavailable due to some reason such as:
- 0x01010002-0x01010004 in cluster1 has a mismatch of the setting information of the pamonitoradm command.
- 0x01010005 is not connectable.

3.7.3.4 Checking the Application State of the Package

Regarding checking the application state of the package described in the check definition file, the following table shows the output information and the meanings of the execution result file.

Table 3.61 Output Information of the Execution Result File (Checking the Application State of the Package)

Check Result	Output Content of Execution Result	Meaning
SAME	Package name, version, release, and architecture	Check results are the same in specified target range.
DIFF	Package name, version, release, and architecture	Check results are different in specified target range.
FAILED	(NotInstalled)	Not installed
FAILED	(CannotCheck)	Checking is unavailable due to some reason such as the impossibility of connection.

The following is an output example of the execution result file.

```
[ CLST: cluster1 ]

[ PLUGIN ]
[ PACKAGE ]

**************************************************
TARGET:rsh               RESULT:SAME     ACTION:-                         # (*1)
==================================================
0x01010001-0x01010005
==================================================
rsh-0.17-76.el7_1.1.x86_64.aarch64

**************************************************
TARGET:rsh-server        RESULT:DIFF     ACTION:-                         # (*2)
==================================================
0x01010001
==================================================
rsh-server-0.17-76.el7_1.1.x86_64
==================================================
0x01010002-0x01010005
==================================================
rsh-server-0.18-76.el7_1.1.x86_64

**************************************************
TARGET:python           RESULT:FAILED    ACTION:Install python package    # (*3)
==================================================
0x01010001
==================================================
python-0.17-76.el7_1.1.aarch64
==================================================
0x01010002-0x01010005
==================================================
(NotInstalled)
```

(*1) This displays the version of rsh, which is installed on all boot I/O nodes in cluster1.
(*2) This displays each of the versions of the nodes that exist in the model PG in cluster1 and have different rsh-server versions.
(*3) The model FT in cluster1 has a node that has not had python installed.

3.7.3.5 Checking whether the File Exists

Regarding checking whether the file described in the check definition file exists or not, the following table shows the output information and the meanings of the execution result file.

Table 3.62 Output Information of the Execution Result File (Checking whether the File Exists)

Check Result	Output Content of Execution Result	Meaning
OK	Detailed information on the file (Equivalent to the result of the ls -n command)	The specified file exists.
FAILED	Detailed information on the file (Equivalent to the result of the ls -n command)	There is a file whose permission, owner, and ownership group are different.
FAILED	(NotExist)	The specified file does not exist.
FAILED	(CannotCheck)	Checking is unavailable due to some reason such as the impossibility of connection.

The following is an output example of the execution result file.

```
[ CLST: cluster1 ]

[ PLUGIN ]
[ FILE EXIST ]

**************************************************
TARGET:bio1.repo         RESULT:OK       ACTION:-    # (*1)
==================================================
0x01010001-0x01010010
==================================================
rw-r--r-- 1    0    0 250 Jul 15 11:42 2015 /etc/yum.repos.d/bio1.repo

**************************************************
TARGET:libmlx4.conf      RESULT:FAILED    ACTION:-   # (*2)
==================================================
0xFFFF0001-0xFFFF0002
==================================================
rw-r--r-- 1    0    0 250 Jul 15 11:42 2015 /etc/modprobe.d/libmlx4.conf
==================================================
0xFFFF0006
==================================================
rw------- 1    0    0 250 Jul 15 11:42 2015 /etc/modprobe.d/libmlx4.conf
==================================================
0xFFFF0007,0xFFFF0008
==================================================
(NotExist)
==================================================
0xFFFF0009
==================================================
(CannotCheck)
```

(*1) This displays the information of a file which exists on all boot I/O nodes in cluster1.
(*2) This displays the information of a file regarding the following existing nodes:
- Node on which the permission of the file is different (0xFFFF0006)
- Nodes on which the file does not exist (0xFFFF0007 and 0xFFFF0008)
- Node that cannot be checked (0xFFFF0009)

3.7.3.6 Checking the Comparison Result of the Contents of the File

Regarding comparison check of the contents of the file described in the check definition file, the following table shows the output information and the meanings of the execution result file.

Table 3.63 Output Information of the Execution Result File (Comparison Check of the Contents of the File)

Check Result	Output Content of Execution Result	Meaning
OK	SAME	This is the same as the specified file.
FAILED	Difference information appears. For a binary file, the checksum value appears.	This is different from the specified file.
FAILED	(NotExist)	The specified file does not exist.
FAILED	(CannotCheck)	Checking is unavailable due to some reason such as the impossibility of connection.

The following is an output example of the execution result file.

```
[ CLST: cluster1 ]

[ PLUGIN ]
[ FILE MATCH ]

**************************************************
TARGET:hosts   RESULT:FAILED    ACTION:pmscatter /etc/hosts    # (*1)
==================================================
0x00000001-0xFFFF0005
==================================================
SAME
==================================================
- 0x01010001
+ 0x01010002-0x01010010
==================================================
@@ -3,6 +3,6 @@
```

```
- 192.0.2.1    example1
+ 192.0.2.2    example2
192.0.2.2    example3
192.0.2.2    example4

**************************************************
TARGET:sysctl.conf   RESULT:FAILED    ACTION:-                 # (*2)
==================================================
- 0x01010001
+ 0x01010002-0x01010009
==================================================
@@ -3,5 +3,5 @@
- kernel.sem = 250 400000 32 2000
+ kernel.sem = 10 200000 20 1000
==================================================
0x01010010
==================================================
(CannotCheck)
```

(*1) This displays difference information because there is a difference in the contents of the file.
(*2) This displays difference information because there is a difference in the contents of the file.

3.7.3.7 Checking the Execution Result of Any Command

This section describes checking of the execution result of any command, which is described in the check definition file.

The following table shows the output information and the meanings of the execution result file when specifying ReturnCode in Type.

Table 3.64 Output Information of the Execution Result File (ReturnCode)

Check Result	Output Content of Execution Result	Meaning
OK	ReturnCode=*Return value*	This is the same as the specified return value.
FAILED	ReturnCode=*Return value*	This is different from the specified return value.
FAILED	(NoCommand)	The specified command does not exist.
FAILED	(CannotCheck)	Checking is unavailable due to some reason such as the impossibility of connection.

The following table shows the output information and the meanings of the execution result file when specifying Diff in Type.

Table 3.65 Output Information of the Execution Result File (Diff)

Check Result	Output Content of Execution Result	Meaning
OK	Command output information	The output of the execution result has no difference.
FAILED	Difference information of command output	The output of the execution result has a difference.
FAILED	(NoCommand)	The specified command does not exist.
FAILED	(CannotCheck)	Checking is unavailable due to some reason such as the impossibility of connection.

The following is an output example of the execution result file.

```
[ CLST: cluster1 ]

[ PLUGIN ]
[ COMMAND ]

**************************************************
TARGET:ibup              RESULT:FAILED    ACTION:-    # (*1)
==================================================
0x01010001
==================================================
ReturnCode=0
==================================================
0x01010002-0x01010010
==================================================
ReturnCode=1

**************************************************
TARGET:cpuMHz           RESULT:FAILED    ACTION:-    # (*2)
==================================================
- 0x01010001-0x01010007
+ 0x01010008-0x01010010
==================================================
@@ -47,6 +47,6 @@
cpu MHz                : 1596.000
cpu MHz                : 1596.000
- cpu MHz                : 1596.000
- cpu MHz                : 1596.000
- cpu MHz                : 1596.000
- cpu MHz                : 1596.000
+ cpu MHz                : 2596.000
+ cpu MHz                : 2596.000
+ cpu MHz                : 2596.000
+ cpu MHz                : 2596.000
```

(*1) This displays the information of a return value which is not 0 on existing nodes (0x01010002-0x01010010).
(*2) This displays the information on the difference caused by the mismatch of the execution results in the range.

Information

The default locale (LANG=C) executes any command.

3.8 Details of the Installation Function

3.8.1 Initial Settings of the Installation Function

This section describes the details on the initial settings of the installation function.

3.8.1.1 Name and Placement Location of the Configuration File

In the following location, it is necessary to place the configuration file to be used for the initial settings of the installation function. (This file is hereinafter called the environment configuration file of the installation function.)

```
/etc/opt/FJSVtcs/pxinst.conf
```

Information

When introducing the Job Operation Software, the pxinst.conf file does not exist. Copy the sample file (/etc/opt/FJSVtcs/sample/pxinst.conf), and then place it on the path described above.

3.8.1.2 Format of the Configuration File

The format of the configuration file is yaml.

3.8.1.3 Items to be Set

The following table shows the items to be set in the environment configuration file of the installation function. These settings are variable according to the environment. For the items that do not require the system administrator to input manually, the sample file has already set default values to all of these items. This enables you to operate them only with minimum editing.

Table 3.66 Items of the Configuration File

Setting Item	Meaning	Value	Description	Default Value
repodir	Uses absolute path to specify the path of a directory to place a repository.	Any path	Required	/var/repos
tftpdir	Uses absolute path to specify the path of a directory to store the settings and materials of the TFTP server.	Any path	Required	/tftpboot/pxinst
provision_nic	Specifies a device name assigned to the provision network which the DHCP server uses.	Any device name Example: eno1	Required	None
ntp_servers	Specifies an IP address or a host name of the NTP server to which the installer node refers.	Dotted IP address format or host name. If two or more of them exist, specify them by separating them with commas.	Required	None
update_hosts	Sets the automatic update for /etc/hosts. This specifies whether or not to edit the /etc/hosts entry automatically when registering, updating, or deleting node information. Specify False (default value) when cooperating with an external server such as LDAP or managing this setting manually.	True: Enables automatic update. False: Disables automatic update.	Required	false

Table 3.67 Setting Items for Installation Parameters

Setting Item	Meaning	Value	Description	Default Value
lang	Default system language	Language type defined by /Usr/share/system-config-language/locale-list (system-config-language package) Example: en_US.UTF-8 (English) ja_JP.UTF-8 (Japanese)	Unavailable	C.utf8
timezone	Time zone of the system	Specify any of the time zones described in the /usr/share/zoneinfo directory. (You can also use the timedatectl list-timezones command to display the time zones.) Example: Asia/Tokyo	Unavailable	Asia/Tokyo

Setting Item	Meaning	Value	Description	Default Value
keyboard	Keyboard layout	Keyboard layout Example: us	Unavailable	jp106
skipx	X WindowSystem is not configured.	true: Does not configure X WindowSystem. faise: Configures X WindowSystem.	Unavailable	true
passalgo	Hash algorithm for password encryption Specify the same algorithm as the installer node.	Example: md5, sha256, sha512, etc.	Unavailable	sha512

Note

- The skipx false setting is only available for PkgProvision.
- The passalgo setting is not reflected during RHEL8 OS installation. RHEL8 applies sha512.

The following is an example of the configuration file.

```
# Install Function Settings
#------------------------#
# system settings
#------------------------#
# specify the directory path to create RPM repositorys.
# You need to specify an absolute path.
repodir: '/repos'

# Specify the directory for the setting of the tftp server.
# You need to specify an absolute path.
tftpdir: '/tftpboot/pxinst'

# Specify the network interface name for provision network.
provision_nic: 'eno1'

# specify the IP address or host name to be used the Network Time Protcol servers.
# Comma-separated in the case of multiple.
# e.g. xxx.xxx.xxx.xxx,yyy.yyy.yyy.yyy
ntp_servers: 'xxx.xxx.xxx.xxx'

#------------------------#
# install parameters
#------------------------#
# SystemLanguage
lang: C

# SystemTimezone
timezone: 'Asia/Tokyo'

# Keyboard Layout
keyboard: 'jp106'

# X Window System
# true: Not set of the X Window System
# false: To set the X Window System
skipx: true

# Password hash algorithm
# e.g. sha512 md5
passalgo: 'sha512'
```

3.8.1.4 Configure Procedure

This section describes the procedure for the initial configurations of the installation function.

1. Copying the sample file

   Copy the sample file on the arrangement path of the configuration file.

   ```
   # cp /etc/opt/FJSVtcs/sample/pxinst.conf /etc/opt/FJSVtcs/pxinst.conf
   ```

2. Setting the items

   For the items that are necessary to be changed according to the environment, edit the configuration file.

   ```
   # vi /etc/opt/FJSVtcs/pxinst.conf
   ```

3. Initializing the installation function

   To carry out the initial configuring, execute the command for initialization.

   Note

   If you execute pxinst init all or pxinst init ssh, you will not be able to execute remote commands on all nodes that you installed using the installation function.

   ```
   # pxinst init all
   hcp:    Configuring dhcp
   dhcp:      Creating /etc/dhcp/dhcpd.conf  [OK]
   [INFO] INST 0801 pxinst Created a configuration backup.(/tmp/pxinst/dhcpd.conf_20180720105644)
   dhcp:      systemctl enable dhcpd         [OK]
   dhcp:      systemctl restart dhcpd        [OK]
   tftp:    Configuring tftp
   tftp:      Creating /etc/xinetd.d/tftp    [OK]
   [INFO] INST 0801 pxinst Created a configuration backup.(/tmp/pxinst/tftp_20180720105644)
   tftp:      systemctl enable xinetd        [OK]
   tftp:      systemctl restart xinetd       [OK]
   ntp:     Configuring ntp
   ntp:       Creating /etc/chrony.conf      [OK]
   [INFO] INST 0801 pxinst Created a configuration backup.(/tmp/pxinst/chrony.conf_20180720105644)
   ntp:       systemctl enable chronyd       [OK]
   ntp:       systemctl restart chronyd      [OK]
   http:    Configuring http
   http:      Creating /etc/httpd/conf.d/pxinst.conf [OK]
   [INFO] INST 0801 pxinst Created a configuration backup.(/tmp/pxinst/pxinst.conf_20180720105644)
   [INFO] INST 0801 pxinst Created a configuration backup.(/tmp/pxinst/00-mpm.conf_20180720105644)
   http:      systemctl enable httpd         [OK]
   http:      systemctl restart httpd        [OK]
   ssh:     Generating ssh keys
   ssh:       ssh-keygen                     [OK]
   inst:    Configuring pxinst
   inst:      systemctl enable pxinst_virtip [OK]
   inst:      systemctl restart pxinst_virtip [OK]
   inst:      systemctl enable pxinstd       [OK]
   inst:      systemctl restart pxinstd      [OK]
   [INFO] INST 1801 pxinst Initialization completed successfully.(all)
   ```

   To execute all of the configurations collectively, specify "all" as the argument of the pxinst command. You can also execute each of the configurations individually. This enables you to initialize only a specific configuration when an operation becomes abnormal by making an erroneous edit. The following is an example of making individual configurations.

[When configuring only the DHCP server]

```
# pxinst init dhcp
dhcp:    Configuring dhcp
dhcp:      Creating /etc/dhcp/dhcpd.conf  [OK]
[INFO] INST 0801 pxinst Created a configuration backup.(/tmp/pxinst/dhcpd.conf_20180720105741)
dhcp:      systemctl enable dhcpd         [OK]
dhcp:      systemctl restart dhcpd        [OK]
[INFO] INST 1801 pxinst Initialization succeeded.(dhcp)
```

[When configuring only the TFTP server]

```
# pxinst init tftp
tftp:    Configuring tftp
tftp:      Creating /etc/xinetd.d/tftp    [OK]
[INFO] INST 0801 pxinst Created a configuration backup.(/tmp/pxinst/tftp_20180720105906)
tftp:      systemctl enable xinetd        [OK]
tftp:      systemctl restart xinetd       [OK]
[INFO] INST 1801 pxinst Initialization succeeded.(tftp)
```

[When configuring only the NTP server]

```
# pxinst init ntp
ntp:     Configuring ntp
ntp:       Creating /etc/chrony.conf      [OK]
[INFO] INST 0801 pxinst Created a configuration backup.(/tmp/pxinst/chrony.conf_20180720105922)
ntp:       systemctl enable chronyd       [OK]
ntp:       systemctl restart chronyd      [OK]
[INFO] INST 1801 pxinst Initialization succeeded.(ntp)
```

[When configuring only the HTTP server]

```
# pxinst init http
http:    Configuring http
http:      Creating /etc/httpd/conf.d/pxinst.conf [OK]
[INFO] INST 0801 pxinst Created a configuration backup.(/tmp/pxinst/pxinst.conf_20180720105941)
http:      systemctl enable httpd         [OK]
http:      systemctl restart httpd        [OK]
[INFO] INST 1801 pxinst Initialization succeeded.(http)
```

[When configuring only the ssh key]

```
# pxinst init ssh
ssh:     Generating ssh keys
ssh:       ssh-keygen                     [OK]
[INFO] INST 1801 pxinst Initialization succeeded.(ssh)
```

[When configuring only the installer node]

```
# pxinst init inst
inst:    Configuring pxinst
inst:      systemctl enable pxinst_virtip [OK]
inst:      systemctl restart pxinst_virtip [OK]
inst:      systemctl enable pxinstd       [OK]
inst:      systemctl restart pxinstd      [OK]
[INFO] INST 1801 pxinst Initialization succeeded.(inst)
```

Information

If you execute pxinst init all, pxinst init http or pxinst init inst on the system management node with the kernel parameter net.core.somaxconn less than 4096, it changes to 4096. Also, the kernel parameter net.core.somaxconn is set to 4096 for standby system management node and compute cluster sub management node which built by using the Installation function.

3.8.1.5 Configuration Contents

The settings of each service create and update the configuration files of the OS. This section describes these files.
It is not required for the system administrator to change the settings, for the installation function automatically configures required items. This section therefore describes the settings as reference. This section also describes the contents of the configuration files as their setting examples, not showing that the identical settings are actually made.

3.8.1.5.1 Configuring the DHCP Server

Set the operation environment of the DHCP server.
Make only the basic settings for the DHCP server; do not define a client (mapping of the MAC address and the IP address by the host declaration) at this point of time.

When registering node information which is described later, the client definition is additionally registered according to the information of the input file. After configuration using this function, it is unnecessary for the system administrator to make an additional edit manually; only automatically-set contents make it possible to run as a server. In addition, the configuration file is not managed after its creation, which allows you to accordingly customize it by using the automatically-created configuration file as a template.

The following table shows the file that the setting function of the DHCP server creates and updates.

Table 3.68 File Changed by Configuration of the DHCP Server

Setting Item	File	Setting Contents
Setting the DHCP server	/etc/dhcp/dhcpd.conf	Set various kinds of information, including the basic settings for the operations of the server, the definition of subnet used for installation, and the information of the boot loader passed to the client.

The following shows an example of the created configuration file. The installation function sets all of the items.

Configuration Example of the DHCP Server (dhcpd.conf)

```
ignore client-updates;
authoritative;
option vendor-class-identifier code 60 = text;
option space PXE;
option PXE.mtftp-ip    code 1 = ip-address;
option PXE.mtftp-cport code 2 = unsigned integer 16;
option PXE.mtftp-sport code 3 = unsigned integer 16;
option PXE.mtftp-tmout code 4 = unsigned integer 8;
option PXE.mtftp-delay code 5 = unsigned integer 8;
option arch code 93 = unsigned integer 16; # RFC4578
option client-uuid code 97 = string; # RFC4578

class "FX server" {
    match if option client-uuid = c1:10:3e:d6:a8:08:4b:80:be:f9:8f:3b:45:2c:f9:94;

    # FX server
    filename "uefi/grubaa64.efi";
}
class "FX server-INITIMAGE" {
    match if substring (option  vendor-class-identifier, 0, 16)  = "FX server-INITIMAGE";

    # FX server (dhclient executed from init-image)
    filename "uefi/grubaa64.efi";
}

class "pxeclients" {
    match if substring (option  vendor-class-identifier, 0, 9)  = "PXEClient";

    if option arch = 00:0b {
        # AARCH64 UEFI
        filename "uefi/grubaa64.efi";
    } else if option arch = 00:07 {
        # X86_64 UEFI
        filename "uefi/grubx64.efi";
    } else {
        # BIOS
        filename  "pxelinux/pxelinux.0";
    }
}


subnet 192.168.134.0 netmask 255.255.255.0 {
    default-lease-time 600;
    max-lease-time 3600;
    option PXE.mtftp-ip 0.0.0.0;

}
```

3.8.1.5.2 Configuring Function of the TFTP Server

Set the operation environment of the TFTP server. The TFTP server has a role of transferring materials to the client. These materials are necessary for installing the OS.
The following table shows the file that the setting function of the TFTP server creates and updates.

Table 3.69 File Changed by Configuration of the TFTP Server

Setting Item	File	Setting Contents
Setting the TFTP server	/etc/xinetd.d/tftp	Set the operation environment of the tftp server.

The following shows an example of the created configuration file and the items whose settings are changed by the installation function.
A configuration example of the TFTP server (File: /etc/xinet.d/tftp) is as follows.

```
# default: off
# description: The tftp server serves files using the trivial file transfer \
#       protocol.  The tftp protocol is often used to boot diskless \
#       workstations, download configuration files to network-aware printers, \
#       and to start the installation process for some operating systems.
service tftp
{
        socket_type             = dgram
        protocol                = udp
        wait                    = yes
        user                    = root
        server                  = /usr/sbin/in.tftpd
        server_args             = -v -s tftpdir                 (*1)
        disable                 = no                            (*2)
        per_source              = 11
        cps                     = 2000 2
        flags                   = IPv4
        instances               = UNLIMITED
}
```

(*1) The *tftpdir* part is changed to the specified value of "tftpdir," which is an item of the operation environment file of the installation function.
(*2) This is changed to "no."

3.8.1.5.3 Configuring the NTP Server

Set the installer node itself to refer to the NTP server. Also, set the NTP server so a node inside the system can refer to it. The installer node refers to the NTP server that is specified in "ntp_servers" of the environment configuration file of the installation function. The installation function uses chronyd as the NTP server.

The following table shows the file that the configuration function of the NTP server creates and updates.

Table 3.70 File Changed by Configuration of the NTP Server

Setting Item	File	Setting Contents
Setting the NTP server	/etc/chrony.conf	Setting to refer to the NTP server Setting to run as the NTP server

A configuration example of the NTP server (File: /etc/chrony.conf) is as follows.

```
# Use public servers from the pool.ntp.org project.
# Please consider joining the pool (http://www.pool.ntp.org/join.html).
server "ipaddr/hostname" iburst                     (*1)
server "ipaddr/hostname" iburst                     (*1)

# Ignore stratum in source selection.
stratumweight 0

# Record the rate at which the system clock gains/losses time.
driftfile /var/lib/chrony/drift

# Enable kernel RTC synchronization.
rtcsync

# In first three updates step the system clock instead of slew
# if the adjustment is larger than 10 seconds.
makestep 10 3

# Allow NTP client access from local network.
allow all                                           (*2)

# Listen for commands only on localhost.
bindcmdaddress 127.0.0.1
bindcmdaddress ::1

# Serve time even if not synchronized to any NTP server.
#local stratum 10

keyfile /etc/chrony.keys

# Specify the key used as password for chronyc.
commandkey 1

# Generate command key if missing.
generatecommandkey

# Disable logging of client accesses.
noclientlog

# Send a message to syslog if a clock adjustment is larger than 0.5 seconds.
logchange 0.5

logdir /var/log/chrony
#log measurements statistics tracking
```

ipaddr: IP address of the NTP server
hostname: Host name

(*1) Add the setting of the NTP server to which the system management node refers.
(*2) Add the setting to run as the NTP server.

By default, it accepts time synchronization from all clients.

3.8.1.5.4 Configuring the HTTP Server

Configure settings for the HTTP server. Regarding materials that the installation function uses through HTTP, they are not placed in "/var/www/html", the default location of the package, but in the dedicated directory "/var/opt/FJSVtcs/pxinst/www/". This is to avoid conflict against the HTTP setting that the system administrator performs. In this section, configure the settings to enable reference to said directory through HTTP. It also configures the httpd process configuration to handle the large number of HTTP access requests that occur to system management nodes and compute cluster sub management nodes.

The following table shows the file that the configuration function of the HTTP server creates and updates.

Table 3.71 File Changed by Configuration of the HTTP Server

Setting Item	File	Setting Contents
Setting the HTTP server	/etc/httpd/conf.d/pxinst.conf	Set Alias for "/var/opt/FJSVtcs/pxinst/www/".
	/etc/httpd/conf.moudles.d/00-mpm.conf	Change MPM module for use to worker.
	/etc/httpd/conf/httpd.conf	Add ListenBackLog 4096.

A configuration example of the HTTP server (File: /etc/httpd/conf.d/pxinst.conf) is as follows.

```
# This configuration file enables the TC-Suite install function settings.

# pxinst www root directory
Alias /pxinst/ "/var/opt/FJSVtcs/inst/www/"
<Directory "/var/opt/FJSVtcs/inst/www/">
    Options +FollowSymLinks
    Require all granted
</Directory>
# pxinst cgi directory
<Directory "/var/opt/FJSVtcs/inst/www/cgi-bin/">
    AddHandler cgi-script .cgi
    Options +ExecCGI
    SetEnv LANG ja_JP.UTF-8
</Directory>

# paclone www root directory
Alias /paclone/ "/var/opt/FJSVtcs/inst/paclone/www/"
<Directory "/var/opt/FJSVtcs/inst/paclone/www/">
    Options +FollowSymLinks
    Require all granted
</Directory>
```

A configuration example of the HTTP server (File: /etc/httpd/conf.moudles.d/00-mpm.conf) is as follows.

```
# Select the MPM module which should be used by uncommenting exactly
# one of the following LoadModule lines:

# prefork MPM: Implements a non-threaded, pre-forking web server
# See: http://httpd.apache.org/docs/2.4/mod/prefork.html
#LoadModule mpm_prefork_module modules/mod_mpm_prefork.so

# worker MPM: Multi-Processing Module implementing a hybrid
# multi-threaded multi-process web server
# See: http://httpd.apache.org/docs/2.4/mod/worker.html
#
LoadModule mpm_worker_module modules/mod_mpm_worker.so

# event MPM: A variant of the worker MPM with the goal of consuming
# threads only for connections with active processing
# See: http://httpd.apache.org/docs/2.4/mod/event.html
#
#LoadModule mpm_event_module modules/mod_mpm_event.so

<IfModule mpm_worker_module>
    StartServers             4
    ServerLimit              4
    ThreadLimit              4
    MaxRequestWorkers        16
    MinSpareThreads          4
    MaxSpareThreads          4
    ThreadsPerChild          4
    MaxConnectionsPerChild   0
</IfModule>
```

Note

The HTTP server configuration function changes the MPM module from the OS default of prefork to worker to take into account httpd's memory usage.
If you deploy the php package in an environment where you have made this change, the added configuration file (/etc/httpd/conf.d/php.conf) will fail and httpd will not be able to start with the as-is configuration. Therefore, do not deploy php packages on system management nodes or compute cluster sub management nodes.

3.8.1.5.5 Configuring ssh

Create the ssh keys to conduct passwordless communications with the installation target node.

3.8.1.5.6 TConfiguring the Installer Node

Configure settings for the command execution node to run as the installer node.
This function sets the internal information of the installation function. It is therefore unnecessary for the system administrator to be conscious of the set contents.

The following table shows the files that the configuration function of the installer node creates and updates.

Table 3.72 Files Changed by Configuration of the Installer Node

Setting Item	File	Setting Contents
Setting the local node information	/var/opt/FJSVtcs/inst/node_info.yaml	type: "installer" parent: "Representative IP address of the installer node"
Enabling and starting the service	-	Enabling and starting of the pxinstd service
Setting a port number	/etc/services	Adding the port number definition of pxinstd "pxinstd 30405/tcp"

Note

The above settings are the internal settings of the installation function. Do not edit them: Manually changing them may prevent the installation function from operating normally.

3.8.2 Managing the Repository

3.8.2.1 Creating a Repository

Create a new repository from the DVD or the ISO image file of the OS.
The repository is created under the directory that is specified in "repodir" of the environment configuration file of the installation function. According to the OS version of the installation target node, you must create a repository for all of the OS versions to be used.

An error occurs if a specified repository has already existed. If you want to recreate a repository, you must delete the repository and rerun it. Also, the created repository is associated with the actually-referred node when registering the node information.

Information

The PXE boot uses materials such as the kernel image (vmlinux), the boot imange (initrd), and the boot loader (pxelinux.0), which are necessary for the installation. When creating a repository, these materials are copied from the DVD of the OS to the directory specified in "tftpdir" of the environment configuration file of the installation function. At the same time, the settings as the HTTP server necessary for releasing the repository and the settings for local access provided by yum are also implemented.

Procedure for Using the DVD

The procedure for using the DVD is as follows:

1. Inserting the DVD

   Insert the DVD of the OS into the DVD drive of the active system management node.

2. Creating a repository

   ```
   # pxinst repo create -R reponame
   [INFO] INST 0800 pxinst repo completed successfully.(create repository)
   ```

   reponame: Repository name

Procedure for Using the ISO Image File

The procedure for using the ISO image file is as follows:

1. Creating a repository

   ```
   # pxinst repo create -R reponame -S isoimage
   [INFO] INST 0800 pxinst repo completed successfully.(create repository)
   ```

   reponame: Repository name
   isoimage: Path of the ISO image file of the OS

   The full path of the created repository is "*Specified value of repodir*/reponame".

Procedure for Creating a Repository by Specifying Previously-Prepared rootfs

The following is the procedure for creating a repository by specifying previously-prepared rootfs:

1. Creating a repository

   ```
   # pxinst repo create -R reponame -S isoimage --roofs rootfs
   [INFO] INST 0800 pxinst repo completed successfully.(create repository)
   ```

   reponame: Repository name
   isoimage: Path of the ISO image file of the OS
   rootfs: Image file of previously-obtained rootfs

Information

If the installer node cannot create rootfs (such as when there is a different architecture), specify the --rootfs option to use rootfs that is created on another node in advance. For details on how to create the root file system manually, see "3.8.5.7 Manual Creation of the Root File System."

3.8.2.1.1 dnf/yum Settings

When creating a repository, the yum environment setting is conducted to provide a local access to a repository on the system management node. "*Repository name*.repo" is actually created under the /etc/yum.repos.d/ directory. Regarding the yum setting for the installation target node, the remote access to a repository used for the installation is automatically set when installing the OS.

The following is an example of the configuration file of the created yum.

```
# cat /etc/yum.repos.d/rhel72.repo
[rhel72]
name=rhel 7.2 x86_64
baseurl=http://192.0.2.1/repos/rhel72
enabled=0
gpgcheck=0
```

Information

In creating a repository, enabled=1 (meaning that the repository is enabled) is set only when specifying the --enable option in the pxinst command. Without specifying the --enable option, this regards it as a repository for reference from subordinate nodes, setting enabled=0 (meaning that the repository is disabled). enabled=1 is specified only for the repository of the OS that the installer node uses for its own package update.
Also, the created configuration file has a minimum of initial settings for its operation; this has no restrictions for manually changing the settings after its creation.

The following is an example of enabling a repository to create it.

```
# pxinst repo create -R reponame -S filepath --enable
[INFO] INST 0800 pxinst repo completed successfully.(create repository)
```

reponame: Repository name
filepath: Path of the ISO image file of the OS

The created repository is also remotely accessible through a network (using http). The path for remote access is as follows.

```
http://ipaddr/pxinst/repos/reponame
```

ipaddr: IP address of the installer node or the relay node
reponame: Repository name

3.8.2.1.2 dnf Settings

If the package manager of the used OS is dnf, the "strict=False" and "max_parallel_downloads = 1" settings are added to the /etc/dnf/dnf.conf file when creating a repository.

```
[main]
gpgcheck=1
installonly_limit=3
clean_requirements_on_remove=True
strict=False
max_parallel_downloads=1
```

Information

"strict=False" is a function to apply only an existing package. If a package specified at the time of dnf installation does not exist, this function skips the package. (This is equivalent to the --skip-broken option of the yum command.) For details on strict, see the reference of the dnf command.
Also, the "strict=False" setting is automatically made for the /etc/dnf/dnf.conf file on the installation target node.

3.8.2.2 Listing Repositories

Display the list of repositories that have already been created. When displaying the list, it displays repositories existing in a directory that is specified in "repodir" of the environment configuration file of the installation function.

Note

If changing the value of "repodir" after creating a repository or if creating a repository in another directory manually, the list does not display the repository.

The following is an example of the list of repositories.

```
# pxinst repo list
REPONAME        PATH                            PKGDIR                                 OS      VERSION  ARCH
--------        ----                            ------                                 --      -------  ----
rhel72          /var/repos/rhel72                 Packages                                 rhel     7.2
x86_64
rhel80           /var/repos/rhel80                   AppStream/Packages,BaseOS/Packages       rhel
8.0      aarch64
```

REPONAME

The repository name

This displays values specified in reponame.

PATH

The full path of a repository

PKGDIR

The directory where stores packages in a repository

This varies depending on the OS versions , multiple versions are separated by commas.
To add or delete a package, use "PATH" and "PKGDIR." In the sample output above, for rsel 80, either "/ repos/rsel 80/AppStream/ Packages" or "/ repos/rsel 80/BaseOS/Packages" is the path where the package is stored.
When a job operation software package or a user adds a package, store it in the directory shown at the top "/ repos/rsel 80/AppStream/ Packages".
If you are adding an OS update package, store it in the distributor's public repository.

OS

The OS type

VERSION

The OS version

ARCH

The architecture

3.8.2.3 Updating a Repository

The system administrator must update the metadata of a repository to the latest state when adding or deleting the package file to/from the repository. If not updating a repository, note that you cannot apply the package, for mismatch occurs between the metadata managed inside the repository and the information on the actually-arranged RPM package file.

The following is an example of updating a repository.

```
# pxinst repo update -R reponame
[INFO] INST 0800 pxinst repo completed successfully.(update repository)
```

reponame: Repository name

3.8.2.4 Registering the Package Kit

In a specified repository, register an RPM package file group which the package kit contains.
This section describes the package kit of the Technical Computing Suite. This registration requires the "Technical Computing Suite Package Kit." This is the file for collecting RPM package files which the DVD of Technical Computing Suite contains.

To enable registered RPM package files, you must update the repository after the registration.

The following is an example of registering the package kit.

```
# pxinst repo pkgkit --kit packagekit_path -R reponame
[INFO] INST 0800 pxinst repo completed successfully.(add package kit)
```

packagekit_path: Path of the Technical Computing Suite package kit
reponame: Repository name

3.8.2.5 Deleting a Repository

Delete a repository that has already been created. If no target repository exists in the directory that is specified in "repodir" of the environment configuration file of the installation function, the command exits abnormally.

The following is an example of deleting a repository.

```
# pxinst repo delete -R reponame
Do you really want to continue (y/n)? y
[INFO] INST 0800 pxinst repo completed successfully.(delete repository)
```

reponame: Repository name

If deleting a repository to which the installation target node refers, you cannot install the OS and update the package for such node. In this case, you must recreate a repository with the same name or change a repository to be referred to on the installation target node. You can change a repository by changing the configuration file for yum (/etc/yum.repo.d/*repository name*.repo). This changing method is described below.

Perform this work on the node to be changed.

```
# cp /etc/yum.repo.d/old_repo.repo /etc/yum.repo.d/new_repo.repo
# vi /etc/yum.repo.d/new_repo.repo
[new_repo]                                        (*1)
name=rhel x.x  x86_64                             (*2)
baseurl=http://192.0.2.1/pxinst/repos/new_repo   (*3)
enabled=1
gpgcheck=0
```

old_repo: Old repository name
new_repo: New repository name

(*1) This changes the repository name.
(*2) This changes the name to any name.
(*3) This changes the repository name.

3.8.2.6 Adding and Deleting a Package

To add or delete an RPM package file, use a standard OS command to copy or delete the RPM package file to/from the package storage directory inside the repository. There is no dedicated function for this addition/deletion.

Procedure for Update the module definition file

If you add an OS update package of RedHat to the AppStream/Packages directory of the RHEL 8 repository, you also need to update the module definition file as well as add the package.

Before updating the module definition file, create a local mirror of the Red Hat repository on the subscripted server.

Extract the module definition file from the file on the local mirror of the Red Hat repository and place it in the target repository on the system management node.

The following example expands the module definition file on the subscripted server and places it in the target repository on the system management node.

```
# ssh server_ip zcat mirror_path/repodata/*-modules.yaml.gz > repopath/pxinst/
modules_AppStream.yaml
```

server_ip: IP address of the subscripted server
mirror_path: Local mirror path of the AppStream repository
repopath: Repository path of the system management node

See

For subscription servers, see "Appendix C Preparing Subscripted Servers" in "Job Operation Software Setup Guide."

Procedure for Adding a Package

The following is the procedure for adding a package:

1. Getting the repository path and the package storage directory

```
# pxinst repo list
REPONAME      PATH                     PKGDIR                                   OS       VERSION
ARCH
--------      ----                     ------                                   --       -------
----
rhel72        /repos/rhel72             Packages                                 rhel     7.2
x86_64
rhel80         /repos/rhel80              AppStream/Packages,BaseOS/Packages       rhel
8.0      aarch64
```

"PATH" and "PKGDIR" show paths that store the package. If there is more than one, the first directory listed is the target directory.

2. Adding the package

```
# cp xxxx.rpm /repos/rhel72/Packages
```

3. Updating a repository
Update a repository to enable the addition of the package.

```
# pxinst repo update -R reponame
```

reponame: Repository name

Procedure for Deleting a Package

The following is the procedure for deleting a package.

1. Getting the repository path and the package storage directory

```
# pxinst repo list
REPONAME      PATH                     PKGDIR                                OS       VERSION
ARCH
--------      ----                     ------                                --       -------
----
rhel72        /repos/rhel72             Packages                             rhel     7.2
x86_64
rhel80         /repos/rhel80               AppStream/Packages,BaseOS/Packages       rhel
8.0      aarch64
```

   "PATH" and "PKGDIR" show paths that store the package. If there is more than one, the first directory listed is the target directory.

2. Checking the existence of the package

```
# ls /repos/rhel72/Packages/xxxx.rpm
xxxx.rpm
```

3. Deleting the package

```
# rm /repos/rhel72/Packages/xxxx.rpm
```

4. Updating a repository

```
# pxinst repo update -R reponame
```

   reponame: Repository name

3.8.2.7 Changing the Repository Arrangement Path

Normally, repositories are all arranged in the directory that is specified in "repodir" of the environment configuration file of the installation function. However, operating the system involves the following cases due to reasons such as the capacity shortage of the disk:
- Where it is required to move a repository to another directory
- Where it is desirable to distributively arrange repositories in multiple directories
In such a case, implementing the following procedure allows you to move a repository without changing the settings of the installed node that refers to the repository.

The procedure for moving a repository is as follows:

1. Moving a repository

   Move a repository from the directory specified in "repodir" to another directory.

```
# mv Value specified in "repodir"/Repository name Destination directory/Repository name
```

2. Creating a symbolic link

   Create a symbolic link in "repodir" so that you can refer to the moved directory via network.

```
# ln -s Destination directory/Repository name Value specified in "repodir"/Repository name
```

3. Synchronizing the repository

   Execute the synchronization to apply the repository change to the standby installer node and the relay node.

```
# pxinst sync repo
```

Note

The synchronization of the repository synchronizes the destination file of the symbolic link, which you created in "repodir," in "repodir" as an entity. This requires both the standby installer node and the relay node to have disk capacity including an entity.

3.8.3 Creating Settings of the Installation Target Node

3.8.3.1 Information Designing for PG

The information designing of PRIMERGY server does not consider the cluster configuration in the Job Operation Software, which allows you to collectively define nodes to be installed in multiple clusters. Meanwhile, there is no restriction that forces you to enter all node information in one book, which enables you to divide it into multiple books in any unit.

The following table shows the sheets to enter information for installing PRIMERGY server.

Table 3.73 Input Sheets of the Automatic Configuration Creation Function (for PRIMERGY server)

Sheet Name	Description	Input Content
partition	Disk partition definition sheet	Define the configuration of a disk partition set to a node. It is unnecessary to define the configurations of all nodes; define necessary configurations only.
nodes	Node information definition sheet	Enter node-specific information such as a host name and IP address, which is necessary for OS installation. Also, use this sheet to specify associations with the disk partition definition used by a node.

3.8.3.1.1 Disk Partition Definition Sheet

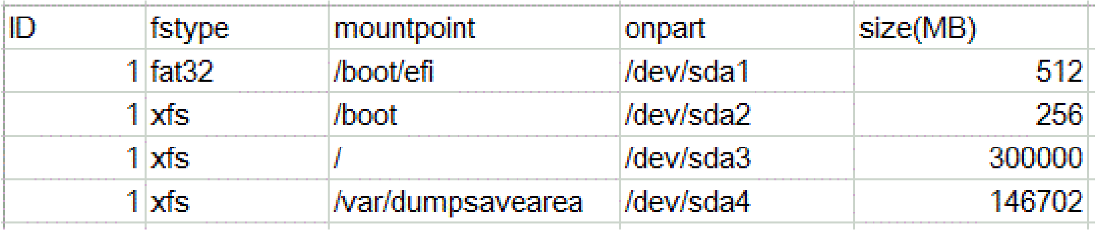
Define the configuration of a disk partition on the partition sheet. This requires you to define all necessary
The UEFI mode requires you to define the EFI system partition in the disk partition definition.
You must define the EFI system partition as the first partition of the disk for booting. You must specify the following contents (fixed). Also, a disk containing the EFI system partition is automatically set as the GUID partition table (GPT).

[Contents to be specified for the EFI system partition]

- fstype: fat32
- mountpoint: /boot/efi
- onpart: The first partition of the disk for booting such as /dev/sda1
- size: 512

The setting example is as follows.

Figure 3.15 Example of the partition Sheet in the UEFI Mode



ID	fstype	mountpoint	onpart	size(MB)
1	fat32	/boot/efi	/dev/sda1	512
1	xfs	/boot	/dev/sda2	256
1	xfs	/	/dev/sda3	300000
1	xfs	/var/dumpsavearea	/dev/sda4	146702

The following table shows the meanings of its input items.

Table 3.74 Input Items of the partition Sheet

Setting Item	Meaning	Required or Optional
ID	Partition ID Specify any value of 1 or more in order to associate this ID with node information on the node information definition.	Required
fstype	Type of a file system This supports the following file systems: - ext3(*)	Required

Setting Item	Meaning	Required or Optional
	- ext4 - xfs - swap - fat32 (supporting EFI system partition only) (*)ext3 is not recommended for RHEL8.	
mountpoint	Mount point of a partition While defining the same partition ID, there must be one partition whose mount point is "/". In the EFI system partition, it is necessary to specify "/boot/efi". If fstype is swap, it is unnecessary to specify it. If specifying it, this is ignored. Example: /, /boot, etc.	Required
onpart	Device arranging partitions Example: /dev/sda1, /dev/sda2, etc. [Note] For mbr partition You cannot specify the 4th partition, for this is an extended partition. (You cannot specify /dev/sda4, etc.) For gpt partition It is unnecessary to consider it, for no extended partition exists. The maximum number of primary partitions is 128.	Required
size	Size of a partition (in MB) With max specified, use up to the upper limit of the disk to create a partition. [Note] You can specify max for only the last partition in the same disk.	Required

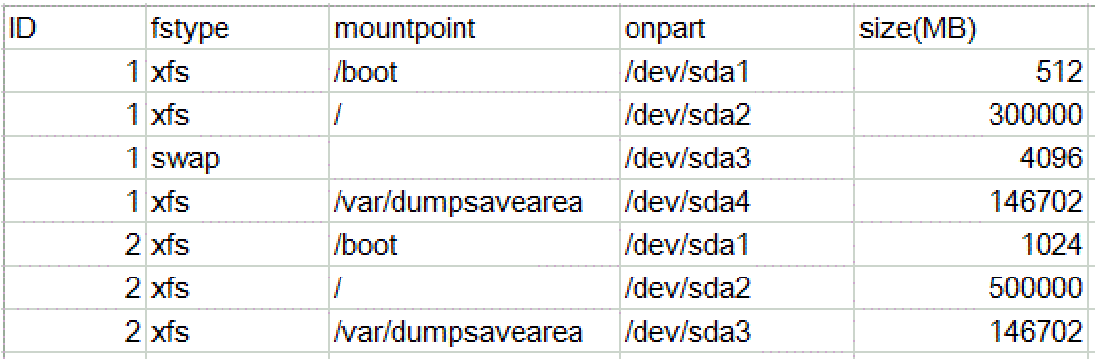
Note

Be sure to include the dump save area on the disk partition definition sheet. The file system types can be ext3(*), ext4, or xfs. (*)ext3 is not recommended for RHEL8.
If you have a support desk contract, obtain a dump support tool (FJSVdumptools) and set the capacity of the dump area according to the attached document.
If a support desk is not available or you do not want to use the above tools, configure them according to your operating system documentation.

Information

If you use BIOS mode, define the disk partition configuration in the partition sheet as follows.

Figure 3.16 Example partition sheet in BIOS mode



ID	fstype	mountpoint	onpart	size(MB)
1	xfs	/boot	/dev/sda1	512
1	xfs	/	/dev/sda2	300000
1	swap		/dev/sda3	4096
1	xfs	/var/dumpsavearea	/dev/sda4	146702
2	xfs	/boot	/dev/sda1	1024
2	xfs	/	/dev/sda2	500000
2	xfs	/var/dumpsavearea	/dev/sda3	146702

3.8.3.1.2 Node Information Definition

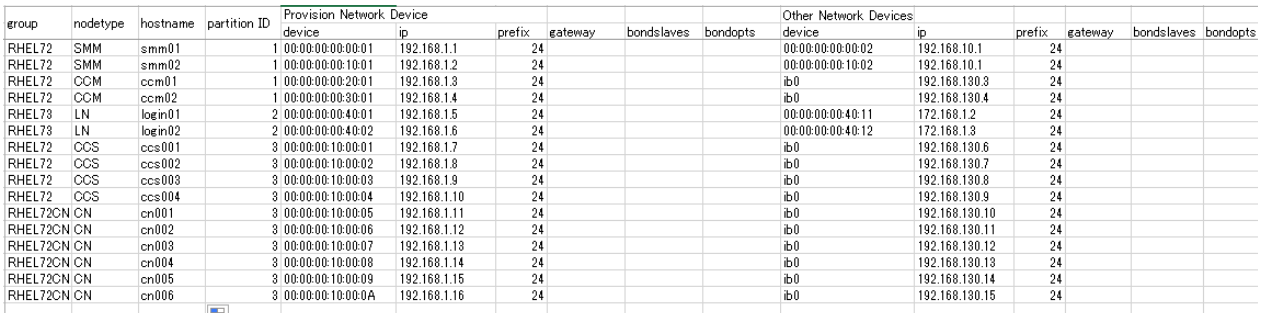
In the node information definition, enter node information to be added to the system. For node information, it is also necessary to enter the information on the active installer node to prepare for recovery or failover in the event of a failure.

Note

In creating the template of the rule file for cluster configuration definition, which is a file of the configuration management function, 1 command execution corresponds to 1 cluster. Therefore, divide the design sheet to be created for each cluster. However, the number of the design sheet must be 1. You can also divide it by dividing output file manually.

A screenshot of the nodes sheet is as follows.

Figure 3.17 Screenshot of the nodes Sheet



group	nodetype	hostname	partition ID	Provision Network Device						Other Network Devices					
				device	ip	prefix	gateway	bondslaves	bondopts	device	ip	prefix	gateway	bondslaves	bondopts
RHEL72	SMM	smm01	1	00:00:00:00:00:01	192.168.1.1	24				00:00:00:00:00:02	192.168.10.1	24			
RHEL72	SMM	smm02	1	00:00:00:00:10:01	192.168.1.2	24				00:00:00:00:10:02	192.168.10.1	24			
RHEL72	CCM	ccm01	1	00:00:00:00:20:01	192.168.1.3	24				ib0	192.168.130.3	24			
RHEL72	CCM	ccm02	1	00:00:00:00:30:01	192.168.1.4	24				ib0	192.168.130.4	24			
RHEL73	LN	login01	2	00:00:00:00:40:01	192.168.1.5	24				00:00:00:00:40:11	172.168.1.2	24			
RHEL73	LN	login02	2	00:00:00:00:40:02	192.168.1.6	24				00:00:00:00:40:12	172.168.1.3	24			
RHEL72	CCS	ccs001	3	00:00:00:10:00:01	192.168.1.7	24				ib0	192.168.130.6	24			
RHEL72	CCS	ccs002	3	00:00:00:10:00:02	192.168.1.8	24				ib0	192.168.130.7	24			
RHEL72	CCS	ccs003	3	00:00:00:10:00:03	192.168.1.9	24				ib0	192.168.130.8	24			
RHEL72	CCS	ccs004	3	00:00:00:10:00:04	192.168.1.10	24				ib0	192.168.130.9	24			
RHEL72CN	CN	cn001	3	00:00:00:10:00:05	192.168.1.11	24				ib0	192.168.130.10	24			
RHEL72CN	CN	cn002	3	00:00:00:10:00:06	192.168.1.12	24				ib0	192.168.130.11	24			
RHEL72CN	CN	cn003	3	00:00:00:10:00:07	192.168.1.13	24				ib0	192.168.130.12	24			
RHEL72CN	CN	cn004	3	00:00:00:10:00:08	192.168.1.14	24				ib0	192.168.130.13	24			
RHEL72CN	CN	cn005	3	00:00:00:10:00:09	192.168.1.15	24				ib0	192.168.130.14	24			
RHEL72CN	CN	cn006	3	00:00:00:10:00:0A	192.168.1.16	24				ib0	192.168.130.15	24			

The following table shows the meanings of its input items.

Table 3.75 Input Items of the nodes Sheet

Setting Item	Meaning	Required or Optional
group	Any group name for grouping nodes Grouping nodes outputs the output file of node information for each group. When there are nodes using different OS versions, divide them into groups and then use the files that are output by the division. This makes it easier to register node information for each OS version.	Optional
nodetype	Specifying a node type This is reference information for workers referring to the design sheet for installation to identify node types. This does not affect the operations of the installation function, regardless of specifying this or not. [Note] To cooperate with the configuration management function in the system	Optional

Setting Item		Meaning	Required or Optional
		management, enter the node type defined by the job operation software in the nodetype field. Node types defined by the Job Operation Software are as follows: SMM, CCM, CCS, LN, CN, SCM, MGS, MDS, and OSS In a configuration serving for multiple nodes, specify two or more of them by separating them with commas. (Example) For the node serving as both SMM and SCM: "SMM,SCM" If entering anything other than a node type name of the Job Operation Software, it is treated as the multiuse node. Nodetype can be between 3 and 5 uppercase letters and numbers.	
hostname		Host name The only valid symbol for a hostname is hyphen(-).	Required
partition ID		Partition definition ID Specify the partition ID which is defined on the partition sheet.	Required
Provision Network		Information on network to be used for installation	Required
	device	Network device name According to a network device to be used, specify it using either of the following methods: a. For Ethernet interface Specify the MAC address of the interface. Example: 01:23:45:67:89:ab b. When using the bonding device Specify any device name. Example: bond0	Required
	ip	IP address set to the interface	Required
	prefix	Network mask Specify it in bit format. Example: For 24-bit mask (255.255.255.0), specify 24.	Required
	gateway	Default gateway Specify it when setting a default gateway. This is required when the provision network on the installation target node is different from the provision network on the installer node. Without specifying it, communications become unavailable, resulting in failure of the installation.	Optional
	bondslaves	Slave device for using the bonding device Specify them by separating them with commas. This has the same available formats as device. Example: 00:00:00:00:00:01,00:00:00:00:00:02	Optional
	bondopts	Options of the bonding device Specify them by separating them with semicolons. Example: mode=active-backup;primary=00:00:00:00:00:01 [Note] You can specify the mode parameter by using either a full mode name such as balance-rr and active-backup, or a numerical value such as 0 and 1.	Optional

Setting Item		Meaning	Required or Optional
Other Network		Define information on a network other than the network to be used for installation. Define an external network or the network of infiniband. To automatically set the IP address of IPMI for BIOS or UEFI when installing the OS, define the IP address of BMC here.	Optional
	device	Network device name According to a network device to be used, specify it using either of the following methods: For Ethernet interface Specify either the MAC address or the device name of the interface. Example of specifying the MAC address: 01:23:45:67:89:ab Example of specifying the device name: ens6f0 When using the bonding device Specify any device name. Example: bond0 For IPoIB interface Specify either the MAC address or the device name of the interface. Example of specifying the device name: ib0 Example of specifying the MAC address: 80:00:00:28:fe:80:00:00:00:00:01:02:03:04:05:06:07:08:09:0a For BMC Specify the fixed name "bmc." When specifying "bmc," IPMI is automatically configured after the OS installation, according to the specified contents. [Note] Suppose that you cooperate with the template creation function of the node configuration definition file, which is a file of the configuration management function. Then, this defines the device specifying "bmc" as the control network managed by the Job Operation Software.	Required
	ip	IP address set to the interface	Required
	prefix	Network mask	Required
	gateway	Default gateway Specify it when setting a default gateway.	Optional
	bondslaves	Slave device for using the bonding device Specify them by separating them with commas. This has the same available formats as device. Example: 00:00:00:00:00:01,00:00:00:00:00:02	Optional
	bondopts	Options of the bonding device Specify them by separating them with semicolons. Example: mode=active-backup;primary=00:00:00:00:00:01 [Note] You can specify the mode parameter by using either a full mode name such as balance-rr and active-backup, or a numerical value such as 0 and 1.	Optional

3.8.3.1.3 Usage Method

This section describes how to use the partition sheet and the nodes sheet.

1. Displaying the input sheet

   Perform [Add-in] - [Technical Computing Suite] - [init]. The partition sheet and the nodes sheet appear.

2. Defining the configuration of a disk partition

   Input the configuration of a disk partition in the partition sheet.

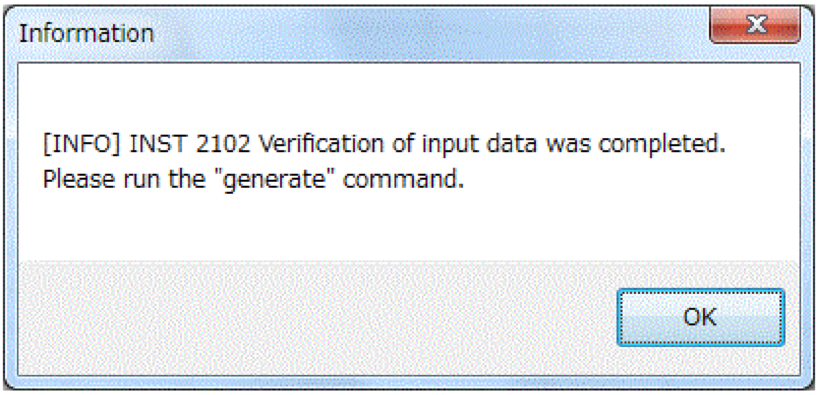
3. Defining node information

   In the nodes sheet, input the information on the installation target node.

4. Checking the consistency of the input information

   Perform [Add-in] - [Technical Computing Suite] - [verify].
   If the input information has no problem, the following dialog message appears.

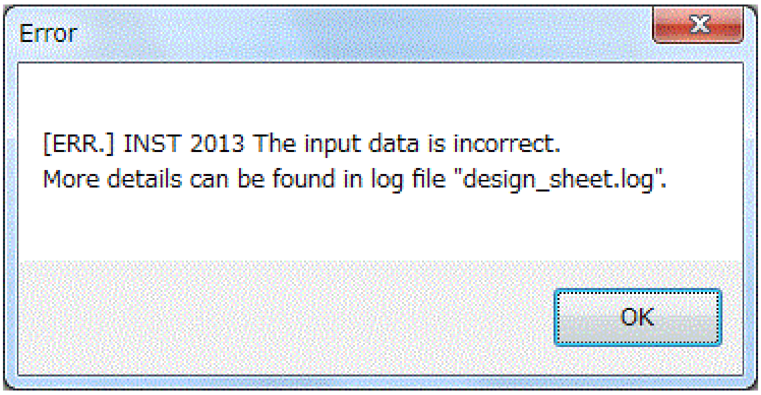
   Figure 3.18 Message of verify Completion

   [INFO] INST 2102 Verification of input data was completed.
   Please run the "generate" command.

   If the input information has an error, the following dialog message appears.
   Then, you must refer to the log file shown in this message, modify the error, and then verify the consistency of the input information once again.

   Figure 3.19 Message of verify Failure

   [ERR.] INST 2013 The input data is incorrect.
   More details can be found in log file "design_sheet.log".

5. Outputting the configuration file

   Perform [Add-in] - [Technical Computing Suite] - [generate]. This outputs the node information definition file and the information definition file for disk partition to the current directory.

3.8.3.2 Information Designing for FX server

The automatic configuration creation function for FX server uses the same design sheet for installation as PRIMERGY server.
On the PC, the system administrator enters information necessary for installation in the design sheet. Enter the basic information in the dialog box. This automatically creates the template of the input sheet. The following table shows the input sheet.

Unlike PRIMERGY server, FX server requires you to use this function, for FX server makes it necessary to make special settings in accordance with the specification of hardware or that of the Tofu interconnect D.

Table 3.76 Input Sheet of the Automatic Configuration Creation Function (for FX server)

Sheet Name	Description	Input Content
nodes_ft	Node information definition for FX server	Input node-specific information such as a host name, an IP address, and the Tofu interconnect D, which is necessary for OS installation.

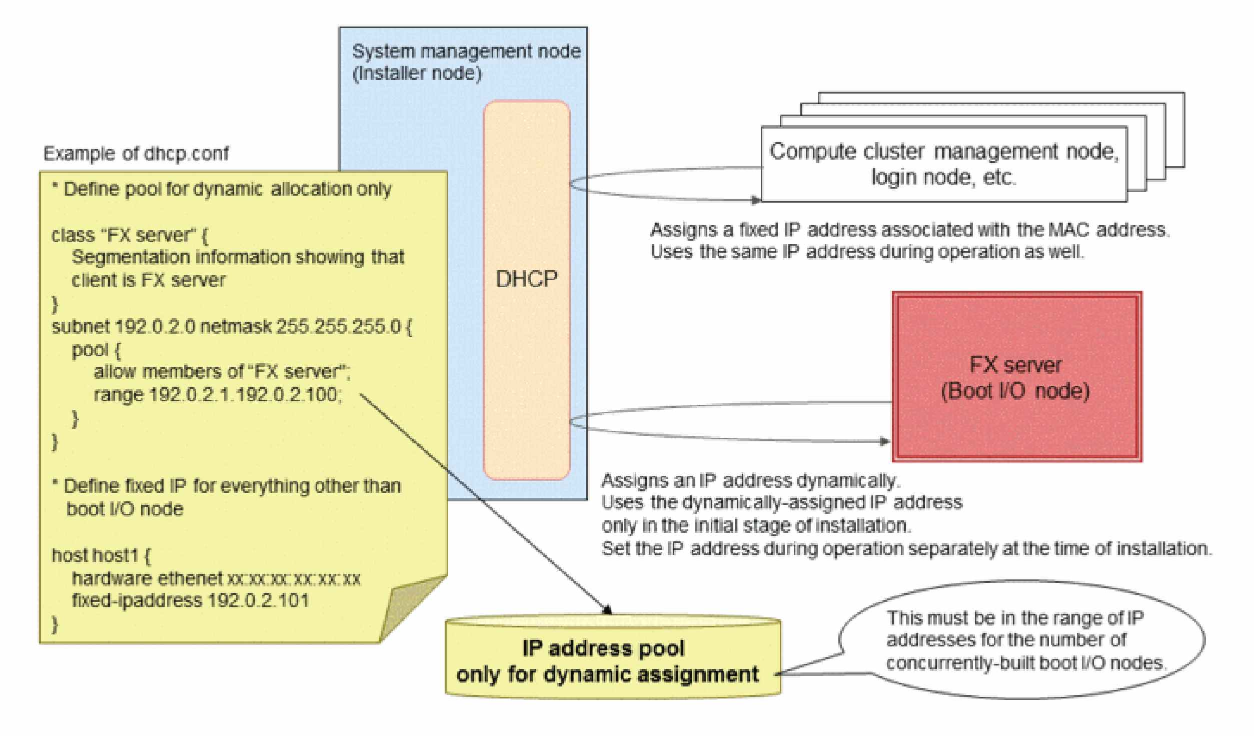
3.8.3.2.1 Assigning a Dynamic IP Address when Installing a BoB

The Job Operation Software makes it necessary to perform operations by assigning a fixed IP address on each node.
You must use the design sheet to enter a MAC address corresponding to an IP address, for the DHCP server adds a fixed IP address. Therefore, FX server requires you to enter the MAC address of the Ethernet on all boot I/O nodes.

Building a large-scale environment requires saving the effort of entering a MAC address and reducing the rework caused by an input error in order to reduce the building time. The installation function therefore supports the method of dynamically assigning the IP address of the management network used for installing the boot I/O node. (This method is hereinafter called dynamic IP address assignment.)

This function makes it unnecessary to enter the MAC address of the management network of the boot I/O node in the design sheet. Add the dynamic IP address only temporarily at the installation time. After the installation, the original IP address specified in the design sheet becomes valid. Also, this function is available only in FX server, not in PRIMERGY server. The following diagram shows the image of the installation by the dynamic assignment.

Figure 3.20 Installing FX server With a Dynamic IP Address



```
* Define pool for dynamic allocation only

class "FX server" {
   Segmentation information showing that
   client is FX server
}
subnet 192.0.2.0 netmask 255.255.255.0 {
   pool {
      allow members of "FX server";
      range 192.0.2.1 192.0.2.100;
   }
}

* Define fixed IP for everything other than
  boot I/O node

host host1 {
   hardware ethenet xx:xx:xx:xx:xx:xx
   fixed-ipaddress 192.0.2.101
}
```

Assigns a fixed IP address associated with the MAC address.
Uses the same IP address during operation as well.

Assigns an IP address dynamically.
Uses the dynamically-assigned IP address
only in the initial stage of installation.
Set the IP address during operation separately at the time of installation.

This must be in the range of IP
addresses for the number of
concurrently-built boot I/O nodes.

Note

- The IP addresses for dynamic assignment must be the IP addresses that are not used (no duplicates) in operations.
- You must secure IP addresses only for installation whose number is the same as that of concurrently-installed boot I/O nodes (BoB).
- To specify the IP address pool for dynamic allocation, use the registration function of the node information (the pxinst node command).
- When there are individual subnets for each node group, the load sharing by the relay node is installed by using the relay node in the same node group as the target node. If the same subnet has multiple node groups, the relay node used for installation is undefined. (A dynamically-assigned IP address automatically sorts the relay node to be referred.)

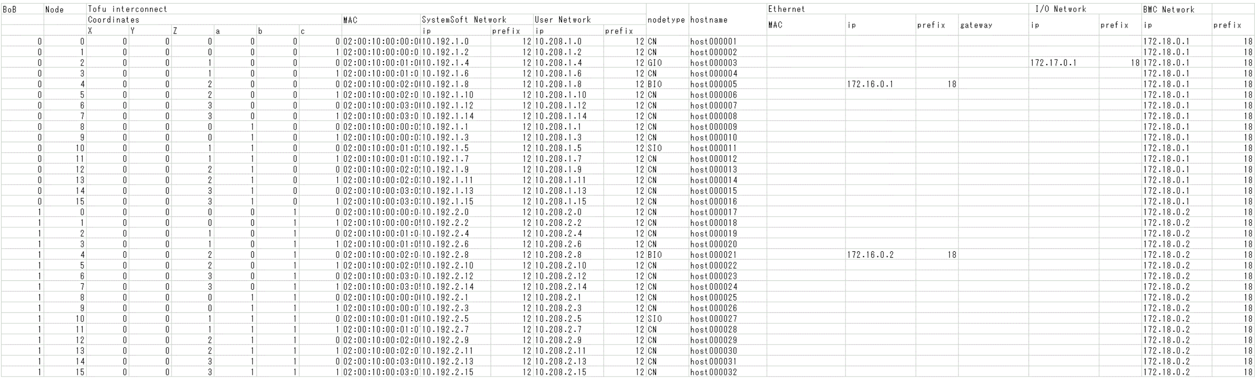
See

- For the setting using the installation by a dynamic IP address, see "3.8.3.2.3 Usage Method" of the automatic configuration creation function.
- For details on creating the IP address pool, see "3.8.8 Registering Node Information."

3.8.3.2.2 Node Information Definition

A screenshot of the nodes_ft sheet is as follows.

Figure 3.21 Screenshot of the nodes_ft Sheet



BoB	Node	Tofu interconnect													Ethernet				I/O Network		BMC Network	
		Coordinates						MAC	SystemSoft Network		User Network		nodetype	hostname	MAC	ip	prefix	gateway	ip	prefix	ip	prefix
		X	Y	Z	a	b	c		ip	prefix	ip	prefix										
0	0	0	0	0	0	0	0	02:00:10:00:00:0	10.192.1.0	12	10.208.1.0	12	CN	host000001							172.18.0.1	18
0	1	0	0	0	0	0	1	02:00:10:00:00:0	10.192.1.2	12	10.208.1.2	12	CN	host000002							172.18.0.1	18
0	2	0	0	1	0	0	0	02:00:10:00:01:0	10.192.1.4	12	10.208.1.4	12	GIO	host000003					172.17.0.1	18	172.18.0.1	18
0	3	0	0	1	0	0	1	02:00:10:00:01:0	10.192.1.6	12	10.208.1.6	12	CN	host000004							172.18.0.1	18
0	4	0	0	2	0	0	0	02:00:10:00:02:0	10.192.1.8	12	10.208.1.8	12	BIO	host000005		172.16.0.1	18				172.18.0.1	18
0	5	0	0	2	0	0	1	02:00:10:00:02:0	10.192.1.10	12	10.208.1.10	12	CN	host000006							172.18.0.1	18
0	6	0	0	3	0	0	0	02:00:10:00:03:0	10.192.1.12	12	10.208.1.12	12	CN	host000007							172.18.0.1	18
0	7	0	0	3	0	0	1	02:00:10:00:03:0	10.192.1.14	12	10.208.1.14	12	CN	host000008							172.18.0.1	18
0	8	0	0	0	1	0	0	02:00:10:00:00:0	10.192.1.1	12	10.208.1.1	12	CN	host000009							172.18.0.1	18
0	9	0	0	0	1	0	1	02:00:10:00:00:0	10.192.1.3	12	10.208.1.3	12	CN	host000010							172.18.0.1	18
0	10	0	0	1	1	0	0	02:00:10:00:01:0	10.192.1.5	12	10.208.1.5	12	SIO	host000011							172.18.0.1	18
0	11	0	0	1	1	0	1	02:00:10:00:01:0	10.192.1.7	12	10.208.1.7	12	CN	host000012							172.18.0.1	18
0	12	0	0	2	1	0	0	02:00:10:00:02:0	10.192.1.9	12	10.208.1.9	12	CN	host000013							172.18.0.1	18
0	13	0	0	2	1	0	1	02:00:10:00:02:0	10.192.1.11	12	10.208.1.11	12	CN	host000014							172.18.0.1	18
0	14	0	0	3	1	0	0	02:00:10:00:03:0	10.192.1.13	12	10.208.1.13	12	CN	host000015							172.18.0.1	18
0	15	0	0	3	1	0	1	02:00:10:00:03:0	10.192.1.15	12	10.208.1.15	12	CN	host000016							172.18.0.1	18
1	0	0	0	0	0	1	0	02:00:10:00:00:0	10.192.2.0	12	10.208.2.0	12	CN	host000017							172.18.0.2	18
1	1	0	0	0	0	1	1	02:00:10:00:00:0	10.192.2.2	12	10.208.2.2	12	CN	host000018							172.18.0.2	18
1	2	0	0	1	0	1	0	02:00:10:00:01:0	10.192.2.4	12	10.208.2.4	12	CN	host000019							172.18.0.2	18
1	3	0	0	1	0	1	1	02:00:10:00:01:0	10.192.2.6	12	10.208.2.6	12	CN	host000020							172.18.0.2	18
1	4	0	0	2	0	1	0	02:00:10:00:02:0	10.192.2.8	12	10.208.2.8	12	BIO	host000021		172.16.0.2	18				172.18.0.2	18
1	5	0	0	2	0	1	1	02:00:10:00:02:0	10.192.2.10	12	10.208.2.10	12	CN	host000022							172.18.0.2	18
1	6	0	0	3	0	1	0	02:00:10:00:03:0	10.192.2.12	12	10.208.2.12	12	CN	host000023							172.18.0.2	18
1	7	0	0	3	0	1	1	02:00:10:00:03:0	10.192.2.14	12	10.208.2.14	12	CN	host000024							172.18.0.2	18
1	8	0	0	0	1	1	0	02:00:10:00:00:0	10.192.2.1	12	10.208.2.1	12	CN	host000025							172.18.0.2	18
1	9	0	0	0	1	1	1	02:00:10:00:00:0	10.192.2.3	12	10.208.2.3	12	CN	host000026							172.18.0.2	18
1	10	0	0	1	1	1	0	02:00:10:00:01:0	10.192.2.5	12	10.208.2.5	12	SIO	host000027							172.18.0.2	18
1	11	0	0	1	1	1	1	02:00:10:00:01:0	10.192.2.7	12	10.208.2.7	12	CN	host000028							172.18.0.2	18
1	12	0	0	2	1	1	0	02:00:10:00:02:0	10.192.2.9	12	10.208.2.9	12	CN	host000029							172.18.0.2	18
1	13	0	0	2	1	1	1	02:00:10:00:02:0	10.192.2.11	12	10.208.2.11	12	CN	host000030							172.18.0.2	18
1	14	0	0	3	1	1	0	02:00:10:00:03:0	10.192.2.13	12	10.208.2.13	12	CN	host000031							172.18.0.2	18
1	15	0	0	3	1	1	1	02:00:10:00:03:0	10.192.2.15	12	10.208.2.15	12	CN	host000032							172.18.0.2	18

The following table shows the meanings of its input items.

Table 3.77 Input Items of the nodes_ft Sheet

Setting Item			Meaning	Required or Optional
BoB			Serial number of BoBs. This is unchangeable, for these numbers are automatically set.	-
Node			Serial number of nodes inside BoB. This is unchangeable, for these numbers are automatically set.	-
Tofu interconnect			Settings about Tofu interconnect D	-
	Coordinates		Tofu coordinates. This is unchangeable, for these numbers are automatically set.	-
		X	X-coordinate	Required
		Y	Y-coordinate	Required
		Z	Z-coordinate	Required
		a	a-coordinate	Required
		b	b-coordinate	Required
		c	c-coordinate	Required
	MAC		The MAC address of IP over Tofu device (hereinafter called as IPoTofu device). This is automatically calculated from Tofu coordinates. The MAC address is common in "SystemSoft Network" and "User Network."	Required
	SystemSoft Network		Interface for system software	-
		ip	IP address	Required
		prefix	Prefix of a subnet mask	Required
	User Network		Interface for user	-
		ip	IP address	Required

Setting Item			Meaning	Required or Optional
		prefix	Prefix of a subnet mask	Required
nodetype			Node type of the Job Operation Software. It is unnecessary to change this, for this is automatically set. However, if the mount locations of the global I/O node and the storage I/O node is set to manual, ("Manually assign" is set in the user form "I/O node(s) configuration" described below) the system administrator must make this setting manually. BIO: Node serving as both compute node and boot I/O node GIO: Node serving as both compute node and global I/O node SIO: Node serving as both compute node and storage I/O node CN: Compute node	Required
hostname			Specifies a host name to be assigned to a node. The only valid symbol for a hostname is hyphen(-).	Required
Ethernet			Information on Ethernet. This is required for boot I/O node.	Required
	MAC		MAC address	Required
	ip		IP address	Required
	prefix		Prefix of a net mask	Required
	gateway		Default gateway Specify it when setting a default gateway.	Optional
I/O Network			Network information for storage I/O. This is required for global I/O node.	Required
	ip		IP address	Required
	prefix		Prefix of a net mask	Required
BMC Network			Control network information for operating BMC. Specify the same information as the information specified at the time of hardware setting (the setting of BMC). This setting must be common in all nodes inside BoB, for this is a setting for each BoB.	Required
	ip		IP address of BMC	Required
	prefix		Prefix of a net mask	Required

3.8.3.2.3 Usage Method

This section describes how to use the nodes_ft sheet.

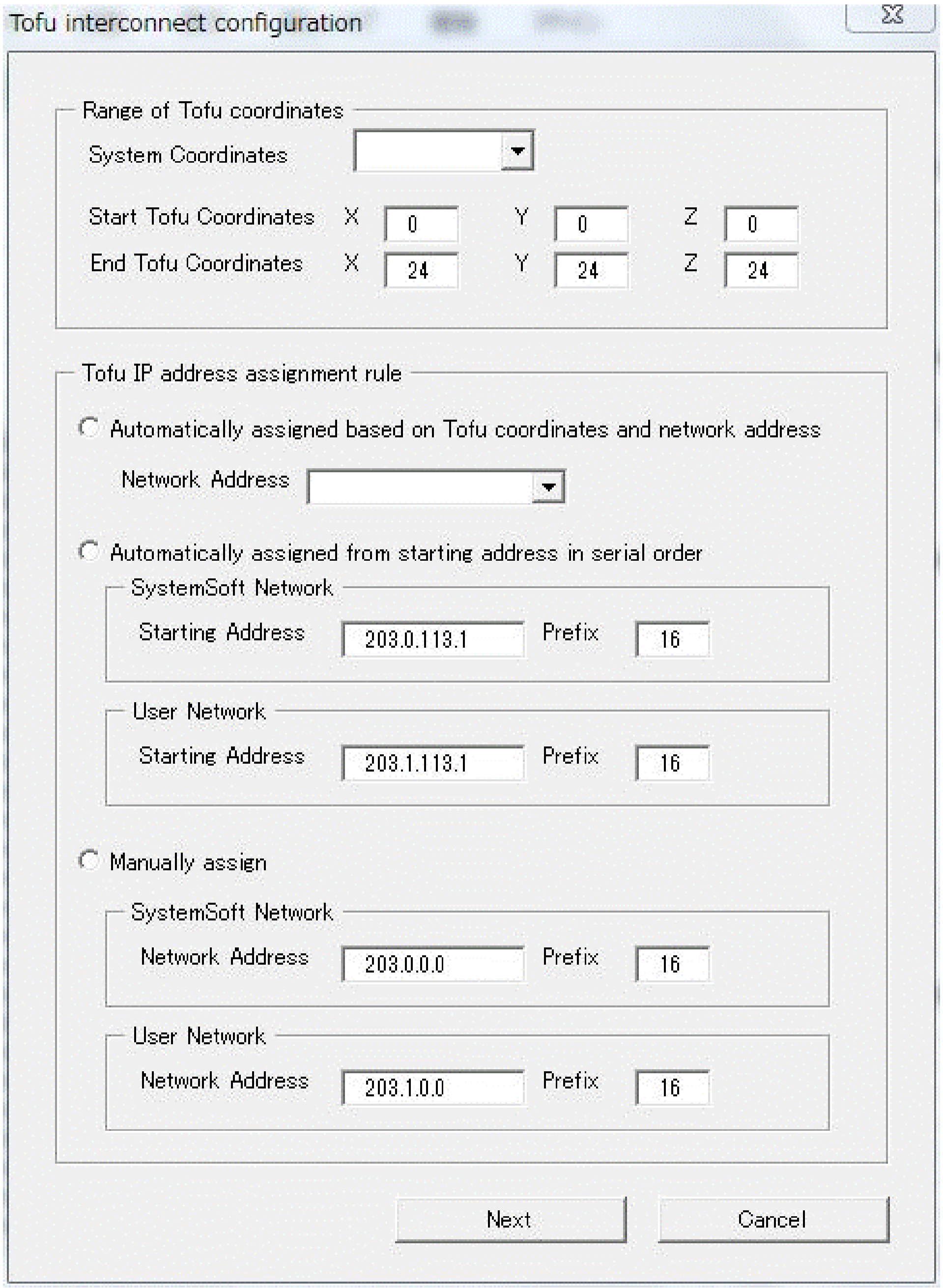
1. Entering the system information

   Perform [Add-in] - [Technical Computing Suite] - [init_ft].

2. Inputting information about the Tofu interconnect D

In the [Tofu interconnect configuration] form, input information about the Tofu interconnect D. After completing this inputting, select [Next].
The following is a screenshot of input information in the [Tofu interconnect configuration] form.

Figure 3.22 Screenshot of the [Tofu interconnect configuration] Form



Tofu interconnect configuration

Range of Tofu coordinates

System Coordinates:

Start Tofu Coordinates X 0 Y 0 Z 0

End Tofu Coordinates X 24 Y 24 Z 24

Tofu IP address assignment rule

○ Automatically assigned based on Tofu coordinates and network address

Network Address:

○ Automatically assigned from starting address in serial order

SystemSoft Network
Starting Address 203.0.113.1 Prefix 16

User Network
Starting Address 203.1.113.1 Prefix 16

○ Manually assign

SystemSoft Network
Network Address 203.0.0.0 Prefix 16

User Network
Network Address 203.1.0.0 Prefix 16

In [Range of Tofu coodinates], specify the Tofu configuration of the entire system and the range of Tofu coordinates of the installation target node. Its setting items are as follows.

Table 3.78 Setting Items of [Range of Tofu coordinates] in [Tofu interconnect configuration]

Setting Item		Meaning	Required or Optional
1	[System Coordinates]	Select either of the following Tofu configurations of the entire system. The assignment method for Tofu coordinates varies depending on a selected	Require d

Setting Item		Meaning	Required or Optional
		configuration. It is therefore necessary to select a configuration appropriate to the actual hardware configuration. [Configuration for small scale (Z=4 configuration)] This configuration consists of up to 8 shelves. 1,1,4: 1 rack composes (X,Y,Z) = (1,1,4). 2,1,4: 1 rack composes (X,Y,Z) = (2,1,4). 2,2,4: 1 rack composes (X,Y,Z) = (2,2,4). 2,3,4: 1 rack composes (X,Y,Z) = (2,3,4). 2,4,4: 1 rack composes (X,Y,Z) = (2,4,4). [Configuration for middle scale (2n,2n,8: Z=8 configuration)] 1 rack composes (X,Y,Z) = (2,2,8). [Configuration for large scale (2n,2n,16: Z=16 configuration)] 2 racks compose (X,Y,Z) = (2,2,16). [Configuration for ultra-large scale (2n,2n,24: Z=24 configuration)] 3 racks compose (X,Y,Z) = (2,2,24).	
2	[Start Tofu Coordinates]	Specify the start coordinates of x-coordinate, y-coordinate, and z-coordinate of the installation target node.	Required
3	[End Tofu Coordinates]	Specify the end coordinates of x-coordinate, y-coordinate, and z-coordinate of the installation target node.	Required

In [Tofu IP address assignment rule], specify a rule to assign an IP address to the Tofu interface of each node. The Tofu interface has two interfaces: the interface for system software and the interface for the user. You must assign an IP address to each of them. It is also necessary to set these two interfaces as different subnets.

The methods for assigning IP addresses are described below.

[Automatically assigned based on Tofu coordinates and network address]

This automatically assigns a unique IP address based on the Tofu coordinates and the network address. (This is called the bit shift method.) This automatically assigns both of the IP address for system software and the IP address for user. The following table shows the item requiring an input when selecting this rule.

Table 3.79 Item of [Automatically assigned based on Tofu coordinates and network address]

Setting Item		Meaning	Required or Optional
1	[Network Address]	Select either of the following network addresses to assign the IP address: 10.0.0.0/12, 10.32.0.0/12, 10.64.0.0/12, 10.96.0.0/12, 10.128.0.0/12, 10.160.0.0/12, 10.192.0.0/12, 10.224.0.0/12, 172.16.0.0/16, 172.18.0.0/16, 172.20.0.0/16, 172.22.0.0/16, 172.24.0.0/16, 172.26.0.0/16, 172.28.0.0/16, and 172.30.0.0/16 You can select the network of the 172.*xx*.0.0/16 system from a system whose end coordinate of the Tofu coordinates of the entire system is equal to or less than (X,Y,Z) = (15,15,15). For details on the assigned IP address, see "3.8.3.2.5 Method for Assigning IP Address by the Bit Shift Method."	Required

[Automatically assigned from starting address in serial order]

This assigns IP addresses from the starting address in serial order.
The following table shows the items requiring an input when selecting this rule.

Table 3.80 Items of [Automatically assigned from starting address in serial order]

Setting Item		Meaning	Required or Optional
1	[Starting Address]	Starting address to assign IP addresses consecutively Specify this when selecting the 2nd rule.	Required
2	[Prefix]	Subnet mask to assign IP addresses consecutively They must fall in the subnet range specified by the number of nodes.	Required

It is required to specify both of the rule for system software and the rule for user. Also, the orders of assigning IP addresses are as follows.

Table 3.81 Assignment Orders of Tofu IP Addresses

Order	Axis	Remarks
1	b axis	To b0 to b2 direction
2	Z axis	To Z0 to Z23 direction
3	X axis	To X axis max direction
4	Y axis	To Y axis max direction

[Manually assign]

The system administrator is to assign any IP addresses manually. It is necessary to enter IP addresses directly on the [node_ft] sheet, for IP addresses are not set automatically. It is required to specify both of the rule for system software and the rule for user. The following table shows the items requiring an input when selecting this rule.

Information

The items specified here are the information for the design sheet to check whether or not an IP address manually entered by the system administrator falls in the network range. They are not used for the assignment.

Table 3.82 Items of [Manually assign]

Setting Item		Meaning	Required or Optional
1	[Network Address]	Network address	Required
2	[Prefix]	Subnet mask This must fall in the subnet range specified by a manually-assigned IP address.	Required

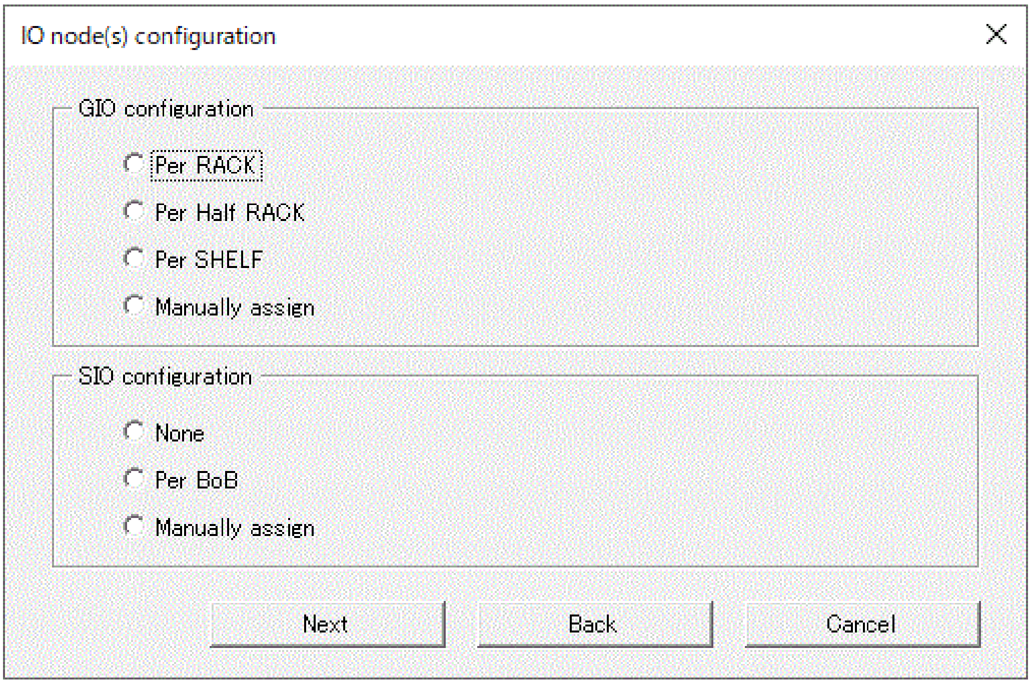
3. Entering the mount information of the global I/O node and the storage I/O node

In the [I/O node(s) configuration] form, enter information about the mount locations of the node serving as both CN and GIO and the node serving as both CN and SIO. (These mount locations mean mounted Tofu coordinates.) After completing this inputting, select

[Next].

The following is a screenshot of the input information in the [I/O node(s) configuration] form.

Figure 3.23 Screenshot of the [I/O node(s) configuration] Form



IO node(s) configuration

GIO configuration

○ Per RACK

○ Per Half RACK

○ Per SHELF

○ Manually assign

SIO configuration

○ None

○ Per BoB

○ Manually assign

In [GIO configuration], specify the rule of mounting the node serving as both the compute node and the global I/O node. There are the following four selectable rules:

a. [Per RACK]

Select this when the two nodes serving as both the compute node and the global I/O node is mounted for per rack. The Tofu coordinates of the node serving as both the compute node and the global I/O node are as follows.

Table 3.83 Location of [Per RACK] node serving as both the compute node and the global I/O node

if x/2 + y/2 is even	if x/2 + y/2 is even
(x+0, y+0, z+1, 0, 0, 0)	(x+1, y+1, z+5, 1, 2, 0)
(x+1, y+1, z+5, 1, 2, 0)	(x+0, y+0, z+1, 0, 0, 0)

(x, y, z are the origin (Minimum) xyz coordinates in the rack.)

b. [Per Half RACK]

Select this when the four nodes serving as both the compute node and the global I/O node is mounted for per rack. The Tofu coordinates of the node serving as both the compute node and the global I/O node are as follows.

Table 3.84 Location of [Per Half RACK] node serving as both the compute node and the global I/O node

if x/2 + y/2 is even	if x/2 + y/2 is even
(x+0, y+0, z+1, 0, 0, 0)	(x+1, y+1, z+5, 1, 2, 0)
(x+1, y+1, z+5, 1, 2, 0)	(x+0, y+0, z+1, 0, 0, 0)
(x+0, y+0, z+5, 0, 0, 0)	(x+1, y+1, z+1, 1, 2, 0)
(x+1, y+1, z+1, 1, 2, 0)	(x+0, y+0, z+5, 0, 0, 0)

(x, y, z are the origin (Minimum) xyz coordinates in the rack.)

c. [Per SHELF]

Select this when the eight nodes serving as both the compute node and the global I/O node is mounted for per rack. The Tofu coordinates of the node serving as both the compute node and the global I/O node are as follows.

Table 3.85 Location of [Per SHELF] node serving as both the compute node and the global I/O node

if x/2 + y/2 is even	if x/2 + y/2 is even
(x+0, y+0, z+1, 0, 0, 0)	(x+1, y+1, z+5, 1, 2, 0)
(x+1, y+1, z+5, 1, 2, 0)	(x+0, y+0, z+1, 0, 0, 0)
(x+0, y+0, z+5, 0, 0, 0)	(x+1, y+1, z+1, 1, 2, 0)
(x+1, y+1, z+1, 1, 2, 0)	(x+0, y+0, z+5, 0, 0, 0)
(x+0, y+1, z+1, 0, 0, 0)	(x+1, y+0, z+5, 1, 2, 0)
(x+1, y+0, z+5, 1, 2, 0)	(x+0, y+1, z+1, 0, 0, 0)
(x+0, y+1, z+5, 0, 0, 0)	(x+1, y+0, z+1, 1, 2, 0)
(x+1, y+0, z+1, 1, 2, 0)	(x+0, y+1, z+5, 0, 0, 0)

(x, y, z are the origin (Minimum) xyz coordinates in the rack.)

d. [Manually assign]

The system administrator is to set the mount location of the node serving as both the compute node and the global I/O node manually. It is necessary to select this rule in the following cases:
- When the number of the mounted nodes serving as both the compute node and the global I/O node differs by rack or shelf, and
- When the mount coordinate is different from the coordinate that is automatically set in [Per RACK] , [Per Half RACK], and [Per SHELF]
When selecting this rule, the node serving as both the compute node and the global I/O node (whose character string is "GIO") is not set in [nodetype] of the [node_ft] sheet automatically. It is therefore necessary to enter it manually and directly.

For [SIO configuration], select None.

Information

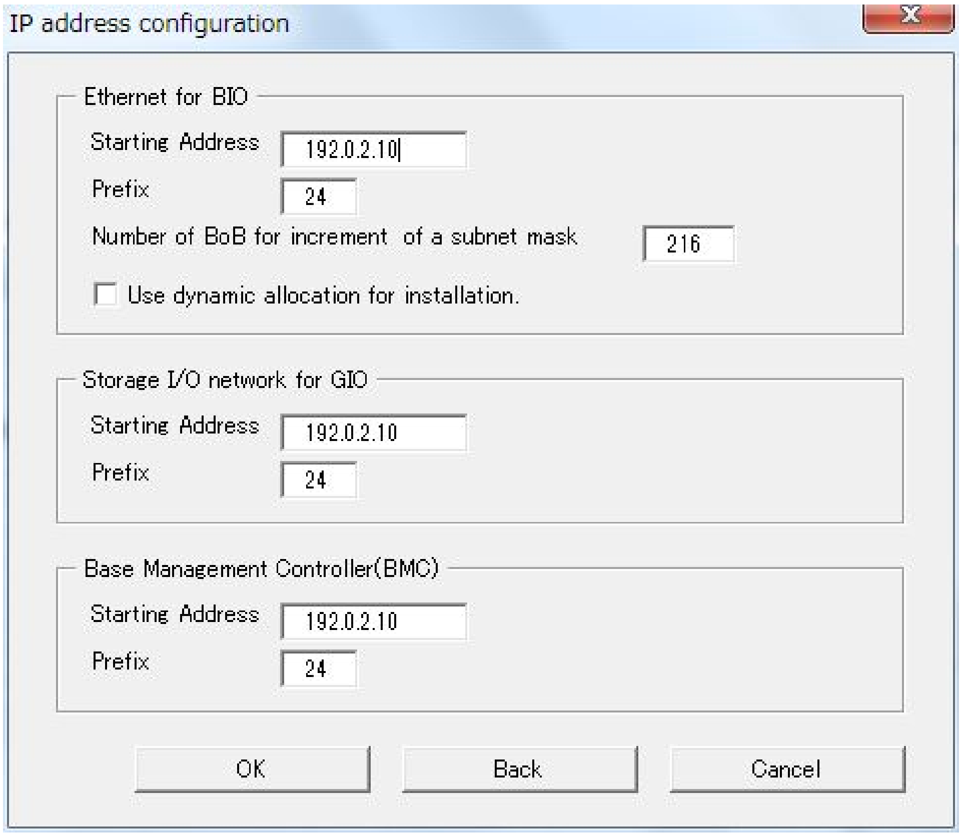
When selecting [Manually assign] in [GIO configuration] and [SIO configuration], you must enter a node type (GIO and SIO) according to the mount locations of the global I/O node and the storage I/O node. (Enter the node type in step "5. Defining node information.") Also, after step "7. Outputting the configuration file," you must edit the output file and enter the coordinate range, in which the global I/O node and the storage I/O node take a role as file I/O.

For details on specifying [Manually assign] in [GIO configuration] and [SIO configuration], see "3.8.3.2.4 How to Describe the Storage I/O Node and the Global I/O Node."

4. Entering the IP address allocation rule

Specify the IP address allocation rule in the [IP address configuration] form. After completing this inputting, select [OK]. The following is a screenshot of the input information in the [IP address configuration] form.

Figure 3.24 Screenshot of the [IP address configuration] Form



IP address configuration

Ethernet for BIO

Starting Address: 192.0.2.10

Prefix: 24

Number of BoB for increment of a subnet mask: 216

□ Use dynamic allocation for installation.

Storage I/O network for GIO

Starting Address: 192.0.2.10

Prefix: 24

Base Management Controller(BMC)

Starting Address: 192.0.2.10

Prefix: 24

In [Ethernet for BIO], specify the rule of IP addresses (the management network of the Job Operation Software) which are allocated to NIC of Gb Ethernet mounted on the boot I/O node. Its setting items are as follows.

Table 3.86 Setting Items of [Ethernet for BIO]

Setting Item	Meaning	Required or Optional
[Starting Address]	Starting address to assign IP addresses consecutively If omitting it, IP addresses are not allocated automatically. It is therefore necessary to enter IP addresses directly on the [nodes_ft] sheet.	Optional
[Prefix]	Subnet mask to assign IP addresses consecutively They must fall in the subnet range specified by the number of nodes. It is required to set this if setting [Starting Address].	Optional
[Number of BoB for increment of a subnet mask]	If exceeding the specified number of BoBs by consecutively allocating IP addresses up to such number, this increments a subnet mask to resume the allocation. If omitting this, the allocation is made without the increment. Example: Starting address: 192.0.2.10 Prefix: 24 Number of BoB for increment of a subnet mask: 216 BoB 000 192.0.2.10 BoB 001 192.0.2.11 ... BoB 215 192.0.2.225	

Setting Item	Meaning	Required or Optional
	BoB 216 192.0.3.10 BoB 217 192.0.3.11 ...	
[Use dynamic allocation for installation.]	Put a check mark here when using the dynamic allocation method for IP addresses at the time of installation. With this checked, it is unnecessary to enter the MAC address of the boot I/O node on the node_ft sheet. For details on the dynamic allocation method, see "3.8.3.2.1 Assigning a Dynamic IP Address when Installing a BoB."	

In [Storage I/O network for GIO], specify the rule of IP addresses allocated to the devices of the storage I/O network, which is mounted on the node serving as both the compute node and the global I/O node. Its setting items are as follows.

Table 3.87 Setting Items of [Storage I/O network for GIO]

Setting Item	Meaning	Required or Optional
[Starting Address]	Starting address to assign IP addresses consecutively If omitting it, IP addresses are not allocated automatically. It is therefore necessary to enter IP addresses directly on the [nodes_ft] sheet.	Optional
[Prefix]	Subnet mask to assign IP addresses consecutively They must fall in the subnet range specified by the number of nodes. It is required to set this if setting [Starting Address].	Optional

In [Base Management Controller(BMC)], specify the rule of IP addresses allocated to BMC that each BoB mounts. Its setting items are as follows.

Note

You must set the IP addresses of BMC so as to become the same as the IP addresses that are actually allocated to hardware. If these IP addresses mismatch each other, note that processes (such as power control) communicating with BMC become unavailable.

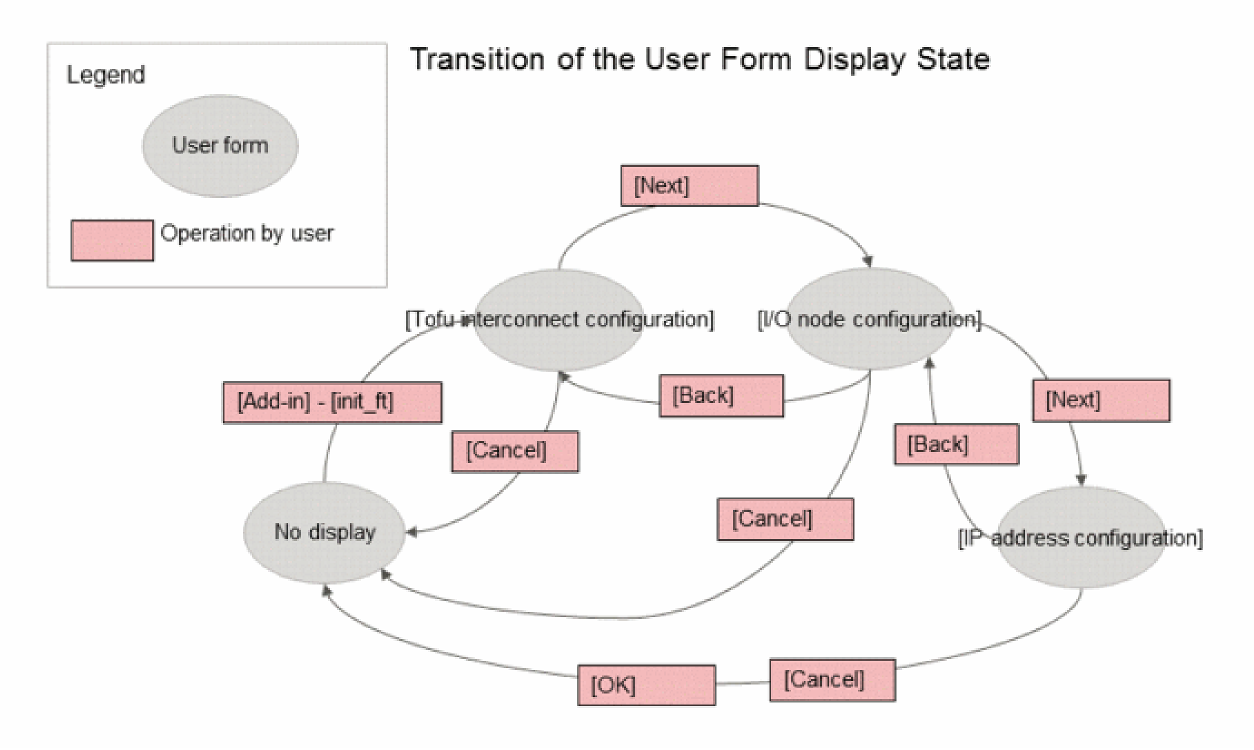
Table 3.88 Setting Items of [Base Management Controller (BMC)]

Setting Item	Meaning	Required or Optional
[Starting Address]	Starting address to assign IP addresses consecutively If omitting it, IP addresses are not allocated automatically. It is therefore necessary to enter IP addresses directly on the [nodes_ft] sheet.	Optional
[Prefix]	Subnet mask to assign IP addresses consecutively They must fall in the subnet range specified by the number of nodes. It is required to set this if setting [Starting Address].	Optional

Information

As shown in the following diagram, the display of the user form transitions by operating the buttons. In the user form of this function, the close button ([X] button on the upper right of the dialog) is unavailable. To cancel and close this form, use the [Cancel] button.

Figure 3.25 Transition of the User Form



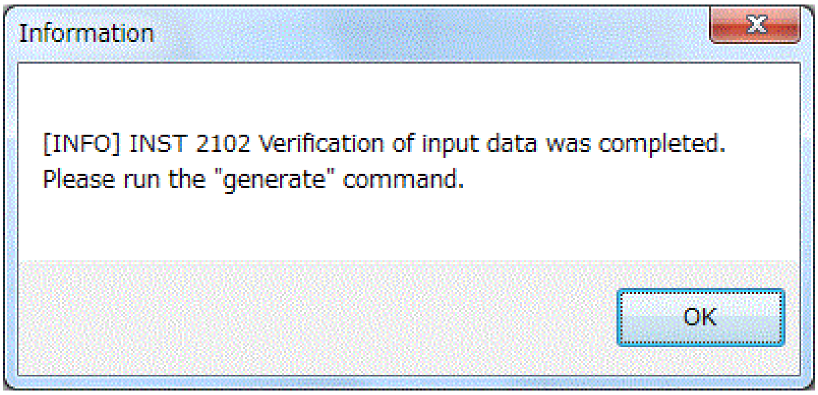
5. Defining node information

   In the nodes_ft sheet, enter the information on the installation target node.

6. Checking the consistency of the input information

   Perform [Add-in] - [Technical Computing Suite] - [verify].
   If the input information has no problem, the following dialog message appears.

Figure 3.26 Message of verify Completion

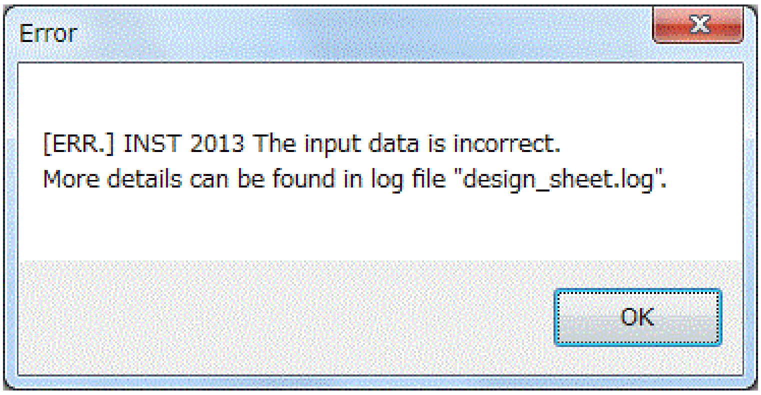


Information

[INFO] INST 2102 Verification of input data was completed.
Please run the "generate" command.

If the input information has an error, the following dialog message appears.
Then, you must refer to the log file shown in this message, modify the error, and then verify the consistency of the input information once again.

Figure 3.27 Message of verify Failure



[ERR.] INST 2013 The input data is incorrect.
More details can be found in log file "design_sheet.log".

7. Outputting the configuration file

   Perform [Add-in] - [Technical Computing Suite] - [generate].
   This outputs the FX server configuration information file to the current directory.

3.8.3.2.4 How to Describe the Storage I/O Node and the Global I/O Node

When selecting [Manually assign] in [SIO configuration] and [GIO configuration] of the [I/O node(s) configuration] form, the system administrator must enter the mount locations of the storage I/O node and the global I/O node manually.

There are the following rules regarding mounting of the storage I/O node and the global I/O node:

- The number of the storage I/O nodes mounted in one shelf is zero, one, or three.
- The number of the global I/O nodes depends on the size of the FX server system as follows:
  - Configuration of Z=4

    The number of the global I/O nodes mounted in one shelf is one.
  - Configuration of Z=8 or more

    The number of the global I/O nodes mounted in one rack is two, four, or eight.

It is also necessary to edit the file output by using the design sheet for installation, and describe the range in which the storage I/O node and the global I/O node are in charge of input and output for the file system.

See

This section describes how to edit the output file of the design sheet for installation. For the meanings of the formats and items of the output file, see "3.8.8.6 Node Information Definition File for FX server."

Describing the Design Sheet for Installation (The Storage I/O Node and the Global I/O Node)

When selecting [Manually assign] in [SIO configuration] and [GIO configuration] of the design sheet for installation, enter the node types ("SIO" or "GIO") of the storage I/O node and the global I/O node in the [node_ft] sheet. Also, enter a value in the [I/O Network] column of the global I/O node.

See the [Coordinates] column of the [node_ft] sheet, and then enter the node type name "GIO" or "SIO" in the row of the coordinates that mount the global I/O node and the storage I/O node.

Information

By default, "BIO" or "CN" has existed in the [nodetype] column of the [node_ft] sheet.

Describing the Output File (Storage I/O Node)

Edit the output file of the design sheet for installation.
Describe the coordinate range of a compute node in the item "iorange" of the output file node_ft_*nn*.conf. (*nn* is a serial number from 01.) In this compute node, the storage I/O node, which you manually entered in the design sheet for installation, is in charge of input and output inside the FX server.

This description method varies depending on the number of mounted storage I/O nodes per shelf.

- When a shelf mounts three storage I/O nodes

  When a shelf mounts three storage I/O nodes, one of the storage I/O nodes is in charge of input and output inside FX server for one BoB. Thus, in the item "iorange," describe the range for one BoB that includes the corresponding storage I/O node.

  When showing the range for one BoB, only the value of Z-coordinate differs between the start point and the end point.
  The following is an example where the coordinate of the storage I/O node is (X,Y,Z,a,b,c) = (0,0,1,1,0,0). Describe the item "iorange" as "0,0,0,0-0,0,3,0".

```
 - node: 10
   tofu: '0,0,1,1,0,0'
   nodetype: 'SIO'
   hostname: 'node00011'
   iorange: '0,0,0,0-0,0,3,0'                     (*)
   networks:
   - {device: 'tofu0', ip: '10.0.1.5', mac: '02:00:10:00:01:02', prefix: 12, boot: true}
   - {device: 'tofu1', ip: '10.16.1.5', prefix: 12}
   - {device: 'bmc', ip: '198.51.100.93', prefix: 16}
```

  (*) Use the "X,Y,Z,b-X,Y,Z,b" format to describe the range that is in charge of input and output inside FX server.

  The following is another example when the coordinate of the storage I/O node is (X,Y,Z,a,b,c) = (2,2,5,1,0,0). Describe the item "iorange" as "2,2,4,0-2,2,7,0".

```
 - node: 10
   tofu: '2,2,5,1,0,0'
   nodetype: 'SIO'
   hostname: 'node01979'
   iorange: '2,2,4,0-2,2,7,0'                     (*)
   networks:
   - {device: 'tofu0', ip: '10.1.9.21', mac: '02:00:10:08:45:02', prefix: 12, boot: true}
   - {device: 'tofu1', ip: '10.17.9.21', prefix: 12}
   - {device: 'bmc', ip: '198.51.100.216', prefix: 16}
```

  (*) Use the "X,Y,Z,b-X,Y,Z,b" format to describe the range that is in charge of input and output inside FX server.

- When a shelf mounts one storage I/O node

  When a shelf mounts one storage I/O node, this storage I/O node is in charge of input and output inside FX server for three BoBs. Thus, in the item "iorange," describe the range for three BoBs that include the corresponding storage I/O node.

  When showing the range for three BoBs, only the values of Z-coordinate and b-coordinate differ between the start point and the end point.

  The following is an example where the coordinate of the storage I/O node is (X,Y,Z,a,b,c) = (0,0,1,1,0,0). Describe the item "iorange" as "0,0,0,0-0,0,3,2".

```
 - node: 10
   tofu: '0,0,1,1,0,0'
   nodetype: 'SIO'
   hostname: 'sio0001'
   iorange: '0,0,0,0-0,0,3,2'                     (*)
   networks:
   - {device: 'tofu0', ip: '10.0.1.5', mac: '02:00:10:00:01:02', prefix: 12, boot: true}
   - {device: 'tofu1', ip: '10.16.1.5', prefix: 12}
   - {device: 'bmc', ip: '198.51.100.93', prefix: 16}
```

  (*) Use the "X,Y,Z,b-X,Y,Z,b" format to describe the range that is in charge of input and output inside FX server.

The following is another example when the coordinate of the storage I/O node is (X,Y,Z,a,b,c) = (2,2,5,1,0,0). Describe the item "iorange" as "2,2,4,0-2,2,7,2".

```
- node: 10
  tofu: '2,2,5,1,0,0'
  nodetype: 'SIO'
  hostname: 'node0001'
  iorange: '2,2,4,0-2,2,7,2'                     (*)
  networks:
  - {device: 'tofu0', ip: '10.1.9.21', mac: '02:00:10:08:45:02', prefix: 12, boot: true}
  - {device: 'tofu1', ip: '10.17.9.21', prefix: 12}
  - {device: 'bmc', ip: '198.51.100.216', prefix: 16}
```

(*) Use the "X,Y,Z,b-X,Y,Z,b" format to describe the range that is in charge of input and output inside FX server.

Describing the Output File (Global I/O Node)

Edit the output file of the design sheet for installation.
Describe the coordinate range of a compute node in the item "iorange" of the output file node_ft_*nn*.conf. (*nn* is a serial number from 01.) In this compute node, the global I/O node, which you manually entered in the design sheet for installation, is in charge of input and output to the shared file system (FEFS).

The global I/O node is in charge of input and output to the shared file system (FEFS) of the compute node in the same rack, regardless of the number of mounted nodes.

For this reason, in iorange of the global I/O node, describe the coordinate range of the rack that mounts the corresponding global I/O node.

The following is an example where the coordinate of the global I/O node is (X,Y,Z,a,b,c) = (0,0,1,0,0,0) in the Z=8 system. Describe the item "iorange" as "0,0,0,0-1,1,7,2".

```
- node: 2
  tofu: '0,0,1,0,0,0'
  nodetype: 'GIO'
  hostname: 'node00003'
  iorange: '0,0,0,0-1,1,7,2'                     (*)
  networks:
  - {device: 'tofu0', ip: '10.0.1.4', mac: '02:00:10:00:01:00', prefix: 12, boot: true}
  - {device: 'tofu1', ip: '10.16.1.4', prefix: 12}
  - {device: 'ib0', ip: '203.0.113.1', prefix: 24}
  - {device: 'bmc', ip: '192.0.2.100', prefix: 24}
```

(*) Use the "X,Y,Z,b-X,Y,Z,b" format to describe the range that is in charge of input and output to the shared file system (FEFS).

For all the global I/O nodes in the same rack, describe values in iorange in the same manner.

Information

In systems with the small-scale configuration (the Z=4 system), iorange of the global I/O node is equivalent to the coordinate range of the system.

3.8.3.2.5 Method for Assigning IP Address by the Bit Shift Method

When selecting the bit shift method for the IP address allocation of the Tofu network, IP addresses are automatically allocated in the following ranges. This allocation is based on the network address selected in [Tofu interconnect configuration] of the design sheet and the Tofu coordinates of a target node.

- For 10.0.0.0/11

  Range of Tofu0: 10.0.1.0 to 10.15.254.127
  Range of Tofu1: 10.16.1.0 to 10.31.254.127

- For 10.32.0.0/11

  Range of Tofu0: 10.32.1.0 to 10.47.254.127
  Range of Tofu1: 10.48.1.0 to 10.63.254.127

- For 10.64.0.0/11

  Range of Tofu0: 10.64.1.0 to 10.79.254.127
  Range of Tofu1: 10.80.1.0 to 10.95.254.127

- For 10.96.0.0/11

  Range of Tofu0: 10.96.1.0 to 10.111.254.127
  Range of Tofu1: 10.112.1.0 to 10.127.254.127

- For 10.128.0.0/11

  Range of Tofu0: 10.128.1.0 to 10.143.254.127
  Range of Tofu1: 10.144.1.0 to 10.159.254.127

- For 10.160.0.0/11

  Range of Tofu0: 10.160.1.0 to 10.175.254.127
  Range of Tofu1: 10.176.1.0 to 10.191.254.127

- For 10.192.0.0/11

  Range of Tofu0: 10.192.1.0 to 10.207.254.127
  Range of Tofu1: 10.208.1.0 to 10.223.254.127

- For 10.224.0.0/11

  Range of Tofu0: 10.224.1.0 to 10.239.254.127
  Range of Tofu1: 10.240.1.0 to 10.255.254.127

- For 172.16.0.0/16

  Range of Tofu0: 172.16.0.64 to 172.16.255.191
  Range of Tofu1: 172.17.0.64 to 172.16.255.191

- For 172.18.0.0/16

  Range of Tofu0: 172.18.0.64 to 172.18.255.191
  Range of Tofu1: 172.19.0.64 to 172.19.255.191

- For 172.20.0.0/16

  Range of Tofu0: 172.20.0.64 to 172.20.255.191
  Range of Tofu1: 172.21.0.64 to 172.21.255.191

- For 172.22.0.0/16

  Range of Tofu0: 172.22.0.64 to 172.22.255.191
  Range of Tofu1: 172.23.0.64 to 172.23.255.191

- For 172.24.0.0/16

  Range of Tofu0: 172.24.0.64 to 172.24.255.191
  Range of Tofu1: 172.25.0.64 to 172.25.255.191

- For 172.26.0.0/16

  Range of Tofu0: 172.26.0.64 to 172.26.255.191
  Range of Tofu1: 172.27.0.64 to 172.27.255.191

- For 172.28.0.0/16

  Range of Tofu0: 172.28.0.64 to 172.28.255.191
  Range of Tofu1: 172.29.0.64 to 172.29.255.191

- For 172.30.0.0/16

  Range of Tofu0: 172.30.0.64 to 172.30.255.191
  Range of Tofu1: 172.31.0.64 to 172.31.255.191

3.8.3.3 Files to be Output

The design sheet for installation outputs the following files.

Table 3.89 Output Information of the Design Sheet

File	Description	Output Condition
partition.conf	Information definition file for disk partition	When entering information on PRIMERGY server
nodes.conf	Node information definition file	When entering information on PRIMERGY server
node_ft_*nn*.conf	Node information definition file for FX server nn is a serial number from 01. This is divided by every 27 racks (in unit of 648 BoBs = 10368 nodes).	When entering information on FX server

On the arrangement path of the design sheet for installation, a folder in the "*YYYYMMDDhhmmss*" format is automatically created and a file is output there. After outputting of the file, copy the file to the installer node in order to use it as the input of the command of the installation function. If the set information has an error, cope with it by performing this output again after modifying the design sheet for installation or by editing the output file manually.

When you have grouped nodes, the node information definition file for each of the groups is output under the "group" folder. Also, the node information definition file of all nodes is always output, regardless of whether or not to group them. The directory structure of the output file is as follows.

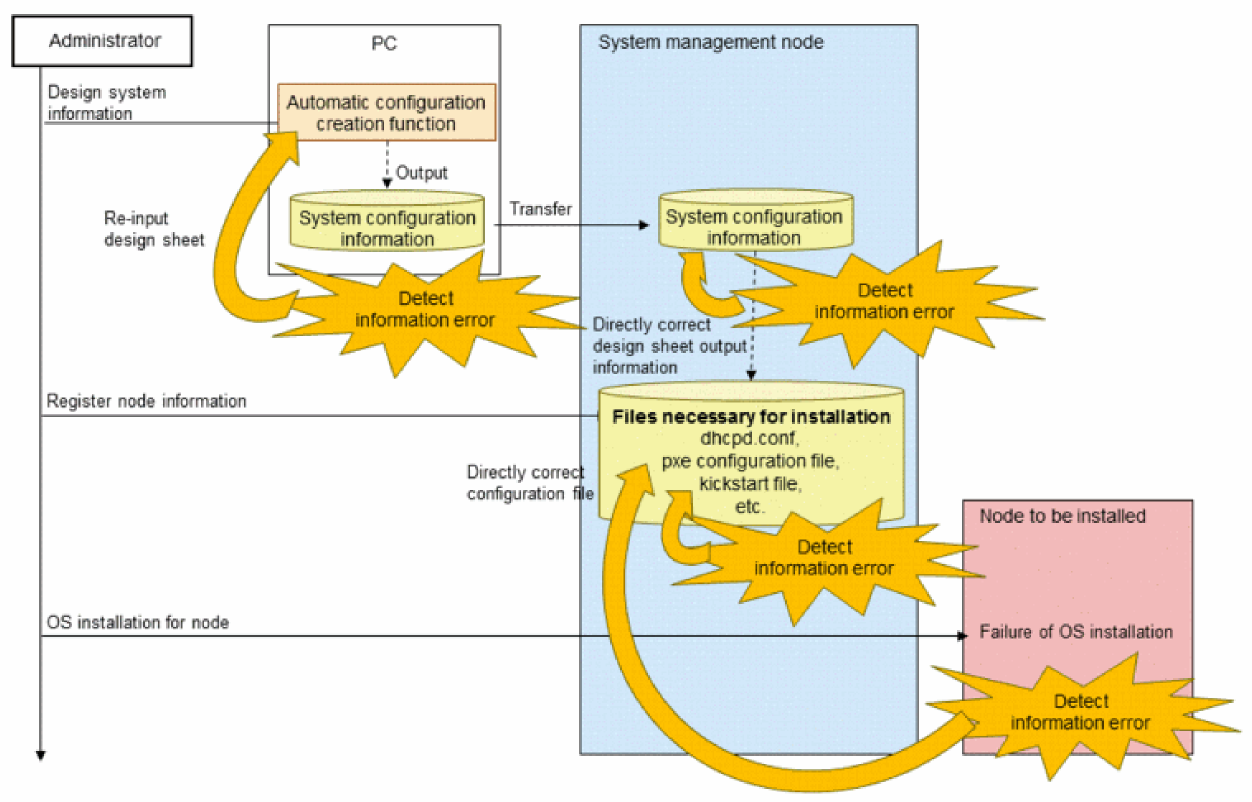
```
\---20160301093015                (*1)
    |   nodes.conf                 (*2)
    |   partition.conf            (*3)
    |
    \--- node_ft                  (*4)
      |    node_ft_01.conf        (*5)
      |    node_ft_02.conf
      |   ...
      |    node_ft_nn.conf
      |
    \---group                     (*6)
           node_RHEL72.conf       (*7)
           node_RHEL73.conf       (*8)
```

(*1) *YYYYMMDDhhmmss* folder
(*2) Node information definition file of all PRIMERGY server
(*3) Information definition file for disk partition
(*4) Folder for the node information definition file of FX server
(*5) Node information definition file of FX server
(*6) Folder for group
(*7) Node information definition file of the group name "RHEL72"
(*8) Node information definition file of the group name "RHEL73"

3.8.3.4 Countermeasure when the Input Information is Incorrect

If the design sheet for installation has an error on its input information, the countermeasure varies depending on the timing of detecting the error. When finding out an error of the information, you do not always have to go back to the creation of the design sheet for installation. This information enables you to cope with the error for each detected phase, making it possible to continue the operation with a minimum of rework. The following is the image of countermeasure for each timing of detecting an error.

Figure 3.28 Countermeasure Image when the Input Information is Incorrect


Administrator
PC
System management node
Design system information
Automatic configuration creation function
Output
Re-input design sheet
System configuration information
Transfer
System configuration information
Detect information error
Detect information error
Directly correct design sheet output information
Register node information
Files necessary for installation
dhcpd.conf,
pxe configuration file,
kickstart file,
etc.
Directly correct configuration file
Detect information error
Node to be installed
OS installation for node
Failure of OS installation
Detect information error

Information

Not all the patterns apply to the above contents. Depending on the contents of an error, it may be faster to redo the work from several steps before or from the beginning. For example, if the IP addresses of all nodes are incorrect because of the wrong starting address of IP addresses allocated to a node, redo the work from creating the design sheet for installation. This allows you to cope with this problem faster than modifying the configuration file manually.

3.8.4 Creating initrd for Installation

Create the initial RAM disk (initrd), which is required to perform the OS installation using the installation function. A directory specified by "tftpdir" of the environment configuration file of the installation function stores the created initrd as "initrd-*Repository name*.img". To use this function, you must create a target repository in advance.

3.8.4.1 Procedure for Creating initrd

The method for creating initrd is described below.

Note

Since initrd in an already-specified repository is overwritten, you must move (rename) or copy the old initrd accordingly.

```
# pxinst initrd create -R reponame
Building the initial ram disk (name=rhel version=7.2 arch=x86_64)
It may take some time...
[INFO] INST 0800 pxinst initrd completed successfully.(create initrd)
```

reponame: Repository name

3.8.4.2 Kernel Update for initrd

To import the update package of the kernel to initrd, register the update package in the repository, and then perform the normal procedure for creating initrd. This automatically imports the update package to initrd. This registration in the repository replaces vmlinuz; it is also necessary to update initrd when registering the update package of the kernel in the repository.

3.8.5 Creating the Root File System

Create the root file system (rootfs), which is necessary to perform the image-based OS installation.
The created rootfs is stored as "*Repository path*/images/rootfs.img" in a repository specified by the command option.

To use this function, you must create a target repository in advance. Also, this function is available only to the OS whose architecture is the same as the installer node.
For rootfs of FX server or a different architecture, handle it by manually placing rootfs, which has already been created on the corresponding architecture, in a repository.

See

For handling of rootfs of FX server or a different architecture, see "3.8.5.7 Manual Creation of the Root File System."

3.8.5.1 Creating a New Root File System

The procedure for creating a new root file system is described below.

Note

Since rootfs in an already-specified repository is overwritten, you must move (rename) or copy the old OS image accordingly.

```
# pxinst rootfs create -R reponame
Building the rootfs image (name=rhel version=7.2 arch=x86_64)
[INFO] INST 0800 pxinst rootfs completed successfully.(create rootfs)
```

reponame: Repository name

3.8.5.2 Adding the Package

The procedure for adding pkg1 and pkg2 to the root file system is described below. You must register pkg1 and pkg2 in a repository beforehand.

```
# pxinst rootfs addpkg -R reponame -P pkg1,pkg2
Adding 2 packages.
[INFO] INST 0800 pxinst rootfs completed successfully.(add packages)
```

reponame: Repository name
pkg1,pkg2: Package names to be added to the root file system

3.8.5.3 Updating the Package

The procedure for updating the root file system is described below. Updating the root file system updates all packages applied on rootfs to the latest version registered in the repository.

```
# pxinst rootfs updpkg -R reponame
updating rootfs
[INFO] INST 0800 pxinst rootfs completed successfully.(update packages)
```

reponame: Repository name

3.8.5.4 Editing the Root File System Manually

The procedure for manually changing a setting of the root file system is described below.

Note

- The nodes to be installed with the same OS version use rootfs in common. It is therefore recommended to implement only settings common to all nodes.
- The installation function cannot assure operations of manually-edited contents, making it possible for installation using rootfs to fail due to making a false setting. It is therefore recommended to create the backup beforehand. You can also recover it by executing the pxinst rootfs create command to recreate rootfs.

The following is an example of editing rootfs by extracting it in /var/tmp/rootfs.

1. Extracting rootfs

   ```
   # pxinst rootfs unpack -R reponame -d /var/tmp/rootfs
   unpacking rootfs
   [INFO] INST 0800 pxinst rootfs completed successfully.(unpack rootfs)
   ```

   reponame: Repository name

2. Editing rootfs

   Perform any setting for rootfs extracted in /var/tmp/rootfs.

3. Creating and placing rootfs

   To enable the edited contents, create rootfs by setting /var/tmp/rootfs as "/" and then place it in the repository.

   ```
   # pxinst rootfs pack -R reponame -d /var/tmp/rootfs
   packing rootfs
   [INFO] INST 0800 pxinst rootfs completed successfully.(pack rootfs)
   ```

   reponame: Repository name

3.8.5.5 Package Imported to the Root File System

Packages imported to rootfs become kernel packages or a basic package group of the OS. (These packages are equivalent to a minimal installation and required for operations of the installation function.)

To rootfs of PRIMERGY server, however, you can import any package by specifying a package name or a group name which the addpkg action adds to rootfs. Regarding FX server, packages imported to rootfs are fixed; the system administrator cannot change them.

Information

The following table shows the packages that are imported to rootfs of PRIMERGY server by default when the OS is RHEL7.*x*.
The individual packages listed here includes packages installed as an extension of group installation. This table does not list packages that are installed by the dependency of each package.

Table 3.90 Packages Imported to rootfs of PRIMERGY server by Default

Package Name and Group Name	Description (Abstract From Description of Group)	Remarks
@core	Minimal installation	
kernel		Applies both of latest erratas registered in GA version and repository.
grub2		
grub2-tools		

Package Name and Group Name	Description (Abstract From Description of Group)	Remarks
grub2-efi		
efibootmgr		
python		This may be python3 depending on distribution.
PyYAML		This may be python3-PyYAML depending on distribution.
libyaml		
python-jinja2		This may be python3-jinja2 depending on distribution.
python-netaddr		Only for python2 environment
python-six		This may be python3-six depending on distribution.
zlib		
rsyslog		
rsync		
chrony		
ipmitool		
kexec-tools		
authconfig		
unzip		
tar		
nfs-utils		
pciutils		
ntp		

3.8.5.6 Update of the Root File System

To import the update package of the OS, register the update package in the repository, and then update the package with the pxinst rootfs updpkg command. This automatically imports the update package to rootfs. However, this imports only packages that have already been applied on rootfs. To add a new package, execute the pxinst rootfs addpkg command.

3.8.5.7 Manual Creation of the Root File System

If rootfs has a different architecture from the installer node, it is necessary to create it on the node of the corresponding architecture. This creation procedure is described below. The format of rootfs must use the tar command to archive the directory tree starting with "/", and must be a zip-compressed.

- Example of creating rootfs (Created on a target machine): This example takes /var/tmp/rootfs as "/".

```
# yum --installroot /var/tmp/rootfs install kernel pkg1 pkg2...
# cd /var/tmp/rootfs
# tar c ./* | gzip -c > /var/tmp/rootfs.img
```

Use the repository management function to register the created rootfs.img.

3.8.6 Linkage With ServerView Suite

To run the OS on PRIMERGY server, you need a dedicated kernel module corresponding to the hardware. If not incorporating a required kernel module, a problem such as not being able to recognize the hard disk or network device may occur, preventing the installation function from operating normally.

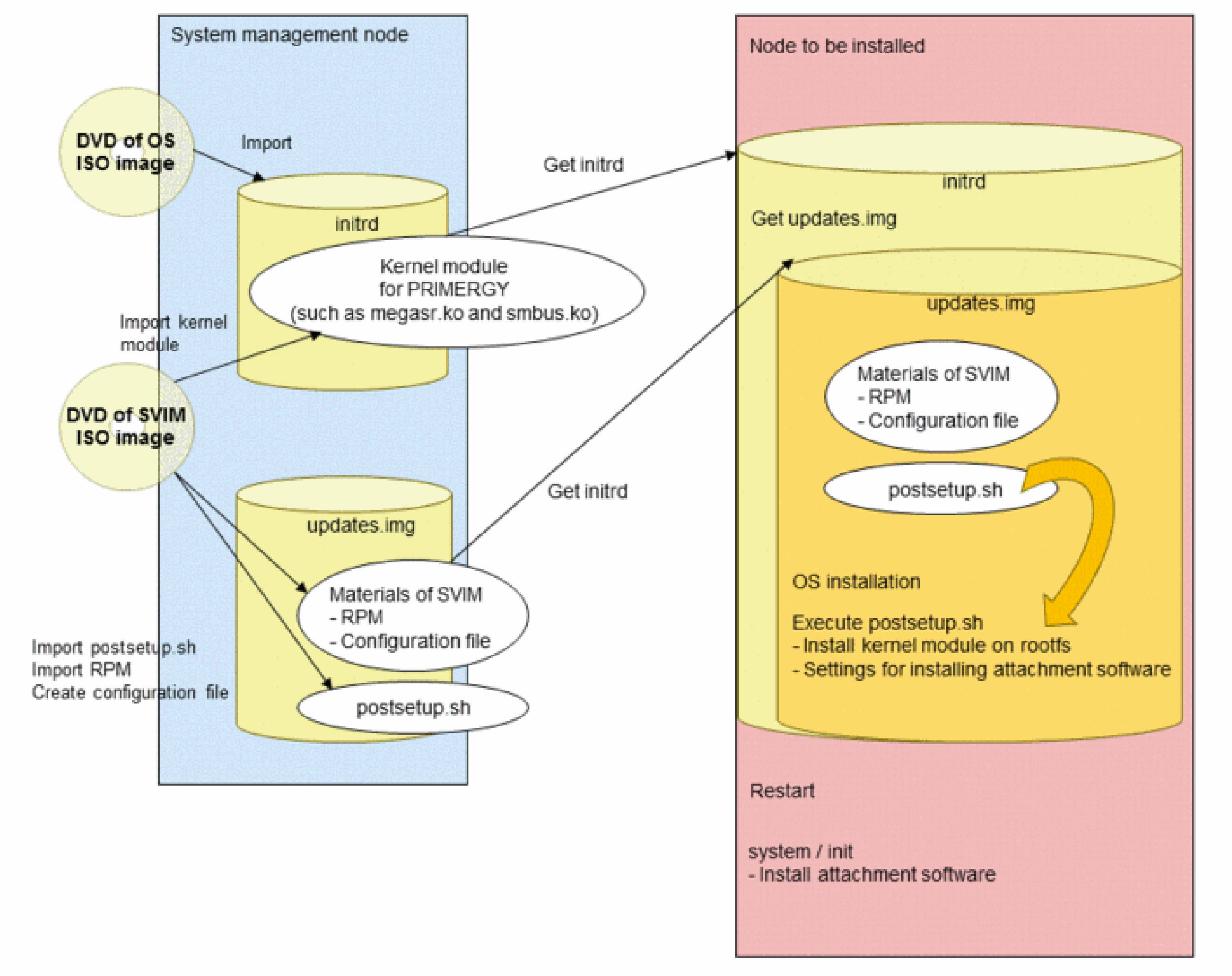
The hardware includes the DVD of the ServerView Installation Manager (SVIM), which packages the kernel module. You need to extract the kernel module from the DVD and then incorporate it into initrd and rootfs. The DVD of the SVIM also includes the ServerView attachment software (such as ServerView RAID Manager and ServerView Agent), which is used for operation and maintenance of PRIMERGY server. To apply it, you must extract it from the DVD.

The installation function provides a function associated with ServerView Suite. This function allows the system administrator to realize the OS installation without considering a kernel module necessary for hardware, and apply the required ServerView attachment software automatically.

3.8.6.1 Flow of ServerView Linkage

The following diagram shows the image of linkage with ServerView Suite during the OS installation.

Figure 3.29 Image of Linkage With ServerView Suite



3.8.6.2 Procedure for Linkage With ServerView Suite

To apply a kernel module and the attachment software, import necessary materials from the ServerView Suite DVD 1. This importing procedure is described below. To carry out this procedure, you must create both the environment configuration file of the installation function and the target repository beforehand.

1. Preparation

   Insert the ServerView Suite DVD 1 into the active system management node. This DVD came with the hardware. Or, place the ISO image file, which has already been created from the DVD, on any path of the active system management node.

2. Incorporating the kernel module and the attachment software

   Extract the kernel module and the attachment software from the DVD or the ISO image of ServerView, and then incorporate them.
   [When using the DVD]

```
# pxinst svs setup -R reponame
Copying the ServerView Suite software and kernel modules
Updating the initial ram disk
[INFO] INST 0800 pxinst svs completed successfully.(setup)
```

reponame: Repository name to import the kernel module and the attachment software

[When using the ISO image file]

```
# pxinst svs setup -R reponame -S isoimage
Copying the ServerView Suite software and kernel modules
Updating the initial ram disk
[INFO] INST 0800 pxinst svs completed successfully.(setup)
```

reponame: Repository name to import the kernel module and the attachement software
isoimage: ISO image file path of the ServerView DVD

Information

The version of the SVIM used for building a new system may be old. If so, when incorporating new hardware into the system to add a node or cluster, you must update the version of the SVIM for the installation. In this case, you can update it by using the media of new SVIM to execute the pxinst svs setup command again.

3.8.6.3 Selecting the ServerView Attachment Software

The ServerView attachment software to be applied may vary depending on system configurations or node types. Due to this reason, the system administrator can select any type of the software to be applied. When registering node information, specify which attachment software to actually apply to which node. Use the command line to specify this.

Note

- The settings of the software to be applied are common to each execution of the registration command of node information.
- For details on registering node information, see "3.8.8 Registering Node Information."

To apply the specified attachment software automatically at the time of node installation, specify the --svs option when registering node information.

The following is an example of registering node information when applying the ServerView attachment software.

```
# pxinst node add -T img -R reponame -N nodefile -P partfile --svs no
```

reponame: Repository name
nodefile: Path of the node information definition file
partfile: Path of the partition definition file
no: Number of the attachment software

Each number specified in the --svs option corresponds to the attachment software. Specify the numbers of the corresponding attachment software by separating them with commas. You can use the pxinst svs list command to confirm the corresponding numbers.

```
# pxinst svs list -R reponame
NO   SOFTWARE
--    --------
1    ServerView Agent
2    ServerView Update Agent
3    ServerView RAID Manager
4    ServerView Operations Manager
```

reponame: Repository name

Note

Depending on the version of the ServerView Suite to be used, applicable software may be different from that of this example.

3.8.7 Incorporating a Kernel Module

Some types of another company's hardware and hardware mounted on PRIMERGY server require two kinds of kernel modules when installing the OS. One is the standard kernel module of the OS, and the other is a special kernel module that is not provided by SVIM. The installation function provides a function to incorporate any kernel module into initrd. If you have already confirmed that hardware does not require the special kernel module, it is unnecessary to perform the work using this function. To import a kernel module into rootfs, use the package addition function provided by the creation function of the root file system.

In addition, this function does not support the kernel module included in SVIM, as the ServerView Suite corporation function incorporates such module.

Information

If the DVD of the ServerView Suite includes an old kernel module, which makes it necessary to incorporate the separately-downloaded RPM package, this function is also available to cope with it.

The procedure for importing a kernel module into initrd is as follows.

```
# pxinst kmod add -R reponame --rpm rpmfile
Building the initial ramdisk (name=rhel version=7.2 arch=x86_64)
[INFO] INST 0800 pxinst kmod completed successfully.(add kernel module)
```

reponame: Repository name
rpmfile: rpm file

To remove an unnecessary kernel module from initrd, perform the following procedure.

```
# pxinst kmod del -R reponame --mod modname
Building the initial ramdisk (name=rhel version=7.2 arch=x86_64)
[INFO] INST 0800 pxinst kmod completed successfully.(delete kernel module)
```

reponame: Repository name
modname: Kernel module name

3.8.8 Registering Node Information

In the configuration files, register information necessary for installing the installation target node.
To install the OS, it is necessary to add the entry of host information to the DHCP server. You also need to create the PXE configuration file describing the settings of the PXE boot, as well as the kickstart file used by anaconda. The registration function of node information provides functions to set information that is necessary for the configuration file of each service.

In registering of the information, consider the file output by the design sheet for installation as the input one. For PRIMERGY server in a small-scale environment, you can create the input file manually without using the design sheet for installation. In a large-scale environment, it is recommended to use the design sheet for installation in consideration of the effort for work.

3.8.8.1 Name and Placement Location of the Input File

There is no rule for the name and the arrangement path of the input file. Place it in any location on the system management node. Its format uses the option of a command to specify the input file when executing the command.

3.8.8.2 Types of Input Files

The configuration files necessary for registering node information are the input file for PRIMERGY server and the input file for FX server. To use this function, you must create and arrange both of them in advance. It is required to add the read-out authority of the arranged files to the root user. In addition, you can use the design sheet for installation to create the input file automatically.

Table 3.91 Input File Necessary for Registering Node Information (PRIMERGY server)

Type of Configuration File	File Name	Setting Contents
Node information definition file	Optional	Defines the configuration information of a node, such as the host name and the IP address.
Information definition file for disk partition	Optional	Defines the configuration information of a disk partition.

Table 3.92 Input File Necessary for Registering Node Information (FX server)

Type of Configuration File	File Name	Setting Contents
Node information definition file for FX server	Optional	Defines the configuration information of a node, such as the host name, IP address, and Tofu coordinates.

Information

The following sample files for the manual creation are provided:

- Node information definition file

  /etc/opt/FJSVtcs/sample/pxinst_nodedef.conf

- Information definition file for disk partition

  /etc/opt/FJSVtcs/sample/pxinst_partdef.conf

- Node information definition file for FX server

  /etc/opt/FJSVtcs/sample/pxinst_nodedef_ft.conf

3.8.8.3 Format of the Input File

Describe the configuration files in the yaml format.

See

Normally, the input file is output by using the design sheet for installation. If the design sheet for installation has incorrect input information, you can edit it manually. The format of the file is listed as reference information.

3.8.8.4 Node Information Definition File

The node information definition file is a configuration file to define the information on the configuration of the installation target node in PRIMERGY server. Based on the contents of the node information definition file, install the OS and perform settings after the installation.

In general, this uses the design sheet for installation; it is unnecessary for the system administrator to consider the formats and input items of the file. The setting items are described below in the case when:
- Manually creating the file when the number of nodes is small, or
- Directly modifying incorrect input information (such as a false host name or IP address) on the design sheet for installation

The items to be defined in the node information definition file are as follows.

Table 3.93 Setting Items of the Node Information Definition File

Key	Meaning	Data Type	Value	Required or Optional	Remarks
nodetype	Node type	Character string	Node type Example: CCM	Optional	Any node type name for the system administrator to identify it If cooperating with the node configuration definition function in the system management, however, you must specify an appropriate node type. For details, see "Table 3.75 Input Items of the nodes Sheet."
hostname	Host name	Character string	Host name Example: ccm01	Required	The only valid symbol for a hostname is hyphen(-).
networks	Network definition section	Array		Required	See "Table 3.94 Network Definition Section."
partition	Partition definition ID	Numerical	ID defined by the partition definition file	Required	Specify "id" of the disk partition definition file.

The following table describes the items of the network definition section.

Table 3.94 Network Definition Section

Key	Meaning	Data Type	Value	Required or Optional	Remarks
device	Network device name	Character string	Specify either device name or MAC address. Example: bond0 Example: 00:00:00:00:00:01 For the installation target device (boot=True), you must specify MAC address. For IPMI, specify "bmc."	Required	
ip	IP address to be allocated	Character string	Specify dotted IP address. Example: 192.168.3.10	Required	
mask	Subnet mask	Character string	Specify dotted IP address. Example: 255.255.255.0	Optional	You cannot specify this together with prefix. You must specify only either of them.
prefix	Prefix of a subnet mask	Numerical	Specify a subnet mask value in the bit format. [For 255.255.255.0] Example: 24	Optional	You cannot specify this together with mask. You must specify only either of them.

Key	Meaning	Data Type	Value	Required or Optional	Remarks
boot	Specify whether or not to use the network for installation.	Boolean	true: Used false: Not used	Optional	Default value: "false"
gateway	Default gateway	Character string	Specify dotted IP address of the default gateway. Example: 192.168.3.10	Optional	
bondslaves	Slave devices for using the channel bonding of multiple network interfaces	Character string	Specify MAC address by separating them with commas. Example: 00:00:00:00:00:02, 00:00:00:00:00:03	Optional	Specifying this key regards them as bonding.
bondopts	Bonding options for using the channel bonding of multiple network interfaces	Character string	Specify the options by separating them with semicolons. Example: 'mode=active-backup;primary=00:00:00:00:00:02;miimon=80;updelay=50000'	Optional	

An example of the node information definition file is as follows.

```
# node definition file
# generated by pxinst
-
  nodetype: LN
  hostname: login1
  networks:
  - {device: '00:23:8b:97:ca:db', ip: 192.168.3.7, prefix: 24, boot: true}
  - {device: '00:23:8b:97:ca:de', ip: 192.168.30.7, prefix: 18}
  partition: 1
-
  nodetype: LN
  hostname: login2
  networks:
  - {device: '00:23:8b:97:ab:20', ip: 192.168.3.8, prefix: 24, boot: true}
  - {device: '00:23:8b:97:ab:21', ip: 192.168.30.8, prefix: 18}
  partition: 1
```

3.8.8.5 Disk Partition Definition File

The disk partition definition file is a configuration file to define the configuration of a disk partition for PRIMERGY server nodes. Use "id" to associate the configuration of a disk partition with a node. You can use the same disk partition definition on multiple nodes.

As with the node information definition file, it is unnecessary for the system administrator to consider the formats and input items of the file when using the design sheet for installation. The settable items are described below in the case when manually creating the file or modifying incorrect input information on the design sheet for installation.

The items to be defined in the disk partition definition file are as follows.

Table 3.95 Setting Items of the Disk Partition Definition File

Key	Meaning	Data Type	Value	Required or Optional	Remarks
id	Identification number to associate it with node information	Numerical	Any numerical value Example: 1	Required	
partition	Partition section	Array		Required	See "Table 3.96 Partition Section."

The items of the partition section are as follows.

Table 3.96 Partition Section

Key	Meaning	Data Type	Value	Required or Optional	Remarks
fstype	File system type	Character string	File system Example: xfs	Required	You can specify the following values: ext3(*), ext4, xfs, fat32 (only the EFI system partition is supported) and swap (*)ext3 is not recommended for RHEL8.
mountpoint	Mount point	Character string	Any mount point Example: /boot	Required	
onpart	Device path to create file system	Character string	Device path Example: /dev/sda1	Required	
size	Size of disk partition	Numerical	Size Example: 512	Required	Specify this in MB.
label	Disk label	Character string	Disk label This definition must be the same in the same disk. Example: msdos and gpt	Required	Disks that do not contain an EFI system partition default to msdos and should be modified directly as needed.

An example of the disk partition definition file is as follows.

```
# disk partition definition file
-
  # for BIOS
  id: 1
  partition:
  - {fstype: 'xfs', mountpoint: '/boot', onpart: '/dev/sda1', size: 256, label: msdos}
  - {fstype: 'swap', mountpoint: '', onpart: '/dev/sda2', size: 4096, label: msdos}
  - {fstype: 'xfs', mountpoint: '/', onpart: '/dev/sda3', size: max, label: msdos}
-
  # for UEFI
  id: 2
  partition:
  - {fstype: 'fat32', mountpoint: '/boot/efi', onpart: '/dev/sda1', size: 512, label: 'gpt'}
  - {fstype: 'xfs', mountpoint: '/', onpart: '/dev/sda2', size: max, label: 'gpt'}
```

3.8.8.6 Node Information Definition File for FX server

The node information definition file for FX server is a configuration file to define the information on the configuration of FX server nodes. Based on the contents of the node information definition file, install the OS and perform settings after the installation. The design sheet for installation automatically generates this file. It is generally unnecessary for the system administrator to consider the formats and input items of the file. Since it can be assumed to edit this file directly in order to modify incorrect input information (such as a false IP address) on the design sheet for installation, the following table describes its setting items.

The items to be defined in the node information definition file for FX server are as follows.

Table 3.97 Setting Items of the Node Information Definition File for FX server

Key	Meaning	Data Type	Value	Required or Optional	Remarks
arpmode	ARP mode of Tofu driver	Numerical	0: Dynamic ARP mode 1: Static ARP mode (10.0.0.0/8) 2: Static ARP mode (172.16.0.0/12) 3: NOARP mode	Required	This is used to set the Tofu driver of the installation target node.
bob	Information on BoB	Array	-	Required	-
bobno	BoB number (serial)	Numerical	Value of 0 or more	Required	This is information for identification; you can specify any value here.
nodes	Node information inside BoB	Array	-	Required	-
node	Node number inside BoB	Numerical	Value from 0 to 15	Required	This must be in accordance with hardware specification.
tofu	Tofu coordinates	Character string	X,Y,Z,a,b,c format Example: '0,0,0,0,0,0'	Required	This must be in accordance with hardware specification.
nodetype	Node type	Character string	Node type Example: BIO	Required	This must be in accordance with hardware specification.
hostname	Host name	Character string	Host name Example: node001	Required	The only valid symbol for a hostname is hyphen(-).
networks	Network definition section	Array		Required	See "Table 3.98 Network Definition Section."
iorange	Range of Tofu coordinates in charge of I/O	Character string	X,Y,Z,b-X,Y,Z,b format Example: For each BoB 0,0,0,0-0,0,3,0 0,0,0,0-0,0,3,1 For each shelf 0,0,0,0-0,0,3,2 0,0,4,0-,0,0,7,2 For each rack 0,0,0,0-1,1,7,2 0,0,8,0-1,1,15,2 (*)	Optional	This is range of GIO group or SIO group. This is required for global I/O node or storage I/O node.

(*) iorange is automatically set according to the input contents of the design sheet of the installation function. (This is set according to the input contents of "NodeConfig" in the user form.) If selecting [Manually assign] in [GIO configuration] or [SIO configuration], however, the system administrator must edit the node information definition file for FX server directly in order to describe iorange.

The items of the network definition section are as follows.

Table 3.98 Network Definition Section

Key	Meaning	Data Type	Value	Required or Optional	Remarks
device	Device name or MAC address of a network device	Character string	For Ethernet, specify a colon-separated MAC address. For Tofu network, set both tofu0 and tofu1 as fixed values. For Infiniband, specify ib0 as a fixed value.	Required	You can set " (a null character) for Gb Ethernet when using a dynamic IP address.
ip	IP address to be assigned	Character string	Dotted IPv4 address Example: 203.0.113.1	Required	
mask	Subnet mask	Character string	Dotted IPv4 address Example: 255.255.255.0	Optional	You cannot specify this together with prefix. You must specify only either of them.
prefix	Prefix of a subnet mask	Numerical	A subnet mask value in the bit format [For 255.255.255.0] Example: 24	Optional	You cannot specify this together with mask. You must specify only either of them.
mac	MAC address	Character string	This is required for tofu0 of boot I/O nodes	Optional	
boot	Whether or not the network is used for installation	Boolean	true: Used false: Not used	Optional	For boot I/O node, specify true in Ethernet. For other node types, specify true in Tofu network. Default value: false

An example of the node information definition file is as follows.

```
# node definition file for FX server
arpmode: 2
bob:
-
  bobno: 0
  nodes:
  - node: 0
    tofu: '0,0,0,0,0,0'
    nodetype: 'CN'
    hostname: 'node0000001'
    networks:
    - {device: 'tofu0', ip: '10.224.1.0', prefix: 11, boot: true}
    - {device: 'tofu1', ip: '10.240.1.0', prefix: 11}
    - {device: 'bmc', ip: '194.0.0.1', prefix: 8}
...
  - node: 4
    tofu: '0,0,2,0,0,0'
    nodetype: 'BIO'
    hostname: 'node0000005'
    networks:
    - {device: 'tofu0', ip: '10.224.1.8', mac: '00:00:00:00:20:01', prefix: 11}
    - {device: 'tofu1', ip: '10.240.1.8', prefix: 11}
    - {device: '00:00:00:00:00:02', ip: '192.168.122.100', prefix: 14,boot: true}
    - {device: 'bmc', ip: '194.0.0.1', prefix: 8}
...
```

3.8.8.7 Registering Node Information

3.8.8.7.1 Registering PRIMERGY server

The procedure for registering node information of PRIMERGY server is as follows.

1. Registering node information

   To register information matching with the provisioning method, specify the --type option when registering node information.

   a. Registering node information (Image-Based OS installation)

   ```
   # pxinst node add --repo reponame --type img --nodedef nodefile --partdef partfile
   [INFO] INST 1803 pxinst Adding nodes completed.(nodenum=xxx)
   ```

   reponame: Repository name
   nodefile: Path of the node information definition file
   partfile: Path of the partition information definition file
   xxx: The number of nodes

   b. Registering node information (Package-Based OS installation)

   ```
   # pxinst node add --repo reponame --type pkg --nodedef nodefile --partdef partfile
   [INFO] INST 1803 pxinst Adding nodes completed.(nodenum=xxx)
   ```

   reponame: Repository name
   nodefile: Path of the node information definition file
   partfile: Path of the partition information definition file
   xxx: The number of nodes

2. Restarting the service

   Restart the DHCP service to enable the settings.

   ```
   # systemctl restart dhcpd
   ```

3. Synchronizing the setting

   Synchronize the settings for the standby installer node and the relay node.

   ```
   # pxinst sync all
   ```

3.8.8.7.2 Registering FX server (When Not Using a Dynamic IP Address)

The following is the procedure for registering node information of FX server when not using a dynamic IP address.

1. Registering node information

   To register node information of FX server, specify the --ft option, which indicates FX server, during this registration.

   ```
   # pxinst node add --repo reponame --nodedef filepath --ft
   [INFO] INST 1803 pxinst Adding nodes completed.(nodenum=xxx)
   ```

   reponame: Repository name
   filepath: Path of the node information definition file for FX server
   xxx: The number of registered nodes

2. Restarting the service

   Restart the DHCP service to enable the setting.

   ```
   # systemctl restart dhcpd
   ```

3. Synchronizing the setting

   Synchronize the setting for the standby installer node and the relay node.

```
# pxinst sync all
```

3.8.8.7.3 Registering FX server (When Using a Dynamic IP Address)

The following is the procedure for registering node information of FX server when using a dynamic IP address to perform the installation.

1. Registering node information

   To register node information of FX server, specify the --ft option, which indicates FX server, during this registration. At the same time, use the --pool option to specify the range of the IP address pool, which allocates IP addresses dynamically. You can specify multiple pools by separating them with commas.

   [An example of setting the pool of dynamic IP addresses from 192.0.2.10/24 to 192.0.2.225/24]

```
# pxinst node add --repo reponame --nodedef filepath --ft --pool 192.0.2.10/24-192.0.2.225/24
[INFO] INST 1803 pxinst Adding nodes completed.(nodenum=xxx)
```

   reponame: Repository name
   filepath: Path of the node information definition file for FX server
   xxx: The number of nodes

2. Restarting the service

   Restart the DHCP service to enable the setting.

```
# systemctl restart dhcpd
```

3. Synchronizing the setting

   Synchronize the setting for the standby installer node and the relay node.

```
# pxinst sync all
```

3.8.8.8 Set Information

The following table shows the files created and updated by setting the registration function of node information.

Table 3.99 Items Set by Registering Node Information (PRIMERGY server)

Setting Item	File	Setting Contents
Settings of the DHCP Server	/etc/dhcp/dhcpd.conf	Add the definition of host of the installation target node.
Settings of the TFTP server (BIOS)	"*tftpdir*"/pxelinux/pxelinux.cfg/*xxx* *xxx*: Hexadecimal IP address for provisioning	Create the PXE configuration file for each node. *"tftpdir"* refers to the directory specified by *"tftpdir"* of the environment configuration file of the installation function.
Settings of the TFTP server (UEFI)	"tftpdir"/uefi/grub.cfg-*xxx* *xxx*: Hexadecimal IP address for provisioning	Create grub.cfg for each node. "tftpdir" refers to the directory specified by "tftpdir" of the environment configuration file of the installation function.
Creating the kickstart file	"tftpdir"/kickstart/ks.cfg.*IP address for provisioning*	Create the kickstart file for each node. "tftpdir" refers to the directory specified by "tftpdir" of the environment configuration file of the installation function.

The following shows examples of the created configuration files.

Settings of the DHCP Server (dhcpd.conf)

```
/* ----- Host definition ----- */
        group {
                next-server 192.168.3.1;
                host smm02 {
                        hardware ethernet 00:19:99:21:80:45;
                        fixed-address 192.168.3.3;
                        option host-name "smm02"
                }
                host ccm01 {
                        hardware ethernet 00:23:8b:97:ca:db;
                        fixed-address 192.168.3.7;
                        option host-name "ccm01"
                }
        }...
/* ------------------------------------ */
```

Settings of the TFTP Server (pxelinux.cfg)

```
# PXE configuration file for: node01
default install
prompt 0

label localdisk
        kernel chain.c32
        append hd0 0

# for install function
label install
        kernel vmlinuz-rhel-7.3-x86_64
        IPAPPEND 2
        append initrd=initrd-rhel-7.3-x86_64 inst.repo=http://192.168.4.110/pxinst/repos/rhel73
inst.ks=http://192.168.4.110/pxinst/kickstart/ks.192.168.4.12 inst.text inst.sshd selinux=0
biosdevname=0 pxinst.type=img pxinst.rootfs=/images/rootfs.img pxinst.updates=/images/updates.img
console=ttyS0 pxinst.nosvs

# for backup function
label backup
        kernel vmlinuz-rhel-7.3-x86_64
        IPAPPEND 2
        append initrd=initrd-rhel-7.3-x86_64 inst.repo=http://192.168.4.110/pxinst/repos/rhel73
inst.ks=http://192.168.4.110/pxinst/kickstart/ks.192.168.4.12 inst.text inst.sshd selinux=0
biosdevname=0 pxinst.type=backup pxinst.rootfs=/images/rootfs.img pxinst.updates=/images/
updates.img console=ttyS0 pxinst.nosvs

# for restore function
label restore
        kernel vmlinuz-rhel-7.3-x86_64
        IPAPPEND 2
        append initrd=initrd-rhel-7.3-x86_64 inst.repo=http://192.168.4.110/pxinst/repos/rhel73
inst.ks=http://192.168.4.110/pxinst/kickstart/ks.192.168.4.12 inst.text inst.sshd selinux=0
biosdevname=0 pxinst.type=restore pxinst.rootfs=/images/rootfs.img pxinst.updates=/images/
updates.img console=ttyS0 pxinst.nosvs
```

kickstart File

```
# Redhat kickstart template
#
# Copyright (C) Fujitsu limited 2016
#
# Licensed under GPL version 2; See LICENSE file for details.
#
```

```
# System authorization information
auth --useshadow --passalgo=sha512 --enablecache

# Install OS instead of upgrade
install
# Reboot after installation
reboot

# Use text mode install
text
# Firewall configuration
firewall --disabled
firstboot --disable
# Keyboard layouts
keyboard jp106
# System language
lang C

# Network information
network --device=00:23:8B:97:CA:36 --bootproto=static --ip=192.168.4.13 --noipv6 --onboot=yes --
netmask=255.255.255.0 --gateway=192.168.4.254 --hostname=ct-bl014 --activate
network --device=ib0 --bootproto=static --ip=192.168.10.8 --noipv6 --onboot=yes --
netmask=255.255.255.0

# Root password
rootpw --iscrypted $6$k9Zu2Cw0lkYHl94N$M/NAlqgJ1N91uFKcEIzGJGFY3TWj5d1O/AVWhg/
PZRJZNDBbFyvlrrYR4JAFRDHVTIP89whCTthNwaKbtp.s30

# SSH configuration
sshpw --username=root --iscrypted $6$k9Zu2Cw0lkYHl94N$M/NAlqgJ1N91uFKcEIzGJGFY3TWj5d1O/AVWhg/
PZRJZNDBbFyvlrrYR4JAFRDHVTIP89whCTthNwaKbtp.s30

# SELinux configuration
selinux --disabled

# configure the X Window System
skipx
# System timezone
timezone Asia/Tokyo

# System bootloader configuration
bootloader --append="rd.retry=360 crashkernel=128M" --location=mbr
ignoredisk --only-use=sda

# Partition clearing information
#
# clearpart --all --drives=sda
#

# Clear the Master Boot Record
# zerombr

# Disk partitioning information
part /boot --fstype=xfs --onpart=/dev/sda1 --size=512 --asprimary
part swap --fstype=swap --onpart=/dev/sda2 --size=4096 --asprimary
part / --fstype=xfs --onpart=/dev/sda3 --size=1 --grow  --asprimary

%packages --ignoremissing
@core
kernel
grub2
grub2-tools
grub2-efi
```

```
shim
efibootmgr
libyaml
ntp
zlib
chrony
rsyslog
ipmitool
kexec-tools
authconfig
rsync
unzip
tar
nfs-utils
FJSVpxinst
python
PyYAML
python-jinja2
python-netaddr
python-six
%end


%pre
LANG=C
if [ "${PXINST_PROV_TYPE}" != "ram" ]; then
dd if=/dev/zero of=/dev/sda count=1 bs=512
/usr/sbin/parted -s /dev/sda mklabel msdos
disk_size=`/usr/sbin/parted /dev/sda unit MB print | grep "^Disk /dev/sda" | awk '{print $3}'`
/usr/sbin/parted -s -a optimal /dev/sda -- mkpart primary 1MB 513MB
/usr/sbin/parted -s -a optimal /dev/sda -- mkpart primary 514MB 4610MB
/usr/sbin/parted -s -a optimal /dev/sda -- mkpart primary 4611MB ${disk_size}
/usr/sbin/parted /dev/sda u MB print
udevadm settle
fi
%end


%post --nochroot
#!/bin/bash
/bin/bash /run/install/pxinst/post/all-exec
%end

%addon com_redhat_kdump --enable
```

3.8.8.9 Deleting Node Information

The procedure for deleting node information is as follows.

[Deleting node information by specifying a host name]

```
# pxinst node del --hostname hostname,...
[INFO] INST 0801 pxinst Created a configuration backup.(/tmp/pxinst/node_20180720140923/)
[INFO] INST 1804 pxinst Deleting nodes is complete.(nodenum=num)
```

hostname: Host name of the node to be deleted

num: The number of deleted nodes

[Deleting node information by specifying host names in serial order]

```
# pxinst node del --hostname hostname[001-100]
[INFO] INST 0801 pxinst Created a configuration backup.(/tmp/pxinst/node_20180720140923/)
[INFO] INST 1804 pxinst Deleting nodes is complete.(nodenum=num)
```

hostname[001-100]: Specifying the host names of nodes to be deleted in serial order

In this example, node information ranging from hostname001 to hostname100 is deleted.

num: The number of deleted nodes

[Deleting node information by specifying a file]

When there are a large number of nodes to be deleted, this specification is recommended.

1. Creating the input file

   Create a file that lists the host names of nodes to be deleted. You can set any name and placement location of the file.
   The following is an example of creating the file in the /var/tmp directory and naming it hostlist.

   ```
   # vi /var/tmp/hostlist
   host001
   host002
   host003
   ```

2. Deleting node information

   ```
   # pxinst node del -f /var/tmp/hostlist
   [INFO] INST 0801 pxinst Created a configuration backup.(/tmp/pxinst/node_20180720140923/)
   [INFO] INST 1804 pxinst Deleting nodes is complete.(nodenum=num)
   ```

   num: The number of deleted nodes

3.8.8.10 Updating Node Information

Update node information when rebuilding a node or using the backup/restore function.

The procedure for updating node information is as follows.

If changing configuration information necessary for the installation (such as IP address and MAC address), register the node information again by specifying add in the pxinst node sub command.

The following is an example of updating node information to make the image-based installation work at the next starting time.

[Updating node information by specifying a host name]

```
# pxinst node update --type img --hostname hostname,...
[INFO] INST 0801 pxinst Created a configuration backup.(/tmp/pxinst/node_20180720140923/)
[INFO] INST 0800 pxinst node completed successfully.(update)
```

hostname: Host name of the node to be updated

[Updating node information by specifying host names in serial order]

```
# pxinst node update --type img --hostname hostname[001-100]
[INFO] INST 0801 pxinst Created a configuration backup.(/tmp/pxinst/node_20180720140923/)
[INFO] INST 0800 pxinst node completed successfully.(update)
```

hostname[001-100]: Specifying the host names of nodes to be updated in serial order

In this example, node information ranging from hostname001 to hostname100 is updated.

3.8.9 Checking Installation States

The installation state checking function checks the progress, success and failure of the installation of a node. If an installation fails, the system administrator is to cope with it according to the displayed details message. This function displays nodes whose information is registered by using the installation function.

Note

The OS installation using the backup image has a different checking method. See "3.9.4 Checking Status" in "3.9 Details of the Backup/Restore Function." Also, node information is still to be displayed after the installation is complete; the system administrator must delete it explicitly if necessary.

3.8.9.1 Displaying Installation States

The following is an example of checking installation states.

```
# pxinst status show
HOSTNAME       OPERATION      STATUS        STATE         DETAIL
--------       ---------      ------        -----         ------
host01         ImgProvision   Completed     Succeeded     -
host02         ImgProvision   InProgress    Installing    -
host03         ImgProvision   Completed     Failed        detail
```

detail: Detailed information

The displayed items are as follows.

Table 3.100 Displayed Items of the Installation State Checking Function

Item	Meaning	Value
HOSTNAME	Host name of the installation target node	Host name
OPERATION	Type of operation	PkgProvision: Package-Based OS installation ImgProvision: Image-Based OS installation Backup: Backup (*) Restore: Restoration (*)
STATUS	Status of an operation	NotStarted: Before starting operation InProgress: Operation in progress Completed: Operation completed
STATE	State of processing	Waiting: Period from node start waiting time to installation processing start time Preparing: During preparing for installation processing Partitioning: During creating disk partition Installing: During installing OS Posting: During processing after installation Rebooting: During restarting Succeeded: Normal end Failed: Abnormal end PostFailed: Failure in post processing
DETAIL	Detailed information	Displays the information on the cause of an abnormality with up to 255 characters. When STATE is anything other than "Failed" and "PostFailed," "-" (hyphen) appears here.

(*) To check the detailed status, use the backup/restore function. See "3.9 Details of the Backup/Restore Function."

When STATE is "Failed" or "PostFailed," see the "Job Operation Software Troubleshooting" in order to take an appropriate action.

Note

When applying the attachment software of the ServerView Suite, it may take 10 to 20 minutes for STATE to transition from "Rebooting" to "Succeeded." This is the time taken to apply, configure, and start the attachment software. The installation function waits for the attachment software to complete the processing, and then makes it transition to the installation completion (Succeeded) status. If you restart a node before transitioning to "Succeeded," note that there is no guarantee of operations.

3.8.9.2 Deleting Unnecessary Status Displays

Once the information on a node becomes a display object due to its installation, it continuously appears unless the system administrator deletes it. The following is the procedure for deleting needless node information, which is no longer necessary to be referred.

[Example of deleting all information whose installation is successfully complete]

```
# pxinst status clear
[INFO] INST 0800 pxinst status completed successfully.(clear)
```

[Example of deleting information on a node whose installation has not started]

```
# pxinst status clear --status NotStarted
[INFO] INST 0800 pxinst status completed successfully.(clear)
```

[Example of deleting information on a specific node]

```
# pxinst status clear --hostname hostname1,hostname2
[INFO] INST 0800 pxinst status completed successfully.(clear)
```

hostname1,*hostname2*: Host name of a node whose information is to be deleted

[Example of deleting information on a specific node by specifying host names in serial order]

```
# pxinst status clear --hostname hostname[001-100]
[INFO] INST 0800 pxinst status completed successfully.(clear)
```

hostname[001-100]: Specifying the host names of nodes to be deleted in serial order

In this example, node information ranging from hostname001 to hostname100 is deleted.

3.8.9.3 Status Transition

The following diagrams show the status transition of OPERATION, STATUS, and STATE for each provisioning method.

Figure 3.30 Status Transition Diagram for Image-Based Installation

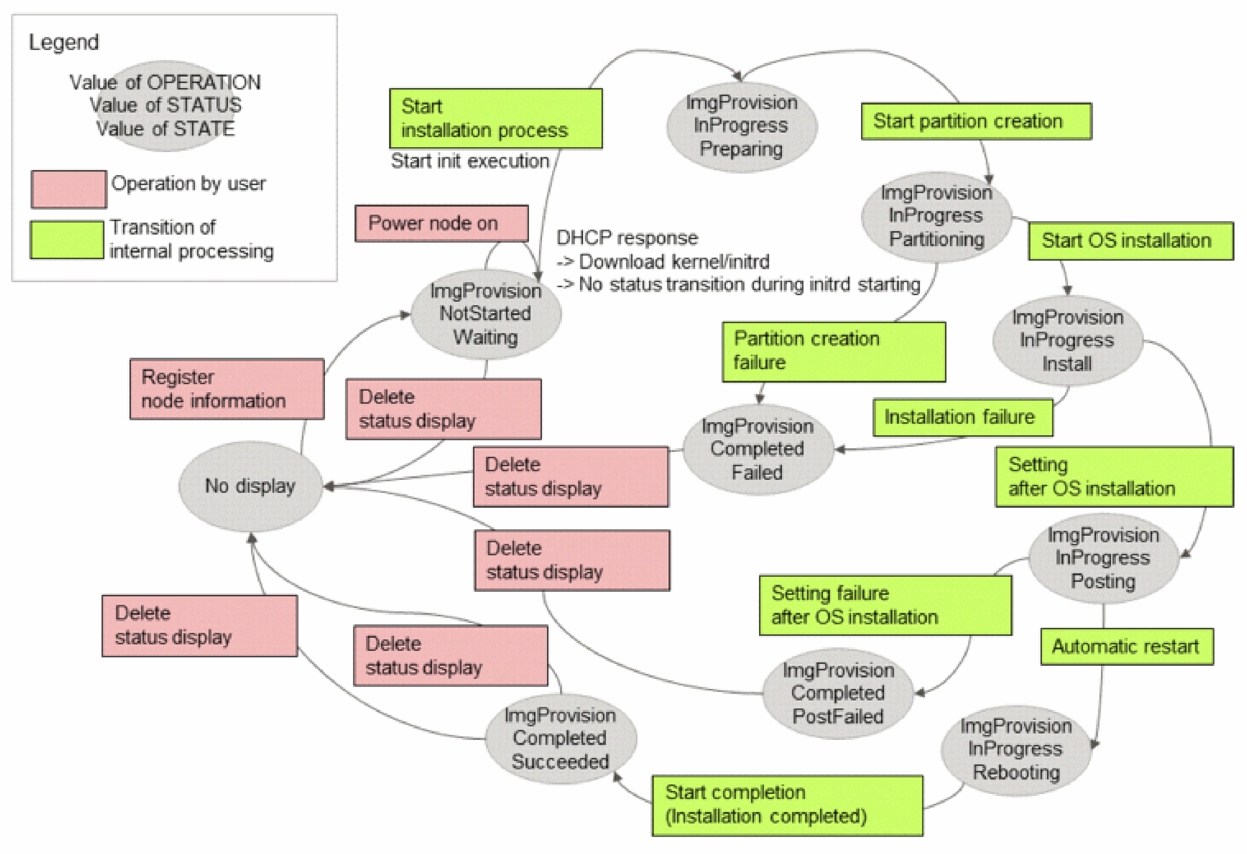
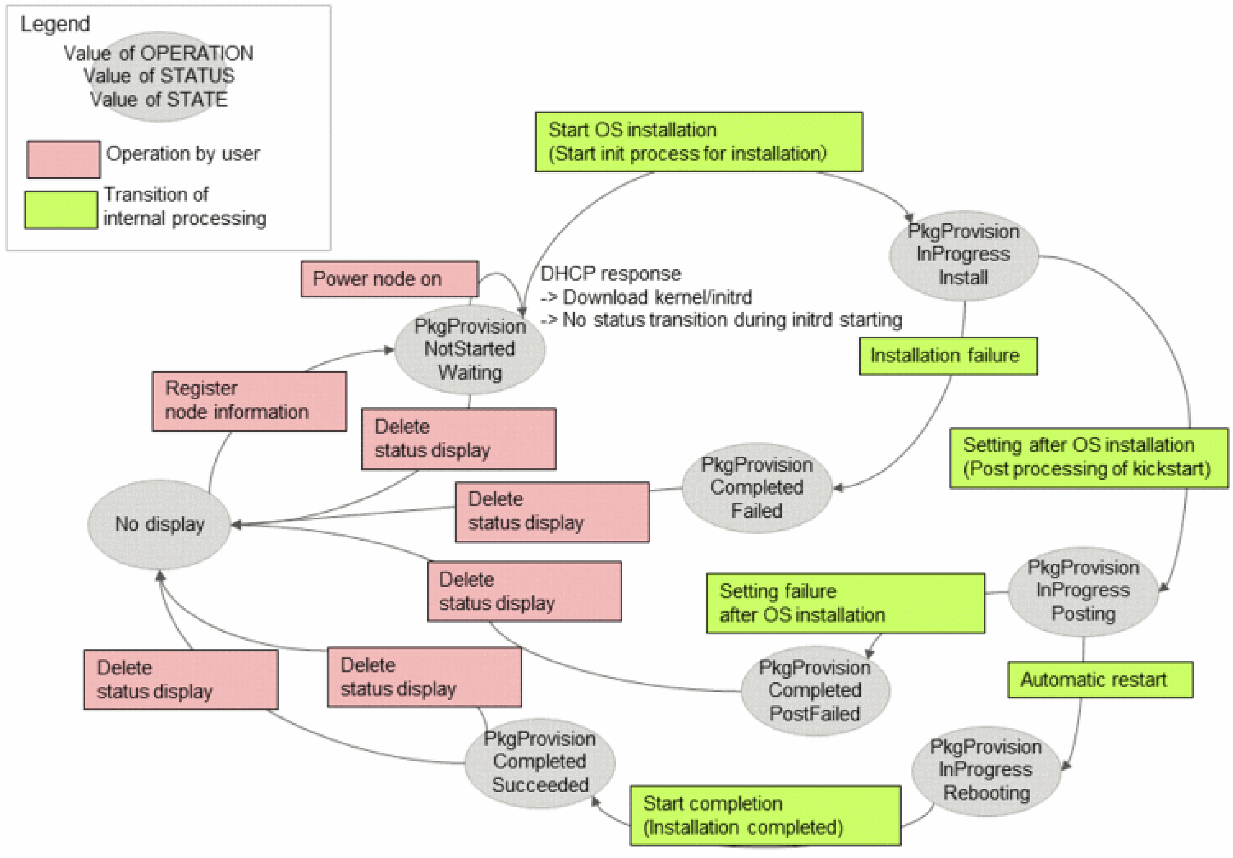

Legend
Value of OPERATION
Value of STATUS
Value of STATE
🟥 Operation by user
🟩 Transition of internal processing
Start installation process
Start init execution
ImgProvision InProgress Preparing
Start partition creation
ImgProvision InProgress Partitioning
Power node on
DHCP response
-> Download kernel/initrd
-> No status transition during initrd starting
Start OS installation
ImgProvision NotStarted Waiting
ImgProvision InProgress Install
Partition creation failure
Register node information
Delete status display
Installation failure
ImgProvision Completed Failed
No display
Delete status display
Setting after OS installation
Delete status display
ImgProvision InProgress Posting
Setting failure after OS installation
Delete status display
Delete status display
Automatic restart
ImgProvision Completed PostFailed
ImgProvision Completed Succeeded
ImgProvision InProgress Rebooting
Start completion (Installation completed)

Figure 3.31 Status Transition Diagram for Package-Based Installation


Legend
Value of OPERATION
Value of STATUS
Value of STATE
🟥 Operation by user
🟩 Transition of internal processing
Start OS installation (Start init process for installation)
PkgProvision InProgress Install
Power node on
DHCP response
-> Download kernel/initrd
-> No status transition during initrd starting
PkgProvision NotStarted Waiting
Installation failure
Register node information
Delete status display
Setting after OS installation (Post processing of kickstart)
PkgProvision Completed Failed
No display
Delete status display
Delete status display
Setting failure after OS installation
PkgProvision InProgress Posting
Delete status display
Delete status display
PkgProvision Completed PostFailed
Automatic restart
PkgProvision Completed Succeeded
PkgProvision InProgress Rebooting
Start completion (Installation completed)

If restarting a node after transiting to a specific status during the OS installation, the installation processing works from the beginning again, which returns the status to the start status of installation. (The start status for image-based installation is ImgProvision InProgress Partitioning.) Also, if an event accompanied with the status transition of a node occurs after deleting node information (such as when rebuilding a node), this becomes an object of status display again.

3.8.9.4 Synchronization Between the Active System and the Standby System

The active system of the installer node holds the master of status information, and the standby system holds its copy. The status information is not synchronized automatically. If necessary, execute the pxinst sync sub command to synchronize it.
The installation function also provides the function to automatically synchronize it with the standby system at regular intervals (every five minutes). Enable the service accordingly.

[How to set the periodical synchronization]

```
# systemctl enable pxinst-replicate.timer
# systemctl start pxinst-replicate.timer
```

3.8.10 Synchronization of the Materials and Settings for Installation

This section describes the details on synchronization of the materials and settings for installation. This synchronization uses the replication function.

3.8.10.1 Files to be Synchronized

Files to be synchronized vary depending on the actions that the command specifies. The files to be synchronized for each specified action are shown below.

See

For details on the pxinst command, see "pxinst command" in the "Job Operation Software Command Reference."

Table 3.101 List of Actions of the Replication Function

Action	Description
all	Synchronizes all settings of the installation function. This operation is equivalent to the operation when executing both repo and srv.
repo	Synchronizes the repository.
srv	Synchronizes the server setting (*) necessary for the installation function.

Information

In the srv specification, the following files are to be synchronized:

- Host entry information of DHCP (/etc/dhcp/dhcpd.conf)

  This synchronization target is the standby installer node.

- Status management DB (/var/opt/FJSVtcs/inst/db/install_status.json)

  This synchronization target is the standby installer node.

- Materials to be used for installation

  They apply to those under the directory that "tftpdir" of the configuration file in the installation environment specifies.
  These synchronization targets are the standby installer node, the active system relay node, and the standby system relay node.

  - kernel image

  - initrd image

  - kickstart file

- PXE configuration file and grub.cfg (PRIMERGY server)
- grub.cfg (FX server)

3.8.10.2 Synchronization Procedure

The synchronization procedures are as follows.

[Procedure for synchronizing all settings]

```
# pxinst sync all
Standby Installer node                           1/1          [OK]
Relay node[s]                                    2/2          [OK]
[INFO] INST 1805 pxinst File replication is complete.
```

[Procedure for synchronizing all repositories]

```
# pxinst sync repo
Standby installer node
Synchronize the repository      1/1           [OK]
Relay node[s].
Synchronize the repository      10/10         [OK]
[INFO] INST 1805 pxinst File replication is complete.
```

[Procedure for synchronizing only a specific repository *repoame*]

```
# pxinst sync repo --repo reponame
Standby installer node
Synchronize the repository      1/1           [OK]
Relay node[s].
Synchronize the repository      10/10         [OK]
[INFO] INST 1805 pxinst File replication is complete.
```

reponame: Repository name

[Procedure for synchronizing a synchronization destination node by specifying it]

```
# pxinst sync repo --hostname hostname1,hostname2
Standby installer node
Synchronize the repository      1/1           [OK]
Standby installer node
Synchronize the repository      1/1           [OK]
[INFO] INST 1805 pxinst File replication is complete.
```

hostname1,hostname2: Host name of the synchronization destination

[Procedure for synchronizing synchronization destination nodes by specifying them in serial order]

```
# pxinst sync repo --hostname hostname[001-100]_
[INFO] INST 1805 pxinst File replication is complete.
```

hostname[001-100]: Specifying host names of nodes to be synchronized in serial order

In this example, nodes ranging from hostname001 to hostname100 are to be synchronized.

If a node fails in synchronization, a warning message appears with the path of a file to which the detailed information is output.

```
[WARN] INST 1500 pxinst Replication failed node exists.(filename)
```

filename: File to which both the host name of a node failing in synchronization and the failure cause are output

filename is output as /tmp/pxinst/sync_error_*YYYYMMDDhhmmss*.log.
An example of the output file is as follows.

```
HOSTNAME: hostname1, DETAIL: ssh: connect to host xxx.xxx.xxx.xxx port 22: No route to host
HOSTNAME: hostname2, DETAIL: ssh: connect to host xxx.xxx.xxx.xxx port 22: No route to host
```

Either the host name or the IP address is output to HOSTNAME. The above example shows a case where hostname1 and hostname2 are output.

3.8.11 Checking the Service Operation State

3.8.11.1 Procedure for Checking the Service Status

The following shows the procedure for checking the service status.

[Procedure for checking the service status for all of the installer node, the relay node, and BIO]

The following is an example displayed when the service status of all nodes is normal.

```
# pxinst check
Installer node[s]          2/2         [OK]
Relay node[s]             10/10        [OK]
Boot IO node[s]          100/100       [OK]
[INFO] INST 1806 pxinst Service state check is complete.
```

When all the services are operating normally, "the number of normal nodes/the number of all nodes" and [OK] appear.

The following is an example displayed when there is a node having an abnormal service.

```
# pxinst check
Installer node[s]           2/2          [OK]
Relay node[s]               8/10         [FAILED]
  Failed node[s]
  HOSTNAME           SERVICE
  hostname1          tftp, http
hostname1            tftp, http
Boot IO node[s]             100/100     [OK]
[WARN] INST 1501 pxinst Abnormal node exist.
```

If there is a node in which the service stops or cannot make communications, a warning message appears. Also, the information on the abnormal node appears as "Failed node[s]." In such a case, the system administrator must start the service on this node. If it is impossible to communicate with this node, the SERVICE field displays the error message. The following tables show the items displayed when there is a node whose service status is abnormal.

Table 3.102 Displayed Items in Common

Item	Meaning	Value
Installer Node[s]	Status of the installer node	The number of normal nodes/the number of nodes to be checked
Relay Node[s]	Status of the relay node	The number of normal nodes/the number of nodes to be checked
Boot IO Node[s]	Status of the node serving as both compute node and boot I/O node	The number of normal nodes/the number of nodes to be checked

Table 3.103 Displayed Items of a Node Whose Service Status is Abnormal

Item	Meaning	Value
HOSTNAME	Host name or IP address of an abnormal node	The host name or the IP address appears.
SERVICE	Name of an abnormal service	Service name or error content In the case of a connection failure to a target node, the error content appears.

3.9 Details of the Backup/Restore Function

3.9.1 About the Disk Image

This section describes the disk image used for backup and restoration.

3.9.1.1 Information on the Disk Image

The backup function creates a file (the image.info file) in the directory which collects the disk image. This file describes information on the disk image. The following are the purposes for creating the file which describes information on the disk image:

- Managing the disk image by the system administrator
- Checking information of each partition by the system administrator
  The restore function uses this information during a restoration. The system administrator can refer to this information to check the partition information during a restoration.
- Using information on the disk image by the system administrator while restoring a specific partition
  The system administrator is to use it to refer to the use area size (USED_SIZE) of a specific partition and then confirm that it is possible to restore it to a restoration destination partition.

Note

- Do not change the contents of the file which describes information on the disk image.

The following is an example of the image.info file when executing a backup by specifying ccm1 as the host name, 192.0.2.1:/work/backup as the backed-up directory, and sda as the disk.

```
IMAGE_DIR: 192.0.2.1:/work/backup/ccm1_20171231153510
HOSTNAME: ccm1
CLSTNAME:-
NODE_ID:-
BOOTGID:-
IMAGE_ARCH: x86_64
IMAGE_DATE: 2018-01-01 15:35:10
IMAGE_SIZE: 10036MB
BACKUP IMAGE:
PART        FSTYPE   MOUNTPOINT     PART_SIZE       USED_SIZE
/dev/sda1   fat32    /boot/efi      512MB           30MB
/dev/sda2   xfs      /boot          508MB           71MB
/dev/sda4   xfs      /           994595MB        25340MB

RESTORE DISK:
Disk: sda
Partition Table: gpt
Number  Start    End        Size      File system     Name                  Flags
1       1.05MB   513MB      512MB      fat32          EFI System Partition  boot, esp
2       514MB    1026MB     512MB      xfs
3       1027MB   5123MB     4097MB     linux-swap(v1)
4       5124MB   1000205MB  995081MB   xfs
```

Table 3.104 Contents Written in the File Which Describes the Information on the Disk Image

Item		Description
IMAGE_DIR		Name of directory storing the disk image
HOSTNAME		Host name
CLSTNAME		Cluster name
NODE_ID		Node ID
BOOTGID		Boot group ID
IMAGE_ARCH		Architecture
IMAGE_DATE		Date
IMAGE_SIZE		Total size of the disk image (MB)
BACKUP IMAGE		Information for each disk image
	PART	Device name
	FSTYPE	Type of a file system
	MOUNTPOINT	Mount point
	PART_SIZE	Partition size (MB)
	USED_SIZE	Size of use area (MB)
RESTORE DISK		Partition information

3.9.1.2 How to Use the Disk Image

- You can restore the disk image for not only the backup source node but also a node with the same hardware configuration and partition configuration.
- The disk image collected from PRIMERGY server is also available to another PRIMERGY server. Likewise, the disk image collected from a BoB of FX server is available to another BoB.
- If the disk image is collected from a node whose node type is in a redundant configuration, it is available to both the active system node and the standby system node with this node type.
- If a node has completed the disclosure setting of the NFS server, you can restore the disk image by copying it on this node. This is regardless of whether or not this node is saved at the time of backup.

3.9.1.3 Deleting the Disk Image

The backup/restore function enables you to save the disk image on any node. The backup/restore function does not have the functions to manage and delete the disk image. This is because managing the disk image on multiple nodes complicates the management of the disk image. The system administrator is to manage and delete the disk image.

To delete an unnecessary disk image, the system administrator must use the rm command of the OS or the like to delete the disk image or a directory that stores the disk image.

3.9.2 Backup of the Disk Image

The restore function restores a disk image required such as when you build a node and a trouble occurs. For this reason, the system administrator must use the backup function to save the disk image in advance.

The backup function saves the contents of the disk unit of a specified node as a disk image. This function collects a disk image for each partition from a specified node. Then, this function saves the disk image as a file on the node where the image is to be saved.

The functions of the backup function are as follows:

- Backup for each node/BoB

  For PRIMERGY server, you can back up a disk image for each node. For FX server, you can back it up for each BoB, which includes nodes under the BoB.

- Backup for nodes in various states

  You can back up and save a node in any state, such as after the OS installation and after application of the modification package. There is no limitation on the number of backups for a node; you can collect multiple backups on one node or BoB.

- Specifying a saving location of a disk image

  You can select a saving location of a disk image. You can specify any node such as a server dedicated to backup, as long as the node has completed the disclosure setting of the NFS server.

- Backup for a specific partition

  If you want to exclude a specific partition such as a dump area, you can perform a backup specifying the partition, not the entire disk.

- Checking the file system before backup

  Before backing up a disk image, execute the command to check the file system. Then, perform the backup after confirming that it has no abnormality.

Note

- The start mode (UEFI or BIOS) of a node to be backed up or restored is in accordance with the contents set when registering node information. You cannot restore a disk image to a node whose start mode differs, due to the difference of the partition configuration. For example, if backing up a disk image from a node in the UEFI mode, you cannot restore the disk image to a node in the BIOS mode. Likewise, you cannot restore it from the BIOS mode to the UEFI mode. It is therefore necessary to collect disk images for necessary start modes (UEFI and BIOS).

- FX server prevents you from making a restoration to an FX server in which the mount locations and existences of node types are different inside a BoB. (This existence means the presence or absence of the global I/O node and the storage I/O node.) To back up a BoB, it is therefore necessary to collect disk images for the configuration patterns related to the existences of node types that exist in operations. For example, if an FX server consists of the following three configuration patterns, collect disk images for these three patterns:
  - Both the global I/O node and the storage I/O node exist.
  - The storage I/O node exists.
  - Neither the global I/O node nor the storage I/O node exists.

3.9.2.1 Backup for Each Node/BoB

For PRIMERGY server, you can back up a disk image for each node. For FX server, you can back it up for each BoB, which includes nodes under the BoB. By backing it up for each BoB, you can collectively restore nodes under a BoB.

The backup function saves a disk image in an NFS-mounted directory. This allows you to specify only one node or one BoB at the time of a backup.

To execute backup for each node, use the host name of a node. If the configuration management function has completed registering the cluster configuration information, you can also use a node ID to make the specification. For backup for each BoB, specify a boot group ID. For details on the specification method, see "paclone command" in the "Job Operation Software Command Reference."

Note

- If the system management node is not in a redundant configuration, you cannot acquire the backup of the active system management node. For the active system management node, use the dump command that the OS provides. For details on the dump command, check the specification of the dump command.

- The management network specified by the node ID must be the same as the interface to be network-booted. (This interface is used for the OS installation provided by the installation function.)

- Backing up or restoring a BoB under the compute cluster sub management node uses each server function of such node. Therefore, start this work after confirming that the active system of the compute cluster sub management node has normally started.

Information

- If node information has already been registered in PRIMERGY server, the backup is possible regardless of the OS installation method for a target node.
- For FX server, you cannot back up nodes on which the OS is manually installed, for FX server requires the OS installation provided by the installation function.
- Regarding backup for each BoB, the partition of the dump area (/var/crash) is not to be backed up.

3.9.2.2 Backup for Nodes in Various States

You can back up and save a node in any state, such as after the OS installation and after application of the modification package. There is no limitation on the number of backups for a node. You can collect multiple backups on one node or BoB.
For example, you can collect the disk image for OS installation and the disk image for introduction of the Job Operation Software respectively for one node. This enables you to use the disk images flexibly when introducing nodes the next time.

3.9.2.3 Specifying a Saving Location of a Disk Image

To specify the saving location of a disk image, use the -d option of the paclone command to specify any directory on any node. When the disk capacity of the saving location of a disk image becomes tight, for example, you can easily switch nodes to save it.

It is optional to provide the host name. Without providing it, the directory of the active system management node is specified.
You can specify a node whose host name information is registered by using the installation function.

- Example of saving a disk image on a node (server1) other than the active system management node

  1. Perform the NFS disclosure setting on server1 (/etc/exports).

```
# vi /etc/exports
/imagedir 192.0.2.0/255.255.255.0(no_root_squash,rw,async)
# exportfs -a
# exportfs
/imagedir   192.0.2.0/255.255.255.0
```

  2. Execute the paclone command on the active system management node.

```
# paclone backup -m hostname -d server1:/imagedir -k sda
Do you really want to continue (y/n)? y
[INFO] INST 0801 paclone Created a configuration backup.(filepath)
[INFO] INST 3201 paclone Command succeeded.(backup)
```

  hostname: Host name of a node to be backed up
  filepath: Path which stores the backup of the PXE configuration file

  3. Start the node on the PXE boot. After the backup is executed, a sub directory is created on server1 as follows.

```
/imagedir/hostname_20171231131020
```

The paclone command creates a sub directory under the directory specified by the -d option, and then creates a disk image file for each partition on this sub directory. An error occurs if there is already this directory.

Specify this sub directory in the -d option at the time of restoration.

For the directory specified by the -d option, the system administrator must preliminarily carry out the directory disclosure setting of the NFS server and approve the read/write authority for the root user. If specifying a directory that cannot be NFS-mounted, such as a directory that has already mounted the NFS, an error occurs when mounting the NFS on a target node.

Note

Only a node specified by the -d option saves a backed up disk image. This prevents it from being synchronized to the standby system management node, even though you specify the saving destination as the active system management node. Assume that you want to save a disk image in the standby system management node as well after considering that failover switches the system management node. In such a case, the system administrator must use the pmscatter command or any similar command to copy the directory containing the disk image on the standby system management node.

See

For the detailed procedure of the directory disclosure setting of the NFS server, see the OS manual.

3.9.2.4 Backup for a Specific Partition

PRIMERGY server requires you to specify a disk or partition of a node to be backed up. You can specify it in units of either disk, partition, by-id, or UUID. For BoBs, it is unnecessary to specify a disk or partition.

The following table shows the usage scenes of the methods for specifying a disk or partition to be backed up.

Table 3.105 Usage Scenes of the Methods for Specifying a Disk or Partition to be Backed Up

Specification Method	Description
Disk	Specify disk in normal cases.
Partition	Use this such as when you want to exclude a partition for a dump area or back up only a specific partition.
by-id	Use this such as when you want to prevent device displacement.
UUID	Use this such as when you want to prevent device displacement.

3.9.2.5 Checking the File System Before Backup

The backup/restore function is based on the assumption that a perfect restoration becomes possible by backing up a partition in which the file system has no abnormality. Therefore, the backup function executes the command to check the file system before backing up a disk image, confirms that it has no abnormality, and then performs the backup.

Note

If the check of the file system detects an abnormality, this function outputs an error message and exits without restoring it automatically. This is because this function cannot determine the necessity of this restoration. The system administrator must determine the necessity of this restoration from the error message. Then, he/she must perform the restoration using the ipmitool command or the console connection function (the paconsole command) to connect to the console. After the restoration, perform the backup again and confirm that the backup is successful. When it has already turned out that it is necessary to restore the file system, specify the --fsck option in the paclone command. This restores the file system automatically. For details on the --fsck option, see "paclone command" in the "Job Operation Software Command Reference."

See

For details on the console connection function, see "3.5.6 Connecting to the Console of a Node."

3.9.2.6 Executing a Backup

Execute a backup in the following order.

1. Executing the paclone command

   Execute the backup sub command of the paclone command on the active system management node.
   After executing the paclone command, an inquiry for confirming the execution appears. The command ends after entering your response.

2. Synchronizing the setting

   Executing the paclone command rewrites the PXE configuration file on the active system management node.

Synchronize this file with the compute cluster sub management node.

This procedure is unnecessary when there is no compute cluster sub management node.

3. Starting a target node

   The system administrator is to start a target node manually or using either the ipmitool command or the papwrctl command.
   He/She must start it on the PXE boot.
   The backup starts by starting a target node on the PXE boot.
   It automatically restarts after the backup is complete.

4. Confirming that the backup is complete

   Use the status sub command of the paclone command in order to confirm that the backup is complete. This sub command is a function to check the status.

Note

- Before starting a backup, confirm that it meets the preconditions. For details on the preconditions, see "2.9.1 Preconditions to Use the Backup/Restore Function."
- If a node to be backed up is conneced to ETERNUS, remove the connection before starting the backup. Connect them again after the backup is complete.

See

- For details on the power control command papwrctl, see "3.2.2 Power Control for Nodes."
- For details on status checking, see "3.9.4 Checking Status."

An example of executing the backup is as follows.

1. Executing the paclone command

   Execute the backup sub command of the paclone command on the active system management node.

   - For FX server BoB

     Specify a boot group ID.

     ```
     # paclone backup -c clstname --bootgrp bootgid -d imagedir
     Do you really want to continue (y/n)? y                              <- (*1)
     [INFO] INST 0801 paclone Created a configuration backup.(filepath)
     [INFO] INST 3201 paclone Command succeeded.(backup)
     ```

     clstname: Cluster name
     bootgid: Boot group ID
     imagedir: Directory storing a disk image
     filepath: Path which stores the backup of the PXE configuration file

     (*1) This requires your confirmation before the execution.

     A sub directory is automatically created under the directory specified by the -d option. The disk image is backed up on this sub directory. Without specifying a host name or IP address, the active system management node stores it.

     ```
     /imagedir/hostname_YYYYMMDDhhmmss
     ```

     hostname_YYYYMMDDhhmmss: Sub directory
     hostname: A target node specified by the paclone command (which varies depending on the specified option)
     YYYYMMDDhhmmss: Date

   - For PRIMERGY server

     Specify a host name or a node ID. This requires you to specify a disk or partition.

- When specifying a host name

```
# paclone backup -m hostname -d /imagedir -k disk
Do you really want to continue (y/n)? y
[INFO] INST 0801 paclone Created a configuration backup.(filepath)
[INFO] INST 3201 paclone Command succeeded.(backup)
```

hostname: Host name
imagedir: Directory storing a disk image
disk: Disk to be backed up
filepath: Path which stores the backup of the PXE configuration file

- When specifying a node ID

```
# paclone backup -c clstname -n nodeid -d /imagedir -k disk
Do you really want to continue (y/n)? y
[INFO] INST 0801 paclone Created a configuration backup.(/tmp/paclone)
[INFO] INST 3201 paclone Command succeeded.(backup)
```

clstname: Cluster name
nodeid: Node ID
imagedir: Directory storing a disk image
disk: Disk to be backed up

2. Synchronizing the setting

This procedure is unnecessary when there is no compute cluster sub management node.

```
# pxinst sync all
[INFO] INST 1805 pxinst File replication is complete.
```

3. Starting a target node

The system administrator is to execute the PXE boot for a target node manually or using either the ipmitool command or the papwrctl command.
The following is an example of using the papwrctl command.

```
# papwrctl -c clstname -n nodeid on --boottype pxe
[INFO] PSM 2081 papwrctl All power control was executed.
```

The backup starts by starting the node.
The node automatically restarts after the backup is complete.

4. Confirming that the backup is complete

Use the status checking function to confirm that the backup is complete.

```
# paclone status show
CLSTNAME   NODE         HOSTNAME       OPERATION     STATUS        STATE        DETAIL
---------- ----------   --------       ---------     ------        -----        ------
clstname   nodeid       hostname       Backup        Completed     Succeeded    -
```

Note

If you have the standby system management node or the standby compute cluster sub management node, note the following.

The standby system management node or the standby compute cluster sub management node is unable to communicate during node startup after completing a backup. As a result, the synchronization of the configuration files from the system management node may fail, and the following message may be output to the state (DETAIL) of the backed up node.

```
# paclone status show
CLSTNAME   NODE         HOSTNAME       OPERATION     STATUS        STATE        DETAIL
---------- ----------   --------       ---------     ------        -----        ------
```

```
clstname   nodeid        hostname        Backup        Completed     Failed        Update PXE (*)
configuration file failed. (detail) Please change default label to "localdisk" manually.
```

detail: Detailed error message
A new line starts just after the (*) part because of the space limitation of this document, but only one line is actually displayed.

If STATUS is Completed and STATE is Failed, the configuration files failed to synchronize after the backup operation completes successfully. After confirming that the target node is started, execute "pxinst sync all" to synchronize the configuration file. Note that even if this command is successful, the STATE Failed display does not disappear, but recovery is complete. To clear the Failed display above, execute "pxinst status clear" with a hostname to clear the backup completion status display.

Information

- For details on checking the progress of a backup, see "3.9.4 Checking Status." To check the detailed status, use the ipmitool command or the console connection function to connect to the console. For details on the console connection function, see "3.5.6 Connecting to the Console of a Node."

- One paclone command can back up one node or one BoB. To back up multiple nodes at the same time, use a script or the like to execute the paclone command in parallel.

- The size of a backed up disk image is smaller than the actual use size, for it is compressed when saving it.

- In a redundant configuration, the standby system nodes are to be backed up. This enables you to perform backup/restoration while continuing job operations.

- In an environment for PRIMERGY server in which the disk usage amount is about 13.5 GB, it takes about 13 minutes to process a backup, including the restart time of a node.

See

If an error message is output or if STATE becomes Failed in the status checking function, remove the cause of this failure, and then execute the backup again. For details on the output message and the coping method, see "Problems With Backup/Restore" in the "Job Operation Software Troubleshooting."

3.9.3 Restoring a Disk Image

The system administrator uses the restore function such as when copying a node or restoring it to its previous state when a trouble occurs. After the backup function collects a disk image, the restore function restores it to a specified node. During the restoration, restore a disk image and set node-specific information (such as ifcfg-*xx*, the network configuration file, and /etc/fstab, the mount configuration file).
The functions of the restore function are as follows:

- Restoration for each node/BoB

  In restoration for each BoB, restore nodes under a BoB collectively. Also, set node-specific information to each node.

- Restoring a disk image saved in any node

  You can restore a disk image that was saved to any node during the backup. You can restore it from any node that has completed the disclosure setting of the NFS server and has a disk image.

- Restoring a specific partition

  If you want to exclude a specific partition such as a dump area, you can perform a restoration by specifying the partition, not the entire disk.

- Executing a restoration plugin

  After restoring a disk image, you can execute a user's arbitrary plugin command to a node to be restored.
  Use a restoration plugin to perform work such as deleting an unnecessary log restored from a disk image. This makes it possible to reduce the effort and time for work that the system administrator performs after the restoration is complete.

Note

- You cannot restore the disk image of PRIMERGY server to an FX server BoB. You also cannot restore the disk image of an FX server BoB to PRIMERGY server.
- You cannot restore a disk image to a node whose architecture is different from the node that is the source of the disk image.

3.9.3.1 Node-Specific Information Set at the Time of Restoration

Set node-specific information when restoring a disk image. The setting contents are the OS information, the information used in the installation function, and the node ID of the Job Operation Software.
The following table shows the setting contents and the target files.

Table 3.106 Node-Specific Information Set at the Time of Restoration

Setting Contents	Target File
Boot loader	/boot/efi/EFI/*xxx*/grub.cfg
Network	/etc/sysconfig/network-scripts/ifcfg-*xxx*
Host name	/etc/hostname
Mount information	/etc/fstab
NTP server	/etc/chrony.conf
ssh setting of root user	Under /root/.ssh
Package manager	/etc/yum.repos.d/*xxx*.repo /etc/dnf/dnf.conf
Stopping unnecessary service	Stop of firewalld
Setting of the installation function (Setting according to node type of the installation function)	Under /var/opt/FJSVtcs/inst
Setting of boot I/O node	- Setting of TofuD driver - Setting of arp - Setting of NFS server - Edition of fstab - Setting of DHCP server - Setting of diskless node
Setting of the Job Operation Software	Under /etc/opt/FJSVtcs

xxx: Value according to the setting of the OS or a node

Information

The restore function does not change the configuration of the rpm package. Thus, if nodes of the Job Operation Software have different package configurations or settings for each node type, it is necessary to back them up for each node type.

See

For details on the settings of the installation function, the boot I/O node, and the NFS diskless, see "3.8 Details of the Installation Function."

3.9.3.2 Restoration for Each Node/BoB

After backing up a disk image for each node, you can restore them to each node.
For FX server, you can collectively restore nodes under a BoB by using the disk image that restored each BoB.

To perform a restoration for each node, use the host name of the node. If the configuration management function has completed registering the cluster configuration information, you can use a node ID to make the specification. To backup each BoB, specify a boot group ID. For details on the specification method, see "paclone command" in the "Job Operation Software Command Reference."

Information

If nodes or BoBs have the same configuration, you can restore a disk image by specifying multiple nodes or BoBs at once. For details on the specification method, see "paclone command" in the "Job Operation Software Command Reference."

The following is an example of concurrently restoring the same disk image for PRIMERGY server whose host names are host01 and host02.

```
# paclone restore -m host01,host02 -d /imagedir
Do you really want to continue (y/n)? y
[INFO] INST 0801 paclone Created a configuration backup.(filepath)
[INFO] INST 3201 paclone Command succeeded.(restore)
```

imagedir: Directory storing the disk image
filepath: Path which stores the backup of the PXE configuration file

3.9.3.3 Restoring a Disk Image Saved in Any Node

After saving a disk image to any node during the backup, you can use it for the restoration by using the -d option of the paclone command to specify a node or directory.

When executing the command, the system administrator is to specify the directory that stores a disk image to be restored. (This directory is a sub directory created by the backup sub command.)

Note

In the same manner as the backup, the directory specified by the -d option requires the system administrator to preliminarily carry out the directory disclosure setting of the NFS server. The system administrator must approve the read/write authority for the root user. If specifying a directory that cannot be NFS-mounted, such as a directory that has already mounted the NFS, an error occurs when mounting the NFS on a target node.

See

For the detailed procedure of the directory disclosure setting of the NFS server, see the OS manual.

The following is an example of the procedure for restoring a disk image that was saved to any node.

- Example of the procedure for restoring a disk image saved in a node (server1) other than the active system management node

    1. Perform the NFS disclosure setting on server1 (/etc/exports).

    ```
    # vi /etc/exports
    /imagedir 192.0.2.0/255.255.255.0(no_root_squash,rw,async)
    # exportfs -a
    # exportfs
    /imagedir   192.0.2.0/255.255.255.0
    ```

    2. Restore it on the active system management node.

    ```
    # paclone restore -m hostname -d server1:/imagedir
    Do you really want to continue (y/n)? y
    [INFO] INST 0801 paclone Created a configuration backup.(filepath)
    [INFO] INST 3201 paclone Command succeeded.(restore)
    ```

    filepath: Path which stores the backup of the PXE configuration file

3.9.3.4 Restoring a Specific Partition

The restore function operates the partitions of a target disk on a target node. (This operation creates and formats the partitions.) Then, it restores the disk image of each partition.
To restore only the disk image of a specific partition, use the -p option of the paclone command to specify the name of a partition to be restored. When specifying the -p option, a partition is not operated on a target node. It is therefore necessary to preliminarily make a partition specified by the -p option complete to its format.

Note

The partition operation (which creates and formats partitions) deletes all data in a target disk. Back up necessary data beforehand.

3.9.3.5 Executing a Restoration Plugin

The restore function includes the plugin function. After restoring a disk image, the plugin function executes a plugin command describing any process to a target node. You can use the plugin function to delete an unnecessary log restored from a disk image or make any setting requiring restart. This allows the system administrator to reduce more effort and time than performing such work after the restoration is complete. The following describes the plugin function at the time of a restoration.

- Plugin commands that you can specify
    - The contents of plugin commands are arbitrary; the result is not guaranteed.
    - The upper limit value of the number of executable plugin commands is 10000.
    - Define the file name of the plugin command with up to 63 characters, including single-byte alphanumeric characters (case-sensitive), hyphens ("-"), underscores ("_"), and periods("."). However, the first character cannot be a hyphen.
    - You cannot specify multiple plugin commands whose file names are the same.
- Execution of plugin commands
    - An execution authority is added to plugin commands.
    - Plugin commands are executed with root privileges.
    - The restore function uses the chroot command to execute plugin commands on a target node.
    - The execution directory of plugin commands is the /root directory on a target node.
    - Plugin commands are arranged and executed under the /root/paclone_plugin directory on a target node. They still remain there after their execution. If a plugin command already exists in the /root/paclone_plugin directory, delete the file and then place the plugin command.

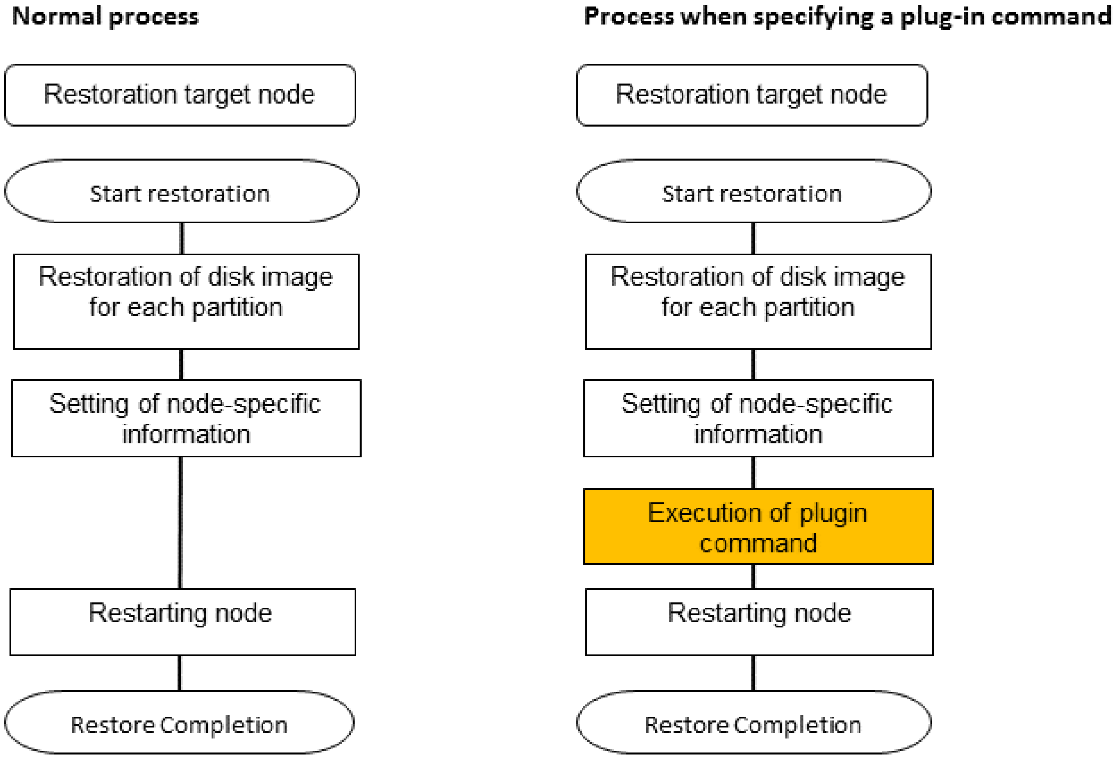
See

For details on the chroot command and the operations under the chroot environment, see the OS manual.

- Execution result of a plugin command
    - You can determine the execution result of a plugin command by checking the return value. The return value is output to a console and log.
    - The execution result of a plugin command never abnormally ends or stops the restoration process of a disk image.
- Operation timing of a plugin command

  Normally, proceed the restoration process in the order of "Restoration of disk image for each partition," "Setting of node-specific information," and "Restarting node."
  In restoration when specifying a plugin command, proceed the process in the order of "Restoration of disk image for each partition," "Setting of node-specific information," "Execution of plugin command," and "Restarting node." The following charts show the flows of the normal process and the process when specifying a plugin command.

Figure 3.32 Operation Timing of a Plugin Command


Normal process
Process when specifying a plug-in command
Restoration target node
Start restoration
Restoration of disk image for each partition
Setting of node-specific information
Restarting node
Restore Completion
Restoration target node
Start restoration
Restoration of disk image for each partition
Setting of node-specific information
Execution of plugin command
Restarting node
Restore Completion

- Examples of plugin commands

  Examples of plugin commands are as follows.

  - A plugin command to delete an unnecessary log

    The restore function also restores the log of a backup source node, which a disk image contains. The following is an example of a plugin command to delete a restored log.

```
# vi log_delete.sh
#!/bin/bash
rm -f /var/log/*.log
rm -f /var/log/sa/sa*
rm -f /var/log/FJSVtcs/*/*.log
```

  - A plugin command to reset FEFS

    FEFS requires you to reset the node-specific information for each node. The following is an example of resetting the node-specific information of FEFS.

```
# vi fefs_setup.sh
#!/bin/bash
/usr/sbin/fefsconfig --setup
```

See

For details on the FEFS settings, see the *FEFS User's Guide*.

3.9.3.6 Executing a Restoration

Execute a restoration in the following order:

1. Executing the paclone command
2. Synchronizing the setting

3. Starting a target node

4. Confirming that the restoration is complete

   Use the status sub command of the paclone command in order to confirm that the restoration is complete. This sub command is a function to check a status.

Note

- Before starting a restoration, confirm that it meets the preconditions. For details on the preconditions, see "2.9.1 Preconditions to Use the Backup/Restore Function."

- If a node to be restored is connected to ETERNUS, remove the connection before starting the restoration. Connect them again after the restoration is complete.

- If the driver used at a backup source node is different from the driver used at a restoration destination node, the OS may not start normally.

- If it is necessary to restore an active system node, switch the active system to the standby system to make it a standby system node before restoring it.

See

- For details on the papwrctl command, see "3.2.2 Power Control for Nodes."

- For details on status checking, see "3.9.4 Checking Status."

The following is an example of executing the restoration.

1. Executing the paclone command

   Execute the restore sub command of the paclone command on the active system management node.
   After executing the paclone command, an inquiry for confirming the execution appears. The command ends after entering your response.

   - For FX server BoB

     Specify a boot group ID.

     ```
     # paclone restore -c clstname --bootgrp bootgid -d /imagedir/hostname_YYYYMMDDhhmmss
     Do you really want to continue (y/n)? y              <- (*)
     [INFO] INST 0801 paclone Created a configuration backup.(filepath)
     [INFO] INST 3201 paclone Command succeeded.(restore)
     ```

     clstname: Cluster name
     bootgid: Boot group ID
     /imagedir/hostname_YYYYMMDDhhmmss: Directory storing a disk image
     filepath: Path which stores the backup of the PXE configuration file

     (*) This requires your confirmation before the execution.

   - For PRIMERGY server

     Specify a host name or a node ID.

     - When specifying a host name

       ```
       # paclone restore -m hostname -d /imagedir/hostname_YYYYMMDDhhmmss
       Do you really want to continue (y/n)? y
       [INFO] INST 0801 paclone Created a configuration backup.(filepath)
       [INFO] INST 3201 paclone Command succeeded.(restore)
       ```

     hostname: Host name

- When specifying a node ID

```
# paclone restore -c clstname -n nodeid -d /imagedir/hostname_YYYYMMDDhhmmss
Do you really want to continue (y/n)? y
[INFO] INST 0801 paclone Created a configuration backup.(filepath)
[INFO] INST 3201 paclone Command succeeded.(restore)
```

nodeid: Node ID

2. Synchronizing the setting

   This procedure is unnecessary when there is no compute cluster sub management node.

```
# pxinst sync all
[INFO] INST 1805 pxinst File replication is complete.
```

3. Starting a target node

   The system administrator is to start a target node on the PXE boot manually or using either the ipmitool command or papwrctl, the power control command. The restoration starts by starting the node. It automatically restarts after the restoration is complete.

   - When using the papwrctl command to start a target node

```
# papwrctl -c clstname -n nodeid on --boottype pxe
[INFO] PSM 2081 papwrctl All power control was executed.
```

4. Confirming that the restoration is complete

   Use the status checking function to confirm that the restoration is complete.

```
# paclone status show
CLSTNAME   NODEID     HOSTNAME       OPERATION    STATUS        STATE        DETAIL
---------- ---------- --------       ---------    ------        -----        ------
cluster   nodeid        hostname       Restore      Completed     Succeeded    -
```

Note

If you have the standby system management node or the standby compute cluster sub management node, note the following.

The standby system management node or the standby compute cluster sub management node is unable to communicate during node startup after completing a restore. As a result, the synchronization of the configuration files from the system management node may fail, and the following message may be output to the state (DETAIL) of the restored node.

```
# paclone status show
CLSTNAME   NODE         HOSTNAME       OPERATION    STATUS        STATE        DETAIL
---------- ----------   --------       ---------    ------        -----        ------
clstname   nodeid       hostname       Restore      Completed     Failed       Update PXE (*)
configuration file failed. (detail) Please change default label to "localdisk" manually.
```

detail: Detailed error message
A new line starts just after the (*) part because of the space limitation of this document, but only one line is actually displayed.

If STATUS is Completed and STATE is Failed, the configuration files failed to synchronize after the restore operation completes successfully. After confirming that the target node is started, execute "pxinst sync all" to synchronize the configuration file. Note that even if this command is successful, the STATE Failed display does not disappear, but recovery is complete. To clear the Failed display above, execute "pxinst status clear" with a hostname to clear the restore completion status display.

Information

- For details on checking the progress of a restore, see "3.9.4 Checking Status." To check the detailed status, use the ipmitool command or the console connection function to connect to the console. For details on the console connection function, see "3.5.6 Connecting to the Console of a Node."

- In the network setting to be made by the processing after the restoration, delete all the existing settings (which are the restored network settings) first. Then, set the network information that is registered in the node information. If requiring a network that is not registered in the node information, the system administrator must set it manually after the restoration is complete. For details on the network setting, see the OS manual.
- In an environment for PRIMERGY server in which the disk usage amount is about 13.5 GB, it takes about 13 minutes to process a restoration, including the restart time of a node.

See

If an error message is output or if STATE becomes Failed in the status checking function, remove the cause of this failure, and then execute the restoration again. For details on the output message and the coping method, see "Problems With Backup/Restore " in the "Job Operation Software Troubleshooting."

3.9.4 Checking Status

The system administrator is to use the status checking function to check the progress, success and failure of a backup/restoration of a node. If a backup/restoration fails, the system administrator is to cope with it based on the displayed details message. This function displays nodes backed up and restored with the backup/restore function.

Information

You can also use the installation function to check the progress of a backup/restoration. For details on the installation status checking function, see "3.8.9 Checking Installation States." Also, node information is still to be displayed after a backup/restoration is complete; the system administrator must delete it explicitly if necessary.

3.9.4.1 Status Checking

Use the status sub command of the paclone command for status checking.

```
# paclone status show
CLSTNAME    NODEID      HOSTNAME     OPERATION    STATUS      STATE      DETAIL
----------  ----------  --------     ---------    ------      -----      ------
cluster1    0xFFFF0005  host01       Backup       Completed   Succeeded  -
cluster1    0xFFFF0006  host02       Backup       InProgress  Cloning    -
cluster1    0xFFFF0007  host03       Backup       Completed   Failed     detail
cluster1    0xFFFF0008  host04       Restore      NotStarted  Waiting    -
```

detail: Details message

The displayed items are as follows.

CLSTNAME

Cluster name of a target node

NODEID

Node ID of a target node

HOSTNAME

Host name of a target node

OPERATION

Type of an operation
Any of the following values appears.

Value	Meaning
Backup	Backup

Value	Meaning
Restore	Restoration

STATUS

Status of an operation

Any of the following values appears.

Value	Meaning
NotStarted	Before starting operation
InProgress	Operation in progress
Completed	Operation completed

STATE

State of processing
Any of the following values appears.

Value	Meaning
Waiting	Period from node start waiting time to backup/restoration processing start time
Checking	Performing prior check before backup/restoration
Cloning	Performing backup/restoration
Posting	Performing processing after backup/restoration
Succeeded	Normal end
Failed	Abnormal end

DETAIL

Detailed information
This displays the information on the reason why STATE shows Failed with up to 255 characters.

See

The compute nodes of FX server (which are the global I/O node, the storage I/O node, and the compute node) do not appear here.

3.9.4.2 Deleting Unnecessary Status Displays

Once the information on a node becomes a display object due to its backup/restoration, it continuously appears unless the system administrator deletes it. The following is the procedure for deleting needless node information, which is no longer necessary to be referred.

- Example of deleting all information whose backup/restoration is successfully complete

```
# paclone status clear
[INFO] INST 3201 paclone Command succeeded.(status clear)
```

- Example of deleting information of a node with a specific status

  The following is an example of deleting the information of a node which succeeds in the backup or restoration. (The STATE of such node is Completed.)

```
# paclone status clear --status Completed
[INFO] INST 3201 paclone Command succeeded.(status clear)
```

- Example of deleting information on a specific node

  The following is an example of deleting the information of a node whose host name is host01.

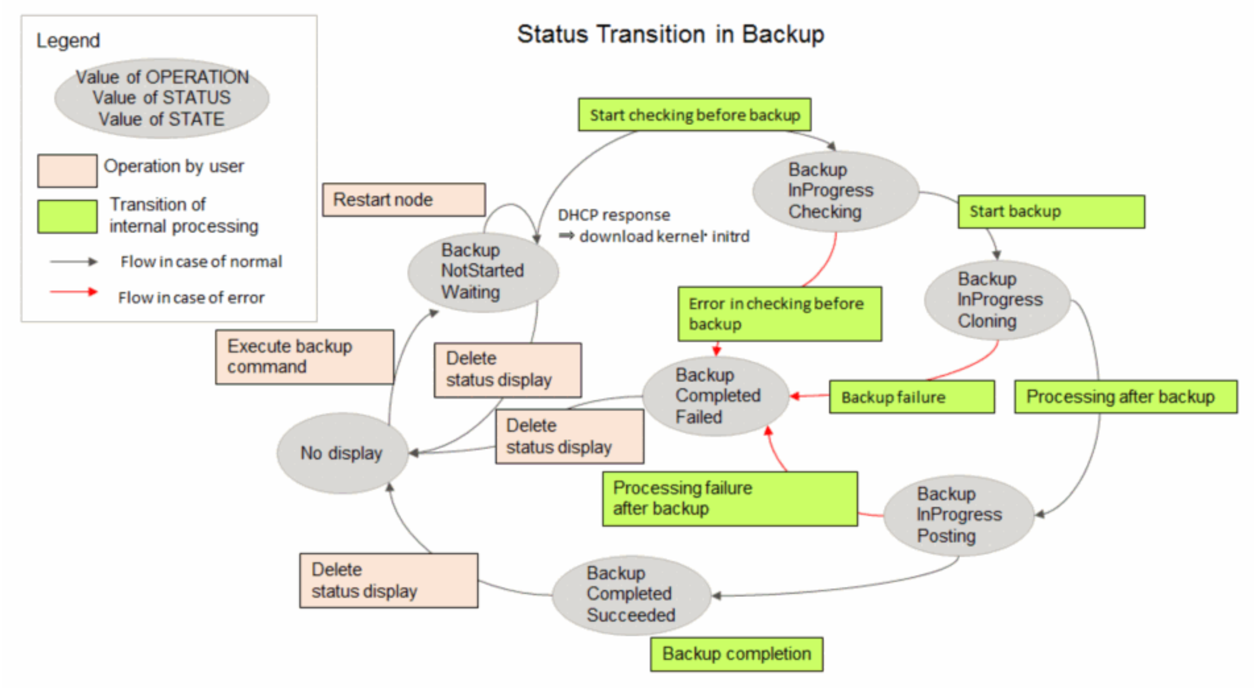
```
# paclone status clear --hostname host01
[INFO] INST 3201 paclone Command succeeded.(status clear)
```

3.9.4.3 Status Transition

The following diagram shows the status transition of OPERATION, STATUS, and STATE.

The transition flow in backup is the same as the transition flow in restoration. The following is an example of the backup process.

Figure 3.33 Status Transition in Backup


Status Transition in Backup
Legend
Value of OPERATION
Value of STATUS
Value of STATE
Operation by user
Transition of internal processing
Flow in case of normal
Flow in case of error
Start checking before backup
Backup InProgress Checking
Restart node
DHCP response
➡ download kernel･ initrd
Start backup
Backup NotStarted Waiting
Error in checking before backup
Backup InProgress Cloning
Execute backup command
Delete status display
Backup Completed Failed
Backup failure
Processing after backup
No display
Delete status display
Processing failure after backup
Backup InProgress Posting
Delete status display
Backup Completed Succeeded
Backup completion