ExpressCluster® X V1 *for Linux*

Getting Started Guide

Revision 1NA

Disclaimer

Information in this document is subject to change without notice. No part of this document may be reproduced or transmitted in any form by any means, electronic or mechanical, for any purpose, without the express written permission of NEC Corporation.

Trademark Information

ExpressCluster® X is a registered trademark of NEC Corporation.
FastSync™ is a trademark of NEC Corporation.
Linux is a registered trademark and trademark of Linus Torvalds in the United State and other countries.
RPM is a trademark of Red Hat, Inc.
Intel, Pentium and Xeon are registered trademarks and trademarks of Intel Corporation.
Microsoft and Windows are registered trademarks of Microsoft Corporation in the United State and other countries.
VERITAS, VERITAS Logo and all other VERITAS product names and slogans are trademarks and registered trademarks of VERITAS Software Corporation.
Other product names and slogans written in this manual are trademarks and registered trademarks of their respective companies.

Table of Contents

Preface

Who Should Use This Guide

ExpressCluster Getting Started Guide is intended for first-time users of the ExpressCluster. The guide covers topics such as product overview of the ExpressCluster, how the cluster system is installed, and the summary of other available guides. In addition, latest system requirements and restrictions are described.

How This Guide is Organized

Section I   Introducing ExpressCluster

Chapter 1   What is a cluster system?
Helps you to understand the overview of the cluster system and ExpressCluster.

Chapter 2   Using ExpressCluster
Provides instructions on how to use a cluster system and other related-information.

Section II   Installing ExpressCluster

Chapter 3   Installation requirements for ExpressCluster
Provides the latest information that needs to be verified before starting to use ExpressCluster.

Chapter 4   Latest version information
Provides information on latest version of the ExpressCluster.

Chapter 5   Notes and Restrictions
Provides information on known problems and restrictions.

Chapter 6   Upgrading ExpressCluster
Provides instructions on how to update the ExpressCluster.

Appendix

Appendix A.   Glossary
Appendix B.   Index

ExpressCluster X Documentation Set

The ExpressCluster X manuals consist of the following four guides. The title and purpose of each guide is described below:

Getting Started Guide
This guide is intended for all users. The guide covers topics such as product overview, system requirements, and known problems.

Installation and Configuration Guide
This guide is intended for system engineers and administrators who want to build, operate, and maintain a cluster system. Instructions for designing, installing, and configuring a cluster system with ExpressCluster are covered in this guide.

Reference Guide
This guide is intended for system administrators. The guide covers topics such as how to operate ExpressCluster, function of each module, maintenance-related information, and troubleshooting. The guide is supplement to the *Installation and Configuration Guide*.

Conventions

Note:
Used when the information given is important, but not related to the data loss and damage to the system and machine.

Important:
Used when the information given is necessary to avoid the data loss and damage to the system and machine.

Related Information:
Used to describe the location of the information given at the reference destination.

The following conventions are used in this guide.

Convention	Usage	Example
Bold	Indicates graphical objects, such as fields, list boxes, menu selections, buttons, labels, icons, etc.	In **User Name**, type your name. On the **File** menu, click **Open Database**.
Angled bracket within the command line	Indicates that the value specified inside of the angled bracket can be omitted.	`clpstat –s[-h` *`host_name`*`]`
#	Prompt to indicate that a Linux user has logged on as root user.	`clpcl  -s -a`
Monospace (courier)	Indicates path names, commands, system output (message, prompt, etc), directory, file names, functions and parameters.	`/server/`
Monospace bold (courier)	Indicates the value that a user actually enters from a command line.	Enter the following: **`clpcl -s -a`**
Monospace italic (courier)	Indicates that users should replace italicized part with values that they are actually working with.	`rpm –i ecxbuilder -<`*`version_number`*`>-<`*`release_number`*`>.i686.rpm`

Contacting NEC

For the latest product information, visit our website below:

http://www.ExpressCluster.com

Section I Introducing ExpressCluster

This section helps you to understand the overview of ExpressCluster and its system requirements. This section covers:

- Chapter 1 What is a cluster system?
- Chapter 2 Using ExpressCluster

Chapter 1 What is a cluster system?

This chapter describes overview of the cluster system.

This chapter covers:

Overview of the cluster system

A key to success in today's computerized world is to provide services without them stopping. A single machine down due to a failure or overload can stop entire services you provide with customers. This will not only result in enormous damage but also in loss of credibility you once enjoyed.

A cluster system is a solution to tackle such a disaster. Introducing a cluster system allows you to minimize the period during which operation of your system stops (down time) or to avoid system-down by load distribution.

As the word "cluster" represents, a cluster system is a system aiming to increase reliability and performance by clustering a group (or groups) of multiple computers. There are various types of cluster systems, which can be classified into the following three listed below. ExpressCluster is categorized as a high availability cluster.

High Availability (HA) Cluster

In this cluster configuration, one server operates as an active server. When the active server fails, a stand-by server takes over the operation. This cluster configuration aims for high-availability and allows data to be inherited as well. The high availability cluster is available in the shared disk type, data mirror type or remote cluster type.

Load Distribution Cluster

This is a cluster configuration where requests from clients are allocated to load-distribution hosts according to appropriate load distribution rules. This cluster configuration aims for high scalability. Generally, data cannot be taken over. The load distribution cluster is available in a load balance type or parallel database type.

High Performance Computing (HPC) Cluster

This is a cluster configuration where CPUs of all nodes are used to perform a single operation. This cluster configuration aims for high performance but does not provide general versatility.
Grid computing, which is one of the types of high performance computing that clusters a wider range of nodes and computing clusters, is a hot topic these days.

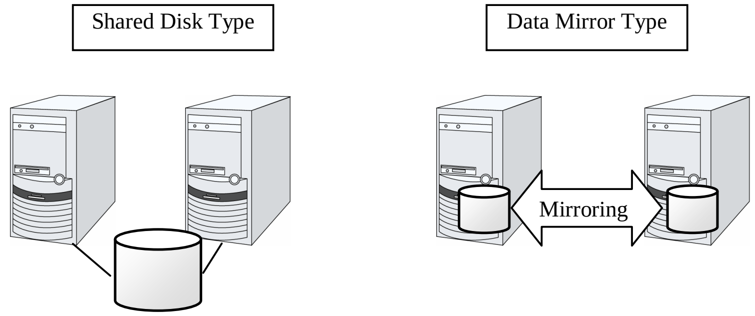
High Availability (HA) cluster

To enhance the availability of a system, it is generally considered that having redundancy for components of the system and eliminating a single point of failure is important. "Single point of failure" is a weakness of having a single computer component (hardware component) in the system. If the component fails, it will cause interruption of services. The high availability (HA) cluster is a cluster system that minimizes the time during which the system is stopped and increases operational availability by establishing redundancy with multiple servers.

The HA cluster is called for in mission-critical systems where downtime is fatal. The HA cluster can be divided into two types: shared disk type and data mirror type. The explanation for each type is provided below.

Shared disk type

Data must be inherited from one server to another in cluster systems. A cluster topology where data is stored in a shared disk with two or more servers using the data is called shared disk type.



- Expensive since a shared disk is necessary.
- Ideal for the system that handles large data

- Cheap since a shared disk is unnecessary.
- Ideal for the system with less data volume because of mirroring.

Figure 1-1: HA cluster configuration

If a failure occurs on a server where applications are running （active server）, the cluster system detects the failure and applications are automatically started in a stand-by server to take over operations. This mechanism is called failover. Operations to be inherited in the cluster system consist of resources including disk, IP address an application.

In a non-clustered system, a client needs to access a different IP address if an application is restarted on a server other than the server where the application was originally running. In contrast, many cluster systems allocate a virtual IP address on an operational basis. A server where the operation is running, be it an active or a stand-by server, remains transparent to a client. The operation is continued as if it has been running on the same server.

File system consistency must be checked to inherit data. A check command (for example, fsck or chkdsk in Linux) is generally run to check file system consistency. However, the larger the file system is, the more time spent for checking. While checking is in process, operations are stopped. For this problem, journaling file system is introduced to reduce the time required for failover.

Logic of the data to be inherited must be checked for applications. For example, roll-back or roll-forward is necessary for databases. With these actions, a client can continue operation only by re-executing the SQL statement that has not been committed yet.

A server with the failure can return to the cluster system as a stand-by server if it is physically separated from the system, fixed, and then succeeds to connect the system. Such returning is acceptable in production environments where continuity of operations is important.

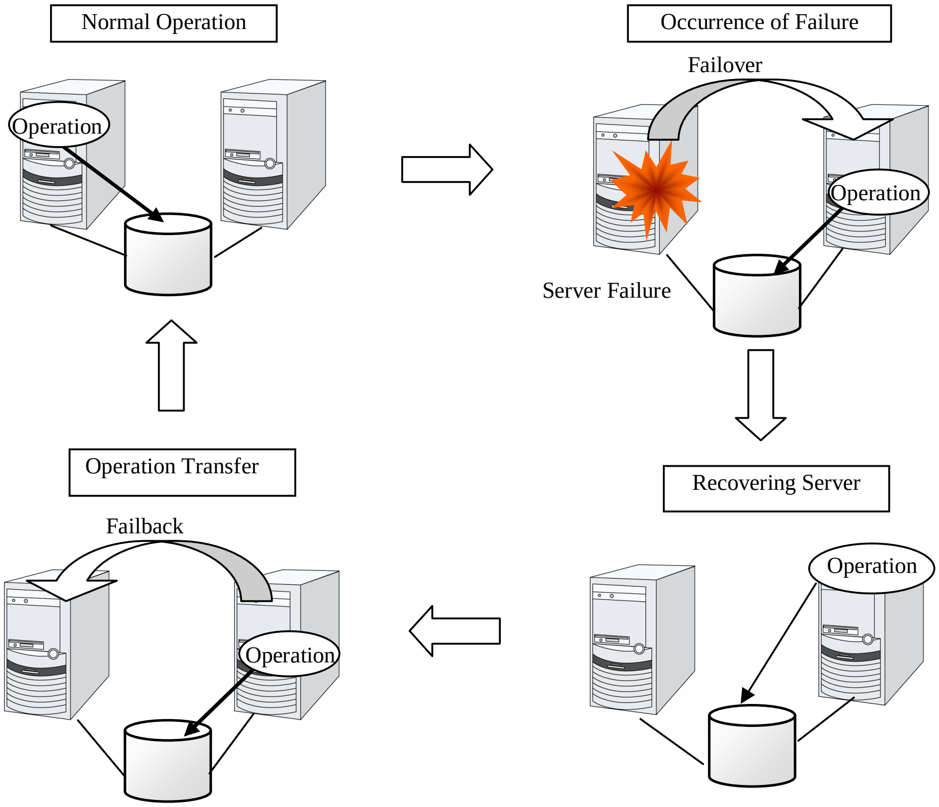


Figure 1-2: From occurrence of a failure to recovery

When the specification of the failover destination server does not meet the system requirements or overload occurs due to multi-directional stand-by, operations on the original server are preferred. In such a case, a failback takes place to resume operations on the original server.

A stand-by mode where there is one operation and no operation is active on the stand-by server, as shown in Figure 1-3, is referred to as uni-directional stand-by. A stand-by mode where there are two or more operations with each server of the cluster serving as both active and stand-by servers is referred to as multi-directional stand-by.

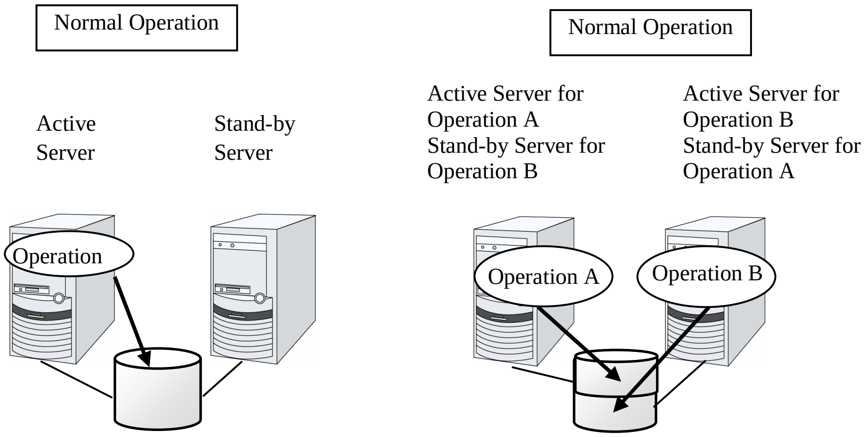


Figure 1-3: HA cluster topology

Data mirror type

The shared disk type cluster system is good for large-scale systems. However, creating a system with this type can be costly because shared disks are generally expensive. The data mirror type cluster system provides the same functions as the shared disk type with smaller cost through mirroring of server disks.

The data mirror type is not recommended for large-scale systems that handle a large volume of data since data needs to be mirrored between servers.

When a write request is made by an application, the data mirror engine not only writes data in the local disk but sends the write request to the stand-by server via the interconnect. Interconnect is a network connecting servers. It is used to monitor whether or not the server is activated in the cluster system. In addition to this purpose, interconnect is sometimes used to transfer data in the data mirror type cluster system. The data mirror engine on the stand-by server achieves data synchronization between stand-by and active servers by writing the data into the local disk of the stand-by server.

For read requests from an application, data is simply read from the disk on the active server.

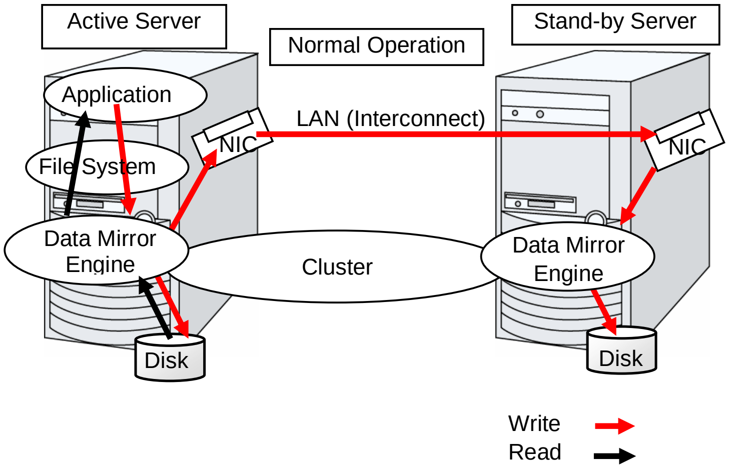


Figure 1-4: Data mirror mechanism

Snapshot backup is applied usage of data mirroring. Because the data mirror type cluster system has shared data in two locations, you can keep the disk of the stand-by server as snapshot backup without spending time for backup by simply separating the server from the cluster.

Failover mechanism and its problems

There are various cluster systems such as failover clusters, load distribution clusters, and high performance computing (HPC) clusters. The failover cluster is one of the high availability (HA) cluster systems that aim to increase operational availability through establishing server redundancy and passing operations being executed to another server when a failure occurs.

Error detection mechanism

Cluster software executes failover (for example, passing operations) when a failure that can impact continued operation is detected. The following section gives you a quick view of how the cluster software detects a failure.

Heartbeat and detection of server failures

Failures that must be detected in a cluster system are failures that can cause all servers in the cluster to stop. Server failures include hardware failures such as power supply and memory failures, and OS panic. To detect such failures, heartbeat is employed to monitor whether or not the server is active.

Some cluster software programs use heartbeat not only for checking whether or not the target is active through ping response, but for sending status information on the local server. Such cluster software programs begin failover if no heartbeat response is received in heartbeat transmission, determining no response as server failure. However, grace time should be given before determining failure, since a highly loaded server can cause delay of response. Allowing grace period results in a time lag between the moment when a failure occurred and the moment when the failure is detected by the cluster software.

Detection of resource failures

Factors causing stop of operations are not limited to stop of all servers in the cluster. Failure in disks used by applications, NIC failure, and failure in applications themselves are also factors that can cause the stop of operations. These resource failures need to be detected as well to execute failover for improved availability.

Accessing a target resource is a way employed to detect resource failures if the target is a physical device. For monitoring applications, trying to service ports within the range not impacting operation is a way of detecting an error in addition to monitoring whether or not application processes are activated.

Problems with shared disk type

In a failover cluster system of the shared disk type, multiple servers physically share the disk device. Typically, a file system enjoys I/O performance greater than the physical disk I/O performance by keeping data caches in a server.

What if a file system is accessed by multiple servers simultaneously?

Since a general file system assumes no server other than the local updates data on the disk, inconsistency between caches and the data on the disk arises. Ultimately the data will be corrupted. The failover cluster system locks the disk device to prevent multiple servers from mounting a file system, simultaneously caused by a network partition.

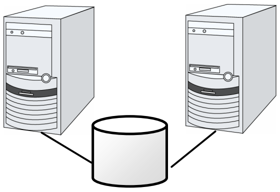


Figure 1-5: Cluster configuration with a shared disk

Network partition (split-brain-syndrome)

When all interconnects between servers are disconnected, failover takes place because the servers assume other server(s) are down. To monitor whether the server is activated, a heartbeat communication is used. As a result, multiple servers mount a file system simultaneously causing data corruption. This explains the importance of appropriate failover behavior in a cluster system at the time of failure occurrence.

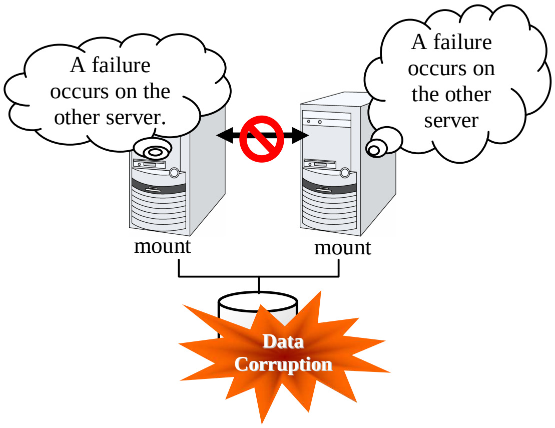

A failure occurs on the other server.
A failure occurs on the other server
mount
mount
Data Corruption

Figure 1-6: Network partition problem

The problem explained in the section above is referred to as “network partition” or “split-brain syndrome.” The failover cluster system is equipped with various mechanisms to ensure shared disk lock at the time when all interconnects are disconnected.

Inheriting cluster resources

As mentioned earlier, resources to be managed by a cluster include disks, IP addresses, and applications. The functions used in the failover cluster system to inherit these resources are described below.

Inheriting data

Data to be passed from a server to another in a cluster system is stored in a partition on the shared disk. This means inheriting data is re-mounting the file system of files that the application uses on a healthy server. What the cluster software should do is simply mount the file system because the shared disk is physically connected to a server that inherits data.

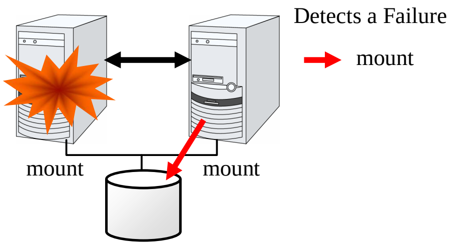


Figure 1-7: Inheriting data

The figure 1-7 may look simple, but consider the following issues in designing and creating a cluster system.

One issue to consider is recovery time for a file system. A file system to be inherited may have been used by another server or being updated just before the failure occurred and requires a file system consistency check. When the file system is large, the time spent for checking consistency will be enormous. It may take a few hours to complete the check and the time is wholly added to the time for failover (time to take over operation), and this will reduce system availability.

Another issue you should consider is writing assurance. When an application writes important data into a file, it tries to ensure the data to be written into a disk by using a function such as synchronized writing. The data that the application assumes to have been written is expected to be inherited after failover. For example, a mail server reports the completion of mail receiving to other mail servers or clients after it has securely written mails it received in a spool. This will allow the spooled mail to be distributed again after the server is restarted. Likewise, a cluster system should ensure mails written into spool by a server to become readable by another server.

Inheriting applications

The last to come in inheritance of operation by cluster software is inheritance of applications. Unlike fault tolerant computers (FTC), no process status such as contents of memory is inherited in typical failover cluster systems. The applications running on a failed server are inherited by rerunning them on a healthy server.

For example, when instances of a database management system (DBMS) are inherited, the database is automatically recovered (roll-forward/roll-back) by startup of the instances. The time needed for this database recovery is typically a few minutes though it can be controlled by configuring the interval of DBMS checkpoint to a certain extent.

Many applications can restart operations by re-execution. Some applications, however, require going through procedures for recovery if a failure occurs. For these applications, cluster software allows to start up scripts instead of applications so that recovery process can be written. In a script, the recovery process, including cleanup of files half updated, is written as necessary according to factors for executing the script and information on the execution server.

Summary of failover

To summarize the behavior of cluster software:

- Detects a failure (heartbeat/resource monitoring)
- Resolves a network partition (NP resolution)[1]
- Switches cluster resources
  - Pass data
  - Pass IP address
  - Application Inheriting

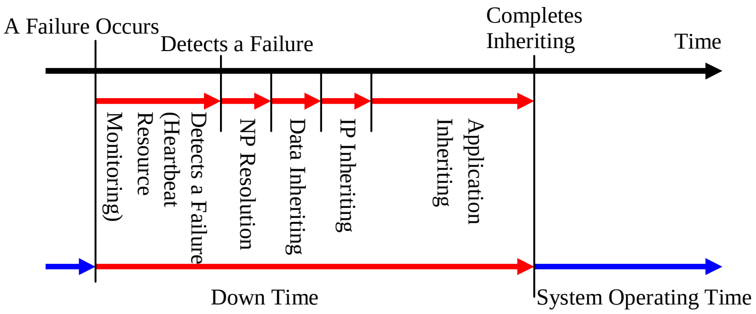


Figure 1-8: Failover time chart

Cluster software is required to complete each task quickly and reliably (see Figure 1-8.) Cluster software achieves high availability with due consideration on what has been described so far.

Note:
There is no "NP Resolution" time as described in Figure 1-8 in Linux.

Eliminating single point of failure

Having a clear picture of the availability level required or aimed is important in building a high availability system. This means when you design a system, you need to study cost effectiveness of countermeasures, such as establishing a redundant configuration to continue operations and recovering operations within a short period of time, against various failures that can disturb system operations.

Single point of failure (SPOF), as described previously, is a component where failure can lead to stop of the system. In a cluster system, you can eliminate the system's SPOF by establishing server redundancy. However, components shared among servers, such as shared disk may become a SPOF. The key in designing a high availability system is to duplicate or eliminate this shared component.

A cluster system can improve availability but failover will take a few minutes for switching systems. That means time for failover is a factor that reduces availability. Solutions for the following three, which are likely to become SPOF, will be discussed hereafter although technical issues that improve availability of a single server such as ECC memory and redundant power supply are important.

- Shared disk
- Access path to the shared disk
- LAN

[1] There is no "NP resolution" in Linux.

Shared disk

Typically a shared disk uses a disk array for RAID. Because of this, the bare drive of the disk does not become SPOF. The problem is the RAID controller is incorporated. Shared disks commonly used in many cluster systems allow controller redundancy.

In general, access paths to the shared disk must be duplicated to benefit from redundant RAID controller. There are still things to be done to use redundant access paths in Linux (described later in this chapter). If the shared disk has configuration to access the same logical disk unit (LUN) from duplicated multiple controllers simultaneously, and each controller is connected to one server, you can achieve high availability by failover between nodes when an error occurs in one of the controllers.

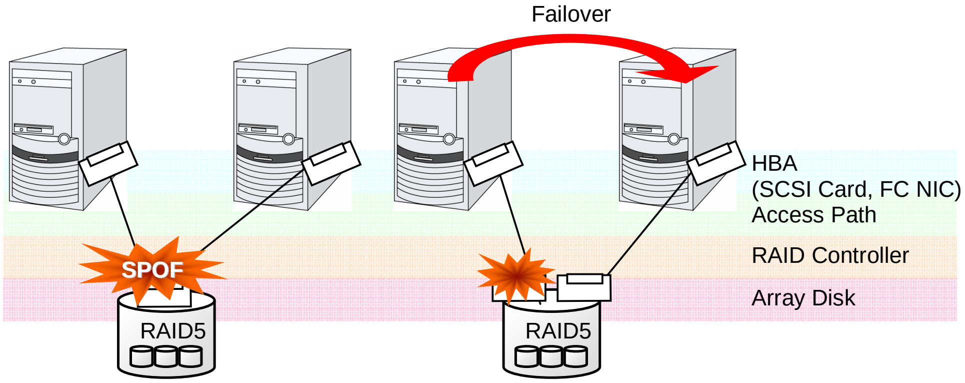


Figure 1-9: Example of the shared disk RAID controller and access paths being SPOF (left) and an access path connected to a RAID controller

With a failover cluster system of data mirror type, where no shared disk is used, you can create an ideal system having no SPOF because all data is mirrored to the disk in the other server. However you should consider the following issues:

- Disk I/O performance in mirroring data over the network (especially writing performance)
- System performance during mirror resynchronization in recovery from server failure (mirror copy is done in the background)
- Time for mirror resynchronization (clustering cannot be done until mirror resynchronization is completed)

In a system with frequent data viewing and a relatively small volume of data, choosing the data mirror type for clustering is a key to increase availability.

Access path to the shared disk

In a typical configuration of the shared disk type cluster system, the access path to the shared disk is shared among servers in the cluster. To take SCSI as an example, two servers and a shared disk are connected to a single SCSI bus. A failure in the access path to the shared disk can stop the entire system.

What you can do for this is to have a redundant configuration by providing multiple access paths to the shared disk and make them look as one path for applications. The device driver allowing such is called a path failover driver. Path failover drivers are often developed and released by shared disk vendors. Path failover drivers in Linux are still under development. For the time being, as discussed earlier, offering access paths to the shared disk by connecting a server on an array controller on the shared disk basis is the way to ensure availability in Linux cluster systems.

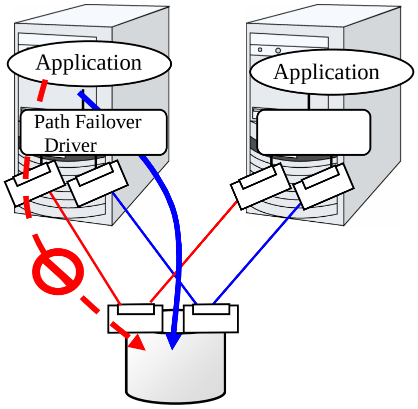


Figure 1-10: Path failover driver

LAN

In any systems that run services on a network, a LAN failure is a major factor that disturbs operations of the system. If appropriate settings are made, availability of cluster system can be increased through failover between nodes at NIC failures. However, a failure in a network device that resides outside the cluster system disturbs operation of the system.

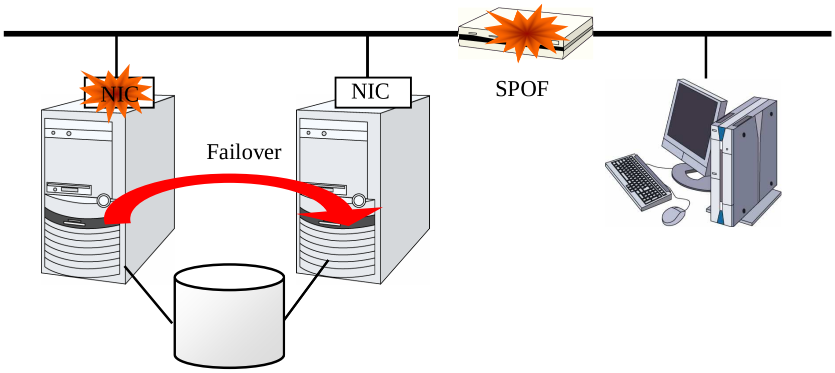


Figure 1-11: Example of router becoming SPOF

LAN redundancy is a solution to tackle device failure outside the cluster system and to improve availability. You can apply ways used for a single server to increase LAN availability. For example, choose a primitive way to have a spare network device with its power off, and manually replace a failed device with this spare device. Choose to have a multiplex network path through a redundant configuration of high-performance network devices, and switch paths automatically. Another option is to use a driver that supports NIC redundant configuration such as Intel's ANS driver.

Load balancing appliances and firewall appliances are also network devices that are likely to become SPOF. Typically they allow failover configurations through standard or optional software. Having redundant configuration for these devices should be regarded as requisite since they play important roles in the entire system.

Operation for availability

Evaluation before staring operation

Given many of factors causing system troubles are said to be the product of incorrect settings or poor maintenance, evaluation before actual operation is important to realize a high availability system and its stabilized operation. Exercising the following for actual operation of the system is a key in improving availability:

- Clarify and list failures, study actions to be taken against them, and verify effectiveness of the actions by creating dummy failures.
- Conduct an evaluation according to the cluster life cycle and verify performance (such as at degenerated mode)
- Arrange a guide for system operation and troubleshooting based on the evaluation mentioned above.

Having a simple design for a cluster system contributes to simplifying verification and improvement of system availability.

Failure monitoring

Despite the above efforts, failures still occur. If you use the system for long time, you cannot escape from failures: hardware suffers from aging deterioration and software produces failures and errors through memory leaks or operation beyond the originally intended capacity. Improving availability of hardware and software is important yet monitoring for failure and troubleshooting problems is more important. For example, in a cluster system, you can continue running the system by spending a few minutes for switching even if a server fails. However, if you leave the failed server as it is, the system no longer has redundancy and the cluster system becomes meaningless should the next failure occur.

If a failure occurs, the system administrator must immediately take actions such as removing a newly emerged SPOF to prevent another failure. Functions for remote maintenance and reporting failures are very important in supporting services for system administration. Linux is known for providing good remote maintenance functions. Mechanism for reporting failures are coming in place. To achieve high availability with a cluster system, you should:

- Remove or have complete control on single point of failure.
- Have a simple design that has tolerance and resistance for failures, and be equipped with a guide for operation and troubleshooting.
- Detect a failure quickly and take appropriate action against it.

Chapter 2 Using ExpressCluster

This chapter explains the components of ExpressCluster, how to design a cluster system, and how to use ExpressCluster.

This chapter covers:

What is ExpressCluster?

ExpressCluster is software that enhances availability and expandability of systems by a redundant (clustered) system configuration. The application services running on the active server are automatically inherited to a standby server when an error occurs in the active server.

ExpressCluster modules

ExpressCluster consists of following three modules:

ExpressCluster Server

A core component of ExpressCluster. Includes all high availability function of the server. The server function of the WebManager is also included.

ExpressCluster X WebManager (WebManager)

A tool to manage ExpressCluster operations. Uses a Web browser as a user interface. The WebManager is installed in ExpressCluster Server, but it is distinguished from the ExpressCluster Server because the WebManager is operated from the Web browser on the management PC.

ExpressCluster X Builder (Builder)

A tool for editing the cluster configuration data. The Builder also uses Web browser as a user interface. The Builder needs to be installed separately from the ExpressCluster Server on the machine where you use the Builder.

Software configuration of ExpressCluster

The software configuration of ExpressCluster should look similar to the figure below. Install the ExpressCluster Server (software) on a Linux server, and the Builder on a management PC or a server. The WebManager does not have to be installed separately because it is automatically installed at the time of ExpressCluster Server installation. The Web browser in which you use the WebManager can be on a management PC.

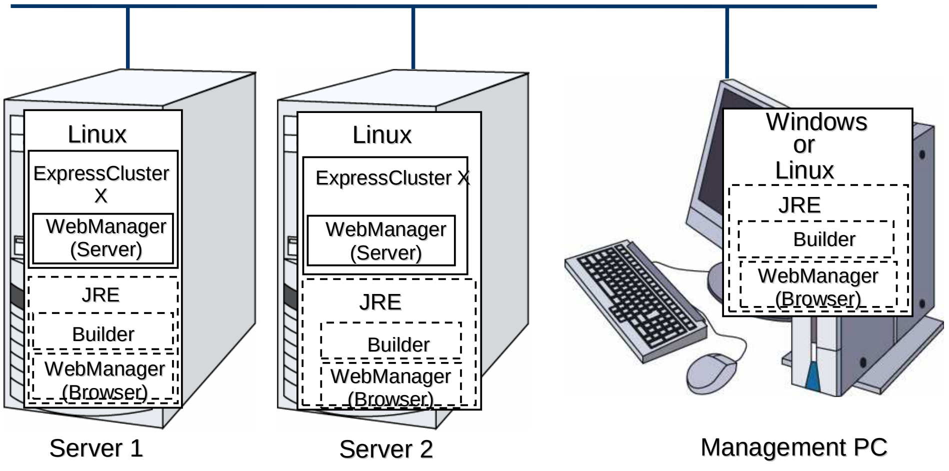


Figure 2-1 Software configuration of ExpressCluster

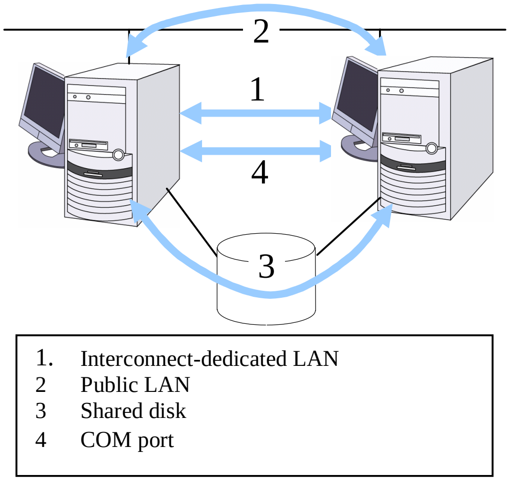
How an error is detected in ExpressCluster

There are three kinds of monitoring in ExpressCluster: (1) server monitoring, (2) application monitoring, and (3) internal monitoring. These monitoring functions let you detect an error quickly and reliably. The details of the monitoring functions are described below.

What is server monitoring?

Server monitoring is the most basic function of the failover-type cluster system. It monitors if a server that constitutes a cluster is properly working.

ExpressCluster regularly checks whether other servers are properly working in the cluster system. This way of verification is called "heartbeat communication." The heartbeat communication uses the following communication paths:



Interconnect-dedicated LAN

Uses an Ethernet NIC in communication path dedicated to the failover-type cluster system. This is used to exchange information between the servers as well as to perform heartbeat communication.

Public LAN

Uses a communication path used for communication with client machine as an alternative interconnect. Any Ethernet NIC can be used as long as TCP/IP can be used. This is also used to exchange information between the servers and to perform heartbeat communication.

Shared disk

Creates an ExpressCluster-dedicated partition (ExpressCluster partition) on the disk that is connected to all servers that constitute the failover-type cluster system, and performs heartbeat communication on the ExpressCluster partition.

COM port

Performs heartbeat communication between the servers that constitute the failover-type cluster system through a COM port, and checks whether other servers are working properly.

Having these communication paths dramatically improves the reliability of the communication between the servers, and prevents the occurrence of network partition.

Note:
Network partition (also known as "split-brain syndrome") refers to a condition when a network gets split by having a problem in all communication paths of the servers in a cluster. In a cluster system that is not capable of handling a network partition, a problem occurred in a communication path and a server cannot be distinguished. As a result, multiple servers may access the same resource and cause the data in a cluster system to be corrupted.

What is application monitoring?

Application monitoring is a function that monitors applications and factors that cause a situation where an application cannot run.

Activation status of application monitoring

An error can be detected by starting up an application from an exec resource in ExpressCluster and regularly checking whether a process is active or not by using the pid monitor resource. It is effective when the factor for application to stop is due to error termination of an application.

Note:
An error in resident process cannot be detected in an application started up by ExpressCluster. When the monitoring target application starts and stops a resident process, an internal application error (such as application stalling, result error) cannot be detected.

Resource monitoring

An error can be detected by monitoring the cluster resources (such as disk partition and IP address) and public LAN using the monitor resources of the ExpressCluster. It is effective when the factor for application to stop is due to an error of a resource which is necessary for an application to operate.

What is internal monitoring?

Internal monitoring refers to an inter-monitoring of modules within ExpressCluster. It monitors whether each monitoring function of ExpressCluster is properly working. Activation status of ExpressCluster process monitoring is performed within ExpressCluster.

Monitorable and non-monitorable errors

There are monitorable and non-monitorable errors in ExpressCluster. It is important to know what can or cannot be monitored when building and operating a cluster system.

Detectable and non-detectable errors by server monitoring

Monitoring condition: A heartbeat from a server with an error is stopped

Example of errors that can be monitored:

- Hardware failure (of which OS cannot continue operating)
- System panic

Example of error that cannot be monitored:

- Partial failure on OS (for example, only a mouse or keyboard does not function)

Detectable and non-detectable errors by application monitoring

Monitoring conditions: Termination of applications with errors, continuous resource errors, and disconnection of a path to the network devices.

Example of errors that can be monitored:

- Abnormal termination of an application
- Failure to access the shared disk (such as HBA[2] failure)
- Public LAN NIC problem

Example of errors that cannot be monitored:

- Application stalling and resulting in error. ExpressCluster cannot monitor application stalling and error results. However, it is possible to perform failover by creating a program that monitors applications and terminates itself when an error is detected, starting the program using the exec resource, and monitoring application using the PID monitor resource.

Network partition resolution

When the stop of a heartbeat is detected from a server, ExpressCluster determines whether it is an error in a server or a network partition. If it is judged as a server failure, failover (activate resources and start applications on a healthy server) is performed. If it is judged as network partition, protecting data is given priority over inheriting operations, so processing such as emergency shutdown is performed.
The following is the network partition resolution method:

- ping method

Related Information:

For the details on the network partition resolution method, see Chapter 9, “Details on network partition resolution resources” in Section II of the *Reference Guide*.

Failover mechanism

When an error is detected, ExpressCluster determines whether an error detected before failing over is an error in a server or a network partition. Then a failover is performed by activating various resources and starting up applications on a properly working server.

The group of resources which fail over at the same time is called a “failover group.” From a user’s point of view, a failover group appears as a virtual computer.

Note:
In a cluster system, a failover is performed by restarting the application from a properly working node. Therefore, what is saved in an application memory cannot be failed over.

From occurrence of error to completion of failover takes a few minutes. See the figure 2-2 below:

[2] HBA is an abbreviation for host bus adapter. This adapter is not for the shared disk, but for the server.

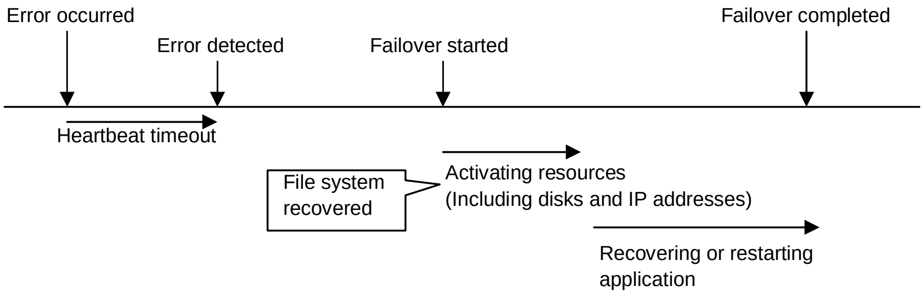


Figure 2-2 Failover time chart

Heartbeat timeout

- The time for a standby server to detect an error after that error occurred on the active server.
- The setting values of the cluster properties should be adjusted depending on the application load. (The default value is 90 seconds.)

Activating various resources

- The time to activate the resources necessary for operating an application.
- The resources can be activated in a few seconds in ordinary settings, but the required time changes depending on the type and the number of resources registered to the failover group. For more information, refer to the *Installation and Configuration Guide*.

Start script execution time

- The data recovery time for a roll-back or roll-forward of the database and the startup time of the application to be used in operation.
- The time for roll-back or roll-forward can be predicted by adjusting the check point interval. For more information, refer to the document that comes with each software product.

Failover resources

ExpressCluster can fail over the following resources:

Switchable partition

- Resources such as disk resource and mirror disk resource.
- A disk partition to store the data that the application takes over.

Floating IP Address

- By connecting an application using the floating IP address, a client does not have to be conscious about switching the servers due to failover processing.
- It is achieved by dynamic IP address allocation to the public LAN adapter and sending ARP packet. Connection by floating IP address is possible from most of the network devices.

Script (exec resource)

- In ExpressCluster, applications are started up from the scripts.
- The file failed over on the shared disk may not be complete as data even if it is properly working as a file system. Write the recovery processing specific to an application at the

time of failover in addition to the startup of an application in the scripts.

Note:
In a cluster system, failover is performed by restarting the application from a properly working node. Therefore, what is saved in an application memory cannot be failed over.

System configuration of the failover type cluster

In a failover-type cluster, a disk array device is shared between the servers in a cluster. When an error occurs on a server, the standby server takes over the applications using the data on the shared disk.

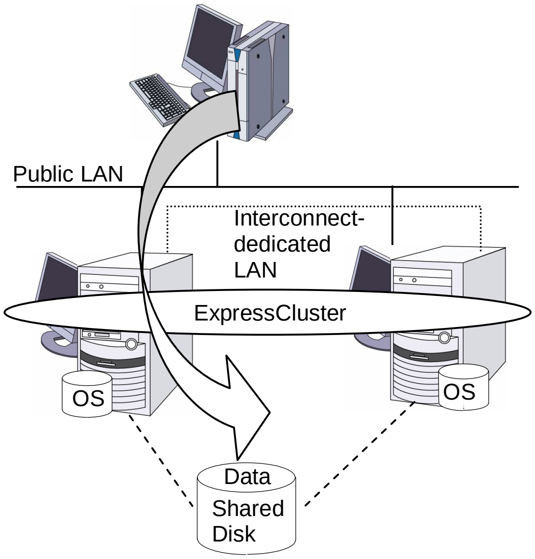


Figure 2-3 System configuration

A failover-type cluster can be divided into the following categories depending on the cluster topologies:

Uni-Directional Standby Cluster System

In the uni-directional standby cluster system, the active server runs applications while the other server, the standby server, does not. This is the simplest cluster topology and you can build a high-availability system without performance degradation after failing over.

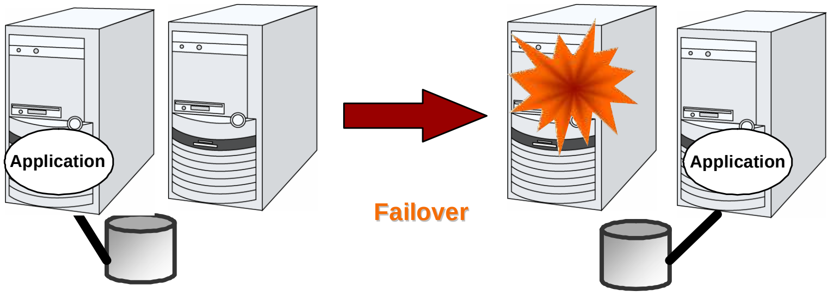
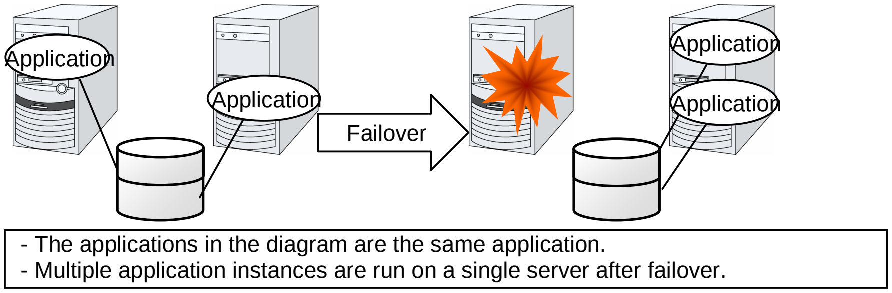


Figure 2-4 Uni-directional standby cluster system

Same Application Multi Directional Standby Cluster System

In the same application multi-directional standby cluster system, the same applications are activated on multiple servers. These servers also operate as standby servers. The applications must support multi-directional standby operation. When the application data can be split into multiple data, depending on the data to be accessed, you can build a load distribution system per data partitioning basis by changing the client’s connecting server.

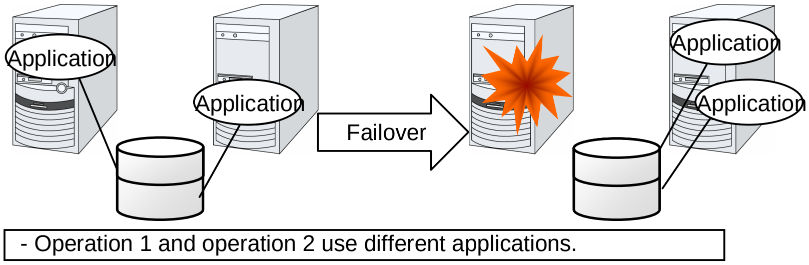


- The applications in the diagram are the same application.
- Multiple application instances are run on a single server after failover.

Figure 2-5 Same application multi directional standby cluster system

Different Application – Multi Directional Standby Cluster System

In the different application multi-directional standby cluster system, different applications are activated on multiple servers and these servers also operate as standby servers. The applications do not have to support multi-directional standby operation. A load distribution system can be built per application unit basis.



- Operation 1 and operation 2 use different applications.

Figure 2-6 Different application multi directional standby cluster system

Node to Node Configuration

The configuration can be expanded with more nodes by applying the configurations introduced thus far. In a node to node configuration described below, three different applications are run on three servers and one standby server takes over the application if any problem occurs. In a uni-directional standby cluster system, one of the two servers functions as a standby server. However, in a node to node configuration, only one of the four server functions as a standby server and performance deterioration is not anticipated if an error occurs only on one server.

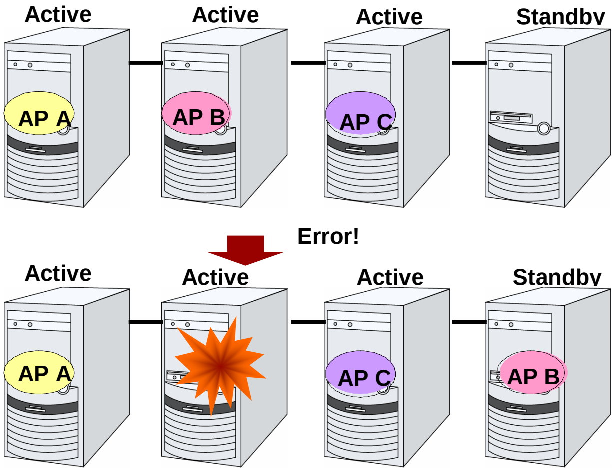


Figure 2-7 Node to Node configuration

Hardware configuration of the shared disk type cluster

The hardware configuration of the shared disk in ExpressCluster is described below. In general, the following is used for communication between the servers in a cluster system:

- Two NIC cards (one for external communication, one for ExpressCluster)
- COM port connected by RS232C cross cable
- Specific space of a shared disk

SCSI or FibreChannel can be used for communication interface to a shared disk; however, recently FibreChannel is more commonly used.

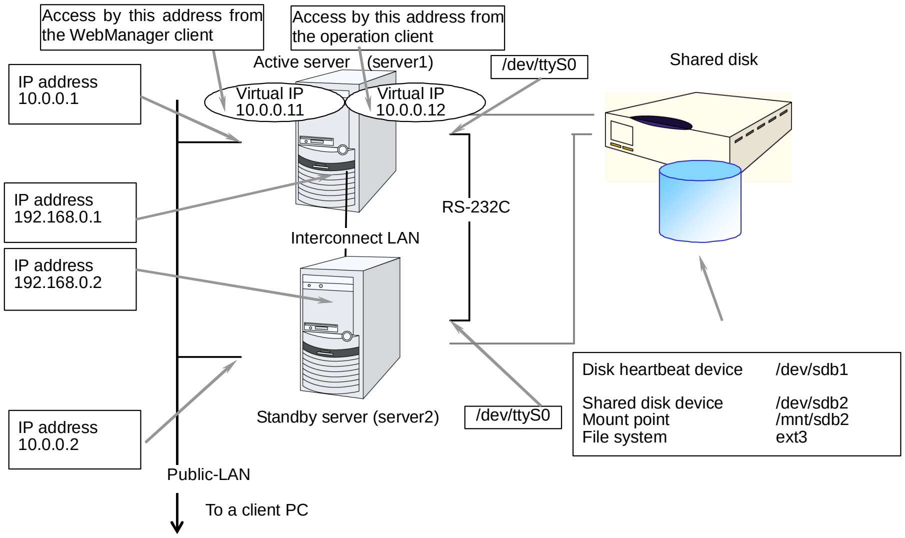


Figure 2-8 Sample of cluster environment when a shared disk is used

Hardware configuration of the mirror disk type cluster

The hardware configuration of the mirror disk in ExpressCluster is described below.

Unlike the shared disk type, a network to copy the mirror disk data is necessary. In general, a network is used with NIC for internal communication in ExpressCluster.

Mirror disks need to be separated from the operating system; however, they do not depend on a connection interface (IDE or SCSI.)

Figure 2-9 Sample of cluster environment when mirror disks are used (when allocating cluster partition and data partition to the disk where OS is installed):

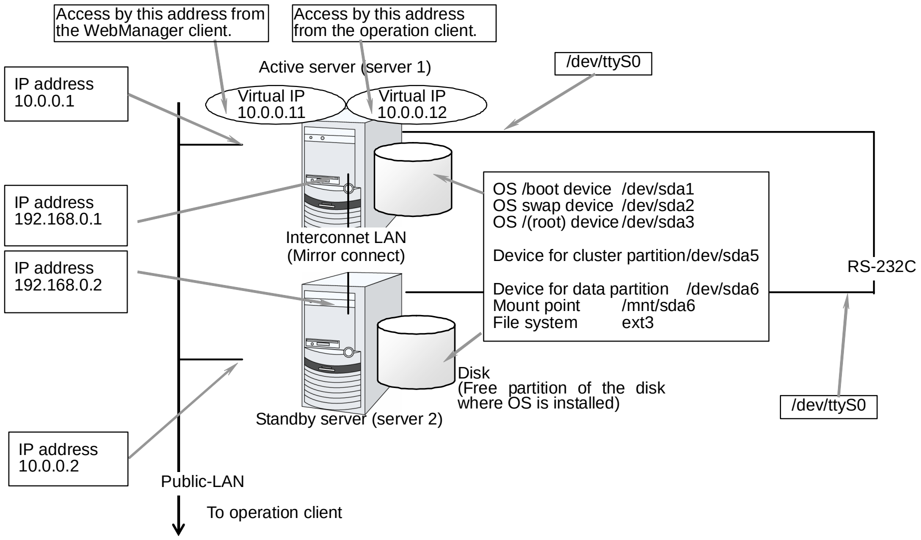
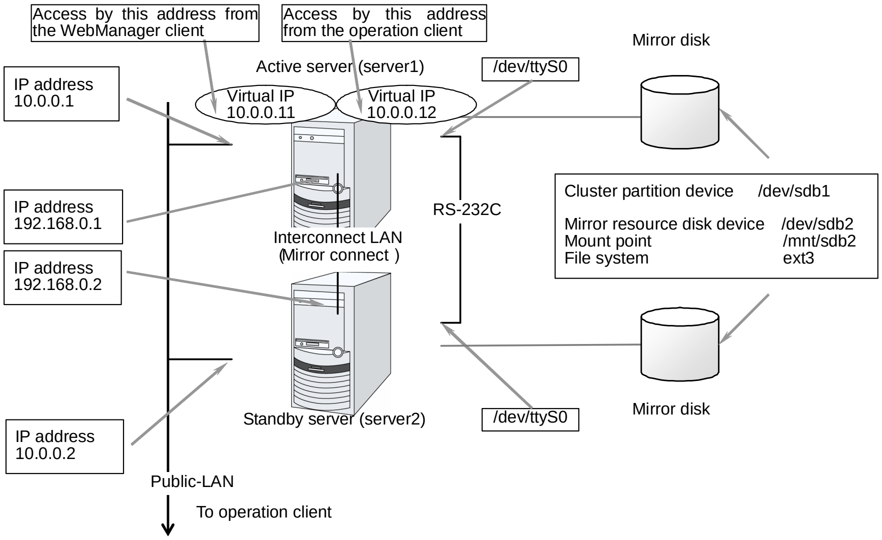

Access by this address from the WebManager client.
Access by this address from the operation client.
Active server (server 1)
/dev/ttyS0
IP address 10.0.0.1
Virtual IP 10.0.0.11
Virtual IP 10.0.0.12
IP address 192.168.0.1
OS /boot device /dev/sda1
OS swap device /dev/sda2
OS /(root) device /dev/sda3
Device for cluster partition/dev/sda5
Device for data partition /dev/sda6
Mount point /mnt/sda6
File system ext3
Interconnet LAN
(Mirror connect)
IP address 192.168.0.2
RS-232C
Disk
(Free partition of the disk where OS is installed)
/dev/ttyS0
Standby server (server 2)
IP address 10.0.0.2
Public-LAN
To operation client

Figure 2-10 Sample of cluster environment when mirror disks are used (when disks for cluster partition and data partition are prepared):



What is cluster object?

In ExpressCluster, the various resources are managed as the following groups:

Cluster object
Configuration unit of a cluster.

Server object
Indicates the physical server and belongs to the cluster object.

Heartbeat resource object
Indicates the network part of the physical server and belongs to the server object.

Group object
Indicates a virtual server and belongs to the cluster object.

Group resource object
Indicates resources (network, disk) of the virtual server and belongs to the group object.

Monitor resource object
Indicates monitoring mechanism and belongs to the cluster object.

What is a resource?

In ExpressCluster, a group used for monitoring the target is called “resources.” There are four types of resources and are managed separately. Having resources allows distinguishing what is monitoring and what is being monitored more clearly. It also makes building a cluster and handling an error easy. The resources can be divided into heartbeat resources, group resources, and monitor resources.

Heartbeat resources

Heartbeat resources are used for verifying whether the other server is working properly between servers. The following heartbeat resources are currently supported:

LAN heartbeat resource
Uses Ethernet for communication.

Kernel mode LAN heartbeat resource
Uses Ethernet for communication.

COM heartbeat resource
Uses RS232C (COM) for communication.

Disk heartbeat resource
Uses a specific partition (cluster partition for disk heartbeat) on the shared disk for communication. It can be used only on a shared disk configuration.

Network partition resolution resources

The resource used for solving the network partition is shown below:

PING network partition resolution resource
This is a network partition resolution resource by the PING method.

Group resources

A group resource constitutes a unit when a failover occurs. The following group resources are currently supported:

Floating IP resource (fip)
Provides a virtual IP address. A client can access virtual IP address the same way as the regular IP address.

EXEC resource (exec)
Provides a mechanism for starting and stopping the applications such as DB and httpd.

Disk resource (disk)
Provides a specified partition on the shared disk. It can be used only on a shared disk configuration.

Mirror disk resource (md)
Provides a specified partition on the mirror disk. It can be used only on a mirror disk configuration.

Raw resource (raw)
Provides a raw device on the shared disk. It can be used only on a shared disk configuration.

VxVM disk group resource (vxdg)
Provides a VxVM disk group on the shared disk. It is used with VxVM volume resource and can be used only on a shared disk configuration.

VxVM volume resource (vxvol)
Provides a VxVM volume on the shared disk. It is used with VxVM disk group resource and can be used only on a shared disk configuration.

NAS resource (nas)
Connect to the shared resources on NAS server. Note that it is not a resource that the cluster server behaves as NAS server.

Virtual IP resource (vip)
Provides a virtual IP address. This can be accessed from a client in the same way as a general IP address. This can be used in the remote cluster configuration among different network addresses.

Monitor resources

A monitor resource monitors a cluster system. The following monitor resources are currently supported:

IP monitor resource (ipw)
Provides a monitoring mechanism of an external IP address.

Disk monitor resource (diskw)
Provides a monitoring mechanism of the disk. It also monitors the shared disk.

Mirror disk monitor resource (mdw)
Provides a monitoring mechanism of the mirroring disks.

Mirror disk connect monitor resource (mdnw)
Provides a monitoring mechanism of the mirror disk connect.

PID monitor resource (pidw)
Provides a monitoring mechanism to check whether a process started up by exec resource is active or not.

User mode monitor resource (userw)
Provides a monitoring mechanism for a stalling problem in the user space.

Raw monitor resource (raww)

Provides a monitoring mechanism for the disks. Load to the system can be reduced because a raw device is used and the read size is small. It can be used for monitoring a shared disk.

NIC Link Up/Down monitor resource (miiw)

Provides a monitoring mechanism for link status of LAN cable.

VxVM daemon monitor resource (vxdw)

Provides a monitoring mechanism for a VxVM daemon. It can be used only on a shared disk configuration.

VxVM volume monitor resource (vxvolw)

Provides a monitoring mechanism for a VxVM volume. It can be used only on a shared disk configuration.

Multi target monitor resource (mtw)

Provides a status with multiple monitor resources.

Virtual IP monitor resource (vipw)

Provides a mechanism for sending RIP packets of a virtual IP resource.

ARP monitor resource (arpw)

Provides a mechanism for sending ARP packets of a floating IP resource or a virtual IP resource.

DB2 monitor resource (db2w)

Provides a monitoring mechanism for IBM DB2 database.

http monitor resource (httpw)

Provides a monitoring mechanism for HTTP server.

MySQL monitor resource (mysqlw)

Provides a monitoring mechanism for MySQL database.

nfs monitor resource (nfsw)

Provides a monitoring mechanism for nfs file server.

Oracle monitor resource (oraclew)

Provides a monitoring mechanism for Oracle database.

PostgreSQL monitor resource (psqlw)

Provides a monitoring mechanism for PostgreSQL database.

samba monitor resource (sambaw)

Provides a monitoring mechanism for samba file server.

smtp monitor resource (smtpw)

Provides a monitoring mechanism for SMTP server.

Sybase monitor resource (sybasew)

Provides a monitoring mechanism for Sybase database.

Tuxedo monitor resource (tuxw)

Provides a monitoring mechanism for Tuxedo application server.

Websphere monitor resource (wasw)

Provides a monitoring mechanism for Websphere application server.

Weblogic monitor resource (wlsw)

Provides a monitoring mechanism for Weblogic application server.

Getting started with ExpressCluster

Refer to the following guides when building a cluster system with ExpressCluster:

Latest information

Refer to Section II, “Installing ExpressCluster” in this guide.

Designing a cluster system

Refer to Section I, “Configuring a cluster system” in the *Installation and Configuration Guide* and Section II, “Resource details” in the *Reference Guide*.

Configuring a cluster system

Refer to the *Installation and Configuration Guide.* When using an optional monitoring command, refer to the *Administrator’s Guide* that is available for each target monitoring application.

Troubleshooting the problem

Refer to Section III, “Maintenance information” in the *Reference Guide*.

Section II Installing ExpressCluster

This section provides the latest information on the ExpressCluster. The latest information on the supported hardware and software is described in detail. Topics such as restrictions, known problems, and how to troubleshoot the problem are covered.

Chapter 3 Installation requirements for ExpressCluster

This chapter provides information on system requirements for ExpressCluster.

This chapter covers:

Hardware

ExpressCluster operates on the following server architectures:

- IA-32

General server requirements

Required specifications for ExpressCluster Server are the following:

- RS-232C port 1 port (not necessary when configuring a cluster with 3 or more nodes)
- Ethernet port 2 or more ports
- Shared disk (not required when the Replicator is used)
- Mirror disk or empty partition for mirror (required when the Replicator is used)
- CD-ROM drive

Required for communication with the Builder when configuring and changing the existing configuration are one of the following:

- Removable media (for example, floppy disk drive or USB memory)
- A machine to operate the Builder and a way to share files

Supported disk interfaces

Disk types that are supported as mirror disks include the following:

Disk type	Host side driver	Remarks
IDE	ide	Validated up to 120GB
SCSI	aic7xxx	
SCSI	aic79xx	
SCSI	sym53c8xx	
SCSI	mptbase,mptscsih	
SCSI	mptsas	
RAID	Megaraid (SCSI type)	
RAID	megaraid (IDE type)	Validated up to 275GB
S-ATA	sata-nv	Validated up to 80GB
S-ATA	ata-piix	Validated up to 120GB

Supported network interfaces

The following are the network boards that are supported as a mirror disk connect for the mirror disk of the Replicator:

Chip	Driver
Intel 82557/8/9	e100
Intel 82540EM Intel 82544EI Intel 82546EB Intel 82546GB Intel 82573L Intel 80003ES2LAN Intel 631xESB/632xESB	e1000
Broadcom BCM5701 Broadcom BCM5703 Broadcom BCM5721	bcm5700
Broadcom BCM5721	tg3

Only typical examples are listed above and other products can also be used.

Software

System requirements for ExpressCluster Server

Supported distributions and kernel versions

The environment where ExpressCluster Server can operate depends on kernel module versions because there are kernel modules unique to ExpressCluster. Kernel versions that provide the complying kernel module are listed below.

ExpressCluster Server has only been validated on the kernel versions listed below.

IA-32

Distribution	Kernel version	Replicator support	Run clpka and clpkhb support	Express Cluster Version	Remarks
Red Hat Enterprise Linux AS/ES 4 (update3)	2.6.9-34.EL 2.6.9-34.ELsmp 2.6.9-34.ELhugemem	Yes	Yes	1.0.0-1~	1
Red Hat Enterprise Linux AS/ES 4 (update4)	2.6.9-42.EL 2.6.9-42.ELsmp 2.6.9-42.ELhugemem	Yes	Yes	1.0.1-1~	1
	2.6.9-42.0.3.EL 2.6.9-42.0.3.ELsmp 2.6.9-42.0.3.ELhugemem	Yes	Yes	1.0.2-1~	1
	2.6.9-42.0.8.EL 2.6.9-42.0.8.ELsmp 2.6.9-42.0.8.ELhugemem	Yes	Yes	1.0.2-1~	1
	2.6.9-42.0.10.EL 2.6.9-42.0.10.ELsmp 2.6.9-42.0.10.ELhugemem	Yes	Yes	1.0.2-1~	1
Red Hat Enterprise Linux AS/ES 4 (update5)	2.6.9-55.EL 2.6.9-55.ELsmp 2.6.9-55.ELhugemem	Yes	Yes	1.1.0-1~	1
	2.6.9-55.0.2.EL 2.6.9-55.0.2.ELsmp 2.6.9-55.0.2.ELhugemem	Yes	Yes	1.1.1-1~	1
Red Hat Enterprise Linux 5	2.6.18-8.el5 PAE-2.6.18-8.el5 xen-2.6.18-8.el5	Yes	Yes	1.1.1-1~	1
Novell SUSE LINUX Enterprise Server 9 (SP3)	2.6.5-7.244-default 2.6.5-7.244-smp 2.6.5-7.244-bigsmp	Yes	Yes	1.0.0-1~	
Novell SUSE LINUX Enterprise Server 10	2.6.16.21-0.8-default 2.6.16.21-0.8-smp 2.6.16.21-0.8-bigsmp 2.6.16.21-0.8-xen	Yes	Yes	1.1.1-1~	1
Novell SUSE LINUX Enterprise Server 10 (SP1)	2.6.16.46-0.12-default 2.6.16.46-0.12-smp 2.6.16.46-0.12-bigsmp 2.6.16.46-0.12-xen	Yes	Yes	1.1.1-1~	1

Notes

[1] ExpressCluster X version is 1.0.1-1 or later is required for “vxfs” file system support.

Applications supported by monitoring options

Version information of the applications to be monitored by built-in monitor resources is described below.

For the support information on the monitoring options of command type (that are registered as script resources at setup), which is provided on ExpressCluster 1.0.x-x, see the administrator's guide of each option.

IA32

Monitor resource	Monitored application	ExpressCluster version	Remarks
Oracle monitor	Oracle 10g 10.2.0.1.0	1.1.0-1~	
DB2 monitor	DB2 V9.1 Fix Pack2	1.1.0-1~	
PostgresSQL monitor	PostgresSQL 8.2.3-1	1.1.0-1~	
MySQL monitor	MySQL 5.1.17-0	1.1.0-1~	
Sybase monitor	Sybase 12.5	1.1.0-1~	
Samba monitor	Samba Version 3.0.10 Release 1.4E.9	1.1.0-1~	
NFS monitor	NFS Version 1.0.6 Release 70.EL4	1.1.0-1~	
HTTP monitor	apache Version 2.0.52 Release 25.ent	1.1.0-1~	
SMTP monitor	sendmail Version 8.13.1 Release 3.RHEL4.5	1.1.0-1~	
Tuxedo monitor	Tuxedo 8.1 Patch Level 099	1.1.0-1~	
Weblogic monitor	WebLogic 9.2	1.1.0-1~	
Websphere monitor	WebSphere 6.1.0.0	1.1.0-1~	

Required memory and disk size

	Required memory size		Required disk size	
	User mode	Kernel mode	Right after installation	Max. during operation
IA-32	64MB	When the synchronization mode and a file system other than vxfs are used: 64MB + (2MB x number of mirror resources) When the synchronization mode and the vxfs file system is used: 256MB + (2MB x number of mirror resources) When the asynchronous mode is used: (Total usage of mirror disk resources) Usage of a mirror disk resource: 2MB + (I/O size x number of asynchronous queues)	140MB	640MB

Note: The I/O size is 128 KB for the vxfs file system and 4KB for file systems other than it.

For the setting value of the number of asynchronization queues, see “Understanding mirror disk resources” in the *Reference Guide*.

System requirements for the Builder

Supported operating systems and browsers

Refer to the website, *http://www.ExpressCluster.com*, *f*or the latest information. The following browser and operating system combinations have been validated:

Operating system	Browser	Language
Microsoft Windows® XP SP2 (IA-32)	IE6 SP2	English
Microsoft Windows Vista™ (IA32)	IE7	English
Microsoft Windows Server™ 2003 SP1 or later (IA-32)	IE6 SP1	English
Novell SUSE LINUX Enterprise Server 9 SP2 (IA-32)	FireFox 1.0.6	English
Novell SUSE LINUX Enterprise Server 9 SP3 (IA32)	FireFox 2.0.0.1	English

Red Hat Enterprise Linux AS/ES 4 update3 (IA-32)	FireFox 1.0.7	English
Red Hat Enterprise Linux AS/ES 4 update4 (IA32)	Konqueror3.3.1-5.13	English

Note:
The ExpressCluster Builder is only supported on IA32 systems.

Java runtime environment

Required:

Sun Microsystems, Java ™ Runtime Environment, Version 5.0 Update 6 (1.5.0_06) or later

Required memory and disk size

Required memory size: 32MB or more

Required disk size: 5MB (excluding the size required for Java runtime environment)

Supported ExpressCluster versions

Builder version	ExpressCluster X rpm version
1.0.0-1	1.0.0-1
1.0.1-1	1.0.1-1 1.0.2-1
1.1.0-1	1.1.0-1 1.1.1-1

Note:
When you use the Builder and the ExpressCluster rpm, a combination of their versions should be the one shown above. The Builder may not operate properly if they are used in a different combination.

System requirements for the WebManager

Supported operating systems and browsers

Refer to the website, *http://www.ExpressCluster.com,* for the latest information. The following operating systems and browsers combinations have been validated :

Operating system	Browser	Language
Microsoft Windows® XP SP2 (IA-32)	IE6 SP2	English
Microsoft Windows Vista™ (IA32)	IE7	English
Microsoft Windows Server™ 2003 SP1 or later (IA-32)	IE6 SP1	English
Novell SUSE LINUX Enterprise Server 9 SP2 (IA-32)	FireFox 1.0.6	English
Novell SUSE LINUX Enterprise Server 9 SP3 (IA32)	FireFox 2.0.0.1	English
Red Hat Enterprise Linux AS/ES 4 update3 (IA-32)	FireFox 1.0.7	English
Red Hat Enterprise Linux AS/ES 4 update4 (IA32)	Konqueror3.3.1-5.13	English

Note:
The ExpressCluster X WebManager is only supported on IA32 systems.

Java runtime environment

Required:

Sun Microsystems, Java ™ Runtime Environment, Version 5.0 Update 6 (1.5.0_06) or later

Required memory and disk size

Required memory size: 40MB or more

Required disk size: 600KB (excluding the size required for Java runtime environment)

Chapter 4 Latest version information

This chapter provides the latest information on ExpressCluster.

Enhanced functions

Number	Version	Enhanced point	Details
1	1.1.0-1	Supported kernels have been expanded.	Supported kernels for the following distributions have been expanded. For the versions of supported kernels, see "Supported distributions and kernel versions." RedHat Linux Enterprise Server 4
2	1.1.0-1	The group resources that can be set for individual server have been added.	In addition to a floating IP resource, configuring settings of disk, RAW, virtual IP, mirror disk resources for individual server has become available.
3	1.1.0-1	The function to check how to reflect the cluster configuration data at its distribution has been added.	The functions to check the cluster operation status at a distribution of the cluster configuration data and to display required procedures for distribution have been added.
4	1.1.0-1	The network partition resolution resource has been added.	The PINGNP resource has been added.
5	1.1.0-1	The group resource has been added.	The virtual IP resource has been added.
6	1.1.0-1	The function to switch synchronization methods of mirroring has been added.	Switching a synchronization mode of mirroring (synchronous and asynchronous) has become available.
7	1.1.0-1	Mirror recovery operation has been improved.	Activating mirror disk resources by forcibly recovering only one server when mirror disks on both servers failed has become available. For details, see "Functions of the WebManager" in the *Reference Guide*.
8	1.1.0-1	The function to set multiple mirror disk connects has been added.	Setting a mirror disk connect for backup has become available.
9	1.1.0-1	The function to use mirror resources in a cluster with more than three nodes has been added.	Using mirror resources between two servers in a cluster with more than three nodes has become available.
10	1.1.0-1	The functions to suspend and resume monitoring monitor resources have been added to the WebManager.	Suspending and resuming monitoring monitor resources on the WebManager have been available.
11	1.1.0-1	The functions to start and stop individual group resource have been added to the WebManager.	Starting and stopping only a specified resource have become available.
12	1.1.0-1	The functions to start and stop services have been added to the WebManager.	Starting and stopping the WebManager, mirror agents and cluster services have become available.
13	1.1.0-1	The real-time update function has been added to the WebManager.	Selecting the update method (updating cluster information on a specific interval or on occurrence of events on a cluster) has become available. The real-time update is set by default.

14	1.1.0-1	The function of the Mirror Disk helper has been improved.	User interfaces have been improved. Forcibly activating mirror disks, cancelling mirror recovery and stopping mirror synchronization have become available. For details, see “Functions of the WebManager” in the *Reference Guide*.
15	1.1.0-1	The management group function has been added.	The WebManager group in the previous version has been changed to the management group.
16	1.1.0-1	The linkage function between the WebManager and the Builder has been added.	Starting the online Builder from the WebManager screen has become available.
17	1.1.0-1	The online Builder function has been added.	Viewing and editing cluster configuration data by accessing a server in a cluster have become available.
18	1.1.0-1	Monitor resources have been added.	VIPW monitor resource has been added. ARPW monitor resource has been added. Monitoring agent monitor resource has been added.
19	1.1.0-1	The function to display an alert when the number of normal interconnects become one has been added.	An alert is now displayed when the number of interconnects available for internal communication become one.
20	1.1.0-1	The function to display alerts while waiting for a startup of a server has been added.	Alerts are now displayed when the process waiting for a startup of the server in a cluster is started and finished.
21	1.1.0-1	The final actions when a failure of group resources is detected have been added.	The following settings are now available for a final action when an error is detected on the activation or inactivation of a group resource. Sysrq Panic Keepalive Reset Keepalive Panic BMC Reset BMC Power Off BMC Power Cycle
22	1.1.0-1	Disk lockout process of disk resources has been improved.	Disk lockout process is now not performed on the server which does not start groups.
23	1.1.0-1	The function to set a disk lockout device for individual server has been added.	Setting a disk specified as a disk lockout device to read-only at a cluster startup has become available.
24	1.1.0-1	The communication ports for mirror ACK2 and Keepalive have been separated. The stability of communication process on multiple NMPs has been improved.	The communication ports for ACK2 used for communications between mirror drivers (for confirmation of write completion between active and standby) and for keepalive communication have been separated. Port numbers are set for each mirror disk resource. Additional port numbers are required when updating ExpressCluster from the version 1.1.0-1 or before.
25	1.1.0-1	The function to send SNMP when an alert is displayed has been added.	Running the snmptrap command when an alert is displayed has become available.
26	1.1.0-1	The information to be collected at log collection has been added.	See the command reference of the log collection command.

27	1.1.0-1	The final actions when a failure of monitor resources is detected have been added.	The following settings are now available for final actions when an error of a monitor resource is detected. Sysrq Panic Keepalive Reset Keepalive Panic BMC Reset BMC Power Off BMC Power Cycle
28	1.1.1-1	Supported kernels have been expanded.	Supported kernels for the following distributions have been expanded. For the versions of supported kernels, see “Supported distributions and kernel versions.” RedHat Linux Enterprise Server 5 Novell SUSE LINUX Enterprise Server 10

Corrected information

Number	Corrected version	Version where a failure is detected	Corrected information	Cause
1	1.1.0-1	1.0.0-1 ~ 1.0.2-1	A failover may fail when a failover and a group moving occur at the same time.	The timeout value for a failover process when a failover occurs on a group moving was invalid.
2	1.1.0-1	1.0.0-1 ~ 1.0.2-1	A cluster may stop with the groups activated when an initialization error occurs at cluster resumption.	There was not enough consideration for an error at cluster resumption on the recovery procedure of an initialization error.
3	1.1.0-1	1.0.0-1 ~ 1.0.2-1	A recovery procedure for monitoring error (group moving) may fail if the group to be recovered is being activated.	There was not enough consideration for the group status when it is changed from being started to having been started.
4	1.1.0-1	1.0.0-1 ~ 1.0.2-1	A monitor server cannot be specified for the user space monitor.	This is because the setting information of the monitor server was not used only for the user space monitor.
5	1.1.0-1	1.0.0-1 ~ 1.0.2-1	The group may remain stopped if a destination server stops while the group is being moved on a recovery procedure (group moving) of the monitoring error.	There was not enough consideration for when the group startup process was cancelled on the destination server on the group moving process on the source server.
6	1.1.0-1	1.0.0-1 ~ 1.0.2-1	A process may be terminated abnormally when multiple servers are connecting to the WebManager server.	The range of exclusion process on the xml library was invalid.
7	1.1.0-1	1.0.0-1 ~ 1.0.2-1	A timeout may occur when a cluster stops and servers may shut down.	There was an error on the determination of when an error occurred on stopping some threads.
8	1.1.0-1	1.0.0-1 ~ 1.0.2-1	Writes to a disk may fail on the DISKHB resource.	There was an error on the alignment process of a buffer address used for writes to a disk.
9	1.1.1-1	1.1.0-1	When a log collection command is carried out in the state that does not build cluster construction, file-transfer service is terminated abnormally.	There was not server information, I accessed a dirty buffer address.
10	1.1.1-1	1.1.0-1	Unnecessary failover occurs during server shut down processing by the network partition solution.	There was an error in a failover restraint condition of the server shut down processing inside.
11	1.1.1-1	1.1.0-1	Unjust process ID is set by a CLP_PID environment variable.	There was an error for processing when I set process ID to a CLP_PID environment variable.
12	1.1.1-1	1.1.0-1	When ExpressCluster was installed in reiserfs file system, start of online version Builder.may fail.	readdir() system call does not give back right file type on reiserfs.

13	1.1.1-1	1.1.0-1	When space is included in a group name and a resource name, at the time of online version Builder use, a download of the cluster constitution information may fail.	Space is converted into "+" by a URL encoding function of java and is going to acquire the file which there is not.

Chapter 5 Notes and Restrictions

This chapter provides information on known problems and how to troubleshoot the problems.

This chapter covers:

Designing a system configuration

Hardware selection, system configuration, and shared disk configuration are introduced in this section.

Supported operating systems for the Builder and WebManager

◆ The Builder and WebManager must be accessed from a Java enabled browser running a 32-bit machine.

Hardware requirements for mirror disks

◆ Disks to be used as a mirror resource do not support a Linux md and/or LVM stripe set, volume set, mirroring, and stripe set with parity.

◆ Mirror partitions (data partition and cluster partition) to use a mirror resource.

◆ There are two ways to allocate mirror partitions:

- Allocate a mirror partition (data partition and cluster partition) on the disk where the operating system (such as root partition and swap partition) resides.
- Reserve (or add) a disk (or LUN) not used by the operating system and allocate a mirror partition on the disk.

◆ Consider the following when allocating mirror partitions:

- When maintainability and performance are important:
  - It is recommended to have a mirror disk that is not used by the OS.
- When LUN cannot be added due to hardware RAID specification or when changing LUN configuration is difficult in hardware RAID pre-install model:
  - Allocate a mirror partition on the same disk where the operating system resides.

◆ When multiple mirror resources are used, it is recommended to prepare (adding) a disk per mirror resource. Allocating multiple mirror resources on the same disk may result in degraded performance and it may take a while to complete mirror recovery due to disk access performance on Linux operating system.

◆ Disks used for mirroring must be the same in all servers.

- Disk type

  Mirror disks on both servers and disks where mirror partition is allocated should be of the same disk type

  For supported disk types, see "Supported disk interfaces" on page 34.

  Example

Supported Combination	server1	server2
Yes	SCSI	SCSI
Yes	IDE	IDE
No	IDE	SCSI

- Notes when the geometries of the disks used as mirror disks differ between the servers.
  - Disk size

    Mirror disks on both servers and disks where mirror partition are allocated should be of the same disk size. It is recommended to use the disk of the same model for both servers.
  - When disk size differs

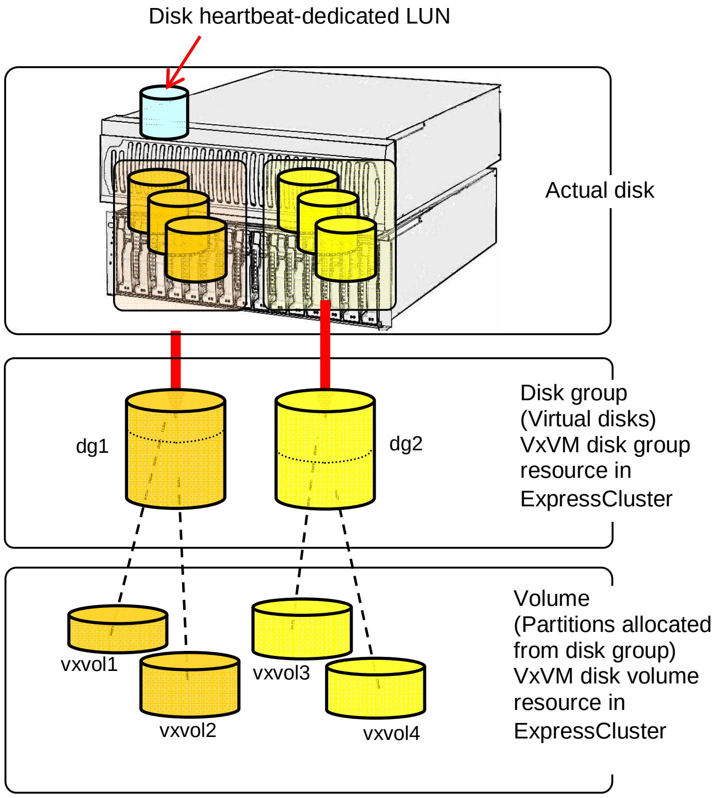
    The partition size allocated by the fdisk command is aligned by the number of blocks (units) per cylinder. Allocate a data partition considering the relationship between data partition size and direction for initial mirror configuration to be as indicated below:

    Source server ≦ Destination server

    “Source server” refers to the server where the failover group that a mirror resource belongs has a higher priority in failover policy. “Destination server” refers to the server where the failover group that a mirror resource belongs has a lower priority in failover policy.

Hardware requirements for shared disks

- A shared disk does not support a Linux md stripe set, volume set, mirroring, and stripe set with parity.
- When a Linux LVM stripe set, volume set, mirroring, or stripe set with parity is used, the following restrictions apply:
  - ExpressCluster cannot control ReadOnly/ReadWrite of the partition configured for the disk resource. Make sure not to mount the same file system on the server where disk resource is not activated.
  - You cannot use any disk monitor resource to monitor LVM logical volumes. Set the device that configures LVM logical volumes as a monitored destination of disk monitor resource. Monitor the device by using the multi target monitor.
- When you use VxVM, LUN that is not controlled by VxVM is required on a shared disk for disk heartbeat of ExpressCluster. You should bear this in your mind when configuring LUN on the shared disk.



NIC link up/down monitor resource

Some NIC boards and drivers do not support required ioctl( ).

The propriety of a NIC Link Up/Down monitor resource of operation can be checked by the ethtool command which each distributor offers.

The execution result of a following ethtool command is "Link detected. : when it is yes", it can be judged that a NIC Link Up/Down monitor resource can be operated.

Keep in mind that it may not correspond to the ethtool command depending on a network driver.

```
ethtool eth0
Settings for eth0:
     Supported ports: [ TP ]
     Supported link modes:   10baseT/Half 10baseT/Full
                             100baseT/Half 100baseT/Full
                             1000baseT/Full
     Supports auto-negotiation: Yes
     Advertised link modes:  10baseT/Half 10baseT/Full
                             100baseT/Half 100baseT/Full
                             1000baseT/Full
     Advertised auto-negotiation: Yes
     Speed: 1000Mb/s
     Duplex: Full
     Port: Twisted Pair
     PHYAD: 0
     Transceiver: internal
     Auto-negotiation: on
     Supports Wake-on: umbg
     Wake-on: g
     Current message level: 0x00000007 (7)
     Link detected: yes
```

The above is as a result of [of the evaluation environment for support] verification. Please look at the website which a distributor offers beforehand before use in user environment.

Write function of the mirror resource

- A mirror disk writes data in the disk of its own server and the disk of the remote server via network. Reading of data is done only from the disk on own server.
- Writing functions shows poor performance in mirroring when compared to writing to a single server because of the reason provided above. For a system that requires through-put as high as single server, use a shared disk.

Installing operating system

Notes on parameters to be determined when installing an operating system, allocating resources, and naming rules are described in this section.

/opt/nec/clusterpro file system

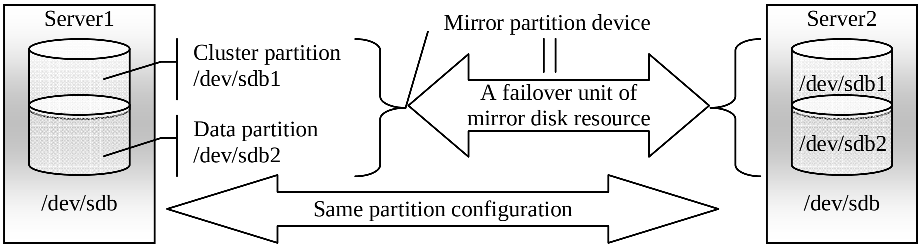
It is recommended to use a file system that has journaling functions to improve tolerance for system failure.

Mirror disks

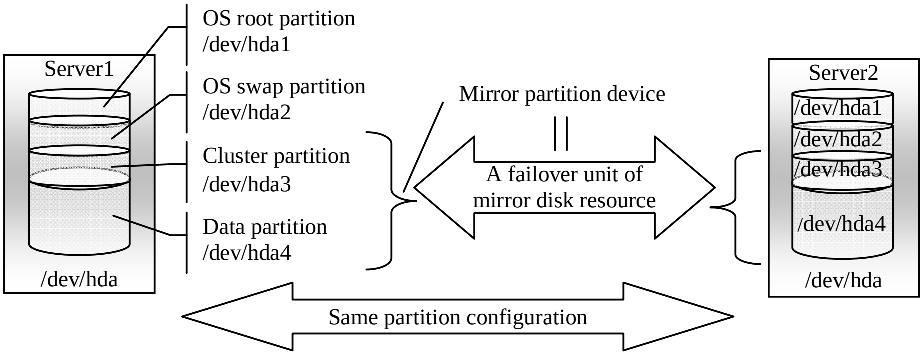
◆ Disk partition

When you make settings of disk partitions, make sure that both servers can access the same partition on the servers by the same device name.

Example: When adding one SCSI disk to each of both servers and making a pair of mirrored disks:



Example: When using free space of IDE disks of both servers, where the OS is stored, and making a pair of mirrored disks:



- Mirror partition device refers to cluster partition and data partition.
- Allocate cluster partition and data partition on each server as a pair.
- It is possible to allocate a mirror partition (cluster partition and data partition) on the disk where the operating system resides (such as root partition and swap partition.).
  - When maintainability and performance are important:

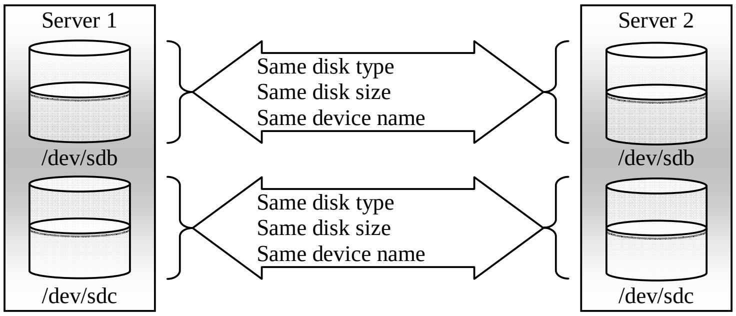
    It is recommended to have a mirror disk that is not used by the operating system (such as root partition and swap partition.)
  - When LUN cannot be added due to hardware RAID specification: or
    When changing LUN configuration is difficult in hardware RAID pre-install model:

    It is possible to allocate a mirror partition (cluster partition and data partition) on the disk where the operating system resides (such as root partition and swap partition.)

- ◆ Disk configurations

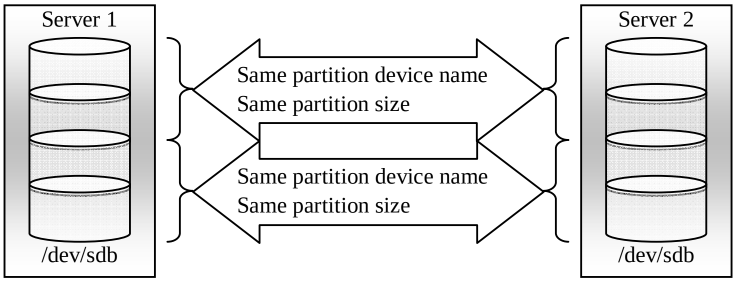
Multiple disks can be used as mirror disks on a single server. Or, you can allocate multiple mirror partitions on a single disk.

Example: When adding two SCSI disks to each of both servers and making two pairs of mirrored disks:



- Allocate two partitions, cluster partition and data partition, as a pair on each disk.
- Use of the data partition as the first disk and the cluster partition as the second disk is not permitted.

Example: When adding one SCSI disk to each of both servers and making two mirror partitions:



- ◆ A disk does not support a Linux md and/or LVM stripe set, volume set, mirroring, and stripe set with parity.

Dependent library

- ◆ libxml2

Install libxml2 when installing the operating system.

Dependent driver

- ◆ softdog

This driver is necessary when softdog is used to monitor user mode monitor resource.

Configure a loadable module. Static driver cannot be used.

Mirror driver

Use mirror partition’s major number 218. Do not use major number 218 for other device drivers.

Kernel mode LAN heartbeat and keepalive drivers

- ◆ Use major number 10, minor number 240 for kernel mode LAN heartbeat driver.
- ◆ Use major number 10, minor number 241 for keepalive driver.

Make sure to check that other drivers are not using major and minor numbers described above.

Raw monitor resource partition

Allocate a partition for monitoring when setting raw monitor resource. The partition size should be 10 MB.

Before installing ExpressCluster

Notes after installing an operating system, when configuring OS and disks are described in this section.

Communication port number

In ExpressCluster, the following port numbers are used by default. You can change the port number by using the Builder except “keepalive between mirror drivers.".

Make sure not to access the following port numbers from a program other than ExpressCluster.

Configure to be able to access the port number below when setting a firewall on a server.

Server to Server					
From			**To**		**Used for**
Server	Automatic allocation[3]	→	Server	29001/TCP	Internal communication
Server	Automatic allocation	→	Server	29002/TCP	Data transfer
Server	Automatic allocation	→	Server	29002/UDP	Heartbeat
Server	Automatic allocation	→	Server	29003/UDP	Alert synchronization
Server	Automatic allocation	→	Server	29004/TCP	Communication between mirror agents
Server	Automatic allocation	→	Server	29006/UDP	Heartbeat (kernel mode)
Server	Automatic allocation	→	Server	XXXX[4]/TCP	Mirror disk resource data synchronization
Server	Automatic allocation	→	Server	XXXX[5]/TCP	Communication between mirror drivers

[3] In automatic allocation, a port number not being used at a given time is allocated.

[4] This is a port number used per mirror disk resource and is set when creating mirror disk resource. A port number 29051 is set by default. When you add a mirror disk resource this value is automatically incremented by 1. To change the value, click **Details** tab in the **[md] Resource Properties** dialog box of the Builder. For more information, refer to Chapter 4, “Group resource details” in the *Reference Guide*.

[5] This is a port number used per mirror disk resource and is set when creating mirror disk resource. A port number 29031 is set by default. When you add a mirror disk resource, this value is automatically incremented by 1. To change the value, click **Details** tab in the **[md] Resource Properties** dialog box of the Builder. For more information, refer to Chapter 4, “Group resource details” in the *Reference Guide*.

Server	Automatic allocation	→	Server	XXXX[6]/TCP	Communication between mirror drivers
Server	Automatic allocation	→	Server	icmp	keepalive between mirror drivers

WebManager to Server					
From			**To**		**Used for**
WebManager	Automatic allocation	→	Server	29003/TCP	http communication

Server connected to the Integrated WebManager to Target server					
From			**To**		**Used for**
Server connected to the Integrated WebManager	Automatic allocation	→	Server	29003/TCP	http communication

Clock synchronization

In a cluster system, it is recommended to synchronize multiple server clocks regularly. Synchronize server clocks by using ntp.

NIC device name

Depending on the ifconfig command specification, there is a limitation in length of NIC device name that is operable in ExpressCluster. The length varies according to the number of floating IP resources.

If you want to change the NIC device name from its default name (such as eth0 and eth1), set the device name within the range of length as indicated below.

There is also a limitation in length of bonding device name. Set the bonding device name within the range of length allowed for NIC device name.

Number of floating IP resources and virtual IP resources	Length of NIC device name
0 to 10	Up to 7 characters
11 to 100	Up to 6 characters
100 and up	Up to 5 characters

Shared disk

- When you continue using the data on the shared disk at times such as server reinstallation, do not allocate a partition or create a file system.
- The data on the shared disk gets deleted if you allocate a partition or create a file system.
- ExpressCluster controls the file systems on the shared disk. Do not include the file systems on the shared disk to /etc/fstab in operating system.

[6] This is a port number used per mirror disk resource and is set when creating mirror disk resource. A port number 29071 is set by default. When you add a mirror disk resource this value is automatically incremented by 1. To change the value, click **Details** tab in the **[md] Resource Properties** dialog box of the Builder. For more information, refer to Chapter 4, “Group resource details” in the *Reference Guide*.

◆ Set a shared disk by following the steps below:

1. Allocate disk heartbeat partition.
   - Create a partition for ExpressCluster on a shared disk. Create it from a server in a cluster that uses a shared disk.
   - Allocate a partition by using the fdisk command.
   - Allocate 83 Linux fo a partition ID.
   - Allocate a partition for disk heartbeat resource on each disk (LUN.)
   - Allocate 10MB (10*1024*1024 bytes) for a disk heartbeat partition. Note that the size actually allocated will be larger than 10MB due to disk size difference. However, it does not cause any problem.
   - Allocate a disk heartbeat partition on each LUN. Make sure to allocate a dummy partition on LUN that does not use disk heartbeat as well because a file system may get corrupted when the device name changes due to a disk failure.
   - Make sure to allocate the same partition number for disk heartbeat partition on each LUN.
   - It is recommended to use one or two heartbeat resources in a cluster even when multiple LUNs are used. Set a disk heartbeat resource considering a disk load because it performs Read/Write to a disk at every heartbeat interval.
2. Allocate disk resource partition.
   - Create a partition that uses disk resource on a shared disk. Create it from a server in a cluster that uses a shared disk.
   - Allocate a partition by using the fdisk command. Allocate a partition ID, 83 Linux.
3. Create a file system.
   - Create a file system for a disk resource partition on a shared disk.
   - Create a file system by using the mkfs command from a server in a cluster that uses a shared disk.
   - A file system for a disk heartbeat partition does not need to be created.
   - In principle, the file system on the shared disk does not depend on others, but a problem may occur depending on fsck specification of the file system.
   - It is recommended to use a file system that has journaling function to avoid system failure.
   - Following is the currently supported file systems on IA-32 machines:

     ext2
     ext3
     xfs
     reiserfs
     jfs
     vxfs (Refer to Chapter 3 Supported distributions and kernel versions)
4. Create a mount point.
   - Create a directory to mount a disk resource partition.
   - Create a mount point on all servers in a cluster that use disk resource.

Mirror disk

- Set a management partition for mirror disk resource (cluster partition) and a partition for mirror disk resource (data partition).
- ExpressCluster controls the file systems on mirror disks. Do not set the file systems on the mirror disks to /etc/fstab in operating system.
- Set a mirror disk by following the steps below. You need to set mirror disks on both servers:

1. Initialize a mirror disk (Required when using a disk that was being used as a mirror disk in the past):
    - Initialization is required because the cluster partition has the old data.
    - For initialization of a cluster partition, refer to the *Reference Guide*.
2. Allocate a cluster partition.
    - Create a partition that ExpressCluster uses on a mirror disk.
    - Allocate a partition by using the fdisk command.
    - Allocate 83 Linux for a partition ID.
    - Allocate a cluster partition for each mirror disk resource.
    - Allocate at least 10MB (10*1024*1024 bytes) for disk heartbeat partition. The size may be more than 10MB depending on the size of the disk; however, it does not cause any problem.
    - For details on cluster partition, refer to the *Reference Guide.*
3. Allocate a data partition.
    - Create a data partition for mirror disk resource on a mirror disk.
    - Allocate a partition by using the fdisk command.
    - Allocate 83 Linux for a partition ID.
    - Allocate more than 1GB for data partition. The partition size should be multiples of 4096 bytes. Block numbers should be multiples of 4.
    - For details on data partition, refer to the *Reference Guide*.
4. Create a file system for the data partition.
    - If **Execute initial mkfs** is set when creating cluster configuration data by using the Builder, ExpressCluster automatically creates a file system.
    - If **Execute initial mkfs** is not set when creating cluster configuration data by using the Builder, ExpressCluster does not create a file system.
    - For information on settings for **Execute initial mkfs**, refer to the *Reference Guide*.
5. Create a mount point.
    - Create a directory to mount a mirror disk resource partition.

Adjusting OS startup time

It is necessary to configure the time from power-on of each node in the cluster to the server operating system startup to be longer than the following:

- ◆ The time from power-on of the shared disks to the point they become available.
- ◆ Heartbeat timeout time.

How to adjust OS startup time when LILO or GRUB is used for OS loader is described below. When other OS loader is used, refer to the configuration guide of that loader.

- ◆ When LILO is used for the operating system loader

1. Edit /etc/lilo.conf.

   Specify the prompt option and timeout=<Startup time (in 1/10 seconds)> option, or specify the delay=<Startup time (in 1/10 seconds)> option without specifying the prompt option. In the following example, change only the underlined part.

```
---(Example 1: Output prompt. Startup time: 90 seconds)---
boot=/dev/sda
map=/boot/map
install=/boot/boot.b
prompt
linear
timeout=900
image=/boot/vmlinuz
          label=linux
          root=/dev/sda1
          initrd=/boot/initrd.img
          read-only

---(Example 2: Not output prompt. Startup time: 90 seconds)---
boot=/dev/sda
map=/boot/map
install=/boot/boot.b
#prompt
linear
delay=900
image=/boot/vmlinuz
          label=linux
          root=/dev/sda1
          initrd=/boot/initrd.img
          read-only
```

2. Run the /sbin/lilo command to make the changes of the setting effective.

- ◆ When GRUB is used for the operating system loader

1. Edit /boot/grub/menu.lst.

Specify the timeout <Startup time (in seconds)> option. In the following example, change only the underlined part.

```
---(Example: Startup time: 90 seconds)---
default 0
timeout 900

title linux
   kernel (hd0,1)/boot/vmlinuz
   root=/dev/sda2   vga=785
   initrd (hd0,1)/boot/initrd

title floppy
   root (fd0)
   chainloader +1
```

Verifying the network settings

- ◆ The network used by Interconnect or Mirror disk connect is checked. It checks by all the servers in a cluster.
- ◆ On all servers in the cluster, verify the status of the following networks using the ifconfig or ping command.
  - Public LAN (used for communication with all the other machines)
  - Interconnect-dedicated LAN (used for communication between ExpressCluster Servers)
  - Mirror disk connect LAN (used with interconnect)
  - Host name
- ◆ The IP address does not need to be set as floating IP resource or virtual IP resource in the operating system.

User mode monitor resource (monitoring method ipmi)

- ◆ When ipmi is used as monitoring method, use ipmiutil.
- ◆ ipmiutil does not come with ExpressCluster. You need to download and install the ipmiutil rpm package.
- ◆ Users are responsible for making decisions and assuming responsibilities. NEC does not support or assume any responsibilities for:
  - Inquires about ipmiutil itself.
  - Tested operation of ipmiutil
  - Malfunction of ipmiutil or error caused by such malfunction.
  - Inquiries about whether or not ipmiutil is supported by servers.
- ◆ Check whether or not your server (hardware) supports ipmiutil in advance.
- ◆ Note that even if the machine complies with ipmi standard as hardware, ipmiutil may not run if you actually try to run it.
- ◆ If you are using a software program for server monitoring provided by a server vendor, do not choose ipmi as a monitoring method. Because these software programs for server monitoring and ipmiutil both use BMC (Baseboard Management Controller) on the server, a conflict occurs preventing successful monitoring.

User mode monitor resource (monitoring method softdog)

- ◆ When softdog is selected as a monitoring method, make sure to set heartbeat that comes with OS not to start.
- ◆ When it sets softdog in a monitor method in SUSE LINUX 10, it is impossible to use with an i8xx_tco driver. When an i8xx_tco driver is unnecessary, please make it the setting that i8xx_tco is not loaded.

Log collection

- ◆ The designated function of the generation of the syslog does not work by a log collection function in SUSE LINUX 10. The reason is because the suffiies of the syslog are different. Please change setting of rotate of the syslog as follows to use the appointment of the generation of the syslog of the log collection function.
  - Please comment out “compress” and “dateext” of the /etc/logrotate.d/syslog file.

Notes when creating ExpressCluster configuration data

Notes when creating a cluster configuration data and before configuring a cluster system is described in this section.

Server reset and server panic

When ExpressCluster performs “Server Reset” or “Server Panic,” servers are not shut down normally. Therefore, the following may occur.

- Damage to a mounted file system
- Lost of unsaved data

“Server reset” or “Server panic” occurs in the following settings:

- Action at an error occurred when activating/inactivating group resources
  -Sysrq Panic
  -Keepalive Reset
  -Keepalive Panic
  -BMC Reset
  -BMC Power Off
  -BMC Power Cycle
- Final action at detection of an error in monitor resource
  -Sysrq Panic
  -Keepalive Reset
  -Keepalive Panic
  -BMC Reset
  -BMC Power Off
  -BMC Power Cycle
- Action at detection of user space monitor timeout
  - Monitoring method softdog
  - Monitoring method ipmi
  - Monitoring method keepalive

Note: “Server panic” can be set only when the monitoring method is “keepalive.”

- Shutdown stall mentoring
  - Monitoring method softdog
  - Monitoring method ipmi
  - Monitoring method keepalive

Note: “Server panic” can be set only when the monitoring method is “keepalive.”

Final action for group resource deactivation error

If you select **No Operation** as the final action when a deactivation error is detected, the group does not stop but remains in the deactivation error status. Make sure not to set **No Operation** in the production environment.

Verifying raw device for VxVM

Check the raw device of the volume raw device in advance:

1. Import all disk groups which can be activated on one server and activate all volumes before installing ExpressCluster.

2. Run the command below:

   # raw -qa

   ```
   /dev/raw/raw2:   bound to major 199, minor 2
   /dev/raw/raw3:   bound to major 199, minor 3
       (A)                         (B)
   ```

   Example: Assuming the disk group name and volume name are:

   - Disk group name: dg1
   - Volume name under dg1: vol1, vol2

3. Run the command below:

   # ls -l /dev/vx/dsk/dg1/

   ```
   brw-------    1 root    root    199,   2  May 15 22:13 vol1
   brw-------    1 root    root    199,   3  May 15 22:13 vol2
                                      (C)
   ```

4. Confirm that major and minor numbers are identical between (B) and (C).

   Never use these raw devices (A) for disk heartbeat resource, raw resource, or raw monitor resource in ExpressCluster.

Selecting mirror disk file system

Following is the currently supported file systems:

- ◆ ext2
- ◆ ext3
- ◆ xfs
- ◆ reiserfs
- ◆ jfs
- ◆ vxfs (Refer to Chapter 3 Supported distributions and kernel versions)

Consideration for raw monitor resource

- ◆ When raw monitor resource is set, partitions cannot be monitored if they have been or will be possibly mounted. These partitions cannot be monitored even if you set device name to “whole device” (device indicating the entire disks).
- ◆ Allocate a partition dedicated to monitoring and set the raw monitor resource to it.

Delay warning rate

If the delay warning rate is set to 0 or 100, the following can be achieved:

- When 0 is set to the delay monitoring rate

  An alert for the delay warning is issued at every monitoring.
  By using this feature, you can calculate the polling time for the monitor resource at the time the server is heavily loaded, which will allow you to determine the time for monitoring time-out of a monitor resource.

- When 100 is set to the delay monitoring rate

  The delay warning will not be issued.

  Be sure not to set a low value, such as 0%, except for a test operation.

Disk monitor resource (monitoring method TUR)

- You cannot use the TUR methods on a disk or disk interface (HBA) that does not support the Test Unit Ready (TUR) and SG_IO commands of SCSI. Even if your hardware supports these commands, consult the driver specifications because the driver may not support them.
- S-ATA disk interface may be recognized as IDE disk interface (hd) or SCSI disk interface (sd) by OS depending on disk controller type and distribution. When it is recognized as IDE interface, all TUR methods cannot be used. If it is recognized as SCSI disk interface, TUR (legacy) can be used. Note that TUR (generic) cannot be used.
- TUR methods burdens OS and disk load less compared to Read methods.
- In some cases, TUR methods may not be able to detect errors in I/O to the actual media.

WebManager reload interval

- Do not set the “Reload Interval” in the WebManager tab for less than 30 seconds.

LAN heartbeat settings

- You need to set at least one LAN heartbeat resource. It is recommended to set two or more LAN heartbeat resources.
- It is recommended to set both LAN heartbeat resource and kernel mode LAN heartbeat resource together.

Kernel mode LAN heartbeat resource settings

- It is recommended to use both LAN heartbeat resource and kernel mode LAN heartbeat resource for distribution kernel of which kernel mode LAN heartbeat can be used.
- It is recommended to register interconnect-dedicated LAN and public LAN as LAN heartbeat resource and kernel mode LAN heartbeat resource. (Registering more than two LAN heartbeat resources and kernel mode LAN heartbeat resources is recommended.)

COM heartbeat resource settings

- It is recommended to use a COM heartbeat resource if your environments allows. This is because using COM heartbeat resource prevents activating both systems when the network is disconnected.

After start operating ExpressCluster

Notes on situations you may encounter after start operating ExpressCluster are described in this section.

Hotplug service

When the hotplug service searches devices, the following log is recorded into the message file:

```
kernel: <liscal liscal_make_request> NMP0 I/O port is close, mount(0),
io(0).
kernel: Buffer I/O error on device NMP1, logical block 0
```

This phenomenon occurs because mirror disk resources are not activated when the hotplug service starts. However, this is not an error.
This phenomenon does not occur when operating this service by changing the settings not to use hotplug and by using coldplug instead.

File operating utility on X-Window

Some of the file operating utilities (coping and moving files and directories via GUI) on X-Window perform the following:

- Checks if the block device is usable.
- Mounts the file system if there is any that can be mounted.

Make sure not to use file operating utility that perform above operations. They may cause problem to the operation of ExpressCluster.

Messages displayed when loading a driver

When loading a mirror driver, the following messages may be displayed at the console and/or syslog. However, this is not an error.

```
kernel: liscal: no version for "struct_module" found: kernel tainted.
kernel: liscal: module license 'unspecified' taints kernel.
```

When loading the clpka or clpkhb driver, the following messages may be displayed on the console and/or syslog. However, this is not an error.

```
kernel: clpkhb: no version for "struct_module" found: kernel tainted.
kernel: clpkhb:  no version for  "strcmp" found: kernel tainted.
kernel: clpkhb: module license 'unspecified' taints kernel.
kernel: clpka: no version for "struct_module" found: kernel tainted.
kernel: clpka: module license 'unspecified' taints kernel.
```

IPMI message

When using ipmi for user mode monitor resources, the following kernel module warning log is recorded many times in the syslog.

```
modprobe: modprobe: Can`t locate module char-major-10-173
```

When you want to prevent this log from being recorded, rename /dev/ipmikcs.

Messages written to syslog when multiple mirror resources are used and activated

When more than two mirror resources are configured on a cluster, the following messages may be written to the OS message files when mirror resources are activated.

This occurs by a fsck command function (function to access a device block which is not a target of fsck) on some distributions.

```
kernel: <liscal liscal_make_request> NMPx I/O port is close,
mount(0), io(0).
kernel: Buffer I/O error on device /dev/NMPx, logical block xxxx
```

This is not a problem for ExpressCluster. If this causes any problem such as heavy use of message files, change the following settings of mirror resources.

- Select “Not Execute” on “fsck action before mount”

- Select “Execute” on “fsck Action When Mount Failed”

Limitations during the recovery operation

Do not control the following commands, clusters and groups by the WebManager while recovery processing is changing (reactivation → failover → last operation), if a group resource is specified as a recovery target and when a monitor resource detects an error.

- Stop and suspend of a cluster
- Start, stop, moving of a group

If these operations are controlled at the transition to recovering due to an error detected by a monitor resource, the other group resources in the group may not be stopped.

Even if a monitor resource detects an error, it is possible to control the operations above after the last operation is performed.

Executable format file and script file not described in manuals

Executable format files and script files which are not described in Chapter 4, “ ExpressCluster command reference” in the *Reference Guide* exist under the installation directory. Do not run these files on any system other than ExpressCluster. The consequences of running these files will not be supported.

Messages when collecting logs

When collecting logs, the message described below is displayed at the console, but this is not an error. Logs are collected successfully.

```
hd#: bad special flag: 0x03
ip_tables: (C) 2000-2002 Netfilter core team
```

("d#" is replaced with the device name of IDE.)

Cluster shutdown and reboot

When using a mirror disk, do not execute cluster shutdown or cluster shutdown reboot from the clpstdn command or the WebManager while a group is being activated.

A group cannot be deactivated while a group is being activated. Therefore, OS may be shut down in the state that mirror disk resources are not deactivated successfully and a mirror break may occur.

Shutdown and reboot of individual server

When using a mirror disk, do not shut down the server or run the shutdown reboot command from the clpdown command or the WebManager while activating the group.

A group cannot be deactivated while a group is being activated. Therefore, OS may be shut down and a mirror break may occur in the state that mirror disk resources are not deactivated successfully.

Scripts for starting/stopping ExpressCluster services

Errors occur in starting/stopping scripts as follows:

- ◆ After installing ExpressCluster (For SUSE Linux)
  When a server shutdown, the error occurs in the following stopping scripts. There is no problem for the error because services have not started.
  - clusterpro_alertsync
  - clusterpro_webmgr
  - clusterpro
  - clusterpro_md
  - clusterpro_trn
  - clusterpro_evt
- ◆ Before start operationg ExpressCluster
  When a server start up, the error occurs in the following starting scripts. There is no problem for the error because cluster confiiguration data has not uploaded.
  - clusterpro_md
- ◆ After start operating ExpressCluster (For SUSE Linux)
  When mirror disk resources are not used, the error occurs in stopping scripts at OS shutdown. There is no problem for the error because mirror agent has not started.
  - clusterpro_md
- ◆ OS shutdown after stopping services manually (Fro SUSE Linux)
  After stopping services manually, the error occurs in the following stopping scripts at OS

shutdown. There is no problem for the error because services have already stopped.

- clusterpro
- clusterpro_md

Scripts in EXEC resources

EXEC resource scripts of group resources stored in the following location.

/opt/nec/clusterpro/scripts/group-name/resource-name/

The following cases, old EXEC resource scripts are not deleted automatically.

- When the EXEC resource is deleted or renamed
- When a group that belongs to the EXEC resource is deleted or renamed

Old EXEC resource scripts can be deleted when unnessesary.

Monitor resources that monitoring timing is “Active”

When monitor resources that monitoring timing is “Active” have suspended and resumed, the following restriction apply:

- ◆ Stop target resource after suspending monitor resuorce
  After monitor resources have resumed, the monitoring differs depending on monitor resource.
  - PID Monitor Resource
    When the EXEC resource is deactivated, a PID monitor resource cannot detect errors if the PID monitor resource has been suspend and is now resumed.
  - Other than PID Monitor Resource
    Monitor resource continue to monitor.
- ◆ Move target resource to other server after suspending monitor resource
  After monitor resources have resumed, the monitoring differs depending on monitor resource.
  - PID Monitor Resource (source server)
    When the EXEC resource is deactivated, a PID monitor resource cannot detect errors if the PID monitor resource has been suspend and is now resumed.
  - Other than PID Monitor Resource (source server)
    Monitor resource continue to monitor.
  - PID Monitor Resource (destination server)
    Monitoring is performed by monitor resource while specified group resource is active.
  - Other than PID Monitor Resource (destination server)
    Monitoring is performed by monitor resource while specified group resource is active.

When monitor resources that recovery target is cluster have suspended and resumed, the following restriction apply:

- ◆ Stop target resource after suspending monitor resource
  After monitor resources have resumed, the action that detected an error differs depending on monitor resource.
  - PID Monitor Resource
    PID monitor resource not be able to detect errors.

  - Other than PID Monitor Resource
    When detecting an error in a target to be monitored, a monitor resource executes final action.

- ◆ Move target resource to other server after suspending monitor resource
  After monitor resources have resumed, the action that detected an error differs depending on monitor resource.

  - PID Monitor Resource (source server)
    PID monitor resource not be able to detect errors.

  - Other than PID Monitor Resource (source server)
    When detecting an error in a target to be monitored, a monitor resource executes final action.

Notes on the WebManager

- ◆ The information displayed on the WebManager does not necessarily show the latest status. If you want to get the latest information, click the **Reload** button.

- ◆ If the problems such as server shutdown occur while the WebManager is getting the information, acquiring information may fail and a part of object may not be displayed correctly. Wait for the next automatic update or click the **Reload** button to reacquire the latest information.

- ◆ When using a browser on Linux, a dialog box may be displayed behind the window managers depending on the combination of the managers. Change the window by pressing the **ALT** + **TAB** keys.

- ◆ Collecting logs of ExpressCluster cannot be executed from two or more WebManager simultaneously.

- ◆ If the WebManager is operated in the state that it cannot communicate with the connection destination, it may take a while until the control returns.

- ◆ If you move the cursor out of the browser in the state that the mouse pointer is displayed as a wristwatch or hourglass, the cursor may be back to an arrow.

- ◆ When going through the proxy server, make the settings for the proxy server be able to relay the port number of the WebManager.

- ◆ When updating ExpressCluster, close the browser. Clear the Java cache and open the browser.

Notes on the Builder

- ◆ ExpressCluster does not have the compatibility of the cluster configuration data with the following products.
  - The Builder of other than ExpressCluster X V1 for Linux
  - The Builder of ExpressCluster for Windows Value Edition
- ◆ Closing the Web browser (by clicking **Exit** from the menu) discards the edited data. Even if the configuration is changed, the dialog box to confirm to save is not displayed. When you need to save the edited data, select **File** from the menu of the Builder and click **Save** before terminating.
- ◆ Reloading the Web browser (by selecting **Refresh** button from the menu or tool bar) discards the current editing data. Even if the configuration is changed, the dialog box to confirm to save is not displayed. When you need to save the editing data, select **File** from the menu bar of the Builder and click **Save** before reloading.
- ◆ When creating the cluster configuration data using the Builder, do not enter the value starting with 0 on the text box. For example, if you want to set 10 seconds for a timeout value, enter “10” but not “010.”

Notes on mirror disks

When changing the size of mirror partitions after the operation is started, see “Changing offset or size of a partition on mirror resource” in Chapter 9 “The system maintenance information” in the *Reference Guide*.

Chapter 6 Upgrading ExpressCluster

This chapter provides information on how to upgrade ExpressCluster.

This chapter covers:

How to upgrade from ExpressCluster V3

Backing up the cluster configuration data

Back up the cluster configuration data on a floppy disk.

Back up the cluster configuration data as root user. Follow the procedure listed below on the master server.

1. Insert a floppy disk on the device.
2. If the floppy disk is unformatted, format it with the fdformat command so that it can be used with the tar command.
3. Run the following command.

   When the Builder is used on Linux machine

   ```
   clpcfctrl --pull –l
   ```

   When the Builder is used on Windows machine

   ```
   clpcfctrl --pull -w
   ```

4. Go to the next step after ejecting the floppy disk from the device. Use the floppy disk after installing the ExpressCluster X V1.

Converting the cluster configuration data

Convert the cluster configuration data that you have backed up to the data for ExpressCluster X V1. Use the Builder whose version supports the server RPM of ExpressCluster X V1 to be installed. Refer to the *Installation and Configuration Guide* for how to install the Builder.

1. Insert the floppy disk on PC or server that you use the Builder.
2. Start up the Builder.
3. From menu of the Builder, click **File(F)**, click **Open(O)**, and then click **Change (C)**.
4. Open the data by selecting the clp.conf on the floppy disk.
5. Open the **Cluster Properties**, and then select the **Info** tab. Select the language used in a cluster. For details on how to set a language, see Chapter 3 “Functions of the Builder” in the *Reference Guide*.
6. From menu of the Builder, click **File(F)**, and click **Save(S)**. Select **Yes(Y)** in the overwrite confirmation dialog box.
7. Exit the Builder.
8. Go to the next step after ejecting the floppy disk from the device. Use the floppy disk after installing the ExpressCluster X V1.

Uninstalling ExpressCluster V3

Uninstall 3.x as root user. Uninstall the ExpressCluster Server by following the procedure listed below.

1. Disable the services by running the **`chkconfig --del`** ***`name`*** in the following order. Specify one of the following services in *`name`*.
   - clusterpro_alertsync
   - clusterpro_webmgr
   - clusterpro
   - clusterpro_md
   - clusterpro_trn
   - clusterpro_evt
2. Reboot the server.
3. Back up the cluster configuration data. For details, see “Backing up the cluster configuration data.”
4. Run the following:

```
rpm -e <ExpressCluster v3 package name>
Example: rpm –e eclan-svr-3.1
```

Installing ExpressCluster X V1

Install the ExpressCluster Server RPM as root user. Install the Server RPM on all the servers by following the procedure listed below.

1. Mount the media of the installation CD-ROM.
2. Install the package file by running the rpm command.
   The RPM for installation is different depending on architecture.

   In the CD-ROM, move to `/server` and run the following:
   `rpm –i --nodeps <ExpressCluster X server package name>`

   Example: `rpm –i ––nodeps ecxlan-svr-1.1.0-1.1.i686.rpm`
3. After completing installation, unmount the installation CD-ROM media.
4. Reboot the server after removing the CD-ROM.
5. Convert the cluster configuration data you backed up when uninstalling ExpressCluster Ver 3.x to the one for X1.0. For details, see “Converting the cluster configuration data.”
6. Distribute the converted cluster configuration data to the servers that configure a cluster by the clpcfctrl command. Run the command on one of the server that configures a cluster. For details, see “Creating a cluster” in Chapter 4, “Installing ExpressCluster” in the *Installation and Configuration Guide*.
7. Register a license and reboot the server. For details on how to register a license, see Chapter 5, “Registering the license” in the *Installation and Configuration Guide*.
8. Check the status of a cluster by using the WebManager. For details, see Chapter 6, “Starting up a cluster system” in the *Installation and Configuration Guide*.

How to update ExpressCluster X to the latest

Follow the steps below to update the Server RPM version 1.0.1-1 - 1.0.2-1 to 1.1.0-1 or later.

Updating the ExpressCluster Server RPM

Install the ExpressCluster Server RPM as root user. Install the Server RPM on all the servers by following the procedure below.

1. Disable the services by running the **`chkconfig --del`** ***`name`*** in the following order. Specify one of the following services in *`name`*.

   clusterpro_alertsync

   clusterpro_webmgr

   clusterpro

   clusterpro_md

   clusterpro_trn

   clusterpro_evt

2. Restart all the servers in a cluster.
3. Mount the media of the installation CD-ROM.
4. Confirm that ExpressCluster services are not running, and then install the package file by executing the rpm command. The RPM for installation is different depending on architecture.

   In the CD-ROM, move to `/server` and run the following:
   `rpm –U ––nodeps <ExpressCluster X server package name>`

   Example: `rpm –U ––nodeps ecxlan-svr-1.1.0-1.1.i686.rpm`

5. After completing installation, unmount the installation CD-ROM media, and remove it.
6. Enable the services by running the **`chkconfig --add`** ***`name`*** in the following order. Specify one of the following services in *`name`*. For SuSE Linux, run the command with the ***–force*** option.

   clusterpro_md

   clusterpro

7. Repeat the step 3-6 on all the servers.
8. Reboot all the servers that configure a cluster.
9. Connect the WebManager to one server that configures a cluster.
10. Start the Builder from the connected WebManager. For details on how to start the online Builder, see the *Reference Guide.*
11. Open the **Cluster Properties**, and then select the **Info** tab. Select the language used in a cluster. For details on how to set a language, see Chapter 3, “Functions of the Builder” in the *Reference Guide*.
12. Confirm that all servers that configure a cluster are started, and then upload the configuration data from the online Builder. For details on how to operate the online Builder, see the *Reference Guide*.
13. Enable the services in the following order by running the **`chkconfig --`*****`add name`*** command. Specify the following services on *name*. For SuSE Linux, run the command with the ***--force*** option.

    clusterpro_md

    clusterpro

14. Perform step 13 on all the servers.

15. Run **Restart Manager** on the WebManager.
16. Run **Start Mirror Agent** on the WebManager.
17. Run **Start Cluster** on the WebManager.

Section Appendix

Appendix A. Glossary

Cluster partition

A partition on a mirror disk. Used for managing mirror disks.
(Related term: Disk heartbeat partition)

Interconnect

A dedicated communication path for server-to-server communication in a cluster.
(Related terms: Private LAN, Public LAN)

Virtual IP address[7]

IP address used to configure a remote cluster.

Management client

Any machine that uses the WebManager to access and manage a cluster system.

Startup attribute

A failover group attribute that determines whether a failover group should be started up automatically or manually when a cluster is started.

Shared disk

A disk that multiple servers can access.

Shared disk type cluster

A cluster system that uses one or more shared disks.

Switchable partition

A disk partition connected to multiple computers and is switchable among computers.
(Related terms: Disk heartbeat partition)

Cluster system

Multiple computers are connected via a LAN (or other network) and behave as if it were a single system.

Cluster shutdown

To shut down an entire cluster system (all servers that configure a cluster system).

Active server

A server that is running for an application set.
(Related term: Standby server)

Secondary server

A destination server where a failover group fails over to during normal operations.
(Related term: Primary server)

Standby server

A server that is not an active server.
(Related term: Active server)

Disk heartbeat partition

A partition used for heartbeat communication in a shared disk type cluster.

Data partition

A local disk that can be used as a shared disk for switchable partition. Data partition for mirror disks.
(Related term: Cluster partition)

Network partition

All heartbeat is lost and the network between servers is partitioned.
(Related terms: Interconnect, Heartbeat)

[7] This applies only for Windows version.

Node

A server that is part of a cluster in a cluster system. In networking terminology, it refers to devices, including computers and routers, that can transmit, receive, or process signals.

Heartbeat

Signals that servers in a cluster send to each other to detect a failure in a cluster.
(Related terms: Interconnect, Network partition)

Public LAN

A communication channel between clients and servers.
(Related terms: Interconnect, Private LAN)

Failover

The process of a standby server taking over the group of resources that the active server previously was handling due to error detection.

Failback

A process of returning an application back to an active server after an application fails over to another server.

Failover group

A group of cluster resources and attributes required to execute an application.

Moving failover group

Moving an application from an active server to a standby server by a user.

Failover policy

A priority list of servers that a group can fail over to.

Private LAN

LAN in which only servers configured in a clustered system are connected.
(Related terms: Interconnect, Public LAN)

Primary (server)

A server that is the main server for a failover group.
(Related term: Secondary server)

Floating IP address

Clients can transparently switch one server from another when a failover occurs.
Any unassigned IP address that has the same network address that a cluster server belongs to can be used as a floating address.

Master server

The server displayed on top of the **Master Server** in **Cluster Properties** in the Builder.

Mirror connect

LAN used for data mirroring in a data mirror type cluster. Mirror connect can be used with primary interconnect.

Mirror disk type cluster

A cluster system that does not use a shared disk. Local disks of the servers are mirrored.

Appendix B. Index